Exploring the Effect of Sampling Strategy on Movement Generation with Generative Neural Networks

Benedikte Wallace
benediwa@ifi.uio.no
RITMO, University of Oslo
Oslo, Norway

Charles P. Martin
Australian National University
Canberra, Australia

Jim Tøressen
RITMO, University of Oslo
Oslo, Norway

Kristian Nymoen
RITMO, University of Oslo
Oslo, Norway

Figure 1: Dance improvisation

ABSTRACT

It is common to explore multiple sampling approaches when generating output from generative deep neural networks for creative applications. Choosing suitable sampling parameters can make or break the realism and perceived creative merit of the output. This is particularly important when attempting to simulate variations of expressive human movement. The process of selecting the correct sampling parameters is often task-specific and under-reported in many publications, which can make the reproducibility of the results challenging. We explore some of the most common sampling techniques in the context of generating human body movement, specifically dance movement, and attempt to shine a light on their advantages and limitations. This work presents a Mixture Density Recurrent Neural Network trained on a dataset of improvised dance motion capture data from which it is possible to generate novel movement sequences. Systematically examining the different sampling strategies allows us to further the understanding of how the sampling parameters affect motion generation, which provides evidence for utility in creative applications.

KEYWORDS

Mixture Density Networks, Movement Generation, Generative Networks, Creative Prediction

1 INTRODUCTION

In this paper, we will outline several common sampling strategies for Mixture Density Recurrent Neural Networks (MDRNNs). We will show that the choice of sampling strategy significantly affects

the output of the model. Building an understanding of these effects could aid when deciding between different approaches in generation of dance motion and other applications.

Expressive movement is an intrinsic part of human life. Hand gestures during speech, body language as well as dance have the ability to efficiently convey an emotional state. Based solely on simple movement patterns such as gait or arm movement it is possible to classify characteristics like gender, personality and mood [15, 18, 20]. As such, motion analysis and generation have use cases in fields such as human-robot interaction, human activity recognition and artificial agent design [12].

While movement generation may involve all of the movement types mentioned above and more, this paper presents a system for generating dance movement. Such a system may be used as an artistic tool for choreographers, or to generate dance movement in video games or animation. In the former, the system must be able to generate novel and varied outputs, while for animation purposes realism may be a more important property than novelty.

For any predictive system, a loss function would typically include some metric indicating the distance from the model output to some ground truth. This approach produces good results when generating basic human motion such as gait [1]. When generating dance movements, or indeed any creative data, there is no single correct answer or ground truth to which the output can be compared as we may consider several predictions equally valid. Thus, rather than attempting to predict a single truth a good model for generating creative data should be able to generate a variety of likely outputs. A common way to approach this problem is to experiment with sampling from the probability distribution learned by the model

with more or less “randomness”, often referred to as stochastic sampling, until the output is found to be satisfactory [6–8, 14]. This is usually performed by including a temperature parameter which is used to reweight the probability distribution. Reweighting the learned distribution allows the model to generate less likely predictions some of the time. If the model consistently outputs the most likely prediction, the output can become repetitive. Conversely, a flat probability distribution where all outcomes are equally likely would be equivalent to sampling at random, invariant to the learned distribution. Allowing for occasional unexpected predictions by implementing stochastic sampling, will in many cases improve the realism of the output and elicit more interesting predictions thereby improving the perceived creative capacity of the model. Another interesting approach to vary the output of a trained model is to prime the model on different unseen movement sequences. Priming consists of running an unseen sequence through the model before generating a prediction for the next time step. [7] demonstrated interesting results using this strategy, and generated handwriting examples in various styles by priming a model on examples by different subjects. With our priming experiments, we aim to examine to what extent the results obtained for handwriting are transferable to full-body motion.

A third type of sampling strategy is to allow sampling only from part of the trained model. For example mixture density networks, learn mixture distributions consisting of several components. These may be constrained by only allowing sampling from a certain mixture component. The idea has been explored by [6] in the context of world models [9], finding evidence to support the theory that each component will model different stochastic events.

In this work, we look at creativity in movement generation, specifically dance movement. We investigate the effect of different sampling strategies for movement generated by a mixture density recurrent neural network. MDRNNs are becoming well-established tools in the generation of creative data. They have previously been applied to musical sketches in two dimensions as part of a smart-phone app [14], to sketches [8], and handwriting [7]. MDRNNs have also previously been applied to motion capture data [5, 17], which show promising results. While variation is a central aspect of generating creative data, there has been little focus on the sampling strategies and how they affect variation in the generated data. We aim to systematically explore the effect of applying the three sampling strategies outlined above: adjusting temperature, priming, and isolating mixture components. The following sections will describe the model and motion capture recordings. The final sections present the results of sampling from the trained MDRNN. The code used to sample from the trained model has been made available in the interest of reproducibility.

2 QUANTIFYING MOVEMENT

Movement of a human body may be represented as real-valued multidimensional time-series data unfolding in space and time. High-precision recordings of movement may be recorded with marker-based optical motion capture technology, where each marker put on the body provides a 3D position vector over time. As mentioned, dance is a particularly complex variant of human movement, with subtle features which can communicate important information. As

Figure 2: Marker setup worn by the dancers during motion capture data collection

such our network must be able to retain the nuances in our dance recordings as real-valued time-series. While other creative data sources such as music, text and images can be simplified into a discrete representation, it is clear that doing so would severely reduce the expressiveness of dance data.

In examining the output of a model which generates creative data the results are often evaluated according to whether or not they are typical examples of their type, be it a piece of music perhaps belonging to a specific genre [21], or a painting in a certain style [23], or in our case a realistic sequence of movement. Realism is naturally a quality that has some degree of importance when generating any type of data, but perhaps especially so for movement; any deviation from what is physically possible for a human to achieve is instantly recognisable. Another important aspect to consider is whether the model is able to produce novel output, different from the examples used in training.

In order to determine the novelty or realism of a generated motion sequence, it is useful to define a notion of similarity. However, the seemingly simple concept of movement similarity becomes quite complicated. In many cases, it may be sufficient to consider motions as similar if they only differ with respect to global rotation or spatial and temporal scaling [16], but for the purpose of comparing dance, such a definition may be unsatisfactory. Consider, for example, one dance sequence where the left arm is raised, and another the right arm. The semantic similarity between the two would not be reflected in metrics like the euclidean distance between global positions or between joint angles in consecutive frames. Further exploration of movement similarity is outside the scope of this article. Instead of comparing generated motion sequences to the motion capture recordings, we will use the terms novelty and realism in reference to the conceptual class of artefacts that this data belongs to, namely dance, and evaluate them by visual inspection.

3 DATASET

Our dataset contains 54 one minute motion capture recordings of improvised dance performed by three female dancers. Their average age is 23 and each has more than 10 years experience within modern, jazz and ballet. Each dancer performs three one

Figure 3: The 43 reflective markers are translated to 22 points

minute improvisations to six different musical stimuli. The dataset was recorded using a Qualisys optical motion capture system with 12 Oqus 300/400 series cameras which capture 43 reflective markers worn by the dancers. Figure 2 shows the front and back of a dancer wearing the motion capture suit with markers. Figure 3 shows how the 43 marker positions were reduced to a 22 point skeleton representation using the MoCap Toolbox 1.5 [3]. Small gaps in the data were spline-filled using Qualisys Track Manager 2019.3 and a 2nd degree Butterworth filter with a .03Hz cutoff was applied to remove any marker jitter. Recordings in our dataset have been normalized so that the root marker (a weighted average of markers 41, 42, 6 and 7 in Figure 3) is centred at the origin where x , y , and z position is 0. Body segment lengths are averaged across the three dancers ensuring that the data is invariant to global position and individual body dimensions. The data was captured at 240Hz, and downsampled to 30Hz before model training to reduce the size of each example. The resulting 54 data tensors consist of 1800 frames (60 seconds at 30Hz) with 3-dimensional positions for each of the 22 points.

Two full motion capture recordings are withheld for testing while the remaining 52 examples are split into two sets, 80% are used for training and 10% for validation. Each example is sliced into overlapping sequences of 256 frames and the spatial dimensions of each of the 22 points are scaled using min-max normalisation. The input to our model thereby consists of 80 288 overlapping sequences of 256 frames. The target value of each training example is the 3D position of each point in the following frame.

4 MOVEMENT PREDICTION WITH MDRNNS

Mixture density networks (MDNs) [2] treat the outputs of a neural network as the parameters of a Gaussian mixture model (GMM), which can be sampled to generate real-valued predictions. Figure 4 shows a simplified mixture distribution with 4 components. A GMM can be derived using the mean, weight and standard deviation of each component. The number of components needed to accurately represent the data is not known and is treated as a hyperparameter for our model. For the study outlined here, we have used 4 components. We can interpret these components as each

Figure 4: Simplified mixture distribution example. The MDRNN presented in this work learns the mean (μ), standard deviation (σ) and weight (π) of 4 components, each having 66 dimensions, one for each of the 22 3D points.

representing a future possible movement. By combining a recurrent neural network (RNN) with an MDN to form an MDRNN we can make real-valued predictions based on a sequence of inputs. Figure 5 shows the model architecture of the MDRNN used in this work. The RNN consists of three layers of LSTM cells [10], known to be effective in modelling temporal sequences such as music, text and speech. The three LSTM layers contain 1024, 512 and 256 hidden units respectively. The outputs of the third LSTM layer are in turn connected to an MDN. The LSTM layers learn to estimate the mean, standard deviation and weight of the 4 Gaussian distributions.

To optimize an MDN, we minimize the negative log-likelihood of sampling true values from the predicted GMM for each example. A probability density function (PDF) is used to obtain this likelihood value. In our case, the GMM consists of $K = 4$ n -variate Gaussian distributions. For simplicity in the PDF, these distributions are restricted to having a diagonal covariance matrix, and thus the PDF has the form:

$$p(\theta; x) = \sum_{k=1}^K \pi_k \mathcal{N}(\mu_k, \Sigma_k; x) \quad (1)$$

where π are the mixing coefficients, μ , the Gaussian distribution centres, Σ the covariance matrices and n is the 66 position values (22 points * 3 dimensions) contained in each frame. This configuration corresponds to 8,540,692 parameters. The loss function in our system is calculated by the `keras-mdn-layer` [13] Python package which makes use of Tensorflow’s probability distributions package to construct the PDF. The model is trained using the Adam optimizer [11] until the loss on the validation set failed to improve for 10 consecutive epochs.

5 SAMPLING FROM THE TRAINED MODEL

In the following sections, three different strategies for examining the creative representational capacity of the MDRNN and their effect on the model output are explored: priming, component selection and temperature adjustments. The code used to sample from the trained model can be found at our working repository together with video examples.¹

¹<https://github.com/benediktewallace/Motion-MDRNN>

Figure 5: Sampling from the MDRNN. A sequence of preceding frames is sent through the model which outputs the parameters of a mixture distribution. By sampling from this distribution the next frame is generated.

5.1 Adjusting temperature

When sampling from our trained model we can choose to alter the value of two temperature parameters, the π -temperature and the σ -temperature. The σ -temperature is used to scale the covariance matrix of each mixture component’s multivariate normal distribution. Adjusting the σ -temperature affects the width of each mixture component. A high σ -temperature allows for samples further from the learned mean of each mixture component to be selected.

For our experiments, we explored a range of sigma values between 0 and 1 to locate at which temperatures the effect on the output changes. Sampling with σ -temperature close to or higher than 1 results in movement sequences that change rapidly between frames. The movements are “shaky” and jump in unnatural ways between time steps. We observe that this shaking effect becomes noticeable with $\sigma > 0.05$. Conversely, a σ -temperature closer to 0 allows for smoother motions. An example of the extremes can be seen in Figures 6a and 6b. These figures show the trajectories of hand and toe markers, with horizontal displacement indicating time evolution. Figure 6c shows a sequence wherein the temperature rises over time. σ -temperatures between 10^{-7} and 10^{-4} cause small variations in the motion sequence while remaining realistic. Higher values result in sequences which contain a fair amount of noise which can also distort the form of the body, while lower values contain less overall movement.

The π -temperature adjusts the probability of sampling from individual mixture components. High π -temperatures reweight the probability of sampling from each component in such a way that each component is an equally likely choice, while sufficiently low temperatures will ensure that only a single component is selected.

For the experiments presented here, we examine π -temperatures between 0 and 10. By selecting a π -temperature close to or higher than 1.0 we create a higher likelihood of sampling from a different mixture component at every time step. Sampling with π -temperature close to 0, on the other hand, ensures that only the component which has the highest learned weight will be sampled from. This is equivalent to isolating this component and sampling from it at every time step. Figure 7b shows a sequence generated with a π -temperature close to 0. Figure 7c shows how the output changes as the π -temperature is increased over time. At temperatures around 1.0 the component with the highest weight is still sampled from most frequently but, we also intermittently sample from the other 3 components according to their respective weights. Figure 7a shows the result of sampling with a high π -temperature. Here, we sample from a different component at almost every time step. Shifting between mixture components when generating motion sequences causes abrupt changes in the position of the body, alluding to the notion that each component may learn a slightly different movement sequence. We examine this more closely in the upcoming section.

5.2 Isolating mixture components

For these experiments, we disregard the π -temperature and instead manually select which of the 4 mixture components to sample from. This ensures that each new frame is sampled from a single component. We observed in the previous section that the entire position of the body changed as we sampled with a higher π -temperature, indicating that individual components emphasise different features.

In order to examine this more closely the σ -temperature is kept at a low value to make certain that we sample close to the mean of each component and each sequence is given the same starting position. Figure 8 shows the 100th frame from sequences sampled from each of the 4 components using the same starting frames and temperature parameters. These figures show how each component has predicted a different outcome, with component 1 being the

(a) Movement generated with $\sigma=1.0$. Movements are noisy, positions of points jerk between frames.

(b) Movement generated with $\sigma = 10^{-9}$. Movements are smooth and realistic.

(c) Movement generated with σ -temperature rising over time.

Figure 6: Sequences generated using different σ -temperatures.

most dissimilar to the other components. Sequences sampled from component 1 contain almost no movement at all, while component 2 and 3 produces examples which show the highest amount of realism. The third component consistently produces the most realistic output and is the component with the highest learned π -value. Thereby making it the component which is most likely sampled from when no adjustments are made using the π -temperature parameter. Sampling from component 1 and 4 shows little overall movement and a fair amount of distortion in the body, as joints

are placed at unnatural angles. From these experiments, it seems that each component learns a slightly different movement pattern. Thereby, selecting to sample from a single component when generating a movement sequence ensures that certain learned patterns appear in the predicted movement.

(a) Movement generated with $\pi=10$. Positions of the body jerk between frames as the we alternate between mixture components.

(b) Movement generated with $\pi = 10^{-9}$. Movements are smooth and realistic.

(c) Movement generated with π -temperature rising over time.

Figure 7: Sequences sampled with different π -temperatures.

5.3 Primed sampling

Being able to generate motion in a certain style is useful both in the context of animation and as a creative tool for choreography. Previous work using MDRNNs to generate handwriting [7] show that it is possible to produce examples which display the characteristics of a particular writer when the model is primed on an example. [4] have recently shown how the characteristics of a participant’s individual style of movement are sufficiently unique to allow for classification using machine learning and as such, it is intriguing to investigate whether priming a model on an example may allow the model to generate movements in the style of a specific participant or performance.

When generating motion with priming, a movement sequence which has not been used in training is given as input to the model. The next frame is then generated and the process is repeated. The model always predicts the next frame for a previously unseen real sequence, as opposed to non-primed sampling, wherein the models’ previous predictions become part of the sequence used to generate the following frame. We explore the effect of priming on two performances by different individuals using the examples that were withheld during training. The first example, hereafter referred to as *primer A*, was performed to rhythmical musical stimuli with a strong beat presence. The second example, *primer B*, was performed to slow, non-rhythmic musical stimuli. As such, the two priming examples have differing characteristics. Primer A contains more

Figure 8: 100th frame from sequences generated by different mixture components. Keeping all other parameters the same, each component generates a different movement sequence.

movement overall and a higher speed of movement, while primer B consists of slower movements. For these experiments π and σ -temperatures are both set to values close to zero ensuring that we sample from close to the mean of the component with the highest weight, component 3.

Figure 9a shows a series of frames from primer A. Figure 9b shows the corresponding movement sequence generated using primer A. As can be seen in the figure, the generated movement to a large degree follows the movement of the priming sequence. Movement aspects such as the speed of motion and the overall amount of movement are similar to the priming sequence as well. Similar results were also found for primer B.

6 DISCUSSION

In the above results, we have explored how the output of the trained MDRNN can be affected using different sampling strategies. There are, of course, countless parameter settings and combinations to examine. Here, we have focused on outlining the extremes as well as the threshold points wherein the parameters effect changes.

When sampling from only the component with the highest weight, we ensure that the model outputs smooth and realistic motion sequences. While stochastic sampling of mixture components is possible, our results indicate that this approach, at least for movement, does not necessarily improve the output. Periodic changes in the choice of mixture component cause abrupt changes and thereby more jagged movement. Even slight deviations from

what is feasible for the human form breaks the perceived realism of the generated movement. Sampling from a single mixture component for a given sequence and altering the amount of deviation we allow away from the mean using the σ -temperature instead can give varied and realistic results. Slight adjustments of σ -temperature allows for some interesting variations to occur. This could have useful applications as a user-controlled parameter, both in a creative tool and for animation. When examining the output generated by each mixture component in isolation it is apparent that the components which achieve the highest learned weight also produce the most realistic movement sequences. It is possible that training on a larger dataset would cause the components which now show a lot of distorted body poses to learn more realistic and useful features. This would then be an interesting parameter in a creative tool in order to create variations on a theme. Examining the output from each component also shows us the capabilities of this particular model to produce several possible futures with different degrees of variation and realism.

From our experiments with priming, we found that the output of the model was able to continue the motion of unseen examples with some variation. This indicates that the style of movement is indeed continued when sampling with priming. Priming the model on unseen data could be useful when using the model as a creative tool to assist choreographers and animators, as it allows for priming the model with their own data and achieve predictions in a desirable style closer to their own. By combining priming and σ -temperature one could produce several variations on a theme.

7 CONCLUSION

We have presented results from exploring several common sampling techniques used when generating output from MDRNNs and how these techniques affect generated dance movement sequences. Our findings show that the priming technique allows the output to be shaped by an unseen example, indicating that the model could be used to generate movement sequences in a certain style. Further, the results show that changing the learned probability distribution by including temperature parameters has the potential to greatly affect the output. However, the range of viable parameter values is small, as any unnatural movement is easily detected by an observer. As such, temperature adjustment may be less suited for human motion than for other creative data. Alternating between mixture components when generating a motion sequence results in unnatural shifts in body positions between consecutive frames. Still, the components with lower learned weight may also contain interesting movement sequences. Therefore, we believe this approach warrants further exploration and analysis.

In future work, we will perform a study wherein trained dancers evaluate the generated movement sequences [22] using rating schemes [19] designed to evaluate the representational and creative capacity of the model. This work is also part of an ongoing research effort to examine how generative deep learning can be used to capture salient features of human movement, and especially dance movement, using full-body motion capture data. Dance and music are intrinsically intertwined, and as such, we will be including the music stimuli in the training data in order to produce a sound-to-movement system.

(a) Frames from priming sequence A.

(b) Motion sequence generated when priming on priming sequence A.

Figure 9: Priming sequence and corresponding output. The sequence generated by the model shows similar motion unfolding over time.

ACKNOWLEDGMENTS

This work was partially supported by the Research Council of Norway through its Centres of Excellence scheme, project number 262762.

REFERENCES

- [1] Omid Alemi and Philippe Pasquier. 2017. WalkNet: A neural-network-based interactive walking controller. In *International Conference on Intelligent Virtual Agents*. Springer, 15–24.
- [2] Christopher M. Bishop. 1994. *Mixture density networks*. Technical Report. Aston University, Birmingham, UK. <http://publications.aston.ac.uk/373/>
- [3] Birgitta Burger and Petri Toiviainen. 2013. MoCap Toolbox-A Matlab toolbox for computational analysis of movement data. In *10th Sound and Music Computing Conference, SMC 2013, Stockholm, Sweden*. Logos Verlag Berlin.
- [4] Emily Carlson, Pasi Saari, Birgitta Burger, and Petri Toiviainen. 2020. Dance to your own drum: Identification of musical genre and individual dancer from motion capture using machine learning. *Journal of New Music Research* (2020), 1–16.
- [5] Luka Crnkovic-Friis and Louise Crnkovic-Friis. 2016. Generative Choreography using Deep Learning. In *Proceedings of the Seventh International Conference on Computational Creativity*.
- [6] Kai Olav Ellefsen, Charles Patrick Martin, and Jim Torresen. 2019. How do Mixture Density RNNs Predict the Future? *arXiv preprint arXiv:1901.07859* (2019).
- [7] Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850* (2013).
- [8] David Ha and Douglas Eck. 2017. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477* (2017).
- [9] David Ha and Jürgen Schmidhuber. 2018. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*. 2450–2462.
- [10] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-term Memory. *Neural computation* 9 (12 1997), 1735–80. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [11] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [12] Jina Lee and Stacy Marsella. 2006. Nonverbal Behavior Generator for Embodied Conversational Agents. In *Proceedings of the 6th International Conference on Intelligent Virtual Agents (Marina Del Rey, CA) (IVA'06)*. Springer-Verlag, Berlin, Heidelberg, 243–255. https://doi.org/10.1007/11821830_20
- [13] Charles Martin. 2018. keras-mdn-layer: Python Package. <https://doi.org/10.5281/zenodo.1482348>
- [14] Charles Patrick Martin and Jim Torresen. 2018. RoboJam: A musical mixture density network for collaborative touchscreen interaction. In *Int'l. Conference on Computational Intelligence in Music, Sound, Art and Design*. Springer, 161–176.
- [15] Johannes Michalak, Nikolaus F Troje, Julia Fischer, Patrick Vollmar, Thomas Heidenreich, and Dietmar Schulte. 2009. Embodiment of sadness and depression—gait patterns associated with dysphoric mood. *Psychosomatic medicine* 71, 5 (2009), 580–587.
- [16] Meinard Müller. 2007. *Information retrieval for music and motion*. Vol. 2. Springer.
- [17] Mariel Pettee, Chase Shimmin, Douglas Duhaime, and Ilya Vidrin. 2019. Beyond Imitation: Generative and Variational Choreography via Machine Learning. *10th International Conference on Computational Creativity* (2019).
- [18] Frank E Pollick, Helena M Paterson, Armin Bruderlin, and Anthony J Sanford. 2001. Perceiving affect from arm movement. *Cognition* 82, 2 (2001), B51–B61.
- [19] Graeme Ritchie. 2007. Some Empirical Criteria for Attributing Creativity to a Computer Program. *Minds and Machines* 17 (2007), 76–99.
- [20] Liam Satchell, Paul Morris, Chris Mills, Liam O'Reilly, Paul Marshman, and Lucy Akehurst. 2017. Evidence of big five and aggressive personalities in gait biomechanics. *Journal of nonverbal behavior* 41, 1 (2017), 35–44.
- [21] Bob Sturm, Joao Felipe Santos, and Iryna Korschunova. 2015. Folk music style modelling by recurrent neural networks with long short term memory units. In *16th International Society for Music Information Retrieval Conference*.

- [22] Bob L Sturm and Oded Ben-Tal. 2017. Taking the models back to music practice: evaluating generative transcription models built using deep learning. *Journal of Creative Music Systems* 2, 1 (2017).
- [23] Wei Ren Tan, Chee Seng Chan, Hernán E Aguirre, and Kiyoshi Tanaka. 2017. ArtGAN: Artwork synthesis with conditional categorical GANs. In *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 3760–3764.