Reviewers' comments:

Reviewer #1 (Remarks to the Author):

The major claims of this paper are:
1. They use machine learning to accurately predict a chaotic system from noisy data.
2. Their approach is interpretable.
3. Their approach does not require a model of the system or knowledge of the nature of the dynamics.

Novelty/impact:
To my knowledge, this is the first use of machine learning to predict this particular physical system. It may also be the first use of QRNNs for a physics application. The interpretability of QRNNs might appeal to the community. However, there is previous work on using other RNNs to other chaotic physical systems, and that work also does not require a system model or knowledge of the nature of the system [8-11]. The application of QRNNs to this dataset seems fairly straightforward, but interpreting the hidden neurons required creativity.

This paper is primarily about a machine learning method for predicting chaotic dynamics, but you don't mention that there is previous work on machine learning to predict other chaotic systems. You do cite several examples [8-11], but you only mention that they predict physics, not that they are also chaotic systems. It seems that your only comment about how you differ is that you add interpretability.

In my opinion, this paper would be much stronger if there was more comparison with previous work. I frequently use deep learning to predict time series data from physical systems (sometimes chaotic). My current first thing to try would be an LSTM. Can you strengthen an argument that I should start trying QRNNs on my data? It would help to see how an LSTM would perform on your dataset, i.e. in terms of accuracy and training time. How about other methods from previous work? I know that the original QRNN paper [12] found a speedup on their datasets, but they were all language model examples. Would the opportunities for parallelism help a lot on examples like yours?

I think that your interpretability analysis is interesting, but I don't think the findings are strong enough to convince me to use QRNNs. The pattern of the hidden units in the original QRNN [12] paper is stronger, and the interpretation in that paper is more intuitive to me.

Validity:

It is critical in machine learning that there is held-out "test" dataset that you do not use until you are finished choosing the final trained model. This allows you to estimate a generalization error by considering totally unseen data. For deep learning, this is commonly handled by having training, validation, and test sets. The training set is used in the training (optimization) of a model, and the validation data is used to choose between models, decide when to stop training, etc. I'm concerned that you only mention splitting your data into training & evaluation datasets, and that you plot both of their losses over time (Fig. S3). ("This measured data is separated into two data sets, the first of which is used for training, and the second for evaluation (Fig. 2b)") In fact, it looks like you are overfitting between the training & evaluation datasets. If you used the evaluation dataset to make decisions, your error might be much higher on a truly held-out dataset due to two levels of overfitting. I don't think this paper should be published without results on a truly held-out dataset.

More minor comments:

1. “we demonstrate the efficacy of machine learning in predicting chaotic behavior in complex nonlinear mechanical systems” I think this should be softened to “a complex nonlinear mechanical system”.
2. “has the potential to be used for complex scenarios, such as the nonlinear dynamics of thin-walled structures and biological membrane systems” Could you elaborate on the connection here?
3. “QRNNs allow access to hidden layers, offering the ability to scope the data set and assign “meaning” to each neuron.” This sentence is unclear to me. Can’t you “access” the hidden layers for any neural network? What does it mean to “scope the data set”? And I think “assign “meaning” to each neuron” is overstating the case. The QRNN paper simply says “the model’s hidden states are more interpretable,” [12] and in your paper, you’re only able to interpret two neurons.
4. Similarly, on page 3, “This approach has great potential for predicting complex dynamics of structures by allowing access to evolutionary steps and providing more parameters to reach reliable decisions.” What does it mean to allow access to the evolutionary steps? What do you mean by “providing more parameters to reach reliable decisions”?
5. “Due to the element-wise calculation in the pooling function, activation of each neuron does not depend on the past outputs of other neurons.” This isn’t right. See Equation 2 in your supplement.
6. “Since chaos is generally defined for deterministic systems as a high sensitivity to initial conditions” is too simplistic. Both [1] & [2] (which you cited in the previous sentence) require more than sensitivity to initial conditions to have chaos.
7. “we explore the feasibility of extracting meaningful system information from our data-driven approach” What is the system information that you extract?
8. On page 4, the delta symbol is used without being defined.
9. Could you give some context for why you’re plotting the Poincare map in Figure 2d?
10. On page 5, I think “This manifests the capability of our TCO system to form chaotic dynamics” needs some explanation. Why does this mean chaotic dynamics?
11. On page 5, I think “Based on the experimentally measured data, we predict whether a response is chaotic or periodic by employing the QRNN technique” is a confusing way to introduce your methodology. This sounds like a classification problem (chaotic vs. periodic), but your paper is primarily about predicting all 12 variables for the next 32 time steps.
12. On page 6, you talk about using the first 5600 time steps for training data. Each example is an input of 128 time steps and a prediction of the next 32 time steps (so intervals of 160). Then on page 11, you say “Training was carried out on minibatches of 50 data sets by using the Adam optimization algorithm [25] with learning rate of 0.001 and decay rate of 0.8. To create each data set, we randomly chose excitation frequency out of 21 different frequencies (fex = 5 Hz ~ 25 Hz), and from the selected frequency data with 5600 time steps, we randomly selected input data with 128 time steps and next 32 times step data to calculate error of prediction.” Using “data set” in this context is unusual and a little confusing. “Examples” would be better. So for each example, you choose a random frequency and a random interval of 160. You choose 50 examples as a minibatch. Do these intervals overlap? Does each epoch cover all 35 intervals of length 160 in 5600 time steps?
13. For Figure 3, it’s unclear to me what you mean by “Predictions from the QRNN (denoted by red color) are compared with the evaluation data from the experiments,” especially since the x-axis has been renormalized. Before, you used “evaluation data” to mean the last 10 seconds of each time series, in contrast with “training data” referring to the first 23.3 seconds. However, the ratios of grey & red are not right for that meaning. Then I thought maybe the grey was the input 128 steps and that the model was applied recursively some number of times. In any case, this should be documented.
14. On page 6, “the QRNN exhibits quantitatively accurate prediction through approximately 30 excitation cycles, and then the deviation begins growing” doesn’t match the results in Figure 3, especially for the 12 Hz case.
15. On page 8, what does “thick red/blue colors” mean? Probably you mean “dark” instead of “thick”?
16. On page 8, describing Figure 4, “we extract the neuron activation of those two units and calculate the average activation value over the time length of the input data (i.e., 0.53 s)”. Is this just the first 0.53 s of the 33.3 s of experimental data? If so, does this pattern continue over the

later data?
17. On the top of page 9, I think the text should be edited to acknowledge that these patterns are just mostly true. (The 14 Hz case doesn't follow the right pattern for either hidden unit 71 or 313.)
18. For Figure S5, "Average neuron activation of the hidden units in the final third hidden layer over 10 training runs", I'm guessing that you didn't average the same neurons across the 10 training runs? I'm guessing that you're showing a different set of 6 neurons for each training run? I think you might be addressing that by using "from each" here: "We selected six different hidden units from each trained QRNN to create panels (b-g). The difference between panels b and c (or d and e, or f and g) is the different hidden units.", but it's still a little confusing to me. Did you pick six interesting neurons from each run (based on averaging over the activations for the 128 time steps in that run) and then average again?
19. In the Figure S1 caption, "However this operation depends only on c(t_i−1) from the previous time step" is confusing. This is a recurrent relation, so it depends on the previous time steps as well. Similarly, on Page 8 of the Supplement, "Please note that unlike standard RNNs, the calculation of the hidden state (c_t) of the QRNN depends only on the hidden state at previous time step (c_t−1)" should be clarified.
20. In the Figure S1 caption, I don't know what you mean by "Internal calculations of a pooling part consist of only element-wise multiplications, which allows both prediction (Lower left insets show phase planes for periodic and chaotic cases. Red dots represent Poincare map) and visualization of hidden states readily without additional layers or components in the system (Right)."
21. I don't see a code availability statement.

Repeating some citations from your bibliography for reference:
[1] S. H. Strogatz, Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering (Westview Press, 2015), second edition ed.
[2] S. Wiggins, Introduction to applied nonlinear dynamical systems and chaos, no. 2 in Texts in applied mathematics (Springer, 2003), 2nd ed.
[8] J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, Chaos 27, 121102 (2017).
[9] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Physical Review Letters 120, 24102 (2018).
[10] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, Proceedings of the Royal Society A: Mathematical, Physical & Engineering Sciences 474, 20170844 (2018), 1802.07486.
[11] K. Yeo and I. Melnyk, Journal of Computational Physics 376, 1212 (2019).
[12] J. Bradbury, S. Merity, C. Xiong, and R. Socher, in International Conference on Learning Representations (ICLR 2017) (2017), pp. 1–11, 1611.01576.

Reviewer #2 (Remarks to the Author):

In this work, the authors investigate a chaotic mechanical process (origami folding) using neural networks. In particular, they employ quasi-recurrent NNs to predict the periodic and chaotic evolution of the system in time. Overall, I am quite impressed by this work for a number of reasons.

The authors train their algorithm on the first part of the time series and test on the second part, which is much more difficult than what most authors do by simply randomly sampling from the time series. The authors are thus extrapolating, which is much more difficult than the typical interpolation.

Further, the authors make efforts to interpret their findings by analyzing the hidden layers of the NN. I believe that applying NN to physical systems where we have some intuition is a great way for us to learn more about the NN itself. So I am happy to see the authors work for understanding in addition to performance.

The figures are very clear and illuminating and the paper is well written.

I recommend publication, although I have a couple of minor suggestions for the authors:

* I encourage the authors to make their code available as open source, for example on GitHub. This will help other researchers build on these methods and carefully check the model.

* The references are pretty scarce. There are some good references by Mahadevan and others.

Response to Reviewers

Manuscript: Data-driven prediction and analysis of chaotic origami dynamics

First Reviewer (Reviewer's comments in blue font):

The major claims of this paper are: 1. They use machine learning to accurately predict a chaotic system from noisy data. 2. Their approach is interpretable. 3. Their approach does not require a model of the system or knowledge of the nature of the dynamics.

Novelty/impact:

To my knowledge, this is the first use of machine learning to predict this particular physical system. It may also be the first use of QRNNs for a physics application. The interpretability of QRNNs might appeal to the community. However, there is previous work on using other RNNs to other chaotic physical systems, and that work also does not require a system model or knowledge of the nature of the system [8-11]. The application of QRNNs to this dataset seems fairly straightforward, but interpreting the hidden neurons required creativity.

We thank the Reviewer for acknowledging the novelty and impact of the presented manuscript. We also want to note that the Reviewer's insightful review and detailed comments helped us improve the quality of the manuscript significantly. We have revised our manuscript accordingly, and detailed responses to each comment are described below.

This paper is primarily about a machine learning method for predicting chaotic dynamics, but you don't mention that there is previous work on machine learning to predict other chaotic systems. You do cite several examples [8-11], but you only mention that they predict physics, not that they are also chaotic systems. It seems that your only comment about how you differ is that you add interpretability.

To properly acknowledge the previous work on machine learning to predict chaotic dynamics, we have added the following sentences:

(Introduction of the manuscript)

> "Due to such capabilities, RNNs or their variations have been recently employed for physics problems [8, 9]. In particular, these neural networks with recurrent connections have shown successful attempts to predict the dynamics of chaotic systems [10-15]. "

In my opinion, this paper would be much stronger if there was more comparison with previous work. I frequently use deep learning to predict time series data from physical systems (sometimes chaotic). My current first thing to try would be an LSTM. Can you strengthen an argument that I should start trying QRNNs on my data? It would help to see how an LSTM would perform on your dataset, i.e. in terms of accuracy and training time. How about other methods from previous work? I know that the original QRNN paper [12] found a speedup on their datasets, but they were all language model examples. Would the opportunities for parallelism help a lot on examples like yours?

The reviewer raises an excellent point. The authors agree that the ultimate target should be to robustly compare the performance of the QRNN with other ML algorithms. The main focus of this article, however, is to (i) introduce the idea of applying QRNN to complex dynamics of a complex mechanical system and (ii) to demonstrate feasibility. We feel that this is worthy of its own paper for several reasons. First, the parallelism due to convolutional layers in the QRNNs definitely helps to enhance the training and test time as demonstrated in the original QRNN paper [16], which can help us to train a QRNN system for various data including chaotic and periodic data, instead of training the neural networks for a specific case (i.e., to predict other cases, we need to train the neural networks for each corresponding data). Second, in this process, unlike the conventional RNNs, QRNN allows easy interpretation and visualization of the hidden layers, which we discovered hint at whether the mechanical system exhibits chaotic behavior or not. We believe that this unique feature of QRNN and its implication to physical system analysis is at the heart of the research reported in this manuscript. In this respect, our QRNN-based approach has potential to be a self-contained tool, which can offer the prediction and analysis capabilities in a relatively straightforward manner. Ultimately, the authors plan to pursue studies of the performance in future studies, and also hope that this article will motivate others to do the same. Such a study is not included herein as it could significantly lengthen the article and potentially distract from the intended scope.

I think that your interpretability analysis is interesting, but I don't think the findings are strong enough to convince me to use QRNNs. The pattern of the hidden units in the original QRNN [12] paper is stronger, and the interpretation in that paper is more intuitive to me.

In the original QRNN paper [16], they analyzed the hidden states for the sentiment classification (label either "positive" or "negative"). On the other hand, we apply our QRNN-based approach to a mechanical system with multiple degree-of-freedoms (DOFs) and we predict those variables as well as analyze the hidden state at the same time. Therefore, it is very difficult to compare their analysis results with our findings. The application of the QRNN to mechanical systems itself has been relatively unexplored, and the interesting finding we report in this manuscript is that the QRNNs has great potential of offering such interpretable hidden state responses for mechanical systems.

Validity:

It is critical in machine learning that there is held-out "test" dataset that you do not use until you are finished choosing the final trained model. This allows you to estimate a generalization error by considering totally unseen data. For deep learning, this is commonly handled by having training, validation, and test sets. The training set is used in the training (optimization) of a model, and the validation data is used to choose between models, decide when to stop training, etc. I'm concerned that you only mention splitting your data into training & evaluation datasets, and that you plot both of their losses over time (Fig. S3). ("This measured data is separated into two data sets, the first of which is used for training, and the second for evaluation (Fig. 2b)") In fact, it looks like you are overfitting between the training & evaluation datasets. If you used the evaluation dataset to make decisions, your error might be much higher on a truly held-out dataset due to two levels of overfitting. I don't think this paper should be published without results on a truly held-out dataset.

We refer to "evaluation data sets" as "test data" (or holdout data set), therefore, the evaluation data sets have never been used in training. Also, we trained the model for fixed number of epochs (100 epochs) with zoneout ($p = 0.2$), instead of early stopping, by following the original QRNN paper (Ref.[16] in the updated manuscript). We selected zoneout to avoid the overfitting. Therefore, we didn't use "evaluation data sets" to decide when to stop training. The terminology has been modified to make this clearer.

We also want to note in passing that the Second Reviewer mentioned that "The authors train their algorithm on the first part of the time series and test on the second part, which is much more difficult than what most authors do by simply randomly sampling from the time series. The authors are thus extrapolating, which is much more difficult than the typical interpolation." We believe this comment supports the validity of the testing approach that we have taken.

More minor comments:

1. "we demonstrate the efficacy of machine learning in predicting chaotic behavior in complex nonlinear mechanical systems" I think this should be softened to "a complex nonlinear mechanical system".

Thank you for the comment. We have updated the manuscript accordingly.

(Abstract in the updated manuscript)

> "Here, we demonstrate the efficacy of machine learning in predicting chaotic behavior in a complex nonlinear mechanical system. "

2. "has the potential to be used for complex scenarios, such as the nonlinear dynamics of thin-walled structures and biological membrane systems" Could you elaborate on the connection here?

In this study, we employed the origami-based structure as an exemplary system to examine the feasibility of our QRNN approach to predict complex dynamics of a mechanical structure. Given the planar nature of the origami components, we envision the potential use of our data-driven approach for analyzing a wide spectrum of general engineering structures with nonlinear behavior, such as unstable oscillations of an aircraft wing and buckled laminated plates. Also, complex dynamics has been studied in biology, and understanding such nonlinear dynamics such as tissue organization is important to predict its behavior. We expect the lessons learned in this study can be applied to such biological systems as well. We have updated the introduction of the manuscript to elaborate this point as follows:

(Page 3 in the updated manuscript)

> "Given the general nature of the origami system and the QRNN method, this approach has great potential for predicting complex dynamics of various engineered and biological structures (e.g., an aircraft wing vibration, a buckled plate with chaotic snap-through behavior [30], flight trajectory prediction and safety assessment [31], and tissue organization [32]) "

3. "QRNNs allow access to hidden layers, offering the ability to scope the data set and assign "meaning" to each neuron." This sentence is unclear to me. Can't you "access" the hidden layers for any neural network? What does it mean to "scope the data set"? And I think "assign "meaning" to each neuron" is overstating the case. The QRNN paper simply says "the model's hidden states are more interpretable," [12] and in your paper, you're only able to interpret two neurons.

We have revised the sentence by focusing on the ease of visualizing the hidden states and the interpretable feature as follows:

(Page 2 in the updated manuscript)

> "Most notably, the hidden states of a QRNN can be readily visualized and interpreted without additional processing (e.g., introducing self-attention to visualize how the input data is processed [17])"

4. Similarly, on page 3, "This approach has great potential for predicting complex dynamics of structures by allowing access to evolutionary steps and providing more parameters to reach reliable decisions." What does it mean to allow

access to the evolutionary steps? What do you mean by "providing more parameters to reach reliable decisions"?

Since we can readily visualize the hidden states for each time step corresponding to the input data at each time step, we can monitor how QRNNs process that input data (e.g., see Fig. 4c), and we can extract system information, e.g., a parameter indicating whether the system shows chaotic or periodic behavior, which could be useful information for engineering applications such as structural health monitoring. We clarified this in the updated manuscript.

(Page 10 in the updated manuscript)

> "Also, this approach has potential to provide the reason why the QRNN produces chaotic data, and specifically what parts of the input data lead to different responses of the QRNN by monitoring the hidden units' activation."

5. "Due to the element-wise calculation in the pooling function, activation of each neuron does not depend on the past outputs of other neurons." This isn't right. See Equation 2 in your supplement.

If the *fo*-pooling (Eq.(2) in the Supplemental Information) is used for the pooling layers of the QRNN, the hidden state of each hidden unit is calculated by using that of its previous time step value in each pooling layer, instead of using activation values from multiple different hidden units. This prevents the direct interactions among different hidden units as discussed in the original QRNN paper (Ref.[16]). Therefore, this element-wise calculation can enable to visualize hidden states of each pooling layer in a straightforward way.

6. "Since chaos is generally defined for deterministic systems as a high sensitivity to initial conditions" is too simplistic. Both [1] & [2] (which you cited in

the previous sentence) require more than sensitivity to initial conditions to have chaos.

Thank you for this comment. From Ref.[1], chaos is defined as "aperiodic long-term behavior in a deterministic system that exhibits sensitive dependence on initial conditions." We have added "aperiodic behavior" in the sentence as follows:

(Page 2 in the updated manuscript)

> "chaos is generally defined as aperiodic dynamical behavior of deterministic systems that exhibits a high sensitivity to initial conditions [1],"

7. "we explore the feasibility of extracting meaningful system information from our data-driven approach" What is the system information that you extract?

In addition to the prediction of periodic/chaotic folding motion of our origami structure, we examine the feasibility of extracting the parameters which characterize periodic/chaotic behavior by using the QRNN. In this study, we consider what system information can indicate whether a system exhibit periodic or chaotic motion. In general, the Lyapunov exponent is used, however, we analyze the hidden states of the QRNN and we find that the QRNN exhibits different activation pattern of its hidden units for periodic and chaotic data sets. Therefore, we show the feasibility of extracting system information from the QRNN, which can be a similar indicator as the Lyapunov exponent. We revised our manuscript accordingly as below.

(Page 3 in the updated manuscript)

> "Based on the experimentally measured time-series data, we explore the feasibility of extracting meaningful system information, e.g., whether the dynamic response of our origami system is chaotic or not, from our data-driven approach.

”

(Page 9 in the updated manuscript)

> ”We observe that the QRNN allows not only the prediction of origami folding behavior, but also the classification of chaotic/periodic input data of the TCO systems by simply monitoring the neuron activation of the hidden units in the final hidden layer. We show that this neuron activation pattern can successfully mimic the Lyapunov exponent, which is a conventional method to characterize chaotic behavior.”

8. On page 4, the delta symbol is used without being defined.

Thank you for the comment. We have updated the manuscript accordingly.

(Abstract in the updated manuscript)

> “at the initial unstretched state, i.e., no axial displacement $\delta = 0$ (see the inset illustration in Fig. 1d)”

9. Could you give some context for why you’re plotting the Poincare map in Figure 2d?

We show the Poincaré map to qualitatively display the chaotic folding motion of our origami system at the excitation frequency of 12 Hz. We have updated the manuscript by adding the following explanation:

(Page 5 in the updated manuscript)

> “Note that in the Poincaré map, periodic oscillations will show a single fixed point, or N points for a N-periodic response, whereas chaotic behavior will lead to a large collection of points.”

10. On page 5, I think "This manifests the capability of our TCO system to form chaotic dynamics" needs some explanation. Why does this mean chaotic dynamics?

Thank you for the comment. In addition to the explanation on the Poincaré map, we have updated the manuscript by adding the following sentence to guide the readers to the detailed analysis on the chaotic behavior of the TCO:

(Page 5 in the updated manuscript)

> "This manifests the capability of our TCO system to form chaotic dynamics (further analysis on this chaotic behavior, specifically Lyapunov exponent calculation, to be explained later)."

11. On page 5, I think "Based on the experimentally measured data, we predict whether a response is chaotic or periodic by employing the QRNN technique" is a confusing way to introduce your methodology. This sounds like a classification problem (chaotic vs. periodic), but your paper is primarily about predicting all 12 variables for the next 32 time steps.

We thank the Reviewer for pointing out this. To clarify this point, we have modified the sentence as follows:

(Page 6 in the updated manuscript)

> "Based on the experimentally measured data, we study a data-driven approach to predict multi-DOF folding motion of our origami structure by employing the QRNN technique"

12. On page 6, you talk about using the first 5600 time steps for training data. Each example is an input of 128 time steps and a prediction of the next 32 time

steps (so intervals of 160). Then on page 11, you say "Training was carried out on minibatches of 50 data sets by using the Adam optimization algorithm [25] with learning rate of 0.001 and decay rate of 0.8. To create each data set, we randomly chose excitation frequency out of 21 different frequencies (fex = 5 Hz $\sim$ 25 Hz), and from the selected frequency data with 5600 time steps, we randomly selected input data with 128 time steps and next 32 times step data to calculate error of prediction." Using "data set" in this context is unusual and a little confusing. "Examples" would be better. So for each example, you choose a random frequency and a random interval of 160. You choose 50 examples as a minibatch. Do these intervals overlap? Does each epoch cover all 35 intervals of length 160 in 5600 time steps?

Thank you for the comment. We have updated the main manuscript by using "examples" instead of "data set" as follows:

(Page 11 in the updated manuscript)

> "Training was carried out on minibatches of 50 examples by using the Adam optimization algorithm"

For the 50 examples which constitute a minibatch, there can be overlaps among different examples because we randomly chose an interval of 160 time steps from the training data composed of 5600 time steps. Since there are 21 frequency cases, the total number of possible examples are $(5600 - 159) \times 21 = 114261$. The batch size is 50, therefore, each epoch consists of $114261/50 \approx 2285$ iterations, and one epoch can cover the entire training data.

13. For Figure 3, it's unclear to me what you mean by "Predictions from the QRNN (denoted by red color) are compared with the evaluation data from the experiments," especially since the x-axis has been renormalized. Before, you

used "evaluation data" to mean the last 10 seconds of each time series, in contrast with "training data" referring to the first 23.3 seconds. However, the ratios of grey & red are not right for that meaning. Then I thought maybe the grey was the input 128 steps and that the model was applied recursively some number of times. In any case, this should be documented.

In our approach, we feed 128 time steps to the QRNN and then it predicts the next 32 time steps. Then, by using the predicted 32 time steps and the last 96 time steps from the previous input data of 128 time steps, we predict another 32 time steps. We repeat this process to predict the dynamic behavior of our origami structure for 10 seconds. In Fig.3 in the main manuscript, this predicted value is denoted by red color and the grey colored data indicate the evaluation data (the last 10 seconds) from the experimentally measured data, which has never been used during the training process. Please note that the training data is not used in Fig.3, and the grey is not the input 128 steps.

14. On page 6, "the QRNN exhibits quantitatively accurate prediction through approximately 30 excitation cycles, and then the deviation begins growing" doesn't match the results in Figure 3, especially for the 12 Hz case.

We have updated the manuscript as follows:

(Page 7 in the updated manuscript)

> "the QRNN exhibits quantitatively accurate prediction through approximately 30 excitation cycles compared with the later part of the prediction, especially for 17 Hz case,"

15. On page 8, what does "thick red/blue colors" mean? Probably you mean "dark" instead of "thick"?

We have updated the manuscript accordingly.

(Page 8 in the updated manuscript)

> "Dark red/blue colors indicate"

16. On page 8, describing Figure 4, "we extract the neuron activation of those two units and calculate the average activation value over the time length of the input data (i.e., 0.53 s)". Is this just the first 0.53 s of the 33.3 s of experimental data? If so, does this pattern continue over the later data?

Figure 4 in the main manuscript shows the visualization of the hidden state for the first 0.53 s of the evaluation data which does not include the training data. Please note that we divide the experimentally measured data (33.3 s) into training data (23.3 s) and evaluation data (10.0 s). We feed only the initial 0.53 s data of the evaluation to the QRNNs, and we predict the later data. For the neuron activation in response to the initial input data as shown in Fig.4, as the Reviewer pointed out, the hidden unit cells show the similar activation patters for the later data. To check this, we feed the input data not only from the initial 0.53 s but also from different time regimes as shown in Fig. 1. In this figure, we show the input data from the three different regimes: (**a**) 0.00 $\sim$ 0.53 s, (**b**) 4.17 $\sim$ 4.70 s, and (**c**) 8.33 $\sim$ 8.86 s. The upper plots show the displacements (u_1 and u_2) as a function of time, and the lower plots show how the 71st (orange) and 313th (green) hidden units response to the input data. We show (*Left*) 7 Hz (periodic) and (*Right*) 12 Hz (chaotic) cases. The activation changes of both hidden units show qualitatively similar behavior in response to these three input data sets. In addition, we also calculate the average responses of the hidden units for various frequency cases. Figure 2 shows the average activation values for the 71st hidden unit obtained for the above three different time regions. Our analysis indicates that the behavior of the hidden unit response for the periodic and chaotic cases is consistent even if we feed the input data from different time regions.

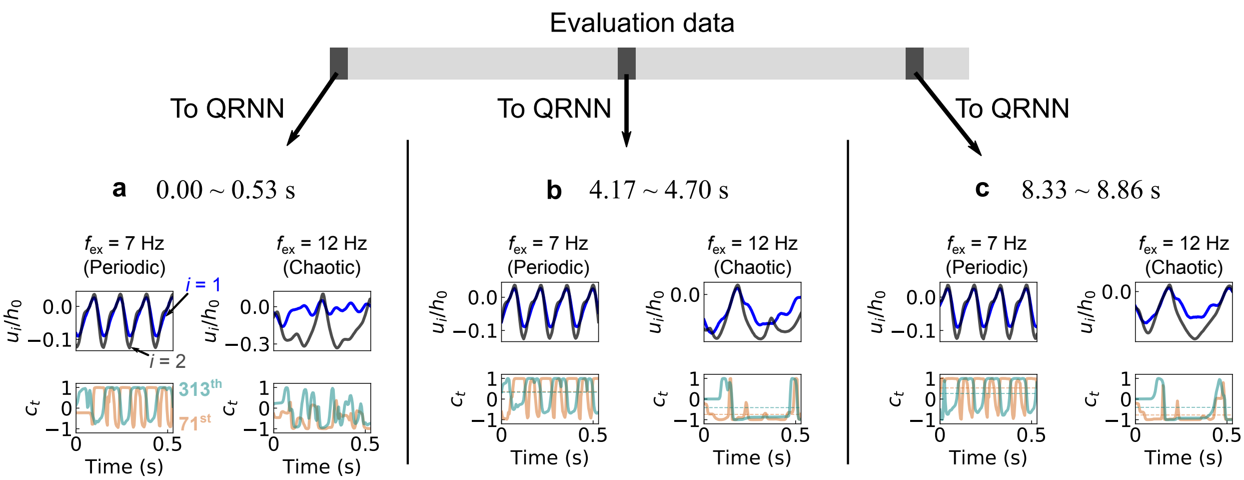

Figure 1: **Activation of 71st and 313th hidden units for different input data sets** We initially input the data from the evaluation data (128 time steps, corresponding to 0.53 s duration) to the QRNN to start the prediction. We extract this initial input data from the following different time regions: (**a**) 0.00 $\sim$ 0.53 s, (**b**) 4.17 $\sim$ 4.70 s, and (**c**) 8.33 $\sim$ 8.86 s. To show how the 71st (orange) and 313th (green) hidden units response to the input data, (*Upper*) the displacement-time plots of the input data, u_1 (blue) and u_2 (grey), and (*Lower*) the neuron activation of the 71st (orange) and 313th (green) hidden units are shown. We show the two different excitation frequency cases: (*Left*) 7 Hz and (*Right*) 12 Hz.

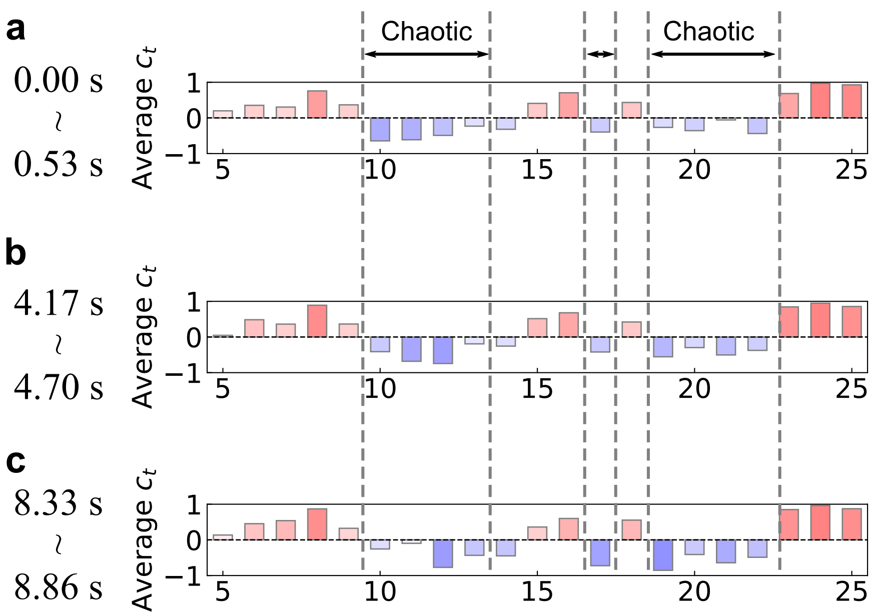

Figure 2: **Average hidden unit response for different input data sets** We extract and calculate the average value of neuron activation value C_t for the 71st hidden unit obtained for the following different time regions: (**a**) 0.00 ∼ 0.53 s, (**b**) 4.17 ∼ 4.70 s, and (**c**) 8.33 ∼ 8.86 s.

17. On the top of page 9, I think the text should be edited to acknowledge that these patterns are just mostly true. (The 14 Hz case doesn't follow the right pattern for either hidden unit 71 or 313.)

We have modified the text to mention the 14 Hz case as follows:

(Page 9 in the updated manuscript)

> "Note that although there are variations between different frequencies in the same periodic/chaotic regimes (especially $f_{ex} = 14$ Hz), we confirm this unique response of hidden units from 10 different training runs, and these multiple trained systems could enhance the capability of distinguishing between periodic and chaotic cases "

18. For Figure S5, "Average neuron activation of the hidden units in the final third hidden layer over 10 training runs", I'm guessing that you didn't average the same neurons across the 10 training runs? I'm guessing that you're showing a different set of 6 neurons for each training run? I think you might be addressing that by using "from each" here: "We selected six different hidden units from each trained QRNN to create panels (b-g). The difference between panels b and c (or d and e, or f and g) is the different hidden units.", but it's still a little confusing to me. Did you pick six interesting neurons from each run (based on averaging over the activations for the 128 time steps in that run) and then average again?

Thank you for this detailed review. As the Reviewer pointed out, we use different sets of 6 neurons from each training to construct Fig. S5 because convolutional filters are initialized with random seeds for each training run. From each run, we select 6 hidden units and calculate the average activation value C_t over 128 time steps (e.g., Fig.4g in the main text). Then, we collect this averaged value from 10 different training runs, and calculate the average values and error bars over these 10 runs. This has been clarified in the updated Supplementary Information.

(Fig.S5 caption in the updated Supplementary Information)

> "Note that we use different sets of 6 neurons from 10 training runs, then calculate the average values and error bars over these 10 runs. "

19. In the Figure S1 caption, "However this operation depends only on c_{t_i-1} from the previous time step" is confusing. This is a recurrent relation, so it depends on the previous time steps as well. Similarly, on Page 8 of the Supplement, "Please note that unlike standard RNNs, the calculation of the hidden state (c_t) of the QRNN depends only on the hidden state at previous time step (c_{t-1})" should be clarified.

Thank you for the comment. As the Reviewer mentioned, c_t is calculated from the previous time "steps" due to the recurrent pooling function. Our intention of the original explanation was to emphasize the difference between a LSTM, in which the calculation of the hidden state requires both $c(t_{i-1})$ and $h(t_{i-1})$ and a QRNN in which $h(t_{i-1})$ is not used to calculate c_t. To clarify this point, we have updated the manuscript as follows:

(Fig. S1 caption in the updated Supplementary Information)

> "However this operation depends only on $c(t_{i-1})$ from the previous time step instead of using both $c(t_{i-1})$ and $h(t_{i-1})$ from the previous time step."

20. In the Figure S1 caption, I don't know what you mean by "Internal calculations of a pooling part consist of only element-wise multiplications, which allows both prediction (Lower left insets show phase planes for periodic and chaotic cases. Red dots represent Poincare map) and visualization of hidden states readily without additional layers or components in the system (Right)."

We have modified the caption of Fig.S1 to clarify the element-wise calculation of the recurrent pooling function as follows:

(Fig. S1 caption in the updated Supplementary Information)

> "Within each pooling layer, the calculations consist of only element-wise multiplications without direct interactions between different hidden units "

21. I don't see a code availability statement.

We modified "Data availability" section as below:

> "Data and code supporting the findings of this study are available from the corresponding author on request. "

Repeating some citations from your bibliography for reference:

[1] S. H. Strogatz, Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering (Westview Press, 2015), second edition ed.

[2] S. Wiggins, Introduction to applied nonlinear dynamical systems and chaos, no. 2 in Texts in applied mathematics (Springer, 2003), 2nd ed.

[8] J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, Chaos 27, 121102 (2017).

[9] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Physical Review Letters 120, 24102 (2018).

[10] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, Proceedings of the Royal Society A: Mathematical, Physical & Engineering Sciences 474, 20170844 (2018), 1802.07486.

[11] K. Yeo and I. Melnyk, Journal of Computational Physics 376, 1212 (2019).

[12] J. Bradbury, S. Merity, C. Xiong, and R. Socher, in International Conference on Learning Representations (ICLR 2017) (2017), pp. 1–11, 1611.01576.

We modified the references of the SI to use consistent reference numbers in the main text and SI.

Second Reviewer (Reviewer's comments in blue font):

In this work, the authors investigate a chaotic mechanical process (origami folding) using neural networks. In particular, they employ quasi-recurrent NNs to predict the periodic and chaotic evolution of the system in time. Overall, I am quite impressed by this work for a number of reasons.

The authors train their algorithm on the first part of the time series and test on the second part, which is much more difficult than what most authors do by simply randomly sampling from the time series. The authors are thus extrapolating, which is much more difficult than the typical interpolation.

Further, the authors make efforts to interpret their findings by analyzing the hidden layers of the NN. I believe that applying NN to physical systems where we have some intuition is a great way for us to learn more about the NN itself. So I am happy to see the authors work for understanding in addition to performance.

The figures are very clear and illuminating and the paper is well written.

I recommend publication, although I have a couple of minor suggestions for the authors:

We thank the Reviewer for appreciating the novelty and the impact of our manuscript. We also appreciate the Reviewer's suggestions, and we have revised our manuscript accordingly, with detailed responses to each comment provided below.

*** I encourage the authors to make their code available as open source, for example on GitHub. This will help other researchers build on these methods and carefully check the model.**

Thank you for the suggestion. We modified "Data availability" section as below:

> "Data and code supporting the findings of this study are available from the corresponding author on request. "

*** The references are pretty scarce. There are some good references by Mahadevan and others.**

We thank the Reviewer for this suggestion. We have added the following references in the updated manuscript:

[7] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning (MIT Press, 2016).

[8] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Reviews of Modern Physics 91, 045002 (2019).

[10] H. Jaeger and H. Haas, Science304, 78 (2004).

[14] Z. Y. Wan, P. Vlachas, P. Koumoutsakos, and T. Sapsis, PLoS ONE 13, e0197704 (2018).

[17] Z. Lin, M. Feng, C. N. Dos Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, inInternationalConference on Learning Representations (ICLR 2017)(2017), pp. 1–15.

[21] H. Fang, S. Li, H. Ji, and K. W. Wang, Physical Review E 95, 052211 (2017).

[30] H. G. Kim and R. Wiebe, Composite Structures 235, 111743 (2020).

[31] X. Zhang and S. Mahadevan, Decision Support Systems131, 113246 (2020).

[32] M. Serra, S. Streichan, M. Chuai, C. J. Weijer, and L. Mahadevan, Proceedings of the National Academy of Sciences 117, 11444 (2020).

[34] G. V. Rodrigues, L. M. Fonseca, M. A. Savi, and A. Paiva, International Journal of Mechanical Sciences 133, 303 (2017).

[35] M. T. Rosenstein, J. J. Collins, and C. J. De Luca, 65, 117 (1993).

REVIEWERS' COMMENTS:

Reviewer #1 (Remarks to the Author):

Thank you for addressing most of my comments. I am especially glad to hear that you had held-out data.

A few downsides of this paper remain:
The main difference between this approach to predicting chaotic systems and previous work on predicting chaotic systems is using a QRNN instead of a more classical RNN like an LSTM. The potential benefits of the QRNN are speed due to opportunities for parallelism within layers and interpretability. As I mentioned before, I think this paper would be stronger if it included a comparison of the speed & accuracy of this approach on this type of data to a more classical RNN. And, although there is some interpretability here, I don't think it is as strong as in the original QRNN paper.

A few of my minor comments haven't been addressed.
5. I still think "Due to the element-wise calculation in the pooling function, activation of each neuron does not depend on the past outputs of other neurons" isn't correct, but it would be correct if it said "Due to the element-wise calculation in the pooling function, activation of each neuron does not depend on the past outputs of other neurons in the same layer." The original paper distinguishes that the convolutional layers apply in parallel across time and the pooling layers apply in parallel across channels. They also say, "This is a consequence of the elementwise nature of the recurrent pooling function, which delays direct interaction between different channels of the hidden state until the computation of the next QRNN layer." The activations of each neuron need to depend on past outputs of other neurons (from other layers), because otherwise context would be lost. In the conclusion, they mention "...some aspects require long-distance context and must be computed recurrently. Many existing neural network architectures either fail to take advantage of the contextual information or fail to take advantage of the parallelism." I think think the authors of the original QRNN paper saw these nuances as important, but this paper is less clear. Especially since this is the paragraph where you introduce QRNNs, I think it's important to make this precise.

13. Thanks for explaining this point to me, but I still think it should be documented in the paper.

19. This one was fixed in one place but not the second place.