Transfer of Status Report

Adam Kosiorek[1]

***Abstract*— Abstract goes here.**

I. Introduction

We spend our lives roaming through the space-time continuum and our brains have evolved to use omnipresent temporal dependencies. They determine the evolution of the outside world, but they also, or maybe because of that, facilitate learning in humans. Even when time itself is not important, temporally-ordered sequence of informative events can introduce context and condition further information processing. Time dependencies permeate our existence to the extent that it is hard to find examples of reasoning not involving them. And yet, the majority of machine learning (ML) algorithms either do not use temporal dependencies at all or do so in a very limited sense. With no reliance on temporal dependencies, supervised ML algorithms are forced to heavily depend on expensive human-labelled data, while accounting for all redundancies therein. I am going to argue that explicit time-series modelling and interaction with the environment are enough to create a powerful signal for self-supervised learning. The resulting feedback loop eliminates the need for huge human-tagged datasets, while improving performance of the existing approaches at the same time.

To this end, I am going to explore the use of probabilistic neural networks for time-series modelling. While my focus is on unsupervised or rather self-supervised learning, I will also explore the connection between learning and interacting with the environment. Specifically, I will argue that structured time-series prediction can lead to learning rich neural representations of sensory inputs. By following the probabilistic approach, it is possible to remove redundancies from the data and model attention mechanisms similar to those present in the human brain. I will also argue that enforcing particular model structure is equivalent to introducing prior information into the model, thereby constraining the learning problem and making it easier to learn certain properties of the dataset. In doing so, I will build on the recent advances in variational inference and construct a generative framework for unsupervised single object tracking in videos. Finally, I will use the learned model for model-based reinforcement learning (RL), with the purpose of improving sample efficiency of RL on one hand, and boosting the learning speed of the perception module by interaction with the environment on the other.

My work as a DPhil student at Oxford started with the problem of single object tracking in videos, which resulted in a NIPS submission (Kosiorek, Bewley, and Posner, 2017). This project gave me an opportunity to explore learning in the presence of temporal dependencies and to explore the concept of self-supervision: how to make the system learn better without using any additional external (e. g., ground-truth) information? The rest of this paper is structured as follows: Section II covers prior work related to the areas in question. Specifically, I summarise the tasks of sequence prediction, predictive coding, variants of unsupervised learning and present a number of relevant approaches. In the section III, I describe the work on object tracking and how it ties with my interests and the planned future work on structured unsupervised learning for videos, predictive coding and model-based reinforcement learning. Section IV describes my future research plans, related risks and expected outcomes. Section V concludes this work.

[1]Oxford Robotics Institute, Dept of Engineering Science, University of Oxford.

II. Related Work

A. Unsupervised Learning via Generative Modelling

While data in general is abundant and cheap, data for supervised learning is often expensive and time-consuming to gather. The majority of ML algorithms require relatively large amounts of labelled training data. One of the explanation states that they start learning without any prior knowledge of the world . This is in stark contrast to humans, who not only have a vast amount of knowledge about the world, but also expand it continuously and without any supervision (Friston, 2009). One alternative is to perform generative modelling of the probability distribution $p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{z})\, \mathrm{d}\mathbf{z}$ of observations $\mathbf{x}$ in terms of some latent variables $\mathbf{z}$. The latent variables *explain* the observations and can make the joint distribution $p(\mathbf{x}, \mathbf{z})$ tractable even in the case of an intractable marginal distribution. The latent encoding can be used in downstream tasks e. g., for transfer or semi-supervised learning (Pan and Yang, 2010). Hinton, Osindero, and Teh, 2006 introduced Deep Belief Networks (DBN) which explain the observations in terms of Bernoulli latent variables. Alternatively, we can approximate the true data distribution by deriving the evidence lower bound (ELBO) on the log probability of the data, which results in variational autoencoders (VAE) (Kingma and Welling, 2014; Danilo J Rezende, Mohamed, and Wierstra, 2014). VAEs are much more flexible than DBNs as they allow latent variables from arbitrary probability distribution functions (pdf) and can be trained end-to-end with off-the-shelf gradient-based methods. These approaches are primarily suited to modelling datasets of independent and identically distributed (*i.i.d.*) points.

Add reference(s

B. Sequence Modelling

Traditional approaches to sequence modelling often consider inference of latent variables that explain the data e. g., linear

dynamical systems or hidden markov models (Bishop, 2006). They often require dynamics of the system to be known and often have too little capacity to model complex and high-dimensional real-world data. Neural networks, on the other hand, can learn both features and state dynamics from data and they can approximate functions of arbitrary complexity with arbitrary precision. Even early works on the topic demonstrated how useful neural networks are for prediction of chaotic time-series (Lapedes and Farber, 1988). Since then, neural networks have been successfully applied for sequence classification and prediction in different domains: written natural language, speech and audio, motion capture data or brain waves (Längkvist, Karlsson, and Loutfi, 2014). Unsupervised learning can be also done as sequence prediction, where the task is to predict the observation at time $t+1$ given a sequence of observations $\mathbf{x}_{1:t}$ up to time t. This task is flexible in that it admits many different model types, including Gaussian processes, support vector machines or feed-forward neural networks, although models which can explicitly use temporal structure of data such as Gaussian process dynamic models (GPDM; Wang, Fleet, and Hertzmann, 2008) or recurrent neural networks (RNN) tend to perform better. Recently, sequential counterparts of VAEs have been proposed, which allow efficient generative modelling of sequences (Fabius and Amersfoort, 2015; Bayer and Osendorfer, 2015; Karl et al., 2017).

C. Predictive Coding

Modern sequential predictive models tend to update its hidden state at every time-step. It can be argued, however, that if a model is able to predict the world perfectly, it should not update its state. On the contrary, if a perfect prediction is available, the model should be capable of evolving its hidden state so as to reflect the change of the world w. r. t. the prediction. Predictive coding formalises this behaviour by using only the prediction errors to update the hidden state. The idea dates back at least to the Kalman filter (Kalman, 1960). Recent advances in neural networks allow to frame it as an auto-regressive neural network (Lotter, Kreiman, and Cox, 2016; Canziani and Culurciello, 2017), which uses the difference between the prediction and the input at the following time step to update the hidden state. As there are many futures possible, this approach leads to an ill-posed problem. The prediction, which is a maximum-likelihood estimate of what might happen, is only a single instantiation thereof. It would be theoretically more sound to normalise prediction errors by the covariance matrix of errors, an approach adopted by Kalman filtering. Friston, 2009 argues that the human brain might also follow this approach, whereby the computational architecture of the brain forms a hierarchical system, whose every layer constantly tries to predict the output of the lower levels of the hierarchy in a fully Bayesian fashion. The normalised predictive error in this setup gives rise to surprise, which is the negative log-likelihood of the inputs under the predictive distribution of the model and where the normalisation can be understood as an attention mechanism.

D. Learning of Abstract Ideas

The utility of sequence prediction as an unsupervised learning approach can be intuitively explained by the fact that predicting the future, if it is to be done well, requires very good understanding of the present. If, for example, a model can learn an idea of an object and the laws of physics, it should be able to constrain its prediction to those physically plausible: e. g., a car should not dissolve into thin air. The majority of neural models are over-parametrised (Denil et al., 2013), however, which makes learning abstract notions from data extremely sample inefficient. Eslami et al., 2016 introduce AIR, a VAE with a variable-length latent encoding for image reconstruction. This model imposes a geometric prior on the encoding length which encourages sparse solutions, therefore learning to decompose the scene into a number of independent parts — the objects. It is worth noting that, along the main model, the authors introduce difference-AIR, which exploits the specific structure of the problem and adheres to the predictive coding paradigm, thereby achieving better performance. In the extension of this work, Danilo Jimenez Rezende et al., 2016 learn to reconstruct three-dimensional (3D) structure of an object from even a single two-dimensional (2D) view by imposing 3D latent representation and structuring the decoder as a projection of the latent space into the 2D output space; they show that their model is able to infer the idea of an object from data. Häusser, Mordvintsev, and Cremers, 2017 learn the idea of an object and its class by learning to associate similar objects with each other in the embedding space, which is very much like a child learning about its identity by comparing itself with others (Decety and Chaminade, 2003). In case of reinforcement learning, a complex environment might itself be a cue which leads to learning abstract ideas. Heess et al., 2017 shows that articulated agents can learn real-world motion patterns by interacting with the environment. Specifically, they learn to crouch, jump, turn and run while maximising a very simple reward function based on forward progress. Using a specific model structure as a method of learning abstract ideas was also demonstrated by Battaglia et al., 2016. The authors propose an interaction network, a highly complex model that operates on a graph of objects and relations between them and acts as a physics simulator. The particular model structure enables learning invariants (e. g., energy conservation) and inferring latent variables describing the system as a whole (e. g., potential energy).
In the following we put these ideas together.

III. Hierarchical Attentive Recurrent Tracking

During my first year as a DPhil student at Oxford I developed the Hierarchical Attentive Recurrent Tracking (HART) framework. This RNN-based model learns to track objects in videos by focusing on small image regions, usually not much bigger than the tracked objects. It does so by using a differentiable attention mechanism, which can effectively crop a part of the image, thereby quickly removing irrelevant parts of the input. Upscaling HART to a challenging real-world dataset proved difficult, as end-to-end training on a randomly initialised

neural network was very unstable and converged to poor results. To address this issue, I resorted to transfer learning (Pan and Yang, 2010). Using AlexNet (A. Krizhevsky, I. Sutskever, and Hinton, 2012) as a feature extractor has stabilised the training and improved performance.

Feature extractors pre-trained on static image analysis tasks are often used for processing video sequences (see e. g., Ning et al., 2016). This approach, while effective, has little justification in neuroscience. In contrary, there is a growing body of evidence indicating the importance of temporal connections in the human visual cortex (Kastner and Ungerleider, 2000), which suggests that the temporal integration of information is vital for building up high resolution representation of the world.

Modern single-object-tracking approaches are based on either metric learning or bounding box regression. Not only do they need to rely on heuristics (non-differentiable image cropping, explicit scale search) to achieve computational efficiency and accuracy, but they are also fully dependent on a single error signal for learning. HART, on the other hand, exploits its structure to make more efficient use of the error signal. It uses the ground-truth bounding boxes to derive three related but distinct learning signals, one of each of the model parts. One of them, the object masks for foreground-background segmentation of extracted attention glimpses, can be seen as self-supervision. It serves different purposes: (a) it forces the model to store object appearance information in the hidden state, (b) it encourages better spatial attention prediction, as computing the object mask is easier (lower relative penalty for any mistakes) if the object covers a bigger part of the attention glimpse and (c) since the ground-truth object mask is computed on the fly, it serves as data-augmentation, in the sense that the errors and the learning signal is dependent on the model parameters and is different in every iteration of training even if the input data is the same Despite being trained under full supervision, HART was a test bench I used to learn about learning in the presence of temporal dependencies and to experiment with different structures of the objective function so as to maximise learning from a limited amount of data.

IV. Research Proposal

Drawing from my experiences with HART, I am going to focus on representation learning for videos. I believe that imposing a specific model structure can lead to better, easier and faster learning and that generative modelling is able to produce neural representations that are easily transferable to other tasks. To this end, I would like to explore the predictive coding paradigm, or rather its instantiation within the variational inference framework. Going further, I believe it is possible to combine attentive recurrent tracking with an approach similar to AIR (Eslami et al., 2016) to create a generative model of a moving object, capable of inferring intuitive physics without any supervision. Finally, I would like to merge these two branches to learn a model of the environment and use it in a model-based reinforcement learning. In the following, I will describe the three ideas, explore connections between them and evaluate associated risks.

A. Variational Inference for Predictive Coding

Predictive coding describes a family of models for sequence prediction. If a sequence predictor has a hidden state, one can argue that this state should be updated only in case of imperfect predictions, see section II-C for details. Friston, 2009 argues that this type of sequence modelling is employed in the human brain, where it explains phenomena related to learning. He also represents the view that the brain is Bayesian and that any prediction in the human brain has to be probabilistic. In this case, the model can be optimised by minimising the information-theoretic surprise and the prediction error can be generalised to Mahalanobis distance w. r. t. the predictive probability distribution. This approach has not been explored in the machine learning literature, and yet it gives rise to a family of models shaped after the VAE, but reformulated for prediction as opposed to reconstruction of the input. This formulation has several advantages, including:

Non-stationary priors A probabilistic prediction of the activations in the latent space can be used as a prior for the latent encoding at the next time-step. It maintains its properties as a regulariser while admitting higher flexibility of the approximate posterior distribution.

Self-normalisation Probabilistic predictive coding can be used for normalisation of activations of neural networks. Given that we minimise surprise as the learning criterion, and assuming Gaussian output probability distribution, we can use the statistics of the distribution to whiten latent encoding at layer l before inputting it to layer $l + 1$. It can potentially alleviate or even solve the problem of covariance shift in the encoder part of the model, therefore removing any need for explicit normalisation (e. g., batch normalisation). The validity of this argument is supported by the successful usage of neural baselines for variance reduction in score-function estimators (Mnih and Gregor, 2014). It is unclear how normalisation of the encoder will impact learning of the whole system, nor whether it is possible to devise a similar method of normalising the decoder activations.

While not revolutionary in itself, the predictive coding paradigm imposes structure on the model and constrains the optimisation problem, potentially leading to faster and more sample-efficient learning.

B. Unsupervised Learning to Track Objects

One can define an object in the image space as a patch of an image, where the correlation between pixels within the patch is strong, while the correlation between pixels inside and outside of the patch is weak. This definition, together with the penalty on the encoding length in the latent space, is in fact what makes AIR work. We can extend this definition to video sequences, where correlation between pixels in patches representing the same object at consecutive time-steps should be high under the assumption of high-enough frame

rate. If video frames contain only simple objects on plain background, we can reformulate AIR for frame prediction instead of reconstruction to form a generative model of moving objects. This approach is unlikely to work with rich backgrounds or when an object constitutes only a small part of the scene, however. I believe that these issues can be addressed by background subtraction and soft visual attention, respectively.

1) Background Subtraction: Assuming that the appearance of the object is known and is given by a vector $\mathbf{v}_t$, it is possible to segment it out of the image by using a dynamic filter network (DFN; De Brabandere et al., 2016) in a very similar fashion to the dorsal stream of HART. Moreover, this segmentation model can be easily pre-trained (as proved by unpublished preliminary experiments) in an unsupervised way by cropping two overlapping patches from an image, treating one of them as the object and trying to find it within the second patch. If we assume that the initial tight bounding box is available (ground-truth, provided by an external object detector or AIR), we can easily extract the appearance vector.

2) Visual Attention: If the object is small relative to the image, background subtraction is unlikely to work due to the potentially large amount of noise in the segmentation. We can address this issue by using soft visual attention to crop the object or a small area around it from the image. This approach requires attention parameters, which can be initialised from the bounding box for the first frame or inferred from the sequence seen so far.

Given an attention glimpse $\mathbf{g}_t$ at time t and an appearance vector $\mathbf{v}_t$, we can create an object mask $\mathbf{m}_t$ and compute a masked version of the glimpse $\mathbf{g}_t^m = \mathbf{m}_t \odot \mathbf{g}_t$, where $\odot$ denotes the Hadamard product. Given two masked glimpses at times t and $t+1$, we can predict $\mathbf{g}_{t+1}^m$ from $\mathbf{g}_t^m$ by using an AIR-like model.

To drive learning, we can minimise the prediction loss of $\mathbf{g}_{t+1}^m$ while maximising the area of the object mask $\mathbf{m}_t$ in the image space at the same time. The trade-off here is that $\mathbf{g}_t^m$ represents the appearance of the object and only the object, therefore it cannot be used to predict anything but the object at the next time-step; therefore minimising the prediction loss will also minimise the positive area of the object mask. Maximising the object mask area will prevent it from shrinking to zero. It will also, together with minimisation of the prediction loss, encourage accurate prediction of attention parameters, since if the attention glimpse does not contain the object it is virtually impossible to predict anything in or segment that glimpse.

By combining all of the above, we arrive at a generative model of a moving object, which includes the motion model as well as the appearance model, both conditioned on the image background. Since the model is generative, we can condition it on a short video sequence and generate multiple trajectories in the image space by sampling from the prior; we can therefore examine the model for visual fidelity as well as physical plausibility of the generated paths. One caveat here is that the model will generate only the moving object, without its background. It might be necessary to use an additional component for background prediction, possibly conditioned on the predicted objects.

C. Model-based Reinforcement Learning

Predicting the next time-step in a structured manner might prove to be an effective way of representation learning. If a model learns intuitive physics directly from data, it can be useful model-based reinforcement learning. It remains to ask whether we can make the learning of the model faster, more general or more efficient by coupling it with an agent which can interact with its environment. Specifically, I would like to investigate the following:

1) Is it possible to use a policy for surprise-maximisation in a predictive coding setting? Can it lead to faster learning by exposing the model to otherwise rare events?
2) Can a surprise-minimisation policy be used in the absence of any explicit goal? How to avoid degenerate solutions in this case?
3) Does predicting the next time-step lead to faster learning in RL, especially when rewards are sparse? Does it encourage or discourage exploration?

Approaches described in sections IV-A and IV-B can be used for model learning in a model-based reinforcement learning setting. They can be used for pretraining or as unsupervised auxiliary tasks. While interactions between proposed methods and reinforcement learning remain unknown, they can potentially improve sample efficiency and scalability of reinforcement learning approaches and I am interested in exploring this topic.

V. CONCLUSIONS

This paper summarises the contributions I have made during my DPhil studies so far and details my future research plan. For the remainder of my studies I would like to explore representation learning for videos, with the focus on next-frame prediction by using models that impose a problem-specific structure. Finally, I would like to investigate the applicability of proposed solutions for model learning for model-based reinforcement learning.

REFERENCES

A. Krizhevsky, I. Sutskever, and Geoffrey E. Hinton (2012). "ImageNet Classification with Deep Convolutional Neural Networks". In: *NIPS*, pp. 1097–1105.

Battaglia, Peter et al. (2016). "Interaction Networks for Learning about Objects, Relations and Physics". In: *Nips*, pp. 4502–4510. arXiv: 1612.00222.

Bayer, Justin and Christian Osendorfer (2015). "Learning Stochastic Recurrent Networks". In: *ICLR*. arXiv: 1411.7610.

Bishop, Christopher M. (2006). *Pattern recognition and machine learning*. Springer, p. 738.

Canziani, Alfredo and Eugenio Culurciello (2017). "CortexNet: a Generic Network Family for Robust Visual Temporal Representations". In: arXiv: 1706.02735.

De Brabandere, Bert et al. (2016). "Dynamic Filter Networks". In: *NIPS*. arXiv: 1605.09673.

Decety, Jean and Thierry Chaminade (2003). "When the self represents the other: A new cognitive neuroscience view on psychological identification". In: *Consciousness and Cognition*. Vol. 12. 4, pp. 577–596.

Denil, Misha et al. (2013). "Predicting Parameters in Deep Learning". In: *NIPS*, pp. 2148–2156. arXiv: 1306.0543.

Eslami, S. M. Ali et al. (2016). "Attend, Infer, Repeat: Fast Scene Understanding with Generative Models". In: *NIPS*. arXiv: 1603.08575.

Fabius, Otto and Joost R. van Amersfoort (2015). "Variational Recurrent Auto-Encoders". In: *Iclr* 2013, pp. 1–5. arXiv: 1412.6581.

Friston, Karl (2009). "The free-energy principle: a rough guide to the brain?" In: *Trends in Cognitive Sciences* 13.7, pp. 293–301.

Häusser, Philip, Alexander Mordvintsev, and Daniel Cremers (2017). "Learning by Association - A versatile semi-supervised training method for neural networks". In: *CVPR*. arXiv: 1706.00909.

Heess, Nicolas et al. (2017). "Emergence of Locomotion Behaviours in Rich Environments". In: arXiv: 1707.02286.

Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh (2006). "A Fast Learning Algorithm for Deep Belief Nets". In: *Neural Computation* 18.7, pp. 1527–1554.

Kalman, R E (1960). "New Approach to Linear Filtering and Prediction Problems". In: *Fluids Engineering* 82.82 (Series D), 35–45 (1960) (11 pages).

Karl, Maximilian et al. (2017). "Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data". In: *ICLR*. arXiv: 1605.06432.

Kastner, Sabine and Leslie G. Ungerleider (2000). "Mechanisms of visual attention in the human cortex". In: *Annual Reviews of Neuroscience* 23.1, pp. 315–341.

Kingma, Diederik P and Max Welling (2014). "Auto-Encoding Variational Bayes". In: *ICLR*. arXiv: 1312.6114.

Kosiorek, Adam R., Alex Bewley, and Ingmar Posner (2017). "Hierarchical Attentive Recurrent Tracking". In: *NIPS*. arXiv: 1706.09262.

Längkvist, Martin, Lars Karlsson, and Amy Loutfi (2014). "A review of unsupervised feature learning and deep learning for time-series modeling". In: *Pattern Recognition Letters* 42, pp. 11–24.

Lapedes, AS and RM Farber (1988). "How neural nets work". In: *NIPS*.

Lotter, William, Gabriel Kreiman, and David Cox (2016). "Deep Predictive Coding Networks for Video Prediction and Unsupervised Learning". In: arXiv: 1605.08104.

Mnih, Andriy and Karol Gregor (2014). "Neural Variational Inference and Learning in Belief Networks". In: *ICML*. arXiv: arXiv:1402.0030v2.

Ning, Guanghan et al. (2016). "Spatially Supervised Recurrent Convolutional Neural Networks for Visual Object Tracking". In: *arXiv preprint arXiv:1607.05781*. arXiv: 1607.05781.

Pan, Sinno Jialin and Qiang Yang (2010). *A survey on transfer learning*. arXiv: PAI.

Rezende, Danilo J, Shakir Mohamed, and Daan Wierstra (2014). "Stochastic backpropagation and approximate inference in deep generative models". In: *ICML*. Vol. 32, pp. 1278–1286. arXiv: arXiv:1401.4082v3.

Rezende, Danilo Jimenez et al. (2016). "Unsupervised Learning of 3D Structure from Images". In: *NIPS*.

Wang, Jack M., David J. Fleet, and Aaron Hertzmann (2008). "Gaussian process dynamical models for human motion". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.2, pp. 283–298.