Transfer of Status Report

Adam Kosiorek[1]

***Abstract*— Abstract goes here.**

I. Introduction

We spend our lives roaming through the space-time continuum. Our senses have evolved to make use of the temporal dependencies omnipresent in real-world data. And yet the majority of machine learning (ML) algorithms either do not use temporal dependencies at all or rely on features extracted by models which do not take them into account. I am interested in and will focus on using neural networks for probabilistic time-series modelling, with the emphasis on unsupervised and self-supervised learning and the connection between learning and interacting with the environment. I am going to argue that time dependencies in data and the interaction of an agent with its environment are enough to create a powerful signal for self-supervised learning. My work as a PhD student at Oxford started with the problem of single object tracking in videos, which resulted in a successful NIPS 2017 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.

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 upstream 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, 2013; 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, e. g., linear dynamical systems or hidden markov models, that explain the data (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).

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

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 attention.

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 where the car follows some trajectory and does not dissolve into thin air. The majority of neural networks are over-parametrised (Denil et al., 2013) and it is difficult to expect learning any abstract notions from data without imposing any structure. 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, Rezende, Eslami, 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 it 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 PhD 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 labelled training data. HART, on the other hand, uses self-supervision in the form of object masks computed as foreground-background segmentation of extracted attention glimpses. is that even true? Computing the object mask serves several 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 mistake) 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 on HART, I would like to focus my future research on representation learning for videos. I believe it is possible to combine attentive recurrent tracking with an approach similar to AIR of Eslami et al., 2016 to create a system capable of learning the idea of an object and to track that object without any supervision. In parallel, I would like to explore the predictive tracking paradigm, or rather its instantiation within the variational inference framework. 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. Unsupervised Learning to Track Objects

Let $\mathbf{x}_{1:T}$ be an image sequence containing a moving object o and let $\boldsymbol{b}_1$ be a bounding box around that object in the initial frame. Assuming that the bounding box is tight (i. e., it contains the maximum amount of the object and the minimum amount of background possible) and assuming that the object appearance changes slower than the appearance of the object's background, it is possible to learn to track that object without any supervision. The learning framework to do so consists of the following steps:

1) extract an attention glimpse $\mathbf{g}_t$ containing the object o in the image $\mathbf{x}_t$,
2) predict attention parameters and extract the attention glimpse $\mathbf{g}_{t+1}$ at time $t+1$,
3) reconstruct $\mathbf{g}_{t+1}$ from $\mathbf{g}_t$ using an approach similar to AIR. If the reconstruction is constrained to be rectangular and done in a single step, the reconstruction gives the bounding box coordinates $\boldsymbol{b}_{t+1}$ at time $t+1$.

Beside the already mentioned assumptions, the above approach relies on the assumption that it is possible to predict attention parameters for the next time-step and that it is possible to reconstruct contents of a glimpse based on the contents of the glimpse at the previous time-step. Moreover, for the step (3) to work, that is, to have a bounding box as a result of reconstruction, we need to enforce that the object is contained within the glimpse. It is not immediately clear whether theses assumptions hold and it is one of the purposes of this project to determine their validity. We believe, however, that it is possible to enforce them and we will now describe how.

Given the initial tight bounding box and the initial frame, it should be clear that it is possible to reconstruct the object at the next time-step given its location. If we assume small object motions (high frame rate), we can assume that the object at time $t+1$ lies within the neighbourhood of the object at time t and it is therefore enough to expand the size of the attention glimpse to cover the object at the next time step.

Given that we know how the object looks like and given that we have a rough idea of where the object might be in the next frame, we can determine the object location in that frame: we can compute the object map (mechanism very similar to dorsal stream from HART).

Once we have the object map, we can reconstruct not the whole glimpse, but only the parts of the glimpse indicated by the object mask as the part containing the object.

Given that we can compute the object mask, we can learn the attention by maximising the positive area of the object mask (area containing the object), as this should encourage the attention to contain the object. There exists a degenerate solution, however. If the attention glimpse shrinks to a very small size, the object can easily cover the entire area of the glimpse. The solution would be to maximise not the size of the object within the glimpse, but the size of the object as indicated by the object mask but projected back to the image coordinates.

This objective should be enough to learn to predict attention parameters.

Reconstruction can be learned given object masks.

It should be possible to learn object masks end-to-end with the other components of the system without encouraging them by any explicit loss function. The object mask should emerge from that fact that the background behind the object is relatively less correlated with the object itself, whereas the correlation between object appearance in subsequent frames is high. In case it proves to be impossible, it is possible to pretrain the object mask component without any supervision by...

B. Variational Inference for Predictive Coding

Unsupervised learning to track objects might prove to be an effective way for learning representation for videos. One drawback is that it is domain specific and can be applied only to image sequences. In the following I would like to explore a framework for learning representations for any time of time-series.

Predictive coding describes a family of models for sequence prediction, typically the next time-step prediction. Since the majority of sequence predictors maintain a (Markovian) hidden state, it is possible to impose additional structure. If the model is able to perfectly predict the next time-step, one can argue that it does not have to update its hidden state as it contains perfect information about the world, see section II-C for details. Friston, 2009 argues that this type of modelling can explain various learning-related phenomena in the human brain. He also supports the view that the brain is Bayesian and therefore any predictive coding there is implemented in a Bayesian way. Specifically, predictions of the sensory inputs are probabilistic, with minimisation of surprise as the learning criterium. This view 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, namely:

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.

Self-normalisation Probabilistic predictive coding can be used for normalisation of activations of neural networks. Given that we minimise surprise as the loss criterion, and assuming Gaussian output probability distribution, we can use the statistics of the distribution to whiten latent encoding at later l before inputting them to later $l+1$. 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 Mnih et al., 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.

C. *Model-based Reinforcement Learning*

Predicting the next time-step might prove to be an effective way of representation learning. Predictive coding can make it more efficient by imposing specific structure on the model and can alleviate some optimisation problems. Going a step further, we can force learning a concept of an object (and laws of physics?) by enforcing the notions of consistency and coherence between views of the same object at subsequent time-steps. It remains to ask whether we can make the learning faster, more general or more efficient by using an agent which can interact with its environment.

1) Predictive coding with a policy: can a policy learn to minimise the surprise in a different way then a degenerate solution of avoiding any motion?
2) Can we use a policy to maximise surprise, which exposes the perception model to otherwise rare events?
3) Can this approach be used to learn a policy with the absence of any goal?
4) Does predicting the next time-step lead to faster learning in RL, especially when rewards are sparse? Does it lead better regularisation? 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 PhD 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*, pp. 1–9. 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.

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: *Neural Information Processing Systems*. 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*. arXiv: `1703.03129`.

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 (2013). "Auto-Encoding Variational Bayes". In: 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: *Neural information processing systems*.

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

Mnih, a. et al. (2014). "Neural Variational Inference and Learning in Belief Networks". In: *ArXiv stat.ML* 32.October, pp. 1–20. 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 Jimenez, S. M. Ali Eslami, et al. (2016). "Unsupervised Learning of 3D Structure from Images". In: *NIPS*.

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra (2014). "Stochastic Backpropagation and Approximate Inference in Deep Generative Models". In: arXiv: `1401.4082`.

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.