Learning Neural-Symbolic Descriptive Planning Models via Cube-Space Priors: The Voyage Home (to STRIPS)

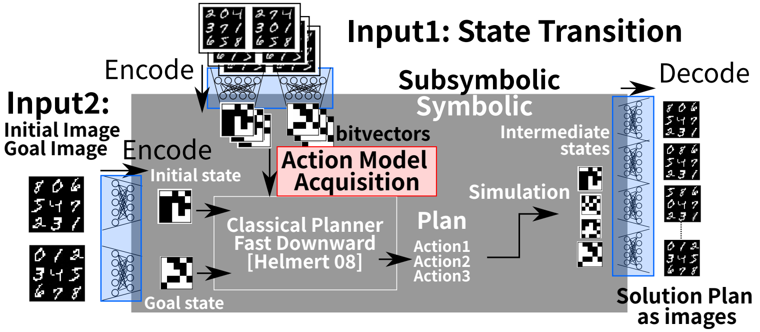
MIT-IBM Watson AI Lab

Masataro Asai, IBM Research, MIT-IBM Watson AILab
Christian Muise, U. Queens

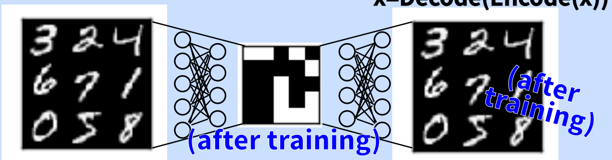
1. Our Objectives

Visually presented classical planning domains

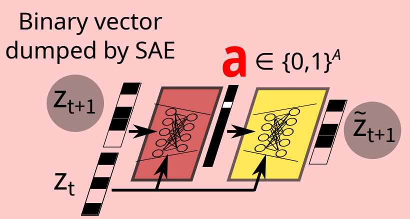
2. Image-based planning: LatPlan (AAAI18)



3. State AutoEncoder(SAE): Binary Conrete VAE



4. Action AutoEncoder(AAE): Gumbel Softmax VAE with Skip Connection



action $(z_t, z_{t+1}) = a$
"What happened in this pair ?"

apply $(z_t, a) = \tilde{z}_{t+1}$
"What will happen with this action?"

a: *A*-dim vector → will be one-hot (*A* actions)

Minimize $|z_{t+1} - \tilde{z}_{t+1}|$ → Action symbols $\{1,2,\dots A\}$

(reconstruction error)

5. LatPlan incompatible with SotA heuristics e.g. LMcut/M&S(Helmert et al, 09,14)

because AAE is a black-box function

→ Custom planner w/ Goal-Count
→ Not ideal

To improve the scalability of Latplan
We need a STRIPS-compatible AAE

5. The core elements missing in AAE

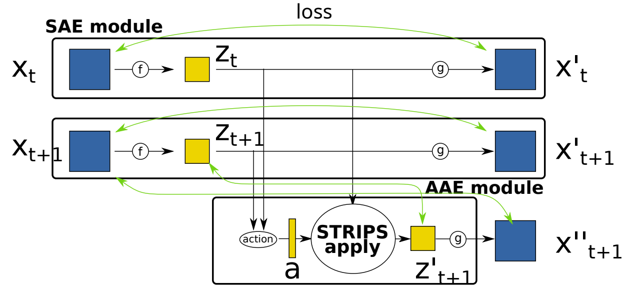
$$\text{AAE:} \quad z_{t+1} = \textit{apply}(z_t, a)$$
$$\text{STRIPS:} \quad z_{t+1} = (z_t \,/\, \textit{del}(a)) \cup \textit{add}(a)$$

Issue1: Effects must be disentangled from z_t.
→ apply may have conditional effects

Issue2: Z is fixed / Z and A are trained separately
→ Z may not have a compact STRIPS model

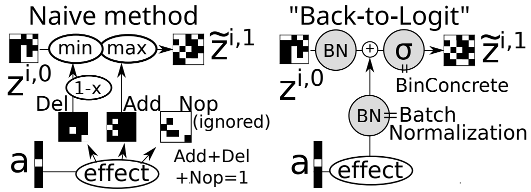
Solution: Train Z and A end-to-end
+ Restrict A to STRIPS
→ Z must adapt to A (STRIPS)

6. Cube-Space AE: end-to-end SAE+AAE



7. How to enforce STRIPS?

✗ Naive method (add/del using min/max)



Back-to-Logit: NNs prefer continuous vectors
Convert discrete → continuous (logit)
→ using Batch Normalization (Ioffe, Szegedy, 2015)
Continuous addition to compute effects
Discretize the result to obtain successor $z^{i,1}$
Guarantees STRIPS encoding (see theorem)

8. Results

Training performance

BtL ≫ min/max. Batchnorm is a must. End-to-end is a must

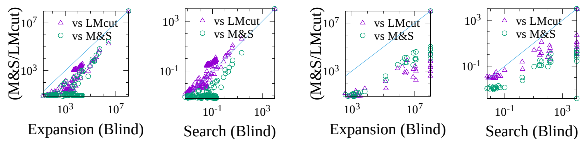
	(1) BtL Cube-Space AE			(2) Total loss			(3) Ablation study			(4) Direct loss under A = 300 → 100	
Domain	Rec.	Succ.	Direct	MinMax	Smooth	**BtL**	-BN	-Direct	-Succ.	Cube-Space AE	SAE+Cube-AAE
Hanoi	.001	.002	.001	.439	.436	**.003**	.019	.088	.501	.001 → .001	.001 → .002
LOut	.000	.000	.000	.506	.506	**.000**	.002	.239	.412	.001 → .008	.010 → .040
Twisted	.000	.001	.000	.458	.487	**.001**	.002	.154	.498	.000 → .002	.001 → .011
Mandrill	.000	.001	.001	.500	.600	**.002**	.035	.046	.495	.001 → .002	.000 → .006
Mnist	.000	.000	.000	.512	.506	**.001**	.002	.158	.462	.000 → .001	.002 → .005
Spider	.001	.001	.001	.607	.563	**.003**	.026	.073	.376	.003 → .002	.001 → .003

Planning performance

blind			gc			blind			gc				blind			gc			lama			LMcut			M&S		
f	v	o	f	v	o	f	v	o	f	v	o	Domain	f	v	o	f	v	o	f	v	o	f	v	o	f	v	o
0	0	0	8	4	3	1	0	0	19	0	0	LOut(30)	30	30	30	30	30	17	30	21	5	30	30	30	30	30	30
0	0	0	4	0	0	0	0	0	14	2	2	Twisted(30)	30	25	25	30	26	5	30	20	4	30	29	29	30	25	25
15	14	14	17	17	16	15	15	15	13	12	12	Mandrill(30)	30	29	13	30	18	9	30	23	11	30	29	13	30	30	13
0	0	0	0	0	0	0	0	0	0	0	0	Mnist(30)	30	25	24	30	4	4	30	7	7	30	25	24	30	26	24
8	8	8	4	4	4	3	3	1	13	3	0	Spider(30)	30	28	16	30	27	12	28	22	14	30	28	16	30	28	16
SAE+AAE+AD						SAE+CubeAAE+AD						15puzzle(40) →	32	32	29	28	9	9	38	13	3	38	38	33	38	38	33

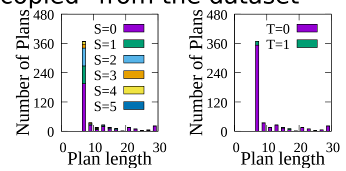
Plans f(ound), v(alid) and o(ptimal) by (left) AMA_2 and (right) Cube-Space AE. Best result except > 2 ties are in bold.

4x4 LightsOut and 8 puzzle   Mandrill 15-puzzle



- Cube-Space AE outperforms previous baselines (in f, v, o)
- LMcut/M&S significantly improves the performance vs. blind search

Strong generalization:
Plans do not contain examples "copied" from the dataset



9. Conclusion

- We proposed how to learn symbolic STRIPS model using NNs completely unsupervised.
- SotA planning heuristics **work out of the box w/o extra training + has optimality guarantee**
- ***Strong candidate for replacing model-free RL which require huge amout of training***