Response to Reviewers for Seismica submission titled: "Seismology in the cloud for template matching and machine-learning earthquake detection"

Dear Yen Joe Tan and reviewers,

We appreciate your time in carefully reviewing our manuscript and providing constructive feedback. We have carefully made revisions addressing all comments and concerns and believe the manuscript is now much improved and ready to be re-considered for publication.

Below we directly address all comments from both Reviewer A and B one-by-one. Reviewer comments are copy and pasted and given in *red text, Italic, Arial font*, and our responses are in black Times New Roman.

Our edits to the majority of the text consist of sentence reorganization or rewording strictly to provide clarity or more strongly highlight main points, as suggested by both reviewers. The largest changes are in Sections 5 and 6, where we have followed the suggestions of Reviewer B: we removed one figure that was unnecessary, and shortened and focused the text to be more in line with the rest of the paper. We also believe the manuscript title should be modified to reflect this new refocusing of the text, from “Seismology in the cloud for template matching and machine-learning earthquake detection” to “Seismology in the cloud: guidance for the individual researcher”.

Reviewer A

1. CPU and GPU cost. I think the authors should separate the analysis of training and inference, so that the readers can have a better understanding if using CPU or GPU for different tasks.

We do not analyze the timing of training, only the timing of inference.
To make this clearer from the very beginning, we have edited the Abstract at Line 18 to specify that we apply pre-trained machine learning models. We also edit Line 24 to say that CPU-only processing “can be” faster than GPUs, not “is” faster.
We edited Line 107-8 to specify “pre-trained” machine learning models and on Line 171 added “we do not perform model training, but only model inference”.

2. Azure cloud. The authors only used Azure cloud in this work. It is better to explain why they chose Azure cloud and if the workflow can be easily transferred to other cloud platforms.

We had already stated in the manuscript that the workflow can be easily transferred to other cloud platforms, previously Line 217: “While we focus on Azure, we report that the framework is similar to other major cloud providers.”
We moved this statement to the Introduction to make it clearer earlier on, and edited it to explain why we choose to use Azure. Line 110-113 now reads:

"We use Microsoft Azure because of resources available to us through our home institution, but we report that the core framework is similar to other major cloud providers, provided that researchers adapt for provider-specific storage and compute systems."

To Lines 664-668, the last sentence of the Discussion, we have also added:
"It would also be possible for researchers to emulate this workflow on other cloud platforms such as Amazon Web Services or Google Cloud Platform; the codes to interact with the cloud resources (Azure Blob storage, Azure Batch) would need to be adapted to the interface specific to the cloud provider, but no significant changes would need to be made to the code base itself."

3. Speedup. One advantage of using cloud computing is that the workflow can be easily scaled up. It is better to add a plot to show the speedup of using different numbers of CPUs and GPUs. Are there any overheads when using more CPUs and GPUs?

We already have a plot showing the speedup of using different numbers of CPUs and GPUs: Figure 4a and 4c. This same Figure also shows the costs associated with the different numbers of CPUs and GPUs.

As for overhead time associated with spinning up more CPUs or GPUs, we have added a sentence in Section 4 to address this. Lines 412-413 now read:
"Since start-up processes are run in parallel across nodes on the Pool, overhead start-up times do not tend to increase with number of CPUs or GPUs."

4. The authors mentioned the learning curve of using cloud computing. Compared with other cloud workflows of earthquake detection, what is the advantage of this workflow?

In our view, the main advantage of our report is that we do not use advanced distributors such as Kubernetes (as mentioned on Line 98) and instead focus on building a beginner-level cloud workflow from scratch rather than presenting something for others to use that is already in the cloud, and therefore a bit of a black box. That is what we believe is missing in the literature currently: we focus instead on migrating over code you already have.

To make this more clear, we have edited the penultimate paragraph of the introduction, namely the first sentence (Lines 102-104), which now reads:
"This report is distinct from other published seismological cloud-based workflows (Zhu et al., 2023) in that we aim to help the average seismological researcher build their own cloud computing version of their local processing algorithms from the ground up."

We also think that changing the title of the manuscript to reflect its refocusing (newly "Seismology in the cloud: guidance for the individual researcher") helps to make this report distinct.

5. Based on my understanding, the approach proposed by this work is very similar to setting up a slurm cluster on cloud. If so, it is better to explain the difference between running on a slurm cluster locally and running on cloud.

To address this, we have added a short paragraph to the end of Section 3.2 (Lines 374-380) where Pool-based parallelization is explained. It reads:
"We choose to use Batch services rather than a workload manager such as SLURM (Yoo et al., 2003), as is typically used in high performance computing (HPC), for several reasons. To use such a system on the cloud would require running a persistent machine to handle orchestrations, which would add cost and complexity. With Batch services, orchestration is instead done by the cloud provider at the time of job submission. Batch services also tend to be focused on independent containerized workflows whereas HPC scheduling systems are designed for non-containerized workflows on closely networked hardware."

Further, to make the distinction between running locally and running on the cloud more clear, we have edited the penultimate paragraph of the introduction (Lines 102-117). We now first introduce that we build the workflow to run in parallel locally, and then separately introduce that we migrate it to the cloud by building a group of cloud resources.

Reviewer B

General suggestion:

While I found the first part of the manuscript well written and with a clear goal, I thought that the presentation of the detection results was a bit confusing and perhaps not necessary since the example shown here is not meant to deliver a comprehensive earthquake catalog. The conclusions do not even mention anything from Section 5 (and most of Section 6). I suggest shortening Section 5 and shifting the focus from an unfair comparison with a much more thorough previous study to the basic description of the detected events and how the workflow can be applied to produce a detailed catalog. After addressing these suggestions and the comments that follow, I think the paper will be ready for publication.

In our responses to the following suggestions, we address the shortening of Sections 5 and 6 and shifting the focus of these sections to more basic results.

1. Lines 407-414: The authors should emphasize that the conclusions about GPUs' cost efficiency are application dependent. For example, a large-scale template matching application with tens or hundreds of thousands of templates and a software like Fast Matched Filter (FMF) will greatly benefit from GPUs. From personal experience, my template matching applications on a single A100 GPU are faster than the 256 AMD EPYC 7742 CPUs that are at my disposal. Then, the cost-advantage might be different. Unless I am unaware of a "recent" update in EQcorrscan, I think that EQcorrscan doesn't run on GPUs.

On Lines 459-462, we have added a sentence that emphasizes how our GPU vs. CPU comparison is application dependent:

"It should also be noted that these results are application dependent: we do not attempt to parallelize the TM branch using GPUs with the capabilities of the Fast Matched Filter method (Beaucé et al., 2017) in order to avoid adding CUDA dependencies to the containers."

We also point out that in paragraph 3 of Discussion Section 6 (now starting on Line 603), we had already specifically referred to the benefit of CPU processing only in the implementation of pre-trained ML models, not template matching.

2. The manuscript could benefit from an additional paragraph with the authors' viewpoint on why cloud-computing is an interesting tool for the scientific community: is it only for computational performances? Cost efficiency (e.g., could researchers in small universities who don't have access to a supercomputer afford working on the cloud)? For now, this kind of information is spread across the manuscript whereas it's quite important to the readers who are considering and weighing the pros and cons of using cloud resources. In some countries, the fact that cloud computing means increasing the carbon footprint of scientific computing is seen as a major drawback, so being more explicit about the pros in a dedicated paragraph could help enlighten the debate.

This is a great suggestion. To drive these points home and focus them more strongly in one place, we reorganized the Discussion section: the first paragraph (Lines 588-595) now points out the strongest pros with cloud computing (accessibility through low costs), and the second paragraph (Lines 596-602) points out the strongest cons (unintuitive and hard to use). We have also edited the last few sentences of the conclusion (Lines 678-682) to be clearer in this same way.

3. In its current shape, Sections 5+6 is neither an in-depth comparison of template matching vs EQ- Transformer nor a simple presentation of the results obtained with the proposed workflow. On one hand, a realistic workflow would use more templates, including EQTransformer's detections, and on the other hand, it is difficult to compare template matching with only 53 templates with any other detection method. For these reasons, Sections 5+6 don't fit well in the paper.

To address this fair point, we have shortened Section 5 to remove Figure 6 and the last two paragraphs discussing it. We agree that this section compared the two methods in a way that was not necessary nor well representative of the two methods.

We have also shortened Section 6 by moving the part of it that discussed our example earthquake results to the last paragraph of Section 5. We have simplified this portion of the text by removing unnecessary comparisons between ML and TM performance, and instead edited to suggest how they can be used together to create an earthquake catalog, found in the paragraph starting at Line 510.

4. Lines 444-446: What is a summed CC of 3.2 in terms of root-mean-square (or median-absolute- deviation) of the CC time series? It is odd to use template matching to detect small events that human analysts typically don't identify but then define the threshold such that all events are identifiable by humans. Giving the threshold in terms of RMS will make it easier to compare to other template matching studies.

We have edited this Line (now Line 491) to also give the threshold in terms of median-absolute-deviation. The sentence now reads:

"We present the TM detections that have a summed absolute cross correlation across the eight template channels greater than 3.2 (Figure 5), equivalent to a median absolute deviation threshold near 8."

5. Lines 459-464: I find that this sentence needs more explanation about why template matching detects such a small number of events in common with the Krauss 2023 catalog. Is it that the reference catalog performs much poorer than template matching? Is it 2% of the TM catalog that was matched with the Krauss 2023 catalog or the other way around?

This sentence and following sentences (now Lines 519-524) have been rewritten for clarity. Only 2% of the detections found by the TM branch are also found in the Krauss 2023 catalog.

The section previously read:

"The TM method has a hugely lower overall rate of picks in common with the Krauss et al. (2023a) catalog, 2% in comparison to 62% for the ML catalog. This is consistent with TM only being able to detect earthquakes with similar waveforms to the subset of templates used, while EQT is likely able to detect a more diverse set of waveforms."

The section now reads:

"The picks found by the TM method are mostly "new" picks: only 2% of the TM picks were also in the Krauss et al. (2023a) catalog. This suggests that the picks found by TM are almost entirely just smaller or noisier examples of the template events that went undetected by traditional methods. In contrast, 62% of the picks found by the ML branch are also in the Krauss et al. (2023a) catalog. Therefore, although EQT finds overall less earthquakes than TM, it likely captures a more complete representation of waveform diversity in the dataset."

6. Lines 500-502: What about EQTransformer's performances on your data? It's likely to be part of why you observe such a large difference in the number of detected events.

We have removed this part of the text which, we agree with the reviewer, makes unnecessary claims about the relative performance of the methods, especially considering we do not intend for our example results to be taken as a thorough, iterative application of them both.

7. Lines 504-506: This is an unsupported statement. First, you have just shown an example where template matching outperforms ML. Second, "may easily outperform", outperform what? Comparing ML and template matching is weird because they are complementary.

Same as the response to suggestion 6 above (this text was removed). The section now emphasizes instead how ML and TM are complementary.

8. Overall, Sections 5 and 6 don't really deliver anything new about template matching or ML techniques and distract from the main focus which is cloud-computing. I think that *simply presenting the results obtained with template matching and ML (perhaps with maps) and then explaining how one could adopt this cloud-based workflow to build an entire catalog would be more in the continuity of the first part of the manuscript.*

To address this point, we have significantly shortened Section 5. We removed Figure 6 and the two paragraphs discussing its result; this figure and section are indeed superfluous to our main result of how machine learning and template matching are complementary methods.

We have also changed the sentence in the Abstract that previously made an interpretation of our catalog results; it now only reports that we simply apply the method and demonstrate how it works.

The sentence previously read:

"When the workflow is applied to one year of continuous data from a mid-ocean ridge, the resulting earthquake catalogs suggest that template-matching and machine learning are complementary methods whose relative performance is dependent on site-specific tectonic characteristics."

It now reads (Lines 20-23):

"We apply the cloud-based workflow to one year of continuous data from a mid-ocean ridge to demonstrate the construction of two earthquake catalogs, one through template-matching and one with a pre-trained machine learning model."