Confronting Domain Shift in Trained Neural Networks

Carianne Martinez CMARTI5@SANDIA.GOV
Sandia National Laboratories
Arizona State University

David A. Najera-Flores DANAJER@SANDIA.GOV
ATA Engineering, Inc.

Adam R. Brink ARBRINK@SANDIA.GOV
Sandia National Laboratories

D. Dane Quinn QUINN@UAKRON.EDU
University of Akron

Eleni Chatzi CHATZI@IBK.BAUG.ETHZ.CH
ETH Zürich

Stephanie Forrest STEPH@ASU.EDU
Arizona State University
Santa Fe Institute

Abstract

Neural networks (NNs) are known as universal function approximators and can interpolate nonlinear functions between observed data points. However, when the target domain for deployment shifts from the training domain and NNs must extrapolate, the results are notoriously poor. Prior work Martinez et al. (2019) has shown that NN uncertainty estimates can be used to correct binary predictions in shifted domains without retraining the model. We hypothesize that this approach can be extended to correct real-valued time series predictions. As an exemplar, we consider two mechanical systems with nonlinear dynamics. The first system consists of a spring-mass system where the stiffness changes abruptly, and the second is a real experimental system with a frictional joint that is an open challenge for structural dynamicists to model efficiently. Our experiments will test whether 1) NN uncertainty estimates can identify when the input domain has shifted from the training domain and 2) whether the information used to calculate uncertainty estimates can be used to correct the NN's time series predictions. While the method as proposed did not significantly improve predictions, our results did show potential for modifications that could improve models' predictions and play a role in structural health monitoring systems that directly impact public safety.

Keywords: Pre-registration, Machine Learning, Reduced Order Model, Uncertainty Quantification, Domain shift

1. Introduction

NNs have seen great success in accurately modeling nonlinear functions by learning directly from observed data. Techniques such as Transformers Vaswani et al. (2017) and Long Short Term Memory (LSTM) Hochreiter and Schmidhuber (1997) models have been applied to sequential data and have demonstrated impressive capabilities in the field of natural language processing (NLP) Otter et al. (2020), excelling at tasks such as language translation Vaswani et al. (2017) and answering text based questions Devlin et al. (2018). These models have been extended to scientific domains where physical laws govern the dynamics of a system Najera-Flores and Brink (2018); Simpson et al. (2020); however, while the performance of a NN may be acceptable when the target domain is closely aligned with the training domain, its performance may degrade when the target domain deviates significantly from the training set. This limitation prevents them from use in high consequence environments such as those monitored by structural health monitoring (SHM) systems, where system failure directly implies that the dominant physics of the system shifts, and indications of this failure must be identified and mitigated to ensure public safety.

Techniques to improve deep learning (DL) model performance on targets that have shifted from the training domain have been proposed in the literature and will be discussed in Section 2. These methods often augment the training data set to more closely match the target deployment domain. They require expensive retraining of models and are not feasible when rapid approximations of system dynamics are necessary. **Our approach removes the need for additional data or training by leveraging information that already exists in the weights of the trained model, realized in the form of uncertainty estimation.** The exemplars set forth herein require efficient approximations of future system states and are critical for understanding the risks associated with deploying systems for industries like aviation Najera-Flores and Brink (2018). Prior work Martinez et al. (2019) introduced a technique to avoid the need for retraining DL models while extending their applicability to shifted target domains. Results from this work indicated that when the most uncertain predictions were flipped, segmentations were significantly improved. An example of a result from this technique is shown in Figure 1, where a NN trained on a particular image domain is extended for use in a shifted domain with improved predictive capability.

We hypothesize that this technique can be extended from binary classification to time-dependent regression, where patterns in the sequential input to the DL model can be used to 1) identify that domain shift is occurring and 2) improve the DL model's prediction without retraining. The anticipated contributions of this work are:

- A practical method for applying DL models to time series data in shifted domains
- New publicly available datasets from the structural dynamics field of well-defined physical systems
- Open source code implementation that allows replication and extension of our experimental results

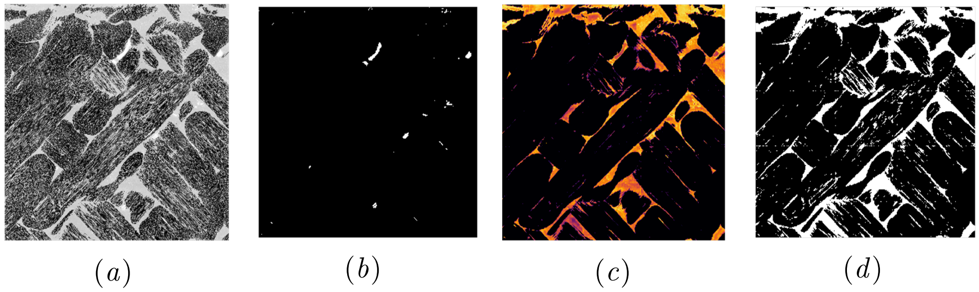


Figure 1: Results from Martinez et al. (2019) showing that uncertainty maps can be used directly to improved trained NN predictions. (a) Slices of CT scan to be segmented. (b) Predicted binary label for the CT slice from the trained NN without UQ correction. (c) Uncertainty map (brighter colors indicate higher uncertainty). (d) Resulting binary labels after UQ informed improvement.

2. Related work

The overfitting of DL models to a specific training domain is a known weakness of NNs, and current research efforts seek to overcome this shortfall. Here we review work on domain shift, DL uncertainty quantification, and the structural dynamics involved in our training domain.

2.1. DL model domain shift mitigation and uncertainty quantification

The problem of domain shift from a training domain to a target domain is an open and active area of DL research. Much of this work focuses on computer vision applications Venkateswara and Panchanathan (2020); for example, Stacke et al. (2019) studied the problem in the context of convolutional NNs and proposed a metric for identifying domain shift in images that leverages information about the NN weights. Other existing works focus on data augmentation, retraining models to better generalize, and training additional models. Sun and Saenko (2016) adds a CORAL loss function that works to effectively transform the features in the network itself to be relevant to a shifted domain. This approach requires unlabeled examples of the shifted domain to learn transformations in the feature space that will reduce the CORAL loss. Li et al. (2018) uses a generative model to Zhou and Li (2005) augment the data necessary to perform well in a shifted deployment domain. CyCADA Hoffman et al. (2017) also employs a generative model to align the shifted domain with the training domain using both pixel-level and feature-level transformations. In Ren et al. (2019), a likelihood ratio is introduced to overcome background statistics that are shown to drive overconfidence in generative model predictions. This method requires training of an additional background-specific model. Domain adaptation techniques Zhu et al. (2017); Sener et al. (2016) can also mitigate shifts in data by training separate models to preprocess the shifted inputs to more closely match the training domain. All of these approaches require additional resources, but in contrast, our proposed method actively uses uncertainty estimates to correct DL model predictions without retraining. Surveys of modern techniques for anomaly detection Wang et al. (2020); Braei and Wagner (2020) are also relevant as these approaches could be applied to detecting domain shift.

Several methods have been proposed to quantify uncertainty in DL model predictions. These include ensemble methods Lakshminarayanan et al. (2017), Bayesian NNs Neal (2012), and dropout networks Gal and Ghahramani (2016). We implement dropout networks in this work to quantify the uncertainty in DL model predictions due to their ease of implementation and their effectiveness with only a single model to be trained.

2.2. Structural dynamics modeling and structural health monitoring (SHM)

We obtain our exemplar datasets from the field of structural dynamics, where applications such as reduced order modeling of complex systems control and SHM of complex systems require real-time detection of anomalous system behavior. In addition to a mechanical example where the system stiffness shifts dramatically, we will utilize experimental data from a jointed structural system. Frictional joints are well-studied Bickford (1926); Vlachas et al. (2020), but current reduced order models (ROMs) cannot practically capture the full extent of the underlying nonlinear physics. To mitigate error accumulation, autoregressive models, a form of NNs Saxén (1997), and k-step ahead prediction Favier and Dubois (1990) are typically used. The proposed corrective mechanism would advance modeling capabilities.

SHM is defined as a four-level hierarchy Rytter (1993); Farrar and Lieven (2007) aiming to detect, localize, quantify, and finally predict damage on the basis of data extracted from operating engineered systems. In doing so, a large body of recent literature explores utilization of ML and DL methods for damage prognosis. Many existing works focus on outlier classification for damage detection Bull et al. (2019). Generative modeling approaches attempt to reproduce joint probabilistic distributions from monitoring data in order to recognize distinct condition regimes Mylonas et al. (2020). For achieving the higher steps in the SHM hierarchy, physics-informed learning incorporates domain knowledge into the learning process Yuan et al. (2020). In this work, we treat this problem as adaptation to shifted domains.

3. Methodology

When a NN is trained to mimic time series data, it learns a mapping from patterns observed in previous time steps to the next data point in the time series. When time series deviates from the expected patterns, the NN could fail to make accurate predictions. If successful, our method will extend the applicability of trained NNs to mitigate domain shift by 1) recognizing that the input domain has shifted and 2) using uncertainty quantification to drive the predictions toward a corrective path.

Our method assumes that a NN with dropout layers used to quantify the uncertainty in its predictions is trained to approximate a real-valued function $f(x,t)$. Input to the model is a sequence of values of f over a series of previous time steps along with the value of x at time t, and output is the value of f over a sequence of subsequent time steps. When the model's uncertainty exceeds a threshold value, instead of returning the model's nominal prediction for f at time t, our method updates the prediction to incorporate information from the calculated uncertainty to improve accuracy. Using the dropout technique set forth in Gal and Ghahramani (2016), we infer several predictions for f at time t with different subsets of neuron outputs dropped from the calculation, resulting in a distribution of predicted output values at each time step. Rather than leaving the uncertainty estimation as a simple indication of the model's confidence at time t, our method actively uses statistical properties of the distribution to serve as a corrective factor for the prediction of f at time t. We will explore two corrective methods in this work: 1) We replace the nominal prediction with the mean of the prediction distribution and 2) We add the standard deviation of the prediction distribution to the nominal prediction in the direction of the distribution skew.

4. Experimental protocol

We will use two structural dynamics datasets to test our hypotheses and report results from two DL models. For both datasets, our intent is to answer the following questions:

- RQ1: Does the uncertainty value correctly detect a significant change in the model's accuracy?
- RQ2: Does the corrective factor informed by the uncertainty improve the accuracy of the prediction?

4.1. Datasets

We will first investigate our method's performance on a toy problem consisting of data drawn from simulations of a mass-spring system with one mass element and a fixed stiffness with varying initial conditions, and loaded under a known force. We will also generate simulated data where the stiffness of the spring abruptly changes during the simulation. The data will consist of a time series of the force on the mass as well as the displacement of the mass, and initial system conditions.

A more challenging dataset will be derived from experimental measurements of a frictional jointed structure subject to a known force. A schematic of the system is shown in Figure 2. This dataset will include the initial conditions, the load on the structure, and accelerometer and displacement measurements from various positions in the structure. We will also develop a reduced order model (ROM) of the system that will predict the displacement over time of structural mass elements. The ability of ROMs for jointed structures to match experimental data is known to degrade as the structural loading on the joint increases and the nonlinear dynamics induced by the joint become more significant.

Each dataset will consist of approximately 100,000 time steps per example, and the simulations will use on the order of 100 different initial conditions.

4.2. Model training

We will implement both a WaveNet with a stack of dilations of size [1,2,4,8] and a receptive field of length 128 as in Oord et al. (2016) and a Transformer with the base model architecture as presented in Vaswani et al. (2017), each of which have seen success in predicting sequential data. For WaveNet, we will apply dropout to all convolutional layers. For Transformer, we will apply dropout only to the decoder portion of the network, since we have observed that dropout in encoding layers removes input information necessary for useful encodings. Each model will be evaluated on both datasets.

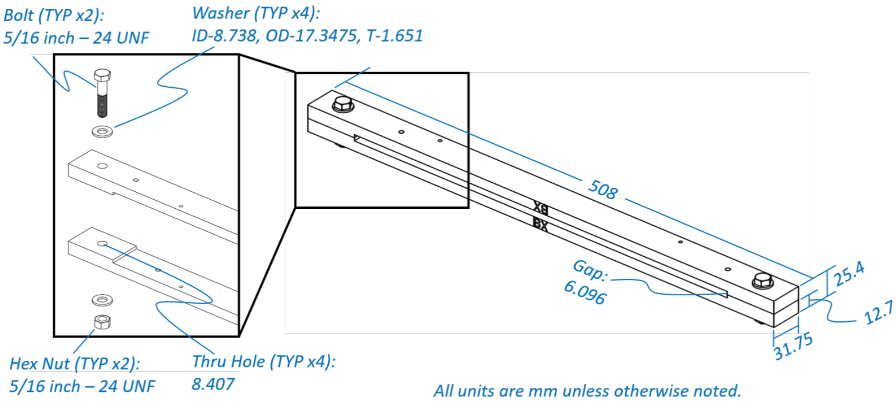


Figure 2: Schematic of jointed structural system from Brink et al. (2020) used to obtain displacement dataset.

For the mass-spring system, our DL models will be given the system's initial conditions, the force on the mass elements at each time step as well as the displacement of the mass elements over a series of previous time steps, and will be used to predict the displacements at the next time step. After training on several examples from this system, we will introduce an input series to the trained model that simulates an abrupt change to the spring's stiffness and apply our corrective factor to improve the predictions of the mass displacement.

For the jointed structure, our DL models will be trained to learn the system dynamics solely from the ROM data and will learn to predict the displacement of each discretized mass element modeled by the ROM. We will then apply our trained DL model on the experimental structure data, where the output with the corrective factor will be used to predict the next time step of the displacements in the real structure. The key idea here is that our ROM will be unable to capture all of the physics necessary to predict the true system dynamics, and that our DL model will identify that the real inputs have shifted from the training domain, and compensate for the missing physics.

One of the primary challenges of employing neural network for predictions in the time domain is the accumulation of error that arises from recursion. To mitigate this challenge, we will enforce physical constraints through the loss function. Terms that require conservation of energy and momentum will encourage the network to learn not only the target output, but its derivatives and the relationship between them. A byproduct of this constraint is that the problem is bounded to produce high-quality predictions in the physical domain in which it was trained. When presented with data from outside its domain, the prediction uncertainty will increase as the physical constraints are harder to enforce.

4.3. Evaluation and significance

We are interested in the impact of our method on the accuracy of sequential predictions, and first must establish baseline behavior of each DL model. We will use the Adam optimizer Kingma and Ba (2015) with learning rate 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon =$ 1e-8 (the default Keras Chollet et al. (2015) settings) with dropout rate of 0.1 for both training and inference to calculate uncertainty. We will exhaustively evaluate each baseline DL model over input sequences of 32, 64, and 128 time steps and output sequence lengths of 1,2,3, and 4. We have proposed these specific hyperparameter settings for concreteness, but we intend to explore other settings such as the dropout rate and the uncertainty threshold value as appropriate to establish baseline performance of the models. We will use the most accurate WaveNet and Transformer model to evaluate the efficacy of our corrective method by calculating the mean squared error with respect to the ground truth sequences with and without our method's corrective factor. From the output of the two competing models, we will estimate the distributions of residuals to quantify the statistical significance of our model improvements using a dependent two-sample t-test or the Wilcoxon-Mann-Whitney U-test as applicable.

When employing our method, we will make 48 predictions for each time step and use the distribution of predictions to correct the prediction if the standard deviation of the distribution of predictions exceeds 10% of the value of the nominal prediction for the next time step. We will explore two corrective factors: 1) the mean of the predictions and 2) the addition of the standard deviation in the direction of the skew of the prediction distribution.

While DL remains a powerful tool for modeling complex systems, its inability to overcome domain shift severely limits its successful deployment. If we achieve a positive result from these experiments, we will unlock the potential of repurposing unused latent features for improved DL generalization.

5. Documented Modifications

In this section, we describe modifications to the experimental protocols that were described in the previous sections.

- **Frictional jointed structure dataset:** For the frictional jointed structure, a ROM of the system was developed based on the experimental data. However, the experimental dataset was insufficient to map accelerometer measurements directly to simulated structure locations in the ROM. Therefore, in lieu of comparing ROM output to the experimental data directly, we trained our DL models on the displacement time series output from the ROM at a point 40% along the beam with fixed initial conditions. The ROM was developed and incorporated a cubic stiffness coefficient k to account for the nonlinearity introduced by the presence of the frictional joint. In the training of the DL models this cubic stiffness coefficient was set to be $k = 1 \times 10^7$, which was representative of the response observed in the experimental system for nominal excitation levels. We then used the ROM to generate 3 displacement time series with the same initial conditions as the training set, but with cubic stiffness $k \in \{2 \times 10^8, 4 \times 10^8, 1 \times 10^9\}$ to simulate a domain shift similar to the domain shift we expected to observe in experimental data.

- **DL Models:** We modified open source PyTorch versions of the Transformer Guhr et al. (2020) and WaveNet Hermann et al. (2018) architectures to add dropout layers for uncertainty quantification. Physics constraints were not needed to achieve high accuracy in each training domain, so we did not incorporate physics into the models.

The models rely on the system response (displacement) at the previous timesteps to make their predictions.

- **Uncertainty threshold criterion:** We originally proposed to consider any prediction with uncertainty greater than ten percent of the predicted value to be an indicator of domain shift requiring corrective action. In running experiments with this uncertainty threshold, we observed that even in the training domain, several predictions had an estimated uncertainty of greater than 10% of the predicted value. We present the results of these experiments, but we also introduce an alternative uncertainty threshold criterion. We use the maximum uncertainty over an inference run of the test data from the training domain as a baseline and consider any prediction with estimated uncertainty that exceeds that maximum value to be an indication of domain shift that triggers corrective action.

- **Abstract:** The abstract was modified to reflect our findings.

6. Results

In this section, we present experimental results from the mass-spring dataset and the jointed structure.

6.1. Dataset details

For the mass-spring dataset, we developed a MATLAB simulator of a single degree-of-freedom mechanical oscillator system with linear and nonlinear stiffness. In particular the stiffness coefficient k of the cubic nonlinearity is varied, introducing the domain shift in the system response. Note that the remaining system parameters are held fixed, so that in essence the underlying linear system is constant and the domain shift is introduced through abrupt changes in the nonlinearity during the simulation. We generated simulated displacement output for $k = 8$, 16, 32, and 64, and we used 245 different sets of initial conditions that vary the initial modal displacements for each cubic stiffness coefficient. Each time series consists of 2,097,151 timesteps, and we downsampled by a factor of 400 for training and inference. We chose the set with $k = 32$ as our training domain and, for each set of initial conditions, trained a model using the first 3000 timesteps for training, the next 1242 steps for validation, and reserved the last 1000 timesteps for testing. To simulate a sudden stiffness change and generate datasets with shifted domains, we concatenated a portion of the time series from the training domain ($k = 32$) with a portion of the time series from a shifted domain $k \in \{8, 16, 64\}$ for each time series with the same initial conditions. This resulted in 3 datasets with different levels of domain shift.

The jointed structure dataset consists of four displacement time series generated by our ROM as described in Section 5. We downsampled these data by a factor of 100 and used the data with $k = 1 \times 10^7$ as the training domain.

6.2. Model selection

We used the mass-spring dataset with cubic stiffness coefficient 32 to select the best performing versions of each model architecture for use in our experiments. For the Transformer, we found that an input window of 128 and an output window of 4 gave the best average mean squared error over the 245 test examples and the WaveNet had the most success with an input window of 64 and output window 1. We tested each model version over 128 total inference steps. For all experiments, we used these best architecture versions. We trained separate models for each of the 245 initial conditions for the mass-spring dataset as well as a model for the jointed structure's training domain.

6.3. Domain shift metrics

For domain shift detection, we calculated the percentage of experiments where the uncertainty of a prediction exceeded the threshold chosen to trigger corrective action. We also report the false positive detection rate: i.e., the percentage of experiments in which the uncertainty exceeded the threshold prior to the domain shift event. Results for both the originally proposed 10% threshold and the maximum training uncertainty threshold are shown in Table 1.

(*a*)

k	**Model**	**Detection**	**FP**
8	WaveNet	0.984	0.980
16	WaveNet	0.984	0.980
64	WaveNet	1.0	0.980
8	Transformer	0.996	0.808
16	Transformer	0.996	0.808
64	Transforer	0.996	0.808

(*b*)

k	**Model**	**Detection**	**FP**
8	WaveNet	0.412	0.347
16	WaveNet	0.416	0.347
64	WaveNet	0.608	0.351
8	Transformer	0.796	0.086
16	Transformer	0.776	0.086
64	Transformer	0.914	0.082

Table 1: Domain shift detection rate (Detection) and false positive rate (FP) for mass-spring datasets with (a) 10% uncertainty threshold and (b) training set max uncertainty threshold. Training set max improves false positive rate.

6.4. Prediction correction metrics

Table 2 shows the results over each of the mass-spring datasets with varying cubic stiffness k. Since the cubic stiffness is driving the domain shift, the data with $k = 16$ is qualitatively the closest to that of the training set with $k = 32$. To assess our method's performance, we calculate the mean squared error over 128 inference steps with respect to the ground truth data for both the Transformer and WaveNet models with 1) no corrective action taken, 2) uncertain predictions replaced with the mean over MC predictions for the timestep, denoted as "Mean correction" and 3) uncertain predictions replaced with the prediction +/- the standard deviation over MC predictions for the timestep, where the sign of the corrective factor is determined by the majority of the MC predictions, denoted as "Skew correction". Table 3 shows results over each of the frictional jointed structure datasets evaluated over 800 timesteps.

(a)

k	**Model**	**No correction**	**Mean correction**	**Skew correction**
8	WaveNet	**0.00148 +/- 0.00181**	0.00159 +/- 0.00187	0.00450 +/- 0.00558
16	WaveNet	**0.00126 +/- 0.00161**	0.00136 +/- 0.00168	0.00413 +/- 0.00531
64	WaveNet	**0.00236 +/- 0.00326**	0.00237 +/- 0.00321	0.00511 +/- 0.00892
8	Transformer	0.07379 +/- 0.07971	0.07349 +/- 0.07991	**0.07252 +/- 0.07898**
16	Transformer	0.04435 +/- 0.04856	0.04408 +/- 0.04814	**0.04354+/- 0.04756**
64	Transformer	0.07689 +/- 0.08116	0.07632 +/- 0.08129	**0.07627 +/- 0.08077**

(b)

k	**Model**	**No correction**	**Mean correction**	**Skew correction**
8	WaveNet	**0.00148 +/- 0.00181**	0.00149 +/- 0.00181	0.00159 +/- 0.00187
16	WaveNet	**0.00126 +/- 0.00161**	0.00126 +/- 0.00160	0.00135 +/- 0.00164
64	WaveNet	0.00236 +/- 0.00326	**0.00234 +/- 0.00323**	0.00248 +/- 0.00325
8	Transformer	0.07379 +/- 0.07971	0.07379 +/- 0.07963	**0.07358 +/- 0.07952**
16	Transformer	0.04435 +/- 0.04856	0.04430 +/- 0.04843	**0.04405 +/- 0.04816**
64	Transformer	0.07689 +/- 0.08116	0.07685 +/- 0.08102	**0.07675 +/- 0.08085**

Table 2: Average mean squared error of model predictions over mass-spring datasets with cubic stiffness k with and without uncertainty-driven correction using (a) the 10% threshold and (b) training set max uncertainty as threshold. Best result for each model type on each dataset shown in bold. Corrections often fail to improve the WaveNet's predictions while the Transformer using the Skew correction strategy achieves the lowest average error, but not with statistically significant improvements to predictions.

k	**Model**	**Threshold**	**No correction**	**Mean correction**	**Skew correction**
1×10^9	Transformer	10%	6.27×10^{-6}	$\mathbf{5.91 \times 10^{-6}}$	6.36×10^{-6}
2×10^8	Transformer	10%	8.93×10^{-6}	8.21×10^{-6}	$\mathbf{6.64 \times 10^{-6}}$
4×10^8	Transformer	10%	8.89×10^{-6}	8.18×10^{-6}	$\mathbf{6.69 \times 10^{-6}}$
1×10^9	Transformer	train max	6.27×10^{-6}	6.26×10^{-6}	$\mathbf{6.04 \times 10^{-6}}$
2×10^8	Transformer	train max	8.93×10^{-6}	8.93×10^{-6}	$\mathbf{8.89 \times 10^{-6}}$
4×10^8	Transformer	train max	8.89×10^{-6}	8.89×10^{-6}	$\mathbf{8.85 \times 10^{-6}}$
1×10^9	WaveNet	10%	$\mathbf{2.37 \times 10^{-7}}$	2.68×10^{-7}	9.29×10^{-7}
2×10^8	WaveNet	10%	2.11×10^{-7}	$\mathbf{1.53 \times 10^{-7}}$	6.25×10^{-7}
4×10^8	WaveNet	10%	2.09×10^{-7}	$\mathbf{1.54 \times 10^{-7}}$	6.36×10^{-7}
1×10^9	WaveNet	train max	$\mathbf{2.37 \times 10^{-7}}$	2.37×10^{-7}	2.39×10^{-7}
2×10^8	WaveNet	train max	$\mathbf{2.11 \times 10^{-7}}$	2.11×10^{-7}	2.11×10^{-7}
4×10^8	WaveNet	train max	$\mathbf{2.09 \times 10^{-7}}$	2.09×10^{-7}	2.09×10^{-7}

Table 3: Average mean squared error of model predictions over frictional jointed structure datasets with cubic stiffness k with and without uncertainty-driven correction. Best result for each model type on each dataset shown in bold. Our models achieve the lowest error on average, but prediction improvements are not statistically significant.

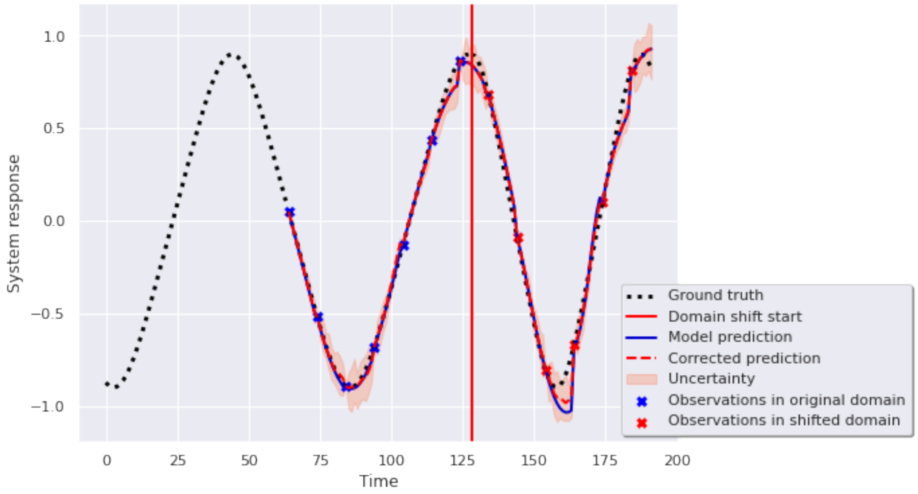


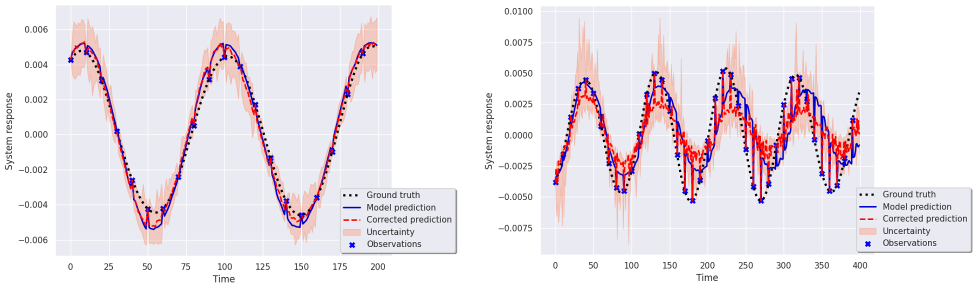
Figure 3: WaveNet result from mass-spring dataset with $k = 64$. Domain shift starts at the vertical red line, where the frequency of the ground truth data increases. Original WaveNet prediction with no corrective action is shown with a solid blue line, and corrected prediction with mean corrective strategy is shown as a dashed red line. Where uncertainty is high after the domain shift occurs between time 150 and 175, the corrective strategy is triggered and the model's prediction is marginally improved.

7. Findings

In this section, we discuss the conclusions based on our results, as well as directions for future work.

To answer RQ1, we review the domain shift detection metrics shown in Table 1. Our results indicate that the originally proposed 10% uncertainty threshold is exceeded by the vast majority of test examples, but the false positive rate is unacceptably high for the proposed use case of detecting domain shift. We calculated an alternative threshold based on the maximum uncertainty of a model's predictions on examples within its training domain. The detection rate decreases significantly for both the WaveNet and Transformer models, but the Transformer's false positive rate also drops to below 10%. These results indicate that the Transformer is a superior detector with an F1 score of 0.87 compared to that of the WaveNet at 0.52. This result points to the importance of selecting the criteria necessary to indicate domain shift. While varying the threshold was not within the scope of this study, our additional experiment indicates that it is possible to improve the method's performance with a more reliable criterion.

For RQ2, we review the results in Table 2 and Table 3. For the mass-spring data, the Transformer with the skew correction method achieved the best performance relative to the uncorrected model predictions on average. The WaveNet model performed better on only the mass-spring dataset with $k = 64$ and the training set maximum uncertainty threshold. For the frictional jointed structure data, again the Transformer achieved a lower mean



(a) (b)

Figure 4: Frictional jointed structure dataset where the entire time series has shifted from the training domain. Original model predictions with no corrective action are shown as a solid blue line, and corrected predictions are shown as a dashed red line. (a) WaveNet result from with $k = 4 \times 10^8$ with mean corrective strategy. (b) Transformer result with $k = 4 \times 10^8$ and the skew correction strategy. WaveNet models handle domain shift well in many cases and need little correction. Transformer models generally accumulated more error and produced noisier predictions. Corrective actions in each case made small improvements to predictions.

squared error on average, but here the correction methods performed differently depending on the dataset. For the data that shifted the most from the training domain with $k = 1 \times 10^9$ using the 10% threshold, the mean correction method performed the best while the skew correction method was on average better for all other data. In all cases, however, the improvements were not statistically significant. As shown in the tables, there is a large amount of variance around the performance of our method. The mass-spring dataset with $k = 64$ shows the largest average margin of improvement over the baseline; however, in 40.8% of the test examples, the mean squared error of the model's predictions was lower without the corrective methods (compared with 49.8% where the skew correction was best). Further investigation is required to determine contributing factors to the discrepancy in performance.

While the improvements are marginal, Figure 3 and Figure 4(a) show examples from the mass-spring and the frictional jointed structure datasets, respectively, where the mean corrective strategy improves the WaveNet's prediction. The small improvement between timesteps 150 and 175 in Figure 3 and at the lower extremes of the system response in Figure 4(a) typify the experiments in which our methods reduce the model error.

We leave the investigation of the discrepancy in our method's performance between the WaveNet and Transformer architectures for future work, but we hypothesize that the differing ability of each model to generalize beyond the training data without correction was a significant factor. Our results indicate that the WaveNet predicted better overall in the shifted domains and was in less need of correction.

Figure 4(b) shows an example of the error accumulation and noisy Transformer prediction in the shifted domain of the frictional jointed structure data. Again, our method makes minor improvements to the prediction.

8. Conclusion

In this paper, we proposed a set of experiments to test our hypothesis that neural network uncertainty could detect domain shift and actively correct time series predictions without retraining. We could often detect domain shift through uncertainty, but this method led to false positives. After changing our threshold from a fixed percentage to one based on the uncertainty of the model's predictions within its training domain, the false positive rate was significantly reduced, and threshold selection presents opportunities for further exploration. While these false positive predictions do not indicate true domain shift, , uncertainty apparently increases over time as error accumulates and the quality of the model's predictions deteriorate. Thus our approach might prove useful for the structural dynamics domain as an indication of model error generally rather than domain shift specifically. Further investigation is necessary to test our method's capability of identifying error rather than explicit domain shift.

Although the corrective factors did not achieve statistically relevant improvement to model predictions, the choice of corrective factor was made a priori. More aggressive corrective factors could improve the results, but further experiments are needed to confirm the limitations of the correction method and to establish an optimal corrective approach. We intend to explore data assimilation techniques to improve our results in future work.

Our code and data will be published at https://github.com/sandialabs upon approval for release.

Acknowledgments

We would like to thank Matthew D. Smith, Veronica Jones, Justin Jacobs, Benjamin Edwards, Drew Levin, Kevin Potter, and Demitri Maestas for their insightful comments and suggestions.

This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525. This manuscript has been authored by National Technology & Engineering Solutions of Sandia, LLC. under Contract No. DE-NA0003525 with the U.S. Department of Energy/National Nuclear Security Administration. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

References

J.H. Bickford. *Introduction to the Design and Behavior of Bolted Joints*, volume 4th Edition. CRC Press, 1926.

Mohammad Braei and Sebastian Wagner. Anomaly detection in univariate time-series: A survey on the state-of-the-art. *arXiv preprint arXiv:2004.00433*, 2020.

Adam R Brink, Robert J Kuether, Matthew D Fronk, Bryan L Witt, and Brendan L Nation. Contact stress and linearized modal predictions of as-built preloaded assembly. *Journal of Vibration and Acoustics*, 142(5), 2020.

LA Bull, TJ Rogers, C Wickramarachchi, EJ Cross, K Worden, and N Dervilis. Probabilistic active learning: An online framework for structural health monitoring. *Mechanical Systems and Signal Processing*, 134:106294, 2019.

François Chollet et al. Keras. https://keras.io, 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Charles R Farrar and Nick AJ Lieven. Damage prognosis: the future of structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851):623–632, 2007.

Gérard Favier and D Dubois. A review of k-step-ahead predictors. *Automatica*, 26(1):75–84, 1990.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning*, 2016.

Oliver Guhr et al. Transformer time series prediction. https://github.com/oliverguhr/transformer-time-series-prediction, 2020.

Vincent Hermann et al. Pytorch-wavenet. https://github.com/vincentherrmann/pytorch-wavenet, 2018.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 30th Conference on Advances in Neural Information Processing Systems*, 2017.

Xiang Li, Wei Zhang, and Qian Ding. Cross-domain fault diagnosis of rolling element bearings using deep generative neural networks. *IEEE Transactions on Industrial Electronics*, 66(7):5525–5534, 2018.

Carianne Martinez, Kevin M Potter, Matthew D Smith, Emily A Donahue, Lincoln Collins, John P Korbin, and Scott A Roberts. Segmentation certainty through uncertainty: Uncertainty-refined binary volumetric segmentation under multifactor domain shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

Charilaos Mylonas, Imad Abdallah, and EN Chatzi. Deep unsupervised learning for condition monitoring and prediction of high dimensional data with application on windfarm scada data. In *Model Validation and Uncertainty Quantification, Volume 3*, pages 189–196. Springer, 2020.

David Aaron Najera-Flores and Adam Ray Brink. Efficient random vibration analysis of nonlinear systems with long short-term memory networks for uncertainty quantification. In *Proceedings of ISMA 2018 International Conference on Noise and Vibration Engineering and USD2018 International Conference on Uncertainty in Structural Dynamics*, 2018.

Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Daniel W Otter, Julian R Medina, and Jugal K Kalita. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pages 14707–14718, 2019.

Anders Rytter. *Vibrational based inspection of civil engineering structures.* PhD thesis, Dept. of Building Technology and Structural Engineering, Aalborg University, 1993.

Henrik Saxén. On the equivalence between arma models and simple recurrent neural networks. In *Applications of Computer Aided Time Series Modeling*, pages 281–289. Springer, 1997.

Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems*, pages 2110–2118, 2016.

Thomas Simpson, Nikolaos Dervilis, and Eleni Chatzi. On the use of nonlinear normal modes for nonlinear reduced order modelling. *arXiv preprint arXiv:2007.00466*, 2020.

Karin Stacke, Gabriel Eilertsen, Jonas Unger, and Claes Lundström. A closer look at domain shift for deep learning in histopathology. *arXiv preprint arXiv:1909.11575*, 2019.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Hemanth Venkateswara and Sethuraman Panchanathan. Domain adaptation in computer vision with deep learning, 2020.

Konstantinos Vlachas, Konstantinos Tatsis, Konstantinos Agathos, Adam R Brink, and Eleni Chatzi. A local basis approximation approach for nonlinear parametric model order reduction. *arXiv preprint arXiv:2003.07716*, 2020.

Ruoying Wang, Kexin Nie, Tie Wang, Yang Yang, and Bo Long. Deep learning for anomaly detection. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 894–896, 2020.

Fuh-Gwo Yuan, Sakib Ashraf Zargar, Qiuyi Chen, and Shaohan Wang. Machine learning for structural health monitoring: challenges and opportunities. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2020*, volume 11379, page 1137903. International Society for Optics and Photonics, 2020.

Zhi-Hua Zhou and Ming Li. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge & Data Engineering*, (11):1529–1541, 2005.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.