As in my previous review, I will mostly concentrate on the methodology of this manuscript, than on the data or context of this study in groundwater literature.

I think that the revised manuscript overall improved by a lot over the original submission in terms of clarity and structure. The authors included information on the hyperparameter search (and made the search space larger than before) and chose the best model based on independent validation data. Although I am surprised by the result of the hyperparameter search, I guess this is something we have to accept.
However, there is one point around the model architecture that I still do not see satisfiably answered (and largely ignored in the manuscript), see Comment 1 + Comment 2.

Comment 1:
In my last review (Reviewer Comment 3.1d), I questioned the use of the convolutional layer. As the layer is used currently, the most recent time steps are ignored for doing the gap filling. To be more concrete: Given an input sequence of length M with N consecutive time steps, which should be gap filled. The first of the N time steps is immediately following the last time step of M. In most autoregressive tasks, the immediate preceding time steps are the most important features, especially with time series of high temporal frequencies. However, due to the choice of the convolutional layer and the filter size, the first day of N is only predicted by the M-N+1 first time steps of M, ignoring probably the most important information. The ARIMA model however, does see these time steps (and performs much better than the LSTM). From the hyperparameter search as described in Section 3.1.1. It does not seem as if the convolution layer at all was optimized. And even if the authors decide to keep this architecture, I think this is a critical point to include into the paper and to explain their decision. I could imagine that people who see this (that the most recent days are ignored in an autoregressive task) will ask why. The answer of the authors in their rebuttal ("*Furthermore, the time steps immediately preceding the current time are not necessarily the most informative information in the presence of dynamical behavior*") might be true, but should definitely be tested as well as discussed in the manuscript.

Comment 2:
This is very related to the comment above: The authors argued in their answer to Reviewer Comment 3.1.d that the (one) reason for the convolutional layer is to map from a sequence with M time steps to a sequence of N time steps. I don't know how this slipped my eye in my previous review, but an important question is "Why do you even map to N?".
On Page 9 L1ff. you say you actually only map M to 1 and, then move M by one time step (integrating the last prediction into the shifted input sequence M) to predict the next time step and so one. So why is the LST-based model not trained to do exactly this? This setting is the most common LSTM setting (called sequence-to-one), and you would simply use the LSTM output at the last time step, to predict the next time step. During inference (= gap filling) you would do exactly what you do now: passing one sequence of M time steps through the model, get the prediction for the M+1 time step, shift M by one time step and include the previous prediction, pass the new sequence again to get the prediction for the M+2 time step, and so on.

The convolutional filter is also not, what makes you model account for spatial correlations (related to the answer of the authors to reviewer comment #1.4), since the LSTM can already account for those correlations. So the framing of the manuscript can remain unchanged.

Comment 3:
I can not follow the conclusion in L7 P 15ff, especially that "*ARIMA cap capture [...] but not changes that occur over a short time window (i.e., at higher frequencies)*". As the authors note themselves, ARIMA is better in every error statistic. It is argued that the LSTM does better at higher frequencies and it is pointed to Figure 9, the first two columns. The figures are small so it might be hard to see, but from what I can see, I don't see the LSTM being better in any well at any point during the entire period. The blue line, which shows the relative error, seems to be always worse for the LSTM, also during periods with higher variance. At this point, I can't see any evidence that backs the statement of the authors and I think, additionally to these plots, some quantification (using some metrics) are needed to support the statement that the LSTM has some advantage over the ARIMA model.

Comment 4:
P18 L11: "significantly" I agree that the improvement seems obvious, however, the use of significant should always be supported by the result of a significance test. Otherwise, maybe rephrase this sentence.

Comment 5:
Isn't it possible to train a multi-well ARIMA(X) model as well? This would be an interesting benchmark for the experiment in Section 4.3, since in the single site the ARIMA model showed superior performance. If the LSTM would be better in the multi-well setting, this would certainly be an interesting result.

Comment 6:
The two sentences in P19 L3ff seem to contradict each other. "*DNNs excel in dealing with high-frequency dynamics (daily and subdaily) or nonlinearities, although they require more training data and computational resources. The DNN approach also appeared to overestimate the high-frequency (daily and subdaily) fluctuations in some wells near the river (i.e., wells 1-1, 1-10A, and 2-2), which was likely caused by the variability in dynamics signatures among the training, validation and test periods.*". They "*excel*" but "*also appear[ed] to overestimate*".