SPECIFIC COMMENTS

1. The authors did not provide any literature review in the Introduction on the use of machine learning models for ice jam prediction, despite being requested to do so by the second reviewer. The authors cite their review article that covers this topic (Madaeni et al., 2020) but do not provide any details on the machine learning methods that have been used for ice jam prediction, which seems essential to highlight in the present study.

I added in the Introduction:

"Although machine learning methods have been widely used in time series forecasting of hydro-meteorological data, they have been used less frequently in the prediction of ice jams (Graf et al., 2022). Semenova et al. (2020) used KNN to predict ice jams using hydro-meteorological variables such as precipitation, snow depth, water level, water discharge, and temperature. They developed their model with data collected from the confluence of Sukhona River and Yug River in Russia between 1960 and 2016 and achieved accuracy of 82%. Sarafanov et al. (2021) presented an ensemble-based model of machine learning methods and a physical snowmelt-runoff model to account for the advantages of physical models (interpretability) and machine learning models (low forecasting error). Their hybrid models proposed an automated approach for short-term flood forecasting in Lena River, Poland, using hydro-meteorological variables (e.g., maximum water level, mean daily water and air temperature, mean daily water discharge, relative humidity, snow depth, and ice thickness). They applied an automated machine learning approach based on the evolutionary algorithm to automatically identify machine learning models, tune hyperparameters, and combine stand-alone models into ensembles. Their model was validated on ten hydro gauges for two years, showing that the hybrid model is much more efficient than stand-alone models with a Nash–Sutcliffe efficiency coefficient of 0.8. Graf et al. (2022) developed an MLP and extreme gradient boosting model to predict ice jams with data from 1983 to 2013, in Warta River, Poland. They employed water and air temperatures, river flow, and water level as inputs to their models, showing that both machine learning methods provide promising results. In Canada, De Coste et al. (2021) developed a hybrid model including a number of machine learning models (e.g., KNN, SVM, random forest, and gradient boosting) for St. John River (New Brunswick). The most successful ensemble model combining 6 different member models was produced with a prediction accuracy of 86% over 11 years of record."

2. Section 2.2: it would be good to mention the software packages used for developing the machine learning models. Without this information, for example, it is difficult to know what is being referred to as 'default values' for the decision tree method.

I added "To develop machine learning models, Scikit-Learn machine learning libraries are used except for NumPy, Pandas, and Scikit-Learn preprocessing libraries."

3. L344-345: I think this is backwards, you do not need a loss function to evaluate model error - you need a prediction and a target. However, in many cases you need the model error to evaluate the loss function (e.g., if the loss function is mean square error or some regularized version of it). Perhaps it was meant that the loss function is used to guide the optimization problem?

I changes it to "Neural networks need a loss function to guide optimization problem resolution."

4. Author's reply to my former comment 8: If grid-search has 'poor coverage in dimension',

its not clear how trial and error overcomes this. How did the authors know which hyper-parameter values to try in trial-and-error (i.e., those reported in Table 3)? Were the values in Table 3 decided upon based on recommendations from the literature? If so, any sources that guided these decisions would be good to cite.

I put the exact values of the parameters that are recommended in the previous studies (added them toTable 3) in trial-and-error. But in grid-search, the model searches for values in grids that have a poor coverage.

Table 3. Common values and selected values for different parameters of the models.

Parameter	Common values	Selected value	Source
Mini-batch size	16, 32, 64	16	Bengio (2012); Devineau et al. (2018b); Masters and Luschi (2018)
Number of convolution filters	32, 64, 128	128	Brownlee (2017); Maggiori et al. (2017)
Filter size	3, 5, 7	(5,1) and (5,3)	Devineau et al. (2018b); Maggiori et al. (2017)
Number of LSTM units	32, 64, 128	128	Brownlee (2017); Karim et al. (2019b); Ordóñez and Roggen (2016)
Number of dense layer units	16, 32, 128, 256	32	Karim et al. (2019a); Livieris et al. (2020); Fawaz et al. (2019b)
Momentum in SGD	0.5, 0.99, 0.9	0.9	Brownlee (2018a)

The authors mention that various combinations of the hyper-parameters (L378) were applied but do not mention what combinations were explored. The authors should provide more information here to enable their experiments to be reproduced. That is, assuming someone had access to the same dataset, sufficient information should be provided by the authors to enable someone to arrive at the same (or at least similar) results.

There is some explanations of the choice of hyperparameters in the Appendix. However, I believe that it is not necessary to include all of the combinations in the paper. As I clearly mentioned the final successful hyperparameters and model structures, which result in promising results, enabling someone to reproduce the same results.

5. Supplemental information file:

a. It would be good for all acronyms (and abbreviations, if any) to be spelled out in full at first use.

Sure, I have done that.

b. It's not clear what is meant by 'channel'. Do the authors mean 'input'?

I replaced that with variables.

c. I think the authors mean 'estimating gradients' rather than 'applying gradients'?

Yes, replaced that.

d. It appears the word 'term' is missing after 'momentum' in the first paragraph of the last section.

Yes, I added that to the text.

e. The authors should appropriately revise ‘high momentums’. Perhaps ‘when using high values for the momentum term’ would be more appropriate.

Done.

6. The referencing format is inconsistent (see, e.g., L 593-594).

The referencing format of conference papers were not consistent with journal papers. I edited them.

7. Authors’ reply to my former comment 10: it’s not clear what is meant by ‘model implementations’ in this context. I suggest removing these words or using terms that better describe the technical matter.

I changed ‘model implementations’ to ‘model developments’.

8. Authors’ reply to my former comment 16:

a. Why not combine Table 11 and 12? It will make it easier for the reader to compare the performance between the deep learning and machine learning models.

I combined them and sort them based on their F1 score.

Table 11. Test F1 scores for the developed deep learning and machine learning models.

Models	F1 score
CNN-LSTM	0.92
CNN	0.80
LSTM	0.80
KNN	0.78
SVM	0.75
DT	0.71
MLP	0.70

b. It would be good for the authors to mention these benchmark machine learning methods in the abstract and include a sentence stating the relative improvement in performance achieved by the deep learning models (in comparison to the benchmarks).

I added to the abstract “We also employed machine learning methods including support vector machine (SVM), k-nearest neighbors classifier (KNN), decision tree, and multilayer perceptron (MLP) for this purpose.” And “The developed deep learning models achieved improvements in performance in comparison to the developed machine learning models.”

9. Authors’ rely to my former comment 17:

a. In the authors’ response A, perhaps ‘time consuming to train’ would be more appropriate than ‘time consuming’?

Done.

b. In the authors’ response B:

i. What characteristics or the model and/or data makes the models transferable to New

Brunswick and Eastern Ontario? Did you run the model on data from these provinces to verify this assertion? If so, this should be mentioned. If not, then the authors should be careful to use appropriate language. For example, the authors may instead mention that they anticipate the deep learning models developed in this research to perform well in these geographical zones for reasons X, Y, and Z.

I added “The developed models in this study can be used to predict future ice jams some days before the event not only for Quebec but can also be transferred to eastern parts of Ontario and western New Brunswick, since these areas have the similar hydro-meteorological conditions.”

ii. Please remove ‘pretty’ and ‘really’.

Done.

iii. In ‘correct predictions with the wrong’, replace ‘with’ with ‘for’.

Done.

References

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. Neural networks: Tricks of the trade, 437-478.

Brownlee, J. (2018a). Better deep learning: train faster, reduce overfitting, and make better predictions. Machine Learning Mastery.

De Coste, M., Li, Z., Pupek, D., & Sun, W. (2021). A hybrid ensemble modelling framework for the prediction of breakup ice jams on Northern Canadian Rivers. Cold Regions Science and Technology, 189, 103302.

Devineau, G., Xi, W., Moutarde, F., & Yang, J. (2018b). Convolutional neural networks for multivariate time series classification using both inter-and intra-channel parallel convolutions. Reconnaissance des Formes, Image, Apprentissage et Perception.

Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2019b). Deep learning for time series classification: a review. Data Mining and Knowledge Discovery, 33(4), 917-963.

Graf, R., Kolerski, T., & Zhu, S. (2022). Predicting Ice Phenomena in a River Using the Artificial Neural Network and Extreme Gradient Boosting. Resources, 11(2), 12.

Karim, F., Majumdar, S., & Darabi, H. (2019a). Insights into LSTM fully convolutional networks for time series classification. IEEE Access, 7, 67718-67725.

Karim, F., Majumdar, S., Darabi, H., & Harford, S. (2019b). Multivariate lstm-fcns for time series classification. Neural Networks, 116, 237-245.

Livieris, I. E., Pintelas, E., & Pintelas, P. (2020). A CNN–LSTM model for gold price time-series forecasting. Neural computing and applications, 32(23), 17351-17360.

Madaeni, F., Lhissou, R., Chokmani, K., Raymond, S., Gauthier, Y., 2020. Ice jam formation, breakup and prediction methods based on hydroclimatic data using artificial intelligence: A review. Cold Reg. Sci. Technol. 174, 103032. https://doi.org/https://doi.org/10.1016/j.coldregions.2020.103032.

Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2017). High-resolution aerial image labeling with convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing, 55(12), 7092-7103.

Masters, D., & Luschi, C. (2018). Revisiting small batch training for deep neural networks. arXiv preprint arXiv:1804.07612.

Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors, 16(1), 115.

Sarafanov, M., Borisova, Y., Maslyaev, M., Revin, I., Maximov, G., & Nikitin, N. O. (2021). Short-Term River Flood Forecasting Using Composite Models and Automated Machine Learning: The Case Study of Lena River. Water, 13(24), 3482.

Semenova, N., Sazonov, A., Krylenko, I., & Frolova, N. (2020). Use of classification algorithms for the ice jams forecasting problem. E3S Web of Conferences.