AC Comment to RC1

We are thankful for the referee's valuable time in helping to improve the manuscript. This document will address all reviewer comments (gray boxes). We believe that the proposed changes will majorly improve the manuscript.

Considering the referee's comments, our planned improvements for the revised submission can be summarized as follows:

(1) In our opinion, the main critique of referee 1 concerned the models selected for comparison. We fully agree that comparing models that use additional exogenous variables, such as precipitation, with models that do not, results in an unfair comparison. To address this, referee 1 suggested to include a total of four models, namely: The original ARIMA and PBHM-HLSTM models as well as an ARIMA model that utilizes exogenous variables (ARIMAX) as well as an LSTM model that uses the same input variables as the original ARIMA model, termed eLSTM. We ran all models suggested by referee 1 and observed that the ARIMAX model did not achieve an improved forecast quality when compared to the already presented ARIMA model (see results below). For this reason, we see no added benefit of incorporating the ARIMAX model into the revised manuscript. Consequently, we suggest the following model selection for the revised manuscript:

   a. **ARIMA**: This is the original ARIMA model presented in the preprint. This model utilizes simulated and observed runoff in the hindcast for correcting the hydrologic model's forecasts.

   b. **eLSTM**: This model posses the proposed hindcast-forecast architecture of the PBHM-HLSTM but will only ingest simulated and observed runoff in the hindcast and simulated runoff in the forecast (similar to ARIMA). This model can be seen as an intermediate between ARIMA and the final PBHM-HLSTM, which in our opinion results in a more fair comparison between the individual models, as it features the data used for ARIMA but the architecture of the PBHM-HLSTM.

   c. **PBHM-HLSTM**: This is the original PBHM-HLSTM model of the preprint.

(2) Another comment concerned the methodology of how we evaluated whether the LSTM is benefiting from the simulated runoff or not. We used overall statistics and the generalization capabilities as a proxy for judging if the LSTM benefited from the PBHM's results or not. Referee 1 suggested to perform this investigation by doing a sensitivity analysis instead (like integrated gradients or ingesting noise in the model), which in our opinion has the potential to majorly improve the manuscript. We already performed some analysis and will incorporate them in the revised manuscript as follows:

   a. The LSTM (HLSTM) that does not include the simulated runoff will be excluded from the manuscript, as referee 1 suggested.

   b. Instead, we will perform a sensitivity analysis using integrated gradients. We will present these results in terms of annual means and at the investigated flooding events (see preliminary results below)

(3) As referee 1 pointed out correctly, the flaw in the study design is also reflected in the formulated research questions. This will be changed accordingly to align with the major updates presented in points (1) and (2).

(4) Another comment addressed the choice of the hyperparameters of the LSTM, i.e., the model was too large. We fully agree that less hidden nodes should be used for regional studies. The models were already retrained and will be updated in the revised version of the manuscript (see below).

(5) We will also address all minor comments as stated below.

In our opinion, the above presented modifications address all major comments of referee 1 and their implementation will majorly benefit the final manuscript.

Dear editor and authors,

The following comment details my review of the manuscript “Long Short-Term Memory Networks for Real-time Flood Forecast Correction: A Case Study for an Underperforming Hydrologic Model” submitted to HESS.

In this preprint the authors present a model comparison study in which two (or three depending on the application) models are compared in their ability to forecast runoff. The models compared are all statistical- or machine learning-based models which take as inputs predictions of an underperforming conceptual model. The preprint is well written and the results are compelling. The scope of the manuscript is well suited for HESS and it has potential to be a great contribution to the literature on runoff forecasting, as well as models which combine physics-based and data-driven approaches.

However, I have a number of major and minor comments/suggestions that should be addressed before final publication and ultimately will benefit the manuscript and overall study.

Major Comments

The comparison is not “fair”

What the ARIMA model is doing is very different than what the LSTM-based models are doing and this “unfair” comparison is apparent in the results. Evidently the model which is able to use data from precipitation in its forecasting step will be better at predicting events that have precipitation as its main driver and not the current or past discharge as calculated by an underperforming PBHM.

What is missing is a model that is in-between the ARIMA and HLSTM-PBHM and bridges the gap between the two approaches. In principle this could be an ARIMA which considers exogenous inputs (ARIMAX) or an LSTM which predicts errors without the aid of external variables for a direct comparison with the presented ARIMA model. This way we see how performance changes from having a model that is only correcting the PBHM (ARIMA), to a model which relies in the PBHM but can use other inputs when the PBHM fails (ARIMAX), to a model that accounts for all available input data and chooses what to use (HLSTM-PBHM).

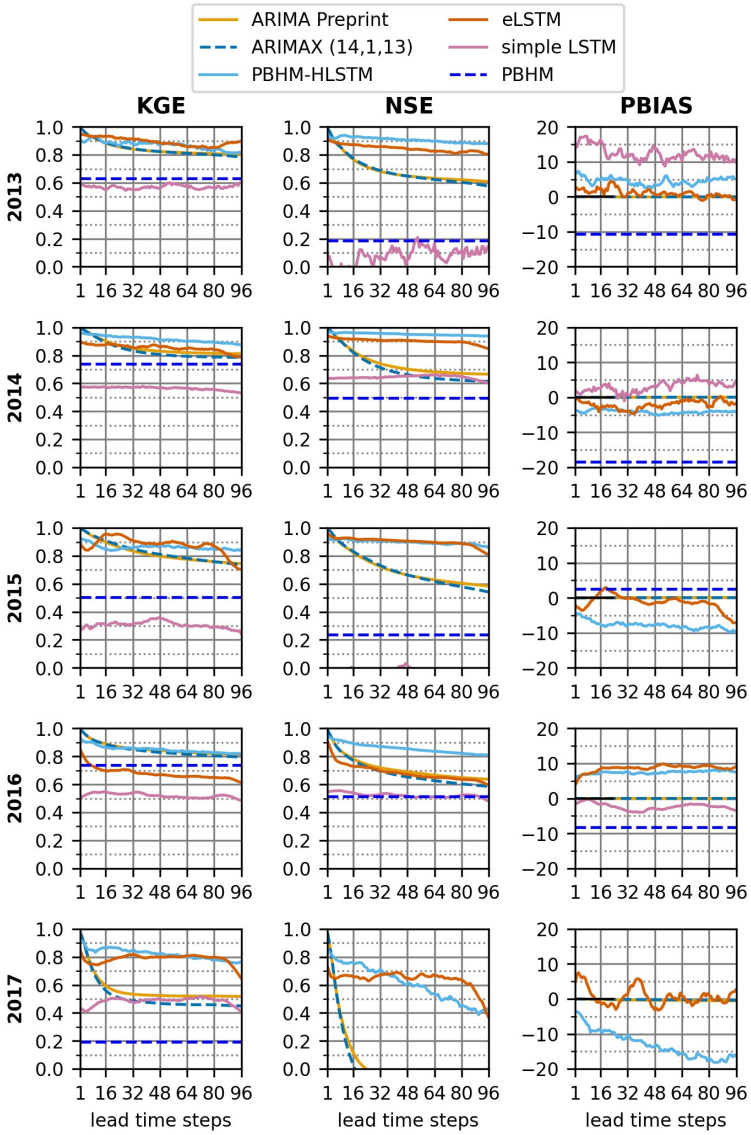
Ultimately, I think the models which should be part of the study are: ARIMA, ARIMAX, LSTM which predicts errors only using Q_{sim} and Q_{obs} (name: $_{e}$LSTM?), and the presented HLSTM-PBHM.

We agree with the referee’s comment that a direct comparison between a model that utilizes exogenous variables (LSTM), especially precipitation, and one that does not (ARIMA), is unfair. The original aim of this study was to compare an approach that can be regarded as more classic (ARIMA), as it is implemented in many existing forecasting systems, with a probably more capable

LSTM model. However, we agree that the manuscript needs a revision to allow for a more fair and hence more objective comparison between both models presented. To achieve this, we tested the models suggested by referee 1, namely: ARIMA, ARIMAX (ARIMA with exogenous variables), eLSTM and the proposed HLSTM-PBHM. For the ARIMAX model, we included the maximum precipitation as an exogenous variable as this is the major driver for flooding in this catchment. We tried multiple model configurations, i.e., using the precipitation in the hindcast and also in the forecast by shifting the precipitation time series. However, it showed that independent of the configuration used, the exogenous variable did not have a positive impact on the forecast quality (see plots below). For this reason, we do not see an added benefit of including the ARIMAX model into the revised manuscript.

Considering this, we suggest comparing the three remaining models suggested by referee 1, namely: ARIMA (identical to the model in the preprint), eLSTM (new model) and HLSTM-PBHM (the proposed model of the preprint). Thereby the eLSTM model can be interpreted as an intermediate model lying between ARIMA and the HLSTM-PBHM. The reasoning for this is that the eLSTM uses the same data as ARIMA (more or less fair comparison between ARIMA and eLSTM) and the same architecture as the HLSTM-PBHM (more or less fair comparison between eLSTM and HLSTM-PBHM).

As for the suggested eLSTM, only ingesting Qsim and Qobs, the results show a significantly better performance compared to ARIMA, but also worse performance compared to the HLSTM-PBHM model, fed also with precipitation data, which aligns with our expectations. In our opinion this will be interesting to add to the revised manuscript and will improve the comparability between the model variants.



> Furthermore, although the HLSTM was added to address a specific concern regarding the combination of the PBHM and an LSTM, I don't think its able to address this issue effectively as the authors also recognize by saying that in their findings: "We did not find strong evidence of whether the inclusion of the PBHM's results benefited the accuracy of the LSTM." My suggestion is that the HLSTM is completely dropped. Give that this model simply serves to check if Q_{sim} is somewhat informative to the LSTM in the HLSTM-PBHM, this could be made clearer through a sensitivity analysis and not using a different model with also a different architecture. I suggest the sensitivity analysis could be done using integrated gradients as Kratzert et al. (2019) or simply by replacing the input of Q_{sim} for noise and seeing the effect it has on the predictions by the model. If the LSTM does not consider the input from Q_{sim} useful, there should be no effect by replacing this input for noise and vice versa.

We fully agree with referee 1 that our original methodology (evaluating overall statistics and the generalization capabilities) was not well suited to demonstrate the impact of the PBHM's results on the proposed LSTM model. We welcome the referee's suggestion to perform a sensitivity analysis instead. We tested both suggested methodologies, replacing the features with noise and also the integrated gradients method. In our opinion the most meaningful insights were gained when applying integrated gradients as follows:

The Integrated Gradient Method is used to evaluate the importance of an input of interest for the evaluated model's output and is defined as follows:

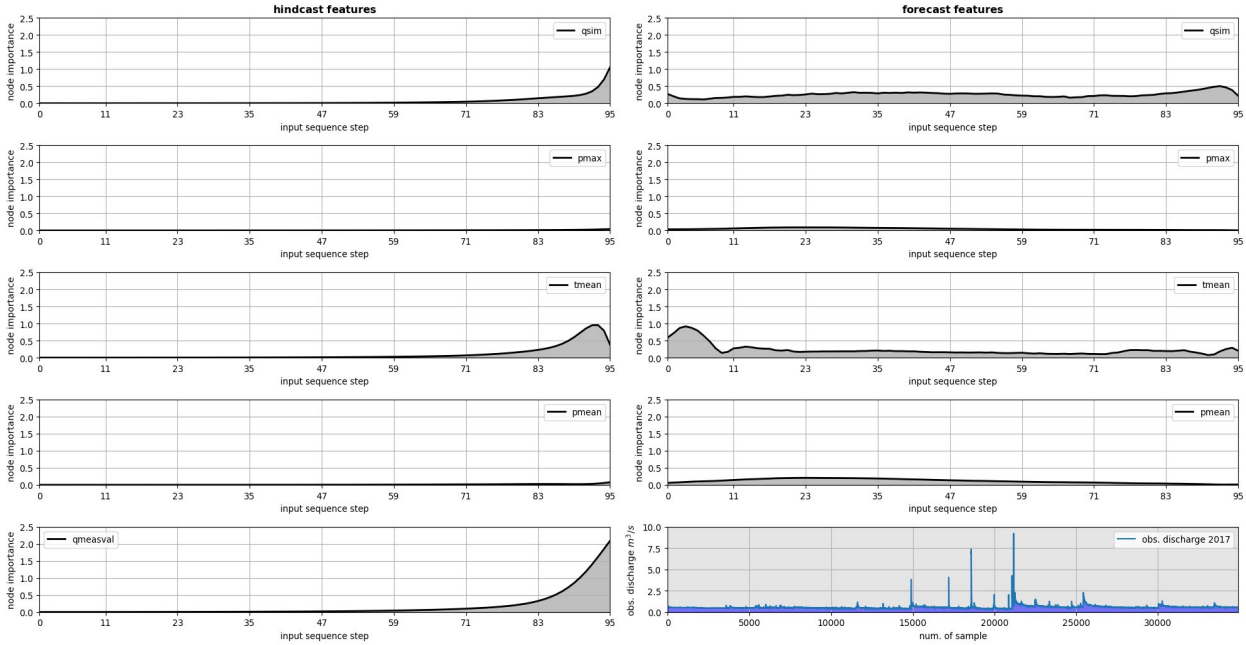
$$IntegratedGrads_i^{approx}(\boldsymbol{x}) := \left(x_i - x_i'\right) \times \sum_{k=1}^{m} \frac{\partial F\left(x' + \frac{k}{m}\left(\boldsymbol{x} - \boldsymbol{x'}\right)\right)}{\partial x_i} \times \frac{1}{m}$$

where $\boldsymbol{x}$ is the input of interest, $F(\circ)$ is the model, $\boldsymbol{x'}$ is the baseline (in our case a sequence of zeros as suggested by Kratzert, 2019), x_i is the input in the i^{th} dimension, i.e. at the i^{th} input node, and m is the step size of the approximation of the integral (here 200, being within the suggested range by Sundararajan, 2017).

In our case, the output of the model is a sequence of size 96, representing the forecasting steps. The number of input dimensions, i.e. input nodes, accumulates from five hindcast and four forecast features, each sequences of size 96, to a total of 864 integrated gradients per output node and sample. To analyze this vast amount of information, we decided on two processing approaches to evaluate (i) **the importance of the nine features for a whole year (in the presented case 2017).** Noteworthy, in the revised manuscript this will be evaluated for all folds, and (ii) **their importance at a flood peak events**, as this is the most relevant in flood forecasting.

(i) The whole year 2017 consists of 34903 samples. First, the integrated gradients for each sample are derived from the sum of the output sequence in respect to all 9x96 input nodes, i.e. dimensions. Then, the absolute values of these gradients are averaged over the whole year leading to 9x96 values, representing the input node importance for the whole output sequence.

A graphical representation of these results is shown in the following figure:



The left side of the figure shows the importance for each hindcast input node for the whole output sequence. It comes clear, that the input nodes at the end of the sequence (being closer to the forecast) have a much higher importance as the once at the beginning of it. Also, the precipitation feature generally shows very low importance, which can be explained due to the precipitation being zero throughout most of the year.

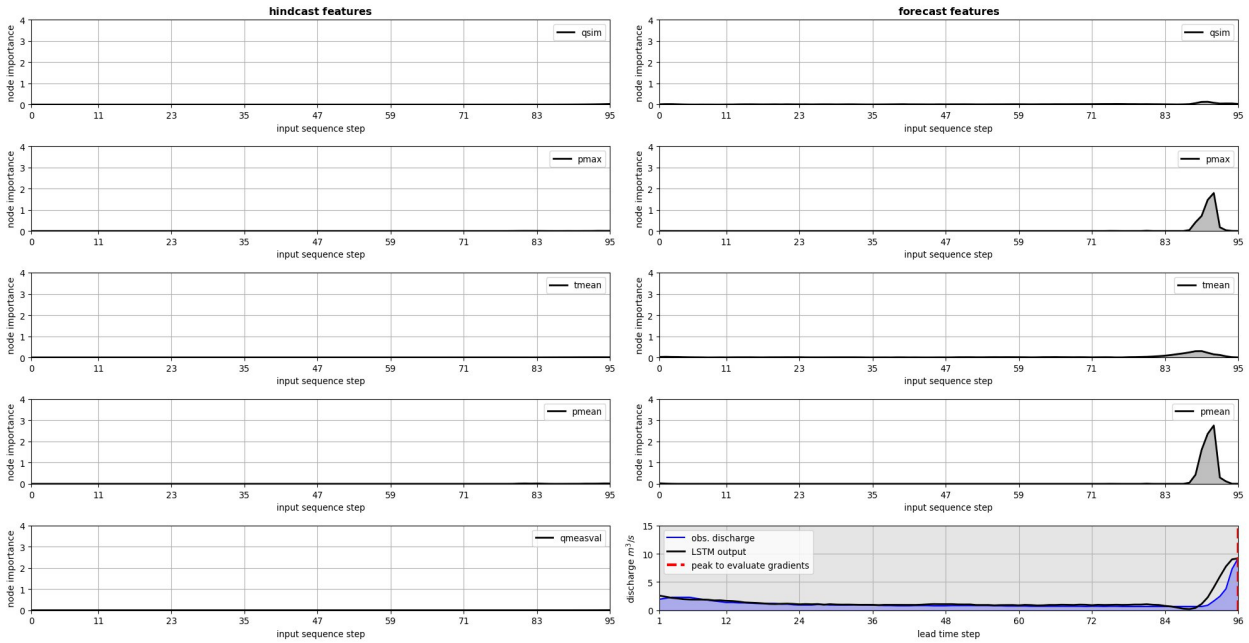
On the right side, the importance of each forecast input node is displayed. Generally, the precipitation features (*Pmean* and *Pmax*) show less impact on the total model output compared to simulation and temperature features (*Qsim* and *Tmean*).

The table below shows the sum of the input node importance for each feature, i.e. feature importance, which equals to the gray area under the curve as displayed in the figure above. Comparing the values for hindcast and forecast features individually – as these are basically two separate networks stacked on one another – the simulation *Qsim* is the most important forecast feature followed by the mean temperature *Tmean*. Regarding the hindcast, the measured runoff *Qmeasval* is the most important feature, followed by the temperature *Tmean*.

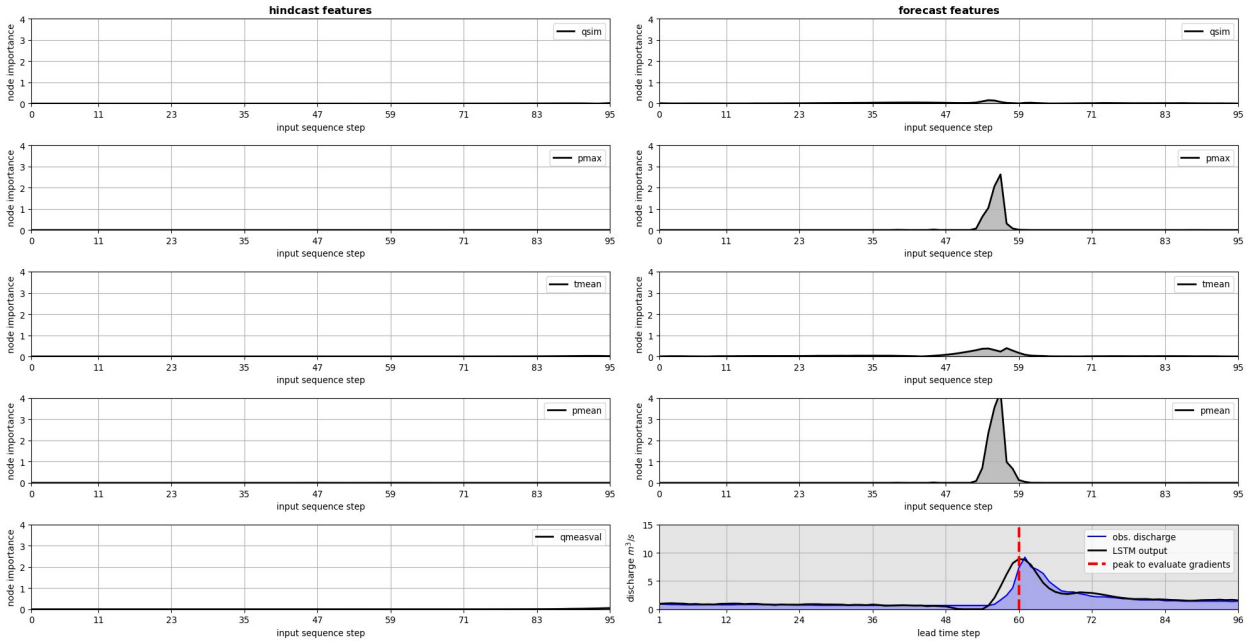
	Qsim	*Pmax*	*Tmean*	*Pmean*	*Qmeasval*
hindcast	5.94	0.45	9.58	0.85	**16.40**
forecast	**24.60**	4.34	21.44	10.76	

(ii) The feature importance for a flood peak event is calculated as the sum of the node importance over 96 consecutive samples. These samples range from the first time the flood peak appears in the forecasting horizon, namely at lead time step 96, until it becomes the next-step-forecast at lead time step 1. The integrated gradients are derived for each sample from the output's maximum. This ensures that the evaluations always take place at the output's peak.

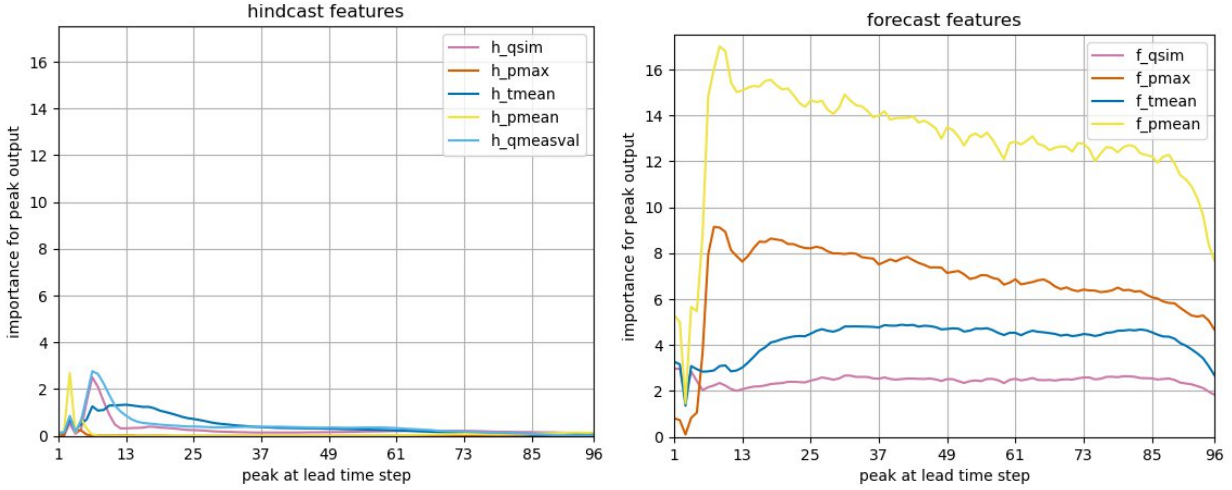
The figure below shows the corresponding node importance for the first sample when the peak of the output is located at lead time step 96. It can be seen, that the importance of all the input nodes of the forecast is highest right before the output peak. Thereby, the precipitation features *Pmean* and *Pmax* show the highest importance.



The next plot displays the node importance when the output peak is located at lead time step 60 (sample 36). Again, the importance of all the input nodes of the forecast is highest right before the output peak, while precipitation features' importance is the highest overall.



To summarize these findings, we calculated the sum of the node importance per feature for each sample and visualized it according to the peak position in the forecast window. The figure below then shows the individual feature importance for a peak output at the corresponding lead time step.



It can be seen, that the precipitation forecast features (*Pmean* and *Pmax*) have the highest impact on the peak in the output sequence of the model, while the simulation *Qsim* has the lowest.

The conclusion of this is that, in general, the LSTM heavily relies on the PBHM's simulations when no precipitation/flooding events are present. However, the PBHM's results loose importance at peak flow predictions, possibly due to the poor initial guess of the underlying hydrologic model. Interestingly, also the temperature has a high importance considering the annual mean, which possible can be attributed to the model gaining some seasonality information from this feature. A more comprehensive analyses of these results will be presented in the revised manuscript.

Kratzert, Frederik, and Herrnegger, Mathew, and Klotz, Daniel, and Hochreiter, Sepp, and Klambauer, Günter. (2019). NeuralHydrology – Interpreting LSTMs in Hydrology. 10.1007/978-3-030-28954-6_19.

Sundararajan, Mukund, and Ankur, Taly, and Qiqi, Yan. (2017). Axiomatic Attribution for Deep Networks. Proceedings of Machine Learning Research. 70. https://arxiv.org/abs/1703.01365.

This flaw in the design of the study is also reflected on the research questions established in the introduction. None of the research questions concern the ARIMA model, so why is it part of the study at all? From the point of view of the RQs, the study should only focus on the HLSTM-PBHM and the previously described *e*LSTM which could be considered a deep-learning adaptation of ARIMA while the HLSTM-PBHM is more akin to an ARIMAX, keeping the scope of the paper within error correcting strategies, and then the discussion can focus on the benefit of precipitation as an input during forecasting, and the difference between years where the PBHM is acceptable (2014 and 2016) in contrast to when it's terrible (2017).

We agree with the referee that the shortcomings in the study design are also reflected in the established research questions. Overall, we do believe that the research questions should clearly reflect the intentions of the study. The original idea of the study was to assess the potential of LSTM models to correct poor forecasts in cases where ARIMA models may result in unfavorable results. In our opinion this very relevant in the field of flood forecasting as many operational forecasting systems currently rely on ARIMA type correction strategies. So we do believe that the research questions should be changed accordingly to still reflect the intentions of the study, while adding the valuable suggestions of the referee:

(1) RQ1: Research question one will deal with an overall comparison of the ARIMA results with the proposed PBHM-HLSTM and the intermediate eLSTM model. ARIMA in this regard can be seen as the classic approach employed in many forecasting systems (We will back this up with more literature of existing forecasting systems).
(2) RQ2: Research question two will deal with an comparison of the peak runoff prediction performance, as this is the most important thing in operational flood forecasting
(3) RQ3: This research question will still deal with the usefulness of the PBHM on the LSTM forecasts but evaluated with the suggestions given by referee 1. In our opinion this is a very important point as the question "why not simply replace the PBHM with a LSTM?" is a very valid and crucial one given the poor predictive skills of the original PBHM.

Hyperparameters of the LSTM-based models

In the supplemental information of the paper by Nevo et al. (2022), their model is described to have an LSTM of 128 hidden units for hindcast and another 128 hidden units for forecast, which is similar to the 96 used in this article, but in the case of Nevo et al. (2022), the model was trained to forecast in at least 165 basins which use LSTM for the "stage forecast model". The architecture presented by Nevo et al. (2022) is also based on the MTS-LSTM presented by Gauch et al. (2021). Although the purpose of that second paper is not forecasting, the idea of "handing" the hidden states from one LSTM to another is the same, and in their case both LSTMs which send and receive the hidden states have 64 hidden nodes. This is also applied in a regional case study in which the amount of data that the model needs to ingest is a lot larger. Finally, in a more recent example by some of the same authors of the previous papers, Kratzert et al. (2024) train single-basin LSTMs using models with hidden nodes ranging from 8 to, at most, 32. This is not a criticism of not using LSTM in a regional setting, in my view LSTMs are

still valid for application in a single basin, acknowledging their limitations, but their size shouldn't be the same as those used in regional modelling.

This could be addressed by adding smaller sizes into the hyperparameter search space and I would encourage the authors to present their results for training/validation in supplemental material of the article in the form of loss curves, metrics in training/validation, etc.

We agree with referee 1 that the amount of nodes should scale with the complexity of the task at hand, i.e., a regional model should be less complex than for cases with multiple basins. We will address this issue by retraining (most of it has been done already as shown above) the models with a decreased search space using 4 to at most 32 units.

We agree with the referee that it is beneficial to also publish training/validation results (losses, metrics, etc.), which will be added to the revised manuscript.

Checking the code repository and additional files provided by the authors, the '\tb' folder was not included so the TensorBoard logs cannot be checked.

On checking the code further, there are remainder classes and functions that were not part of the study such as the models that include 'CNN' layers. I would suggest a general cleanup, but the code definitely runs and I was able to generate one of the trials done for the study and the corresponding logs.

We agree with the referee that we should have published all logs of the hyperparameter tuning from the beginning. We are going to include all TensorBoard logs together with a cleaned-up version of the code.

Taking a look at some of the post-processing notebooks, I find the information presented in the `notebook_aux_peak_events.ipynb` to be informative and would encourage the authors to include some of the plots for peak events in section 4.2 in the manuscript. The colors for the ARIMA model need to be adjusted though, because they are the same as the "measured" data.

We decided to remove these plots from the manuscript due to structural reasons but kept them in the code files as we also find them quite informative. For the final submission, we will try to include them in section 4.2 as suggested, or at least add them to the supplement material.

Minor Comments

General: I suggest that the name of the HLSTM-PBHM should be flipped around to PBHM-HLSTM as the PBHM is the initial step in the pipeline.

We agree and will change this accordingly in the revised manuscript.

Line 72: The hindcast-forecast LSTM cannot be called "novel" as it is adapted from the approach of Nevo, et al. (2022), which in-itself is adapted from other sources.

We agree and will change this accordingly.

Line 146: How was the search-space for the hyperparameters of the ARIMA model defined?

The search space for ARIMA was defined based on a prior testing. However, in the final version of the manuscript, we will retrain the model with an updated search space. The search space will be determined to also align with the comments of referee 2. Specifically, we will determine the p and q orders based on the PACF and ACF functions. As for the differentiation order (d), really

only d=1 produced reasonable results. So this will be also addressed as it makes no sense to still include lower (0) or higher orders to the search space if this does not produce nearly as good results. For more details also see our comments to referee 2.

Line 171: Why was a loss function which combines two metrics chosen? Was minimizing NSE or simply MSE tried? From Appendix B the authors say that this was adapted from Nevo et al. (2022) but in that paper they minimize a negative log-likehood as they have a probabilistic model.

Unfortunately, the citation of Nevo et al. (2022) was misplaced in this context. We apologize for the inconveniences and will fix this in the revised submission.

The first versions of the models were trained using the MSE as a loss function. At some iteration, we switched to the presented loss functions.To us it made sense to use the same function for model training (loss function) and for the tuner objective. Nevertheless, throughout the rerun of our experiments on the smaller search space, we compared the differences between our employed loss function and a simple MSE more extensively. The results showed no added benefit of the employed loss function (presented in the manuscript) over a simple MSE. Interestingly, the MSE even produced slightly better results overall. The figure below shows the histogram of the scores of the 50 trials during the hyperparameter tuning. For both, the eLSTM and the PBHM-HLSTM, the model trained on the MSE shows a better score on average as well as at the best trial.

For the final submission, we will therefore choose the models trained using the MSE as the loss function.

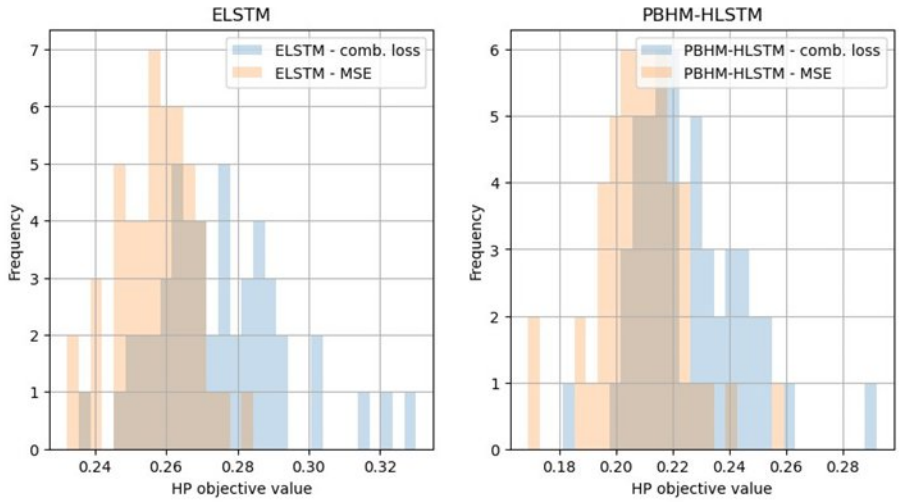


Figure 4: On the right-hand side of the figure I'm missing the $Q_{forecast}$ as in Figure 3. Further the "Targets" should be dropped from this figure or added to Figure 3 to have somewhat equivalent descriptions of the model architectures.

We will do a rearrangement of the graphic to include Qforecast for the final submission. Furthermore, we will remove the targets from Figure 3, as suggested by the referee.

Table 2: Although PBIAS is a good overall metric to include, these results would greatly benefit of including metrics which directly target low- and high-flows like FHV and FLV. See Gauch et

al. (2021) or directly Yilmaz et al. (2008). Some of the results described in the text using PBIAS are more suited to be described using these two metrics.

We agree that the suggested metrics FHV and FLV are more suited to argue about performances for base flow and peak flow conditions, respectively. For this reason we will include the FHV and FLV metrics. We will further investigate if it makes sense for the revised manuscript to replace the PBIAS or if it still has a benefit for interpreting the presented results.

Also, I find the reported PBIAS metrics for the ARIMA models to be strange given their KGE and NSE per year. From the code, in the `run_arima.py` I see it calls the `calculate_bias` function but only prints values to the screen while in `notebook_tab1-4_table_results.ipynb` the metrics are read directly from a file. I'm guessing that `metrics.txt` is generated using the data in the other files in that folder `metric_nse.txt`, etc. but I also don't see where in the code those files are dumped. `run_arima.py` dumps the `all_fc_df` as a pickle, but I'm not sure if that DataFrame is used to generate the metrics `.txt` files.

Yes, we were also surprised that the BIAS error of the ARIMA model was that low. However, upon further investigation, it showed that ARIMA produced many “perfect” forecasts throughout the year. This was especially true in baseflow conditions, which had a large positive impact on the PBIAS values. Noteworthy, this is also the reason why ARIMA in some instances has quite good KGE – KGE has a direct term for the BIAS - statistics, whilst the NSE is not that good. Upon inspecting the results using the FHV and FLV metrics, ARIMA’s forecasts came out as less favorable. So we agree with the referee, that the PBIAS might not be a good metric in this case to draw conclusions from, which will be addressed in the revised manuscript.

The referee is right. For the ARIMA model, the metrics are computed from the ‘all_fc_df’. We will change this accordingly such that the metrics are computed directly, which majorly improves the reproducibility of the results.

Table 3: Generalization refers to a models ability to predict in previously unseen data drawn from the same distribution as the one used to train the model. In doesn’t have to do with differences between validation and testing. I would consider correcting this table with differences between the testing and training sets, or not including it at all as most of the discussion cantered around these results appears or can be included in other sections.

We agree with the referee that the term generalization was misused in the original version of the manuscript. However, based on the referee’s prior suggestion of including a sensitivity analysis for evaluating the benefit of Qsim on the model results, this evaluation becomes obsolete and will be removed from the revised manuscript.

Fig. 6 and Fig. 7: The legend of both columns should not be shared. Currently it appears as if the “normalized win ratio” has something to do with the standard deviation of each model.

We agree with the referee and we will change the plots accordingly to avoid confusion.

References

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2021). Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network. Hydrology and Earth System Sciences, 25(4), 2045–2062. https://doi.org/10.5194/hess-25-2045-2021

Kratzert, F., Gauch, M., Klotz, D., & Nearing, G. (2024). HESS Opinions: Never train an LSTM on a single basin. Hydrology and Earth System Sciences Discussions, 1–19. https://doi.org/10.5194/hess-2023-275

Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S., & Klambauer, G. (2019). NeuralHydrology—Interpreting LSTMs in Hydrology. arXiv:1903.07903 [Physics, Stat], 11700, 347–362. https://doi.org/10.1007/978-3-030-28954-6_19

Nevo, S., Morin, E., Gerzi Rosenthal, A., Metzger, A., Barshai, C., Weitzner, D., Voloshin, D., Kratzert, F., Elidan, G., Dror, G., Begelman, G., Nearing, G., Shalev, G., Noga, H., Shavitt, I., Yuklea, L., Royz, M., Giladi, N., Peled Levi, N., … Matias, Y. (2022). Flood forecasting with machine learning models in an operational framework. Hydrology and Earth System Sciences, 26(15), 4013–4032. https://doi.org/10.5194/hess-26-4013-2022

Yilmaz, K. K., Gupta, H. V., & Wagener, T. (2008). A process-based diagnostic approach to model evaluation: Application to the NWS distributed hydrologic model. Water Resources Research, 44(9). https://doi.org/10.1029/2007WR006716