Goal Recognition in Latent space

Leonardo Amado*, **João Paulo Aires***, **Ramon Fraga Pereira***
Maurício C. Magnaguagno*, **Roger Granada*** **and Felipe Meneguzzi**†
Pontifical Catholic University of Rio Grande do Sul (PUCRS) – Brazil
✉ *{leonardo.amado, joao.aires.001, ramon.pereira, mauricio.magnaguagno, roger.granada}@acad.pucrs.br
✉ †felipe.meneguzzi@pucrs.br

Pontifical Catholic University of Rio Grande do Sul

Motivation and Goals

- Goal recognition approaches have progressively relaxed requirements regarding:
  - amount and accuracy of domain knowledge; and
  - amount and accuracy of available observations at recognition time.
- However, to recognize goals using raw data, to infer possible goals recent approaches need either:
  - near-flawless human engineered domain knowledge
  - samples of behavior that account for almost all actions being observed
- This is too strong for most real-world applications.
- We develop an approach that leverages advances in recurrent neural networks to perform goal recognition as a classification task.

Main contributions.

- An end-to-end machine learning technique for goal recognition based on training an LSTM network.
- Comparison of the performance and trade-offs of resulting approach with traditional goal recognition approaches.

Goal Recognition in Latent Space

Goal recognition in Latent Space is a technique to apply classical goal recognition algorithms in raw data (such as images) by converting it into latent representation. It consists of four key processes

1. Train an autoencoder capable of creating a latent representation to a state of such image domain.
2. Derive a PDDL domain, by extracting the transitions of such domain when encoded in latent space, obtaining a domain $\mathcal{D}$.
3. Convert to a latent representation a set of images representing the initial state $\mathcal{I}$, the set of facts $\mathcal{F}$, the observations $\mathcal{O}$, and a set of possible goals $\mathcal{G}$, where the hidden goal G^* is included.
4. Apply goal recognition techniques using the computed tuple $\langle \mathcal{D}, \mathcal{F}, \mathcal{I}, \mathcal{G}, O \rangle$

Our goal is to obviate the need of the second process by avoiding the intermediary PDDL representation and train an LSTM to recognize goals, instead of using classical goal recognition algorithms.

LSTM Approach

At the center of our approach, we use an LSTM network to infer goals that reasons over an auto-encoded representation of the domain:

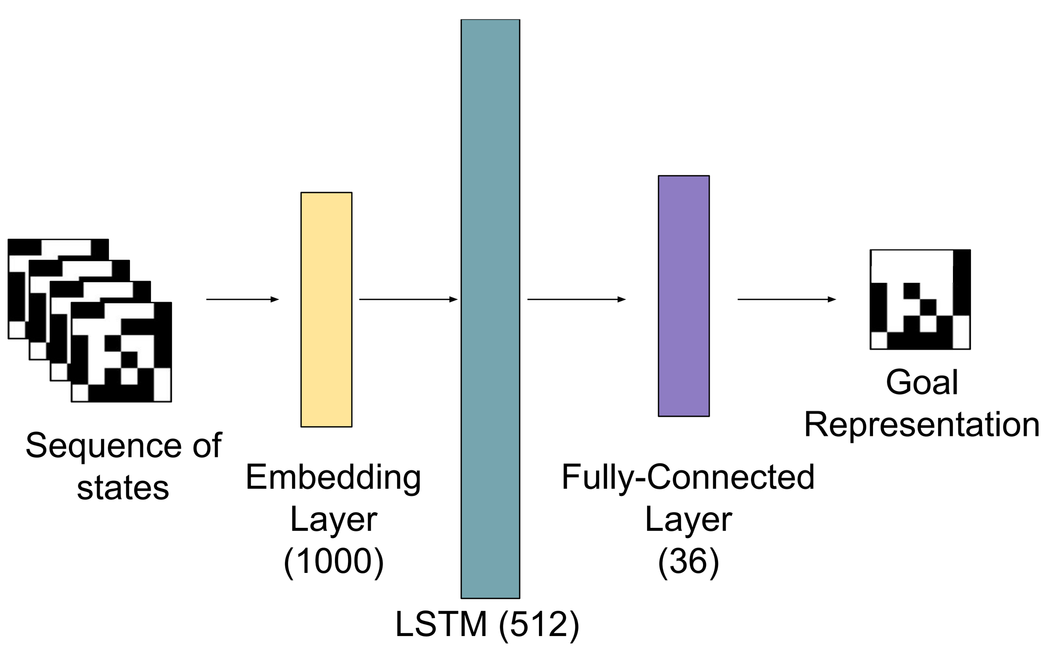

Figure 1: LSTM architecture

This LSTM lies at the center of a latent goal recognition architecture, obviating the need of a PDDL domain.

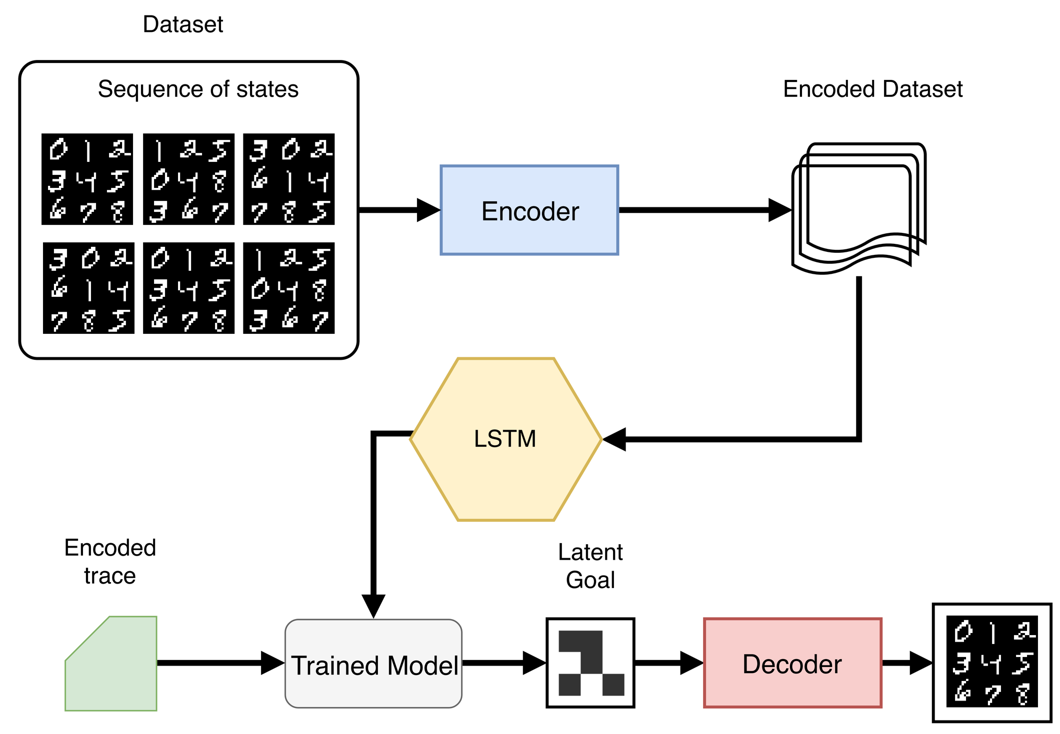

Figure 2: Latent goal recognition structure.

The inferred goal, in the encoded representation, can be decoded to an image representation.

Experiments

To validate our approach, we use six domains from three distinct games. We compare our approach to the state-of-the-art in goal recognition in latent space.

				POM (h_{gc})			LSTM			RG		
Domain	$\|\mathcal{G}\|$	(%) **Obs**	$\|O\|$	**Time (s))** θ (0 / 10)	**Accuracy** % θ (0 / 10)	**Spread in** $\mathcal{G}$ θ (0 / 10)	**Time (s)**	**Accuracy** %	**Spread in** $\mathcal{G}$	**Time (s)**	**Accuracy** %	**Spread in** $\mathcal{G}$
MNIST	6.0	10	1.2	0.591 / 0.603	33.3% / 83.3%	1.6 / 4.0	0.346	16.6%	1.0	21.25	100.0%	6.0
		30	3.0	0.612 / 0.625	33.3% / 83.3%	1.4 / 2.8	0.335	100.9%	1.0	22.26	100.0%	4.8
		50	4.0	0.673 / 0.677	60.0% / 100.0%	2.2 / 3.0	0.326	100.0%	1.0	22.48	100.0%	4.8
		70	5.8	0.698 / 0.703	100.0% / 100.0%	2.4 / 3.0	0.394	100.0%	1.0	23.53	100.0%	3.2
		100	7.8	0.724 / 0.730	100.0% / 100.0%	2.4 / 3.0	0.357	100.0%	1.0	26.34	100.0%	3.4
Mandrill	6.0	10	1.8	0.013 / 0.014	16.6% / 83.3%	1.0 / 3.8	0.335	50%	1.0	1.02	83.3%	5.6
		30	4.8	0.015 / 0.017	16.6% / 100.0%	1.0 / 4.8	0.366	100.0%	1.0	1.38	83.3%	3.8
		50	6.0	0.018 / 0.018	33.3% / 83.3%	1.1 / 4.8	0.389	100.0%	1.0	1.44	83.3%	4.1
		70	8.1	0.020 / 0.021	50.0% / 83.3%	1.3 / 4.3	0.353	100.0%	1.0	1.68	66.6%	1.8
		100	11.3	0.022 / 0.023	66.6% / 100.0%	1.8 / 5.16	0.347	100.0%	1.0	1.71	66.6%	1.8
Spider	6.0	10	1.5	0.166 / 0.178	33.3% / 66.6%	2.3 / 4.8	0.375	83.3%	1.0	1.35	83.3%	4.1
		30	4.0	0.181 / 0.190	66.6% / 66.6%	4.1 / 5.1	0.423	83.3%	1.0	1.57	83.3%	3.0
		50	5.6	0.193 / 0.199	50.0% / 83.3%	3.5 / 5.5	0.431	100.0%	1.0	1.66	83.3%	2.8
		70	7.5	0.201 / 0.205	83.3% / 83.3%	4.6 / 5.5	0.384	100.0%	1.0	1.79	66.6%	2.3
		100	10.5	0.208 / 0.217	100.0% / 100.0%	5.5 / 6.0	0.368	100.0%	1.0	2.04	66.6%	1.1
LO Digital	6.0	10	1.0	0.831 / 0.902	33.3% / 33.3%	1.5 / 3.0	0.315	83.3%	1.0	42.52	100.0%	6.0
		30	1.6	0.884 / 1.09	33.3% / 66.6%	1.5 / 4.3	0.317	100.0%	1.0	43.07	100.0%	5.5
		50	2.5	0.915 / 1.13	33.3% / 83.3%	1.5 / 4.5	0.336	100.0%	1.0	43.41	83.3%	5.1
		70	3.6	0.970 / 1.19	83.3% / 100.0%	3.6 / 4.5	0.371	83.3%	1.0	43.78	100.0%	4.8
		100	4.3	1.12 / 1.24	100.0% / 100.0%	2.6 / 4.3	0.330	83.3%	1.0	43.91	100.0%	4.8
LO Twisted	6.0	10	1.0	1.16 / 1.21	16.6% / 16.6%	1.0 / 3.0	0.376	66.6%	1.0	121.97	100.0%	5.8
		30	1.6	1.25 / 1.39	16.6% / 50.0%	1.0 / 3.8	0.320	100.0%	1.0	123.92	100.0%	5.0
		50	2.1	1.33 / 1.46	16.6% / 50.0%	1.0 / 4.5	0.339	100.0%	1.0	124.42	100.0%	5.6
		70	3.3	1.48 / 1.50	16.6% / 83.3%	1.0 / 3.3	0.312	100.0%	1.0	127.22	100.0%	5.5
		100	4.3	1.57 / 1.62	100.0% / 100.0%	2.3 / 5.0	0.327	100.0%	1.0	129.99	100.0%	5.5
Hanoi	4.0	10	1.0	0.304 / 0.318	33.3% / 66.6%	1.0 / 2.3	0.334	66.6%	1.0	6.08	100.0%	4.0
		30	3.0	0.316 / 0.320	100.0% / 100.0%	4.0 / 4.0	0.365	100.0%	1.0	6.21	100.0%	4.0
		50	4.3	0.322 / 0.337	100.0% / 100.0%	4.0 / 4.0	0.371	100.0%	1.0	7.01	66.6%	3.3
		70	6.0	0.345 / 0.354	100.0% / 100.0%	4.0 / 4.0	0.372	66.6%	1.0	7.26	100.0%	4.0
		100	8.3	0.354 / 0.362	100.0% / 100.0%	4.0 / 4.0	0.329	66.6%	1.0	8.19	100.0%	4.0

The LSTM approach has better accuracy in most domains and does not suffer from high spread. However our approach requires training and is unable to detect goals that are not present the in the training dataset.

Contributions and Conclusions

We developed an approach for goal recognition in latent space using an LSTM network, obviating the need for human engineering to create a task for goal recognition. In summary the advantages of using our LSTM approach to recognize goals are:

- high accuracy and fast prediction time when dealing with known goals;
- no false positive predictions, given that it only predicts a single goal;
- no need of a PDDL domain.

And the disadvantages are:

- performance is tied to the robustness of the training dataset;
- requires training, which is unnecessary for classical goal recognition approaches;
- very limited generalizability with small datasets.