Detect, anticipate and generate: Semi-supervised recurrent latent variable models for human activity modeling

Judith Bütepage[1]
butepage@kth.se

Danica Kragic[1]
dani@kth.se

***Abstract*—Successful Human-Robot collaboration requires a predictive model of human behavior. The robot needs to be able to recognize current goals and actions and to predict future activities in a given context. However, the spatio-temporal sequence of human actions is difficult to model since latent factors such as intention, task, knowledge, intuition and preference determine the action choices of each individual. In this work we introduce semi-supervised variational recurrent neural networks which are able to a) model temporal distributions over latent factors and the observable feature space, b) incorporate discrete labels such as activity type when available, and c) generate possible future action sequences on both feature and label level. We evaluate our model on the Cornell Activity Dataset CAD-120 dataset. Our model outperforms state-of-the-art approaches in both activity and affordance detection and anticipation. Additionally, we show how samples of possible future action sequences are in line with past observations.**

***Index Terms*—Human behavior modeling, activity anticipation**

I. Introduction

Human behavior is often stochastic and therefore difficult to predict over a longer period of time. Even within the context of a given task and a certain environment individuals might act differently based on e.g. intuition, prior knowledge and preferences. For example, if you provide a number of individuals with the task to prepare a meal following the same recipe, one person might follow a different order than specified because they have learned that a certain ingredient needs time to develop flavor. Another person might use only the big green knife instead of the more handy red knife because they prefer the color green and someone else might intentionally leave out a step.

One way to approach this problem is to model different types of human characters [1]. While this method is suitable for a single task setting such as an assembly line application, it might not scale to more general behavior which is distributed over many tasks and environments. A more scalable approach is structured prediction with e.g. conditional random fields (CRFs) [2], [3] which allows to capture the statistical dependencies between human subjects, their activities, objects in the environment and their affordances. However, common CRFs are limited in their capacity to model long-term dependencies due to the Markov assumption. Structural recurrent neural networks (S-RNN) [4] overcome this problem by employing recurrent neural networks (RNNs) as nodes and edges in the structured graph to detect and predict activity and affordance labels at each time step. The expressiveness and representational power of these neural networks increases the predictive power over short time horizons but the model structure prohibits long-term sequence generation. As S-RNNs do not explicitly learn to predict future feature states, they can not generate possible state-action sequences. Additionally, this deterministic model is not able to generate multiple possible sequences but is restricted to predict a single label.

The key contribution of this paper is to address these issues with a generative, temporal model that can capture the complex dependencies of context and human features as well as discrete, hierarchical labels over time. In detail, we propose a semi-supervised variational recurrent neural network (SVRNN), as described in Section II-B, which inherits the generative capacities of a variational autoencoder (VAE) [5], [6], extends these to temporal data [7] and combines them with a discriminative model in a semi-supervised fashion. The semi-supervised VAE, first introduced by [8], can handle labeled and unlabeled data. This property allows us to propagate label information over time even during testing and therefore to generate possible future action sequences. Furthermore, we incorporate the dependencies between human and object

[1]The Authors are with the Robotics, Perception and Learning Lab, EECS, KTH Royal Institute of Technology, Stockholm, Sweden. This work was supported by the EU through the project socSMCs (H2020-FETPROACT-2014) and the Swedish Foundation for Strategic Research.

features by extending the model to a multi-entity semi-supervised variational recurrent neural network (ME-SVRNN), as introduced in Section II-C. The ME-SVRNN propagates information about the current state of an entity to other entities which increases the predictive power of the model. We apply our model to the Cornell Activity Dataset (CAD-120), consisting of 4 subjects who perform ten different high level actions, see Section III for details. Our model is trained to simultaneously detect and anticipate the activities and object affordances and to predict the next time step in feature space. We find that our model outperforms state-of-the-art methods in both detection and anticipation (Section III-A) while being able to generate possible long term action sequences (Section III-B). We conclude this paper with a final discussion of these findings in Section IV.

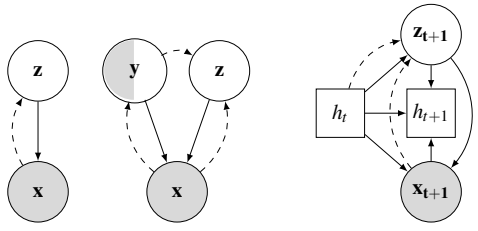
II. Methodology

In this section we introduce the model structure and detail the inference procedure. After a short overview of VAEs, we begin with a description of the general SVRNN before extending it to the multi-entity case.

We denote random variables by bold characters and represent continuous data points by $\mathbf{x}$, discrete labels by $\mathbf{y}$ and latent variables by $\mathbf{z}$. The hidden state of a RNN unit at time t is denoted by h_t. Similarly, time-dependent random variables are indexed by t, e.g. $\mathbf{x_t}$. Distributions p_θ commonly depend on parameters θ. For the sake of brevity, we will neglect this dependence in the following discussion.

A. Variational autoencoders and amortized inference

Our model builds on VAEs, latent variable models that are combined with an amortized version of variational inference (VI). Amortized VI employs neural networks to learn a function from the data $\mathbf{x}$ to a distribution over the latent variables $q(\mathbf{z}|\mathbf{x})$ that approximates the posterior $p(\mathbf{z}|\mathbf{x})$. Likewise, they learn the likelihood distribution as a function of the latent variables $p(\mathbf{x}|\mathbf{z})$. This mapping is depicted in Figure 1a). Instead of having to infer N local latent variables for N observed data points, as common in VI, amortized VI requires only the learning of neural network parameters of the functions $q(\mathbf{z}|\mathbf{x})$ and $p(\mathbf{z}|\mathbf{x})$. We call $q(\mathbf{z}|\mathbf{x})$ the recognition network and $p(\mathbf{z}|\mathbf{x})$ the generative network. To sample from a VAE, we first draw a sample from the prior $\mathbf{z} \sim p(\mathbf{z})$ which is then fed to the generative network to yield $\mathbf{x} \sim p(\mathbf{x}|\mathbf{z})$. We refer to [9] for more details.



a) VAE b) SSVAE c) VRNN

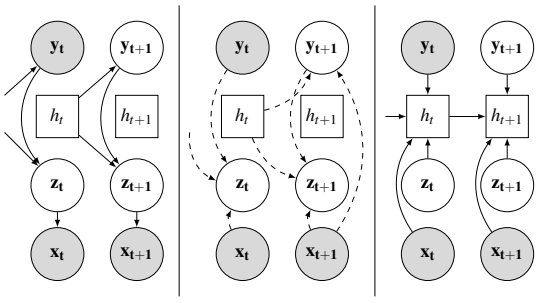
Fig. 1: Model structure of the VAE (a)), its semi-supervised version SVAE (b)), and the recurrent model VRNN (c)). Random variables (circle) and states of RNN hidden units (square) are either observed (gray), unobserved (white) or partially observed (gray-white). The dotted arrows indicate inference connections.

To incorporate label information when available, semi-supervised VAEs (SVAE) [8] include a label $\mathbf{y}$ into the generative process $p(\mathbf{x}|\mathbf{z},\mathbf{y})$ and the recognition network $q(\mathbf{z}|\mathbf{x},\mathbf{y})$, as shown in Figure 1b). To handle unobserved labels, an additional approximate distribution over labels $q(\mathbf{y}|\mathbf{x})$ is learned which can be interpreted as a classifier. When no label is available, the discrete label distribution can be marginalize out, e.g. $q(\mathbf{z}|\mathbf{x}) = \sum_{\mathbf{y}} q(\mathbf{z}|\mathbf{x},\mathbf{y})q(\mathbf{y}|\mathbf{x})$.

VAEs can also be extended to temporal data, so called variational recurrent neural networks (VRNN) [7]. Instead of being stationary as in vanilla VAEs, the prior over the latent variables depends in this case on past observations $p(\mathbf{z_t}|h_{t-1})$, which are encoded in the hidden state of a RNN h_{t-1}. Similarly, the approximate distribution $q(\mathbf{z_t}|\mathbf{x_t},h_{t-1})$ depends on the history as can be seen in Figure 1c). The advantage of this structure is that data sequences can be generated by sampling from the temporal prior instead of an uninformed prior, i.e. $\mathbf{z_t} \sim p(\mathbf{z_t}|h_{t-1})$.

B. Semi-supervised variational recurrent neural network

For SVRNN, we assume that we are given a dataset with temporal structure $D = \{D^L, D^U\}$ consisting of L labeled time steps $D^L = \{\mathbf{x_t},\mathbf{y_t}\}_{t\in L} \sim \widetilde{p}(\mathbf{x_t},\mathbf{y_t})$ and U unlabeled observations $D^U = \{\mathbf{x_t}\}_{t\in U} \sim \widetilde{p}(\mathbf{x_t})$. $\widetilde{p}$ denotes the empirical distribution. Further we assume that the temporal process is governed by latent variables $\mathbf{z_t}$, whose distribution $p(\mathbf{z_t}|h_{t-1})$ depends on a deterministic function of the history up to time t: $h_{t-1} = f(x_{<t}, y_{<t}, z_{<t})$. The generative process follows $\mathbf{y_t} \sim p(\mathbf{y_t}|h_{t-1})$, $\mathbf{z_t} \sim p(\mathbf{z_t}|\mathbf{y_t},h_{t-1})$ and finally $\mathbf{x_t} \sim p(\mathbf{x_t}|\mathbf{y_t},\mathbf{z_t},h_{t-1})$. Here, $p(\mathbf{y_t}|h_{t-1})$ and $p(\mathbf{z_t}|\mathbf{y_t},h_{t-1})$ are time-dependent priors, as shown in Figure 2a). To fit this model to the dataset at hand, we need to estimate the posterior over the unobserved variables $p(\mathbf{y_t}|\mathbf{x_t},h_{t-1})$ and $p(\mathbf{z_t}|\mathbf{x_t},\mathbf{y_t},h_{t-1})$ which is intractable. Therefore we



a) sampling from prior b) inference c) recurrence

Fig. 2: Information flow through SVRNN. a) Passing samples from the prior through the generative network. b) Information passing through the inference network. c) The recurrent update. Node appearance follows Figure 1.

resign to amortized VI and approximate the posterior with a simpler distribution $q(\mathbf{y_t},\mathbf{z_t}|\mathbf{x_t},h_{t-1}) = q(\mathbf{y_t}|\mathbf{x_t},h_{t-1})q(\mathbf{z_t}|\mathbf{x_t},\mathbf{y_t},h_{t-1})$, as shown in Figure 2b). To minimize the distance between the approximate and posterior distributions, we optimize the variational lower bound of the marginal likelihood $\mathcal{L}(p(D))$. As the distribution over $\mathbf{y_t}$ is only required when it is unobserved, the bound decomposes as follows

$$\mathcal{L}(p(D)) \geq \mathcal{L}^L + \mathcal{L}^U + \alpha\mathcal{T}^L \quad (1)$$

$$-\mathcal{L}^L = \sum_{t\in L} \mathbb{E}_{q(\mathbf{z_t}|\mathbf{x_t},\mathbf{y_t},h_{t-1})}[log(p(\mathbf{x_t}|\mathbf{y_t},\mathbf{z_t},h_{t-1}))] - KL(q(\mathbf{z_t}|\mathbf{x_t},\mathbf{y_t},h_{t-1})||p(\mathbf{z_t}|\mathbf{y_t},h_{t-1})) + log(p(\mathbf{y_t}))$$

$$\mathcal{T}^L = -\sum_{t\in L} \mathbb{E}_{\tilde{p}(\mathbf{y_t},\mathbf{x_t})} log(p(\mathbf{y_t}|h_{t-1})q(\mathbf{y_t}|\mathbf{x_t},h_{t-1}))$$

$$-\mathcal{L}^U = \sum_{t\in U} \mathbb{E}_{q(\mathbf{y_t},\mathbf{z_t}|\mathbf{x_t},h_{t-1})}\big[log(p(\mathbf{x_t}|\mathbf{y_t},\mathbf{z_t},h_{t-1}))] - KL(q(\mathbf{z_t}|\mathbf{x_t},\mathbf{y_t},h_{t-1})||p(\mathbf{z_t}|\mathbf{y_t},h_{t-1}))\big] - KL(q(\mathbf{y_t}|\mathbf{x_t},h_{t-1})||p(\mathbf{y_t}|h_{t-1})).$$

$\mathcal{L}^L$ and $\mathcal{L}^U$ are the lower bounds for labeled and unlabeled data points respectively, while $\mathcal{T}^L$ is an additional term that encourages $p(\mathbf{y_t}|h_{t-1})$ and $q(\mathbf{y_t}|\mathbf{x_t},h_{t-1})$ to follow the data distribution over $\mathbf{y_t}$. This lower bound is optimized jointly. We assume the latent variables $\mathbf{z_t}$ to be i.i.d Gaussian distributed. The categorical distribution over $\mathbf{y_t}$ is determined by parameters $\pi = \{\pi_i\}_{i=1:N_class}$). To model such discrete distributions, we apply the Gumbel trick [10], [11]. The history h_{t-1} is modeled with a Long short-term memory (LSTM) unit [12]. For more details, we refer the reader to the related work discussed in Section II-A.

C. Modeling multiple entities

To model different entities, we allow these to share information between each other over time. The structure and information flow of this model is a design choice. In our case, these entities consist of the human H and $o \in [1,N_o]$ additional entities, such as objects or other humans. We denote the dependency of variables on their source by $(\mathbf{x_t}^H,\mathbf{y_t}^H,\mathbf{z_t}^H,h_t^H)$ and $\{(\mathbf{x_t}^o,\mathbf{y_t}^o,\mathbf{z_t}^o,h_t^o)\}_{o\in 1:N_o}$. Further, we summarize the history and current observation of all additional entities by $h_t^O = \sum_o h_t^o$ and $\mathbf{x_t}^O = \sum_o \mathbf{x_t}^o$ respectively. Instead of only conditioning on its own history and observation, as described in Section II-B, we let the entities share information by conditioning on others' history and observations. Specifically, the model of the human receives information from all additional entities, while these receive information from the human model. Let $\mathbf{x_t^{AB}} = [\mathbf{x_t^A},\mathbf{x_t^B}]$ and $h_t^{AB} = [h_t^A,h_t^B]$ for $A,B \in (H,O,o)$. The structure of the prior and approximate distribution then become $p(\mathbf{y_t^H}|h_{t-1}^{HO})$, $p(\mathbf{z_t^H}|\mathbf{y_t^H},h_{t-1}^{HO})$, $q(\mathbf{y_t^H}|\mathbf{x_t^{HO}},h_{t-1}^{HO})$ and $q(\mathbf{z_t^H}|\mathbf{x_t^{HO}},\mathbf{y_t^H},h_{t-1}^{HO})$ for the human, and $p(\mathbf{y_t^o}|h_{t-1}^{oH})$, $p(\mathbf{z_t^o}|\mathbf{y_t^o},h_{t-1}^{oH})$, $q(\mathbf{y_t^o}|\mathbf{x_t^{oH}},h_{t-1}^{oH})$ and $q(\mathbf{z_t^o}|\mathbf{x_t^{oH}},\mathbf{y_t^o},h_{t-1}^{oH})$ for each additional entity $o \in 1:N_o$, We assume that the labels for all entities are observed and unobserved at the same points in time. Therefore, the lower bound in Equation 1 is extended by summing over all entities.

III. Experiments

In this section, we present our experimental results. We evaluate our model on the Cornell Activity Dataset 120 (CAD -120) [2]. This dataset consists of 4 subjects performing 10 high-level tasks, such as *cleaning a microwave* or *having a meal*, in 3 trials each. These activities are further annotated with 10 sub-activities, such as *moving* and *eating* and 12 object affordances, such as *movable* and *openable*. In this work we are focusing on classifying the sub-activities and affordances. We use the features extracted in [2] and preprocess these as in [4]. Our results rely on four-fold cross-validation with the same folds as used in [2]. For comparison, we trained the S-RNN models, for which code is provided online, on these folds and under the same conditions as described in [4]. We use a learning rate of 0.001, a batch size of 10 and the adagrad optimizer. Further, we apply a dropout rate of 0.1 to all units but the latent variable parameters and the output layers. In each batch, we mark ca. 25 % of the labels as unobserved. The object models share all parameters, i.e. we effectively learn one human model and one object model both in the single- and multi-entity case.

A. Detection and anticipation

First, we investigate the ability of our model to detect the current sub-activity and object affordance

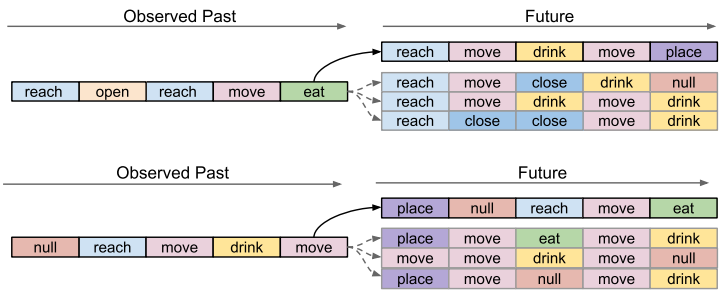


Fig. 3: Sampled sub-activity sequences given the last five observed sub-activities of the high-level actions *taking medicine* (top) and *having a meal* (bottom). Black lines indicate ground truth and gray lines indicate sampled sub-activities. A sub-activity has an average duration of 3.6 seconds.

and to anticipate these labels at the following time step. We compare the performance to the anticipatory CRF reported in [3] and the replicated results of the S-RNN [4]. The F1 score of all models averaged over the cross-validation folds and 20 samples from the latent distributions is reported in Table I. While the SVRNN without information exchange between entities outperforms the baseline methods, the multi-entity model achieves the highest values. Especially the sub-activity detection and anticipation seems to benefit from the information provided by the object states and observations.

	Detection		Anticipation	
Method	Sub-Act	Obj-Aff	Sub-Act	Obj-Aff
ATCRF [3]	86.4	85.2	40.6	41.4
S-RNN [4]	69.6	84.8	53.9	74.3
SVRNN	83.4	88.3	67.7	81.4
ME-SVRNN	**89.8**	**90.5**	**77.1**	**82.1**

TABLE I: Average F1 score for sub-activity and object affordances for detection and anticipation.

B. Generation

In contrast to S-RNN, our SVRNN model is able to generate possible, long-term action sequences. These are generated by propagating a short observation sequence through the network to obtain the summarizing state h_{t-1} and to subsequently sample from the priors $p(\mathbf{z_t}|h_{t-1})$ and $p(\mathbf{y_t}|h_{t-1})$. These samples are used by the generative network to make a prediction of the next observation $\hat{\mathbf{x}}_\mathbf{t}$, which forms the next input to the model. We present a number of sampled sub-activity sequences in Figure 3. Note that a sub-activity has an average duration of 3.6 seconds [3]. Thus, we sample possible sequences for around 18 seconds into the future. The samples are plausible action sequences given the observed past. For example, the model remembers that the action *opening* requires *closing* over several time steps. Additionally, unrelated sub-activities such as *cleaning* are not sampled.

IV. Conclusion

In this work, we presented a generative, temporal model for human activity modeling. Our experimental evaluation shows promising performance in the three tasks of detection, anticipation and generation. In future work, we are planning to evaluate the model more extensively and to extend the model to hierarchical label structures.

References

[1] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah, "Efficient model learning from joint-action demonstrations for human-robot collaborative tasks," in *2015 10th ACM/IEEE International Conference on HRI*. ACM, 2015, pp. 189–196.

[2] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[3] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 14–29, 2016.

[4] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[5] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:*1312.6114, 2013.

[6] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *arXiv preprint arXiv:1401.4082*, 2014.

[7] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *NIPS*, 2015, pp. 2980–2988.

[8] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *NIPS*, 2014, pp. 3581–3589.

[9] C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt, "Advances in variational inference," *arXiv preprint arXiv:1711.05597*, 2017.

[10] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *ICLR*, 2017.

[11] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *ICLR*, 2017.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.