When is an SHM Problem a Multi-Task-Learning Problem?

S. C. BEE[1], L. A. BULL[2], N. DERVILIS[1] and K. WORDEN[1]

ABSTRACT

Multi-task neural networks learn tasks simultaneously to improve individual task performance. There are three mechanisms of multi-task learning (MTL) which are explored here for the context of structural health monitoring (SHM): (i) the natural occurrence of multiple tasks; (ii) using outputs as inputs (both linked to the recent research in population-based SHM (PBSHM)); and, (iii) additional loss functions to provide different insights. Each of these problem settings for MTL is detailed and an example is given.

INTRODUCTION

Structural health monitoring (SHM), is a predictive tool which provides an online damage detection and condition monitoring strategy using data recorded from an individual structure [1]. Data are often unavailable or incomplete, measurements can be limited [2] and, as a consequence, a data set which would be used to train a model could be insufficient to provide reliable results. Part of the increased motivation for SHM systems is from the growing number of structures which are reaching the end of their design life [1]; if the condition of the structures can be accurately assessed online for damage, then the structures can continue to operate. A key element to a SHM system is that it is *accurate*, as the cost of uninformative predictions is not just economical but could have safety implications too.

Multi-task learning (MTL) refers to a suite of algorithms which learn tasks simultaneously, as opposed to in isolation from each other. MTL can be applied to improve generalisation of tasks, hence the accuracy of predictions, and therefore it could be applied

[1]Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK
[2]Department of Engineering, University of Cambridge, United Kingdom, UK, CB3 0FA

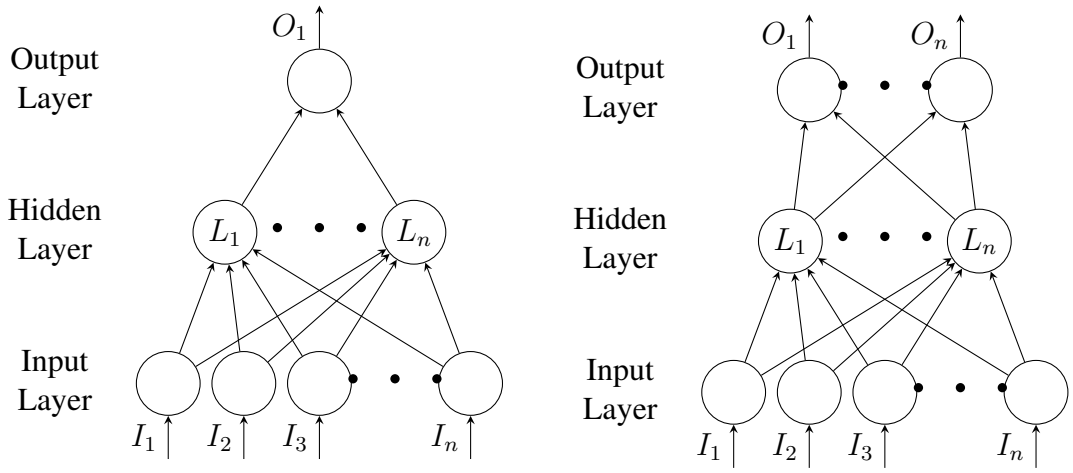

Figure 1. *(Left)* NN with single task. *(Right)* NN with multiple tasks.

to improve SHM systems. Caruana [3], developed one of the first forms of multi-task learning, a neural network (NN) with back propagation, to train tasks simultaneously and improve generalisation between tasks. Since the original work, multi-task learners have been applied to a lot of different machine-learning algorithms, from support vector machines [4] to decision trees [5, 6]. Within this paper, multi-task learning will be discussed purely in relation to NNs. Improved generalisation may improve accuracy of predictions across multiple tasks and hence could be beneficial to SHM systems.

The purpose of this paper is to provide a discussion of the applicability of the different types of multi-task learning problem settings with respect to the field of SHM. To distinguish between whether an NN is multi-task or not, a non-MTL NN will be referred to as an independent learner and a multi-task NN will be referred to as MTL.

BACKGROUND

It is useful to start with the architecture of a simple NN, as shown in Figure 1. Data entered via the input layer, where each node represents an individual feature/measurement that is added to the network. The hidden layer represents a *set* of latent features which are constructed from the measured features; there can be multiple hidden layers and hence multiple sets of latent features. Finally, the output layer generates the final output from the network.

NNs have been applied within the context of SHM in varying applications. Elkordy *et al.* [7] is an early example of the use of NNs in the context of SHM for damage detection. Manson *et al.* [8] and Mustapha *et al.* [9] each use a multi-layer perceptron NN to classify the location of damage, on an aircraft wing and isotropic plate, respectively. Since these early examples of the use of NNs in SHM, there has been a plethora of work utilising NNs, from using auto-associative NNs to detect damage before a crack is visibly seen [10], to using a convolutional NN to classify wind turbine tower vibration health states [11], a systematic review of the use of convolutional NNs in SHM is given in [12].

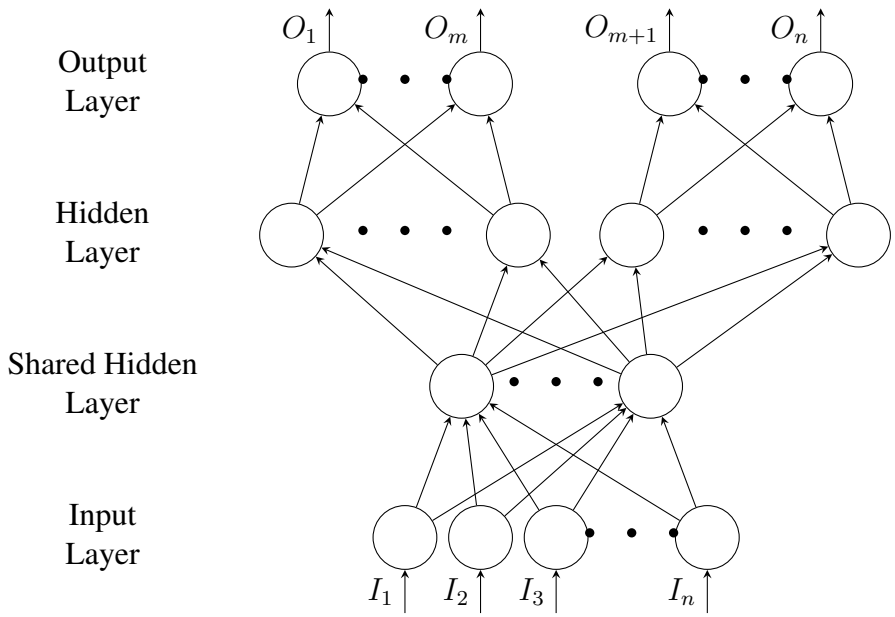

Figure 2. Example of a deep NN with both a shared hidden layer and split hidden layer.

The structure of an MTL NN is an extension of the independent NN, as shown by Figure 1. Just as with the independent formulations, the MTL NN has an input layer, hidden layers and an output layer. The main difference between the two networks in Figure 1 is the additional output nodes in the output layer. The additional output nodes represent different tasks, *i.e.* the multi-task nature of the network. The hidden layer could be more or less unchanged in the MTL setting, or it can take a very different structure. For a deep NN, the MTL NN may share all of the hidden layers or only several of the bottom hidden layers (an example of this structure is shown in figure 2). The input layer may have the same number of input features as with the independent learner; however, it may also have more input nodes to represent more features being added to the model.

The structure of the NN will be dependent on the mechanism of MTL which is being implemented. There are three main themes of multi-task learning discussed here which have been inspired by Caruana's paper on MTL learning [3], all which have applicability within SHM problem settings. Briefly these are:

- Natural occurrence of multiple tasks - additional tasks which make sense to learn together.
- Using outputs as "inputs" - adding data which cannot be added as an input as an output so that the data can influence the model. This can be used when the data is not accessible on the same time scale as the other data inputs.
- Additional loss functions to provide different insights - repeating the output of a NN but using a different loss function.

The following sections will detail each of the mechanisms and provide insight into potential problem settings of them.

PROBLEM SETTING 1: NATURAL OCCURRENCE OF MULTIPLE TASKS

Arguably, one of the most intuitive uses of MTL is when there is a *set* of tasks which are related and therefore make sense to learn together. Often the definition of MTL is given in relation to this problem setting, Zhang *et al.* [13] provides an overview of multitask learning, the paper defines the aims of MTL as: *to leverage useful information contained in multiple learning tasks to help learn a more accurate learner for each task*. For this problem setting of MTL, learning the tasks together provides a synergy that improves the performance of several tasks compared to if an independent learner was applied to each task individually. The structure of an MTL for this problem setting is given in Figure 3.

A example of the natural occurrence of multiple tasks in the context of SHM is the use of data from several nominally-identical structures to obtain information about their damage state, *e.g.* from wind turbines in a wind farm. With multiple structures, generalisation may be improved such that physical changes may be more likely to be identified.

Obtaining information from damage states can be expensive; however, if data from different structures are combined, then damage-state information can be leveraged *between structures* and provide increased information about the structures. One methodology of obtaining damage states for a structure is to synthesise it. Synthesised structures could be used within MTL to improve the performance of the actual structures which feature in the model. Synthesised data has the benefit of being significantly cheaper to obtain than the cost associated with providing damage to actual structures.

The majority of MTL which has been conducted in the field of SHM takes this form. In [14], a multi-task Gaussian process regression is used for missing sensor data reconstruction across a dam sensor network. The data are not missing simultaneously; however, data points were missing from different sensors in the network and by looking at the data together, the reconstruction performance improved. In [15], multi-task sparse Bayesian learning was conducted on two damaged structures with supplementary information provided via simulations of the structures. Learning the two structures together, the damage patterns are more reliably detected. Finally, [16] used an artificial neural network using data from six aluminium plates to predict fatigue crack length and remaining fatigue life. Although not explicitly called MTL in the paper, the neural network has two

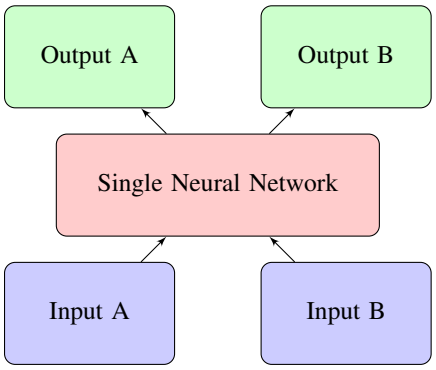

Figure 3. Problem setting 1: two tasks, A and B, put into one neural network.

output tasks which are learnt jointly.

A recent area of research is into population-based SHM (PBSHM) [2, 17–20], which aims to utilise how data can be transferred and shared between *populations* of structures to allow inferences to be shared across the population. An applicable example of a population of structures that may benefit from PBSHM is one of offshore wind farms. There now exist a lot of wind farms and hence a lot of structures which require SHM from a safety perspective, a cost perspective but also from an efficiency-of-power-generated perspective. This problem setting of MTL fits within the remit of PBSHM and could be explored within this context.

PROBLEM SETTING 2: USING OUTPUTS AS INPUTS

The second problem setting of MTL is using outputs as inputs, at first this may appear to be somewhat of an oxymoron; however, there are some features which are not available as inputs to a model, and therefore can only feature as outputs, see the structure in Figure 4. An output influences a NN during back-propagation, which occurs during training, by having an impact on the weights within the model. Hence, although added as an output, the additional task will influence the model and can be viewed as a form of input. There are several different, but linked, forms that this can take: transfer learning, non-operational features, and regression for classification. Each of the different forms is expanded below.

Certain problem settings of MTL may be labelled as *transfer learning*. Gardner *et al.* [21] broadly categorises transfer learning into two categories: training a model with data from an auxiliary task and fine tuning it with the main task data, and, performing domain adaptation such that two tasks can share a latent space. The latter category of transfer learning is applicable to SHM problem settings, with the aim to transfer knowledge between a *source* domain and a *target* domain [22]. Transfer learning does not require tasks to be learnt simultaneously; however, it is when tasks *are* learnt simultaneously that transfer learning is also multi-task learning.

An example of MTL transfer learning is given in [3], in order to predict the medical risk of a patient, there are several features that can be measured as inputs (*e.g.* height),

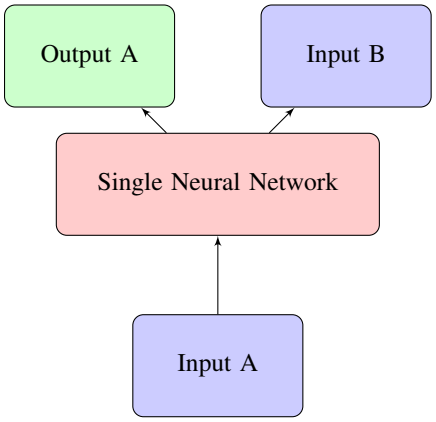

Figure 4. Problem setting 2: Using additional inputs as an output.

however, there also may be medical test results which would be a useful feature to add to the model. Medical test results take time to process and may not be available as an input for all patients. Therefore, the test results can be added as an output to the model and form an auxiliary task. During training, the test results help to inform the medical risk. When tested, the target output is the risk and this can be calculated from the input features which are available. In this example information from the medical test results has been transferred to inform the medical risk. A potential synonymous example for SHM is using the results of model analysis as an additional output for a neural network which is used for system identification.

The examples above are also useful when considering the use of non-operational features during training. A further problem setting of using outputs as inputs to consider features that are available during training (which is likely offline, such as modal analysis), in comparison to features which are available during normal operation. The non-operational features that are available during training can be added as outputs to a NN which could improve the generalisation of the model and therefore improve the performance of the model during online utilisation.

The final category for the problem setting of using outputs as inputs is using additional regression outputs to inform a classification task. For classification tasks, the classification is either True or False, and hence, there is a level of quantisation. Whereas regression tasks are on a sliding scale which can contain a lot more information than simply: True or False. Hence, there may be additional information in a regression task that would inform a classification task. It is a method of using a larger continuous space whilst solving a discrete problem. Additional classification tasks can also be used to improve the main classification task as the quantisation may be different to the original task.

There is little research for this problem setting of MTL within the context of SHM. Aforementioned, there are strong links with transfer learning and there are promising problem settings of using outputs as inputs within the context of SHM.

PROBLEM SETTING 3: ADDITIONAL LOSS FUNCTIONS TO PROVIDE DIFFERENT INSIGHTS

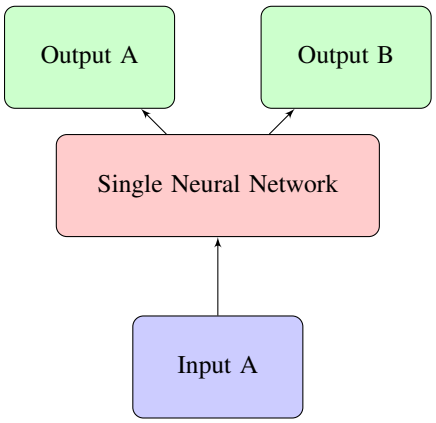

Figure 5. Problem setting 3: Additional outputs for a NN.

When constructing a NN with a single task, the developer has to choose a loss function with which to train the network. During training, the weights and biases of the NN will be tuned to obtain the best performance of the NN, which is determined by minimising the loss function. Different loss functions may have affinities to different values of the weights and biases. Hence, the solution of the NN for different loss functions could provide different solutions and ould provide different insights from the network, even though the inputs into the network are exactly the same.

This problem setting of MTL duplicates outputs but uses different loss functions for each. Caruana [23] demonstrates how the rankprop error metric performs poorly at the lower end of a continuous spectrum. Following this, an MTL NN with both the rankprop error metric and sum-of-square errors metric is used to improve the performance of the main task at the lower end of the spectrum. Adding the additional loss function, in this case, improved performance at a critical end of a spectrum.

A potential example of how this could be implemented within the field of SHM is in the context of fatigue testing. There are two parameters which would be of interest in fatigue testing: the number of cycles and the amplitude of the force experienced. To gauge the overall time signal, an L2-norm might be chosen; overall, it is anticipated that this would give a reasonable average response; however, it might not be sensitive to accurately detect the time when the loading switches from compression to tension. For this related task, a different regularisation parameter may be used, which focuses on when the response passes through 0. To understand the maximum load at the peak tension/compression, an L4-norm could be applied, which is useful at modelling extremities. As different regularisation may be used for different interpretations of the data, this is a good example of where MTL could improve the performance jointly over a range of tasks.

This problem setting could have interesting applications within SHM.

CONCLUDING REMARKS

This paper discusses the different uses of MTL NNs and how then may be applicable in the field of SHM. MTL with multiple tasks arising naturally is the most explored of the problem settings, however, to date, there is still limited work using this approach. Transfer learning has been explored within PBSHM but the intersect of transfer learning and MTL is yet to be explored with regards to NNs. Arguably the least explored problem setting of MTL is the use of additional loss functions to provide different insights. As the cost of error in SHM can be quite high this problem setting could be very beneficial for improving prediction accuracy in the field. Overall there is a lot of potential for the use of MTL NNs in SHM, which can be researched.

ACKNOWLEDGMENT

This research was supported by grants from the Engineering and Physical Sciences Research Council (EPSRC), UK, and Natural Environment Research Council, UK via grant number, EP/S023763/1 and EP/W005816/1 - Revolutionising Operational Safety

and Economy for High-value Infrastructure using Population-based SHM (ROSEHIPS).

REFERENCES

1. Farrar, C. R. and K. Worden. 2013. *Structural Health Monitoring: A Machine Learning Perspective*, John Wiley & Sons, Ltd.
2. Bull, L. A., P. A. Gardner, J. Gosliga, T. J. Rogers, N. Dervilis, E. J. Cross, E. Papatheou, A. E. Maguire, C. Campos, and K. Worden. 2021. "Foundations of population-based SHM, Part I: Homogeneous populations and forms," *Mechanical Systems and Signal Processing*, 148:107141.
3. Caruana, R. 1997. "Multitask Learning," *Machine Learning*, 28:41–75.
4. Shiao, H.-T. and V. Cherkassky. 2012. "Implementation and Comparison of SVM-Based Multi-Task Learning Methods," IEEE.
5. Wang, Q., L. Zhang, C. Mingmin;, and J. Guo. 2008. "MTForest: Ensemble Decision Trees based on Multi-Task Learning," IOS Press, p. 941.
6. Chapelle, O., P. Shivaswamy, S. Vadrevu, K. Weinberger, Y. Zhang, and B. Tseng. 2011. "Boosted multi-task learning," *Machine Learning*, 85:149–173.
7. Elkordy, M. F., K. C. Chang, and G. C. Lee. 1994. "A Structural Damage Neural Network Monitoring System," *Computer-Aided Civil and Infrastructure Engineering*, 9:83–96.
8. Manson, G., K. Worden, and D. Allman. 2003. "Experimental validation of a structural health monitoring methodology: Part III. Damage location on an aircraft wing," *Journal of Sound and Vibration*, 259:365–385.
9. Mustapha, F., G. Manson, K. Worden, and S. G. Pierce. 2007. "Damage location in an isotropic plate using a vector of novelty indices," *Mechanical Systems and Signal Processing*, 21:1885–1906.
10. Dervilis, N., M. Choi, S. G. Taylor, R. J. Barthorpe, G. Park, C. R. Farrar, and K. Worden. 2014. "On damage diagnosis for a wind turbine blade using pattern recognition," *Journal of Sound and Vibration*, 333:1833–1850.
11. Khazaee, M., P. Derian, and A. Mouraud. 2022. "A comprehensive study on Structural Health Monitoring (SHM) of wind turbine blades by instrumenting tower using machine learning methods," *Renewable Energy*, 199:1568–1579.
12. Sony, S., K. Dunphy, A. Sadhu, and M. Capretz. 2021. "A systematic review of convolutional neural network-based structural condition assessment techniques," *Engineering Structures*, 226:111347.
13. Zhang, Y. and Q. Yang. 2018. "An overview of multi-task learning," *National Science Review*, 5:30–43.
14. Li, Y., T. Bao, Z. Chen, Z. Gao, X. Shu, and K. Zhang. 2021. "A missing sensor measurement data reconstruction framework powered by multi-task Gaussian process regression for dam structural health monitoring systems," *Measurement: Journal of the International Measurement Confederation*, 186.
15. Huang, Y., J. L. Beck, and H. Li. 2019. "Multitask Sparse Bayesian Learning with Applications in Structural Health Monitoring," *Computer-Aided Civil and Infrastructure Engineering*, 34:732–754.
16. Lim, H. J., H. Sohn, and Y. Kim. 2018. "Data-driven fatigue crack quantification and prognosis using nonlinear ultrasonic modulation," *Mechanical Systems and Signal Processing*, 109:185–195.
17. Gosliga, J., P. A. Gardner, L. A. Bull, N. Dervilis, and K. Worden. 2021. "Foundations of Population-based SHM, Part II: Heterogeneous populations – Graphs, networks, and com-

munities," *Mechanical Systems and Signal Processing*, 148:107144.

18. Gardner, P., L. A. Bull, J. Gosliga, N. Dervilis, and K. Worden. 2021. "Foundations of population-based SHM, Part III: Heterogeneous populations – Mapping and transfer," *Mechanical Systems and Signal Processing*, 149:107142.
19. Tsialiamanis, G., C. Mylonas, E. Chatzi, N. Dervilis, D. J. Wagg, and K. Worden. 2021. "Foundations of population-based SHM, Part IV: The geometry of spaces of structures and their feature spaces," *Mechanical Systems and Signal Processing*, 157:107692.
20. Bull, L. A., D. D. Francesco, M. Dhada, O. Steinert, T. Lindgren, A. K. Parlikad, A. B. Duncan, and M. Girolami. 2023. "Hierarchical Bayesian modeling for knowledge transfer across engineering fleets via multitask learning," *Computer-Aided Civil and Infrastructure Engineering*, 38:821–848.
21. Gardner, P., L. A. Bull, N. Dervilis, and K. Worden. 2022. "On the application of kernelised Bayesian transfer learning to population-based structural health monitoring," *Mechanical Systems and Signal Processing*, 167:108519.
22. Gardner, P., X. Liu, and K. Worden. 2020. "On the application of domain adaptation in structural health monitoring," *Mechanical Systems and Signal Processing*, 138:106550.
23. Caruana, R. 1998, "A Dozen Tricks with Multitask Learning," .