A Movement in Multiple Time Neural Network for Automatic Detection of Pain Behaviour

Temitayo Olugbade
temitayo.olugbade.13@ucl.ac.uk
University College London

Nicolas Gold
n.gold@ucl.ac.uk
University College London

Amanda C de C Williams
amanda.williams@ucl.ac.uk
University College London

Nadia Bianchi-Berthouze
nadia.berthouze@ucl.ac.uk
University College London

ABSTRACT

The use of multiple clocks has been a favoured approach to modelling the multiple timescales of sequential data. Previous work based on clocks and multi-timescale studies in general have not clearly accounted for multidimensionality of data such that each dimension has its own timescale(s). Focusing on body movement data which has independent yet coordinating degrees of freedom, we propose a Movement in Multiple Time (MiMT) neural network. Our MiMT models multiple timescales by learning different levels of movement interpretation (i.e. labels) and further allows for separate timescales across movements dimensions. We obtain 0.75 and 0.58 average F1 scores respectively for binary frame-level and three-class window-level classification of pain behaviour based on the MiMT. Findings in ablation studies suggest that these two elements of the MiMT are valuable to modelling multiple timescales of multidimensional sequential data.

CCS CONCEPTS

• **Computing methodologies → Neural networks**.

KEYWORDS

time, multiple timescales, neural networks, body movement, pain

ACM Reference Format:
Temitayo Olugbade, Nicolas Gold, Amanda C de C Williams, and Nadia Bianchi-Berthouze. 2020. A Movement in Multiple Time Neural Network for Automatic Detection of Pain Behaviour. In *Companion Publication of the 2020 International Conference on Multimodal Interaction (ICMI '20 Companion), October 25–29, 2020, Virtual event, Netherlands.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3395035.3425969

1 INTRODUCTION

A complete movement, e.g. sit-to-stand, can be defined in terms of layers of sequences of events (such as trunk flexion, vertical hip displacement) with each event involving different (sets of) anatomical segments [5]. Due to changes in goal and/or attention, an observer may group events in the movement differently at each repetition of observation [16]. This multiple timeline of movement both at the points of performance and interpretation motivates this paper on movement encoding and classification at multiple timescales.

We take the problem of automatic detection of bodily-expressed pain behaviours (e.g. stiffness in movement, bracing) [4] as a case study. Pain behaviour detection could be a valuable capability for technology for chronic pain physical rehabilitation as it would enable the technology to call the attention of a person with pain to their unhelpful strategies for performing feared or painful movements [10]. These strategies can make the given movement more challenging, cause non-functional muscle tension, and reinforce avoidance of the use of the painful location [10]. Previous bodily-expressed pain behaviour detection studies have focused on classification at single timescales. For example, in [1] the authors modelled the duration (as a proportion) of pain behaviours in a movement instance while [14] modelled the presence/absence of pain behaviours in fixed window segments of movement instances.

We hypothesize that learning multiple timescales of a pain behaviour label will improve automatic detection of the label at each of the timescales. Further, addressing the earlier-discussed multiple layers of movement by different anatomical segments could additionally increase performance. Thus, we propose the *Movement in Multiple Time* (MiMT) neural network architecture characterised by a distributed time encoding of low level movement features and a joint prediction of pain behaviour at multiple timescales.

In the rest of the paper, we discuss related work and describe our MiMT architecture against this background. We then report its performance on a real dataset of exercise movements of people with chronic pain and results of ablation studies which highlight the value of each of the two main components of the MiMT. We review previous studies in Section 2 and present our architecture in Section 3. The experiments on the MiMT network and their results are discussed in Sections 4 and 5 with a conclusion in Section 6.

2 RELATED WORKS

One of the earliest studies on modelling multiple timescales is the work of [15] who proposed the multiple timescales recurrent neural network (MTRNN) [11]. The MTRNN is a continuous time recurrent neural network with at least two sets of hidden layers, the first of which has a short time delay (e.g. $\tau = 5$) while each additional layer has a longer latency, e.g. $\tau = 70$. Analysis of the activation values at each layer in a movement forecast task suggests that indeed the shorter-time-delay layer(s) learn to encode movement

ICMI '20 Companion, October 25–29, 2020, Virtual event, Netherlands
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8002-7/20/10…$15.00
https://doi.org/10.1145/3395035.3425969

primitives while the longer-time-delay layer(s) encode a higher level of abstraction of the movement. They additionally found in a movement generation task, that error increased the closer to 1 the ratio between the time delays of the two sets of layers, highlighting the importance of learning at multiple time resolutions.

The clockwork recurrent neural network (CWRNN) of [6] takes a similar approach of multiple modules with different update speeds. However, their multiple scales of time are implemented within a single layer. The units in the layer are sectioned into N blocks each with its own speed, and a connection between blocks goes from the slower to the faster. Further, for each time t, an update in a block only occurs if t is divisible by its speed τ. In experiments on audio sequence generation and classification of aural words, with the speed for each block $n \in N$ set to 2^{n-1}, the CWRNN was found to perform better than the long short-term memory neural network (LSTMNN) in learning long sequences.

In the hierarchical multiscale RNN (HM-RNN) [2], each layer's update behaviour depends on its input instead of fixed clocks. For example, a layer resets its state if it detected a boundary in the previous time step; before a reset, the given layer l first passes its output to the layer $l+1$ above it. Boundary detection is learnt during training. The same layer updates its state if instead the layer below it detected a boundary at the previous time step. Otherwise, the layer simply copies its states from the previous time step. The output of all the recurrent layers are then collectively input into further modules. Analysis of the boundaries detected in the HM-RNN for character language modelling showed that while the first LSTM layer learnt to detect a boundary at the start of every word, the second LSTM layer usually detected at the end of multiple words.

Although [7] used the more common fixed clock scheme, with their convolutional multitimescale echo state network (ConvMESN) the recurrent layers are connected in parallel rather than serially; the weights of the recurrent layers are fixed during training. Further, time-dimension convolutions of multiple kernel sizes are applied to the output of each recurrent layer. These convolution outputs are then max-pooled across time and concatenated for each recurrent layer and finally altogether fused using a fully-connected layer. Similar to the CW-RNN [6], the ConvMESN uses a time delay rule of B^{i-1}, $B \in \mathbb{N}$, $B > 1$ for each recurrent layer i. The architecture was explored on human action recognition tasks and the authors found that the ConvMESN performed better than a similar model but with the same time delay across all recurrent layers. Visualisations of the activations of each recurrent unit indeed suggest that the ConvMESN encodes different timescales of movement.

The idea of imposed time delays in recurrent updates are not at all used in the ENHAnCE algorithm of [8]. Instead, their architecture consists of multiple RNNs trained additively. The first is initially trained as a simple, forecasting RNN which is then upgraded to a gated RNN and retrained based on reinforcement learning. The input into the second is a cluster of the embeddings from this gated RNN. The second RNN is trained in a similar manner to the first, and so on until the last RNN in the hierarchy.

While the approaches used in above-discussed studies have proven valuable, there are two aspects of multiple timescales in sequential data that are overlooked by these studies. In this paper, we present our *Movement in Multiple* (MiMT) architecture which is made up of two main components that respectively address these two problems. The first problem is the multiple timescales of description (by an observer) of sequential data. For a simple illustration, consider an observer who provides a single description l for a period t_1 to t_T. Each time step t_i in the given period shares the description l. Conversely, the observer could instead provide a description for each time step t_i and a holistic description then derived for any subset of the period. The findings in the previous studies suggest that jointly learning these multiple timescales of descriptions can improve learning performance for each description timescale. Thus, the MiMT models the same (pain behaviour) label at multiple timescales. For human movement, the second problem is that there exists an additional time structure such that each set of anatomical segments can act in its own timeline in addition to a common timeline that allows these sets of segments to act together when appropriate. For example, I can write with one hand while the other scratches my face (unconnected simultaneous events), my hands could come together in a clap (coordinated simultaneous events), or they could both be typing on a keyboard (coordinated serial events). To further account for this independence-cum-coordination between groups of segments, time encodings for segments of the human body are separated but connected in our MiMT architecture. We describe the network in the next section.

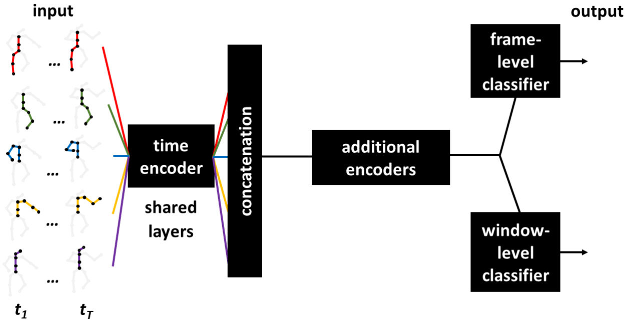

Figure 1: Movement in Multiple Time (MiMT) architecture. Best viewed in colour.

3 MOVEMENT IN MULTIPLE TIME (MIMT) ARCHITECTURE

An overview of the *Movement in Multiple Time* (MiMT) architecture is shown in Figure 1. It has two main features. First, it computes time encodings separately for different groups of anatomical segments but using a shared encoder. Let us consider a human movement specified by $X = [X_1, X_2, ..., X_T]$ where $X_t = \{X_t^1, X_t^2, ..., X_t^J\}$, J is the number of anatomical segments, and T is the duration of the movement. In the MiMT architecture, a common recurrent encoding f is applied to each x where $x_t \subset X_t$, $x_t \neq \emptyset$, and there exists a path in x_t between any pair (x_t^u, x_t^v) in x_t. For example, given 3D full-body joints positions over time as X, we can define five xs, $x_t \subset X_t\ \forall x$, as shown in Figure 1: right lower limb and torso, left lower limb and torso, right upper limb and torso, left upper limb and torso, head and torso. This is the anatomical segments grouping which we used in our implementation of the MiMT architecture in this paper. We use LSTM layers as the time encoder.

The other main element of the MiMT network is its multiple outputs $Y^1, Y^2, ..., Y^N$ at different timescales for the same label Y, i.e. $Y^m = [Y^m_1, Y^m_2, ..., Y^m_{\tau_m}]$, $\tau_i \neq \tau_k \forall i, k \leq N$ and $\tau_m \geq 1$. In the example shown in Figure 1, N=2 with $\tau_1 = T$ and $\tau_2 = 1$. $Y^1, Y^2, ..., Y^N$ are simultaneously learnt in a multi-task learning structure and based on a concatenation of all $f(x)$ $\forall x$. This is processed by additional modules before then being passed to each of the N classifiers (or decoders or regressor). In our use of the MiMT in the experiments reported in the next sections, we use an attention mechanism that borrows from the transformer model of [12] for further encoding. The maximum across activations of three attention heads were additionally encoded by a LSTM layer followed by a fully connected layer shared across time. The corresponding embedding becomes an input into each of a frame-level ($\tau_m = T$) and window-level ($\tau_m = 1$) classifier. For the frame level, we use a sigmoid classifier, after pooling. For the window level, the embedding is multiplied by the concatenated activations of the time encoder which are then passed through a final LSTM and an additional softmax activation.

4 EXPERIMENTS

We explored the MiMT neural network on pain behaviour detection based on the EmoPain dataset [1] which contains 3D positions for 26 full-body joints of people with chronic pain performing exercise movements (e.g. sit-to-stand, bend) and guarding behaviour annotations (guarding present/positive or absent/negative) provided in continuous time by each of 4 clinicians. We excluded eight of the 26 joints (left and right fingertips, ankles, heels, and toes) due to the higher level of noise in their position estimates. Since the remaining joints already include the head and neck, we additionally excluded the crown. This resulted in 17 full-body joints.

We created data instances based on overlapping windows of length of 180 frames (3 seconds) [13] and overlap of 15 frames. To obtain the ground truth for each frame, we took the majority vote across raters and frames with tied votes resolved as having positive label. The window-level label was set as negative if all the frames in the window were negative for guarding, positive if all frames in the window were positive, and mixed otherwise. To manage class imbalance in the training and validation sets, we used data augmentation methods similar to [9] to randomly oversample minority classes. This resulted in 17,185 and 1,394 instances respectively.

Each LSTM in the shared time encoder of the MiMT was set up to have 3 units in line with the dimensions for each joint position. The LSTM and fully connected layer immediately after the attention layers had 15 units, the same size as the channel dimension of their inputs. We trained the network using Adam optimizer and set learning rate and batch size to 0.005 and 200 respectively.

We discuss results based on hold-out validation (with disjoint subject sets in training, validation, and test sets) in the next section.

5 RESULTS AND DISCUSSION

5.1 Performance of the MiMT Model

The F1 scores for each class and the confusion matrices for the unbalanced test set are shown in Tables 1 and 2 respectively.

As can be seen from the confusion matrices, the true positives for the negative and positive classes are better than chance level for both frame and window level classification. In the window level classification, the mixed class is unsurprisingly strongly confused with the other two classes. The two tables further show the level of imbalance in size between the classes, with the negative class having at least 6 times more number of instances at the frame level than the positive class and more than twice and 9 times more instances at the window level than the positive and mixed classes respectively. This high level of imbalance makes it challenging to interpret the results and understand the performance of the MiMT.

To address this, we follow the recommendation in [3] by further reporting performance using test sets balanced with the same data-augmentation-based oversampling technique that we used for the training and validation sets. In Table 1 we present the average result based on 5 of such balanced test sets. There is little variation in the performance across the 5 sets, with less than 0.008 standard deviation in F1 score per class. The results show performances much higher than chance (i.e. F1 score = 0.5) for the frame level classification with average F1 score across the two classes = 0.75. For the window level classification, the average F1 score across the classes is 0.58 also much better than chance (F1 score = 0.33 in this case). In this timescale, while the performance for the negative and mixed classes are each much better than chance, the performance for the positive class is only slightly above chance. We found that the highest confusion (61%) for the positive class is with the mixed class. Interestingly, with the unbalanced test set, this class (the positive class) performs much better than the base rate in that set. It is unclear what this finding implies, but in the envisioned use of the pain behaviour detection model, confusion between the mixed and positive classes is acceptable as it is sufficient to be able know when there has at all been a positive label in a given window.

5.2 Ablation Studies on the MiMT Architecture

To understand the influence of the 2 primary characteristics of the MiMT (separated but shared time encoding, and multiple timescales of the same label) on its performance, we conducted 2 ablation studies. In the first study (MiMT with single input time), the only change to the MiMT was that the time encoder in the architecture was applied to X as a single input rather than the distributed $x \subset X$ $\forall x$. In the second part of the study, the only change to the MiMT was that only one timescale of the label was learnt at a time rather than jointly in a multitask scheme (MiMT with frame time output only and MiMT with window time output only).

As it can be seen in Table 3, assuming a single time line for all anatomical segments (MiMT with single input time) reduces frame level classification performance for each class and the overall window level classification performance although the mixed class is better detected. For the MiMT with frame time output only and the MiMT with window time output only, performance is also reduced per class for frame and window level classification respectively. These findings, which disregard the number of trainable parameters in the network, suggest that having a separate but shared time encoding across groups of anatomical segments as well as multiple timescales of output for the same label are indeed valuable.

Table 1: Pain Behaviour Classification Results based on our MiMT Architecture. The data size (and corresponding proportion) are shown for each class and each classification task, to the neareast thousand for the frame level classification. For the results based on the balanced test set, the average F1 score (and standard deviation) over 5 runs is given.

Class	Unbalanced Test Set				Balanced Test Set			
	Frame level		Window level		Frame level		Window level	
	F1 score	Data size	F1 score	Data size	F1 score	Data size	F1 score	Data size
Negative	0.85	620,000 (0.87)	0.89	3,289 (0.83)	0.70 (0.001)	854,000 (0.48)	0.85 (0.001)	3,289 (0.33)
Mixed	-	-	0.098	371 (0.094)	-	-	0.52 (0.002)	3,283 (0.33)
Positive	0.41	90,000 (0.13)	0.39	291 (0.074)	0.80 (0.001)	920,000 (0.52)	0.37 (0.008)	3,283 (0.33)

Table 2: Confusion Matrices for Pain Behaviour Classification based on Our MiMT Architecture (Unbalanced Test Set)

Window level				
		MiMT		
	Class	Negative	Mixed	Positive
Ground Truth	Negative	2864 (87%)	153 (5%)	272 (8%)
	Mixed	192 (52%)	29 (8%)	150 (40%)
	Positive	83 (29%)	36 (12%)	172 (59%)
Frame level				
		MiMT		
	Class	Negative		Positive
Ground Truth	Negative	484057 (78%)		137491 (22%)
	Positive	30180 (34%)		59452 (66%)

Table 3: Ablation Study Results based (Unbalanced Test Set). The best results per class per classification level in bold.

Architecture (No. of parameters)	Class	F1 score Frame	F1 score Window
MiMT with single input time (8,346)	Negative	0.72	0.71
	Mixed	-	**0.27**
	Positive	0.27	0.098
	Average	0.50	0.34
MiMT with frame time output only (810)	Negative	0.80	-
	Mixed	-	-
	Positive	0.38	-
	Average	0.59	-
MiMT with window time output only (1,038)	Negative	-	0.75
	Mixed	-	0.077
	Positive	-	0.16
	Average	-	0.33
MiMT (1,038)	Negative	**0.85**	**0.89**
	Mixed	-	0.098
	Positive	**0.41**	**0.39**
	Average	**0.63**	**0.46**

6 CONCLUSION

The introduction of multiple but shared timelines across groups of anatomical segments with multiple output timescales seem to improve automatic detection of movement behaviour in each of those timescales. This MiMT architecture can be easily integrated in any of the clocks-based networks discussed in Section 2.

ACKNOWLEDGMENTS

The project was supported by the EU Future and Emerging Technologies Proactive Programme H2020 (Grant No. 824160-EnTimeMent).

REFERENCES

[1] Min Aung, Sebastian Kaltwang, Bernardino Romera-Paredes, Brais Martinez, Aneesha Singh, Matteo Cella, et al. 2016. The automatic detection of chronic pain-related expression: requirements, challenges and the multimodal EmoPain dataset. *IEEE Transactions on Affective Computing* 7, 4 (2016), 435–451.

[2] Junyoung Chung, Sungjin Ahn, and Yoshua Bengio. 2019. Hierarchical multiscale recurrent neural networks. In *Proceedings of ICML*. 1–13.

[3] László A. Jeni, Jeffrey F. Cohn, and Fernando De La Torre. 2013. Facing imbalanced data - Recommendations for the use of performance metrics. In *Proceedings of Conference on Affective Computing and Intelligent Interaction*. IEEE, 245–251.

[4] Francis Keefe and Andrew Block. 1982. Development of an observation method for assessing pain behavior in chronic low back pain patients. *Behavior Therapy* 13, 4 (1982), 363–375.

[5] KM Kerr, JA White, DA Barr, and RAB Mollan. 1994. Standardization and definitions of the sit-stand-sit movement cycle. *Gait and Posture* 2, 3 (1994), 182–190.

[6] Jan Koutník, Klaus Greff, Faustino Gomez, and Jürgen Schmidhuber. 2014. A clockwork RNN. In *International Conference on Machine Learning*. 1863–1871.

[7] Qianli Ma, Enhuan Chen, Zhenxi Lin, Jiangyue Yan, Zhiwen Yu, and Wing Ng. 2019. Convolutional Multitimescale Echo State Network. *IEEE Trans Cybern* (2019), 1–13.

[8] Katherine Metcalf and David Leake. 2019. Unsupervised hierarchical temporal abstraction by simultaneously learning expectations and representations. In *Proceedings of International Joint Conference on Artificial Intelligence*. 3144–3150.

[9] Temitayo Olugbade, Joseph Newbold, Rose Johnson, Erica Volta, Paolo Alborno, Radoslaw Niewiadomski, et al. 2020. Automatic Detection of Reflective Thinking in Mathematical Problem Solving based on Unconstrained Bodily Exploration. *IEEE Transactions on Affective Computing* (2020).

[10] Temitayo A Olugbade, Aneesha Singh, Nadia Bianchi-Berthouze, Nicolai Marquardt, Min SH Aung, and Amanda C De C Williams. 2019. How can affect be detected and represented in technological support for physical rehabilitation? *ACM Transactions on Computer-Human Interaction* 26, 1 (2019), 1–29.

[11] Martin Peniak, Davide Marocco, Jun Tani, Yuichi Yamashita, Kerstin Ficher, and Angelo Cangelosi. 2011. Multiple Time Scales Recurrent Neural Network for Complex Action Acquisition. *Frontiers in Computational Neuroscience* 5 (2011).

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uskoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need Ashish. In *Proceedings of NIPS*. 5998–6008.

[13] Chongyang Wang, Temitayo Olugbade, Akhil Mathur, Amanda C De C. Williams, Nicholas Lane, and Nadia Bianchi-Berthouze. 2019. Recurrent network based automatic detection of chronic pain protective behavior using mocap and semg data. In *Proceedings of International Symposium on Wearable Computers*. 225–230.

[14] Chongyang Wang, Min Peng, Temitayo Olugbade, Nicholas Lane, Amanda C de C Williams, and Nadia Bianchi-Berthouze. 2019. Learning temporal and bodily attention in protective movement behavior detection. In *Proceedings of Affective Computing and Intelligent Interaction Workshops and Demos*. 324–330.

[15] Yuichi Yamashita and Jun Tani. 2008. Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. *PLoS Computational Biology* 4 (2008).

[16] Jeffrey Zacks and Khena Swallow. 2007. Event segmentation. *Current Directions in Psychological Science* 16, 2 (2007), 80–84.