MAGnet: Generating Long-Term Multi-Agent Trajectories

Eric Zhan
Caltech
ezhan@caltech.edu

Stephan Zheng
Caltech
stephan@caltech.edu

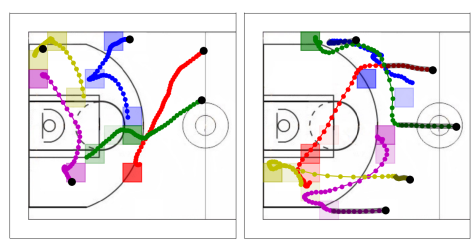
Yisong Yue
Caltech
yyue@caltech.edu

Abstract

We propose MAGnet: a flexible class of generative models that can generate rich multi-agent spatiotemporal trajectories over a long time horizon. Learning generative models in this setting is challenging, as such data can exhibit both complex individual and collective behavior over many time-steps. The key aspects of MAGnet are: 1) its hierarchical latent structure that can jointly represent long-term (macro) and short-term (micro) temporal dependencies; and 2) its shared structure between agents. We experimentally validate that MAGnet can successfully encode and generate realistic long-term basketball team trajectories for 5 players jointly, while state-of-the-art variational models fail on this task.

Learning generative models for multi-agent spatiotemporal trajectories is a challenging problem, as such data can exhibit complex individual behavior and long-term interactions between multiple agents. For example, consider the movement dynamics of 5 basketball players in Figure 1. Players move nonlinearly towards macro-goals (boxes) and display coherent team coordination over long time horizons. This implies the the data distribution is multi-modal, and has nonlinear multi-agent dynamics and long-term non-Markovian dependencies.

Previous approaches have learned sequential generative models for handwriting, speech and image synthesis [Chung et al., 2015, Gregor et al., 2015], but such models fail to generalize to the multi-agent setting (Figure 2a). To address this issue, we propose MAGnet: a flexible model class that can generate complex multi-agent behavior (Figure 1b) and be learned via variational methods. MAGnet has two salient aspects: 1) it uses a hierarchical latent structure that can encode non-Markovian dependencies between macro-goals and agent state predictions; and 2) it uses a shared agent structure, where each agent model conditions its prediction on the goals of other agents. As such, MAGnet can learn compact representations of long-term multi-agent behavior.



(a) Ground truth (b) MAGnet

Figure 1: 5 basketball player trajectories over 8 seconds (50 frames) starting from black dots. MAGnet learns complex multi-agent distributions and consistently generates more realistic behavior over baselines.

In summary, our contributions are:

- We propose a model class of flexible distributions over multi-agent spatiotemporal data.
- We instantiate our model class with VRNNs and show how to train it via variational methods and weak labels for long-term goals.
- We show that our model generates realistic multi-agent trajectories that are semantically meaningful and capture the variability in the data.

Second workshop on Bayesian Deep Learning (NIPS 2017), Long Beach, CA, USA.

1 MAGnet: Multi-Agent Goal-driven Network

The goal of generative models is to learn the data distribution $p(\boldsymbol{x})$ given data $\{\boldsymbol{x}\}$, for instance, by maximizing the log-likelihood of the data with respect to the model parameters. For a sequence $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ of length T, the joint distribution can be factorized using conditional probabilities:

$$p(\mathbf{x}_{\leq T}) = \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{<t}) \tag{1}$$

Markovian approaches (e.g. RNNs) model the conditional probabilities using a hidden state $\mathbf{h}_t$:

$$p(\mathbf{x}_t \mid \mathbf{x}_{<t}) = g_\tau(\mathbf{h}_{t-1}) \quad \text{and} \quad \mathbf{h}_t = f_\theta(\mathbf{x}_t, \mathbf{h}_{t-1}), \tag{2}$$

where τ and θ are model parameters, g returns the parameters of the conditional probability distribution (e.g. mean and covariance of a Gaussian) and f is a deterministic non-linear function.

Variational RNN. Chung et al. [2015] proposes Variational RNNs (VRNN) to better capture the variability of highly structured sequential data. VRNNs combine RNNs with variational autoencoders (VAE) by conditioning the VAE on $\mathbf{h}_{t-1}$. Let $\varphi_\tau^{\text{prior}}$, $\varphi_\tau^{\text{enc}}$, and $\varphi_\tau^{\text{dec}}$ denote the prior, encoding, and decoding functions respectively.[1] The model uses the probabilistic structure:

$$\begin{aligned}
\mathbf{z}_t &\sim \mathcal{N}(\boldsymbol{\mu}_{0,t},\ \operatorname{diag}(\boldsymbol{\sigma}_{0,t}^2)),\ \text{where } [\boldsymbol{\mu}_{0,t}, \boldsymbol{\sigma}_{0,t}] = \varphi_\tau^{\text{prior}}(\mathbf{h}_{t-1}) && \text{(prior)}\\
\mathbf{z}_t \mid \mathbf{x}_t &\sim \mathcal{N}(\boldsymbol{\mu}_{z,t},\ \operatorname{diag}(\boldsymbol{\sigma}_{z,t}^2)),\ \text{where } [\boldsymbol{\mu}_{z,t}, \boldsymbol{\sigma}_{z,t}] = \varphi_\tau^{\text{enc}}(\mathbf{x}_t, \mathbf{h}_{t-1}) && \text{(inference)}\\
\mathbf{x}_t \mid \mathbf{z}_t &\sim \mathcal{N}(\boldsymbol{\mu}_{x,t},\ \operatorname{diag}(\boldsymbol{\sigma}_{x,t}^2)),\ \text{where } [\boldsymbol{\mu}_{x,t}, \boldsymbol{\sigma}_{x,t}] = \varphi_\tau^{\text{dec}}(\mathbf{z}_t, \mathbf{h}_{t-1}) && \text{(generation)}\\
\mathbf{h}_t &= f_\theta(\mathbf{x}_t, \mathbf{z}_t, \mathbf{h}_{t-1}) && \text{(recurrence)}
\end{aligned} \tag{3}$$

Similar to VAEs, the model is trained to maximize a variational lower bound of the log-likelihood:

$$E_{q(\mathbf{z}_{\leq T} \mid \mathbf{x}_{\leq T})}\left[\sum_{t=1}^{T} -\mathrm{KL}\big(q(\mathbf{z}_t \mid \mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \| p(\mathbf{z}_t \mid \mathbf{x}_{<t}, \mathbf{z}_{<t})\big) + \log p(\mathbf{x}_t \mid \mathbf{z}_{\leq t}, \mathbf{x}_{<t})\right] \tag{4}$$

MAGnet. In the multi-agent setting, we want to model N sequences of the same length T: $\mathbf{x}^i = (\mathbf{x}_1^i, \ldots, \mathbf{x}_T^i)$, for agents $i = 1 \ldots N$. MAGnet introduces macro-goal latent variables $\mathbf{m}^i = (\mathbf{m}_1^i, \ldots, \mathbf{m}_T^i)$ for each agent to model the conditional probabilities:

$$\begin{aligned}
p(\mathbf{x}_t^i \mid \mathbf{x}_{<t}^i) &= g_i(\mathbf{h}_{t-1}^i, \mathbf{m}_t) \quad \text{and} \quad \mathbf{h}_t^i = f_i(\mathbf{x}_t^i, \mathbf{h}_{t-1}^i), \quad \text{for agents } i = 1 \ldots N,\\
p(\mathbf{m}_t \mid \mathbf{m}_{<t}) &= g_m(\mathbf{h}_{t-1}^m, \mathbf{x}_{t-1}) \quad \text{and} \quad \mathbf{h}_t^m = f_m(\mathbf{m}_t, \mathbf{h}_{t-1}^m),
\end{aligned} \tag{5}$$

where $\mathbf{m}_t$ is the concatenation of all $\mathbf{m}_t^i$ and is shared among all agents. These conditional distributions can be instantiated using sequential prediction models, such as VRNNs. The aim of using $\mathbf{m}_t$ is to capture non-Markovian dependencies between macro-goals and agent states $\mathbf{x}_t$. In this work, we use weak labels $\hat{\mathbf{m}}$ to learn this behavior from data, which we found to be the most stable approach.

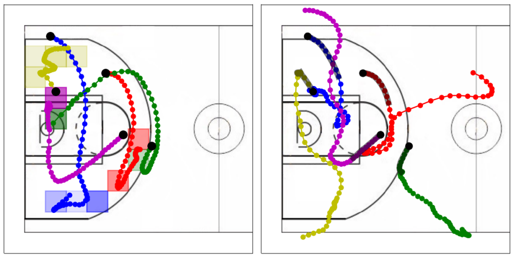
The structure of our model is motivated by the following: 1) standard Markovian approaches with a finite number of hidden states do not naturally capture long-term dependencies commonly exhibited in sequential data [Zheng et al., 2016]; 2) temporal abstractions between long-term goals and short-term state-action predictions have shown promising results in the planning and reinforcement learning communities [Alexander et al., 2016, Lowe et al., 2017]; and 3) an agent's conditional distribution over next state predictions should also depend on the macro-goals of all other agents.[2]

2 Experimental Validation on Multi-Agent Trajectory Generation

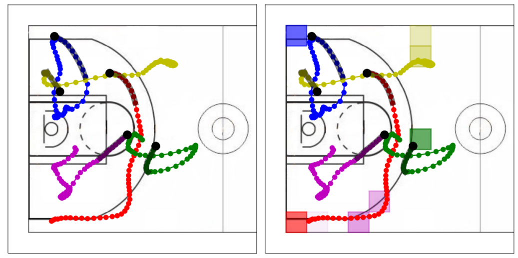
Data. We trained MAGnet on tracking data of professional basketball players (107,146 training and 13,845 test sequences). Each sequence has (x, y)-coordinates of 5 offensive basketball player trajectories for 50 frames (8 seconds) and takes place in the left half-court. We extract weak macro-goal labels $\hat{\mathbf{m}}_t^i$ for each player i as in Zheng et al. [2016]. The domain is segmented into a 10×9 grid of 5ft $\times$ 5ft cells, and $\hat{\mathbf{m}}_t^i$ is defined as the next cell in which player i is stationary (see ground truths in Figures 1, 2a). We represent each $\hat{\mathbf{m}}_t^i$ as a one-hot vector of dimension 90. A key property of macro-goals chosen in this way is that they change slowly over time relative to player positions.

[1] We omit feature extractors φ_τ^x and φ_τ^z from Eq. (3) for clarity.

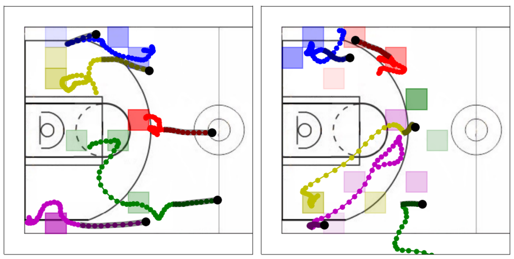
[2] A similar idea was used in MADDPG [Lowe et al., 2017], a multi-agent deep reinforcement learning method in which each agent maintains an approximation of the policies of all other agents.



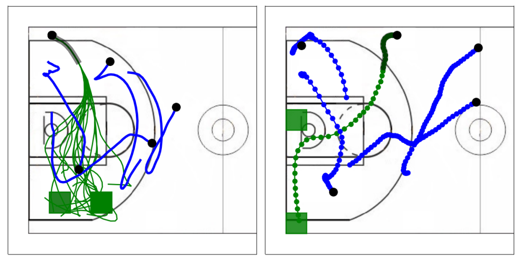
(a) **Left**: Ground truth trajectories from test set with weak macro-goal labels (boxes). Players reach their macro-goals along non-linear paths (green, purple). **Right**: Baseline rollout of representative quality. Common problems include players moving in the wrong direction (red) or out of bounds (purple, yellow, green). Players do not move cohesively as a team.



(b) **Left**: Rollout from MAGnet with the same burn-in as in (a). All players remain in bounds. The green player corrects its trajectory, whereas in (a) it goes off in the wrong direction. **Right**: Rollout from the left shown with its generated macro-goals. The locations of the macro-goals suggest that the players want to set up a formation along the 3-point line.



(c) More rollouts from MAGnet. **Left**: Macro-goal generation is stable and changes only a few times per rollout. Players often reach their macro-goals at some point in their trajectories. **Right**: Rare failure case: the green player moves out of bounds despite macro-goals generated in bounds. This is likely due to an under-representation of starting states in the data.



(d) Blue trajectories are ground truth. **Left**: The green player takes different paths towards the same macro-goals in 15 rollouts, suggesting that MAGnet captures the variability of the data. **Right**: Macro-goals are manually fixed to guide the green player towards the basket and then the bottom-left, demonstrating that macro-goals can control state predictions in rollouts.

Figure 2: 50-frame rollouts starting from the black dots. A 10-frame burn-in period is applied for all rollouts (unless otherwise stated as ground truth), marked by dark shading on the trajectories.

Details of Models. We combine MAGnet with VRRNs by modeling the conditional distributions of the agents and macro-goals in Eq. (5) as separate VRNNs. The baseline is a VRNN whose decoder splits into 5 separate decoders, one for each player, conditioned on the same latent variable $\mathbf{z}_t$. We use memory-less 2-layer fully-connected networks for priors, encoders, and decoders, and 2-layer GRU memory cells for hidden states. Both models have a latent space dimension of 80 (40 for macro-goals and 8 per agent in MAGnet), and are also conditioned on the previous positions of the players. We use a learning rate of 0.0005 and compare models that achieve the best log-likelihood on the test set.

Results. Both models achieve comparable quantitative performance (log-likelihood $\sim$ 2350 nats per test sequence), but rollouts from MAGnet are of significantly higher quality[3], shown and analyzed in Figure 2.[4] For instance, trajectories generated by MAGnet are much more realistic and cohesive as a team, whereas frequent problems exhibited by the baseline involve players moving in the wrong direction or out of bounds. Furthermore, we observe that: 1) macro-goals allow us to interpret each player's long-term goals and how they change over time (Figures 2b, 2c); 2) macro-goals influence a player's trajectory (Figure 2d); and 3) MAGnet captures the variability of the data (Figure 2d).

Future work. Our results suggest several directions for further investigation: 1) developing a better theoretical understanding of the optimal hierarchical latent structure; 2) learning MAGnet without weak macro-goal supervision; 3) validating MAGnet on other modalities and domains; and 4) exploring more probabilistic structures such that the model generalizes better with more agents (e.g. with the ball and defensive players), deeper hierarchies, and over longer time horizons.

[3]Higher log-likelihoods do not necessarily indicate higher quality of generated samples [Theis et al., 2015].

[4]More rollouts can be viewed at `https://ezhan94.github.io`.

References

J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio. A Recurrent Latent Variable Model for Sequential Data. *ArXiv e-prints*, June 2015.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015.

Stephan Zheng, Yisong Yue, and Patrick Lucey. Generating long-term trajectories using deep hierarchical networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1543–1551. Curran Associates, Inc., 2016. URL `http://papers.nips.cc/paper/6520-generating-long-term-trajectories-using-deep-hierarchical-networks.pdf`.

Alexander, Vezhnevets, V. Mnih, J. Agapiou, S. Osindero, A. Graves, O. Vinyals, and K. Kavukcuoglu. Strategic Attentive Writer for Learning Macro-Actions. *ArXiv e-prints*, June 2016.

R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. *ArXiv e-prints*, June 2017.

L. Theis, A. van den Oord, and M. Bethge. A note on the evaluation of generative models. *ArXiv e-prints*, November 2015.