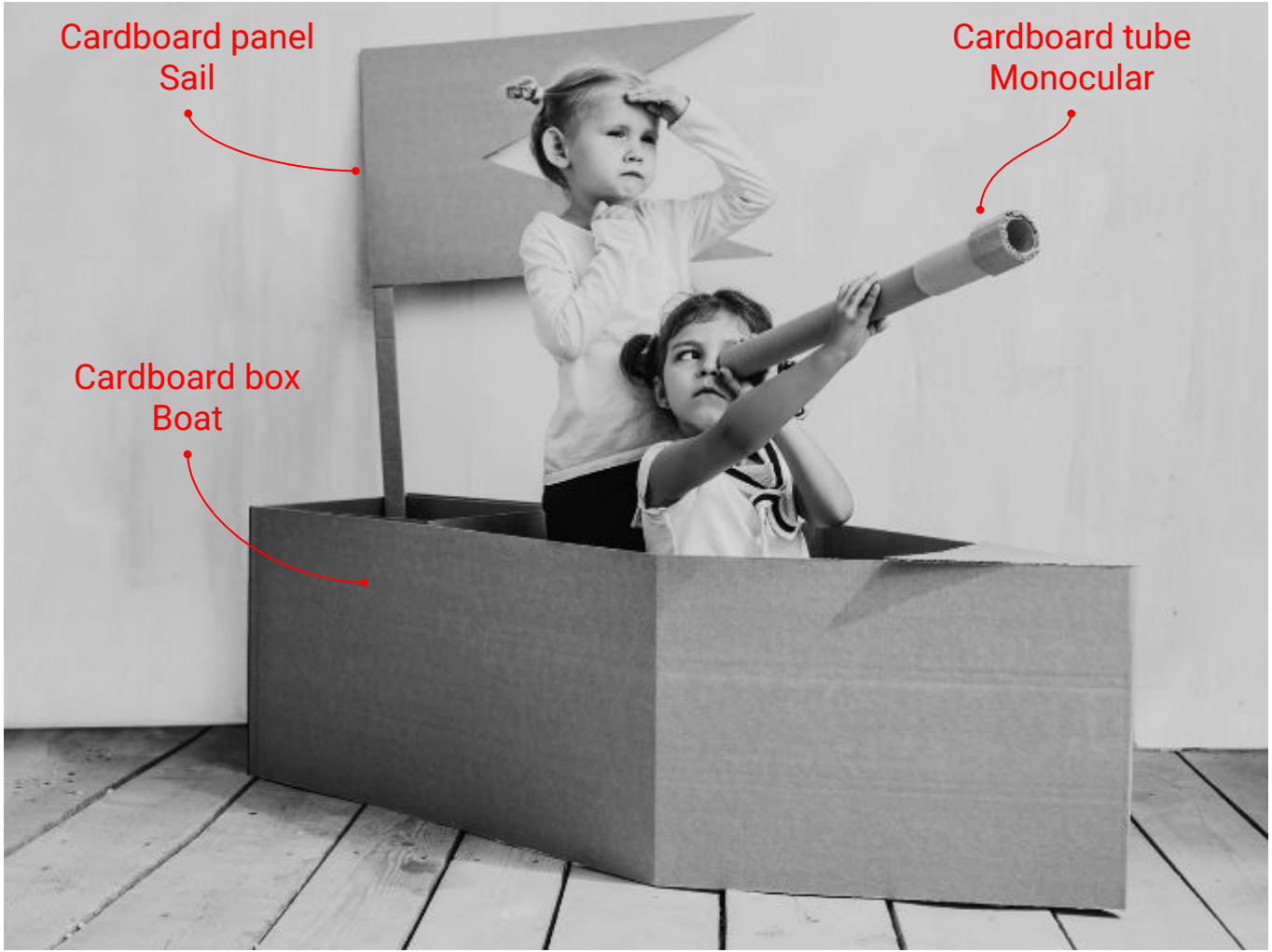
Imperial College London

Learning Invariants through Soft Unification

Nuri Cingillioglu, Alessandra Russo



Source: The Power of Pretend Play by Holly Pevzner Photo Credit: Dana Gallagher

Which symbols are variables?

How can we assign new values to them?

sandra is a lion
bernhard is a frog
mary is a lion
lily is a frog
mary is yellow
lily is green

---

what colour is john

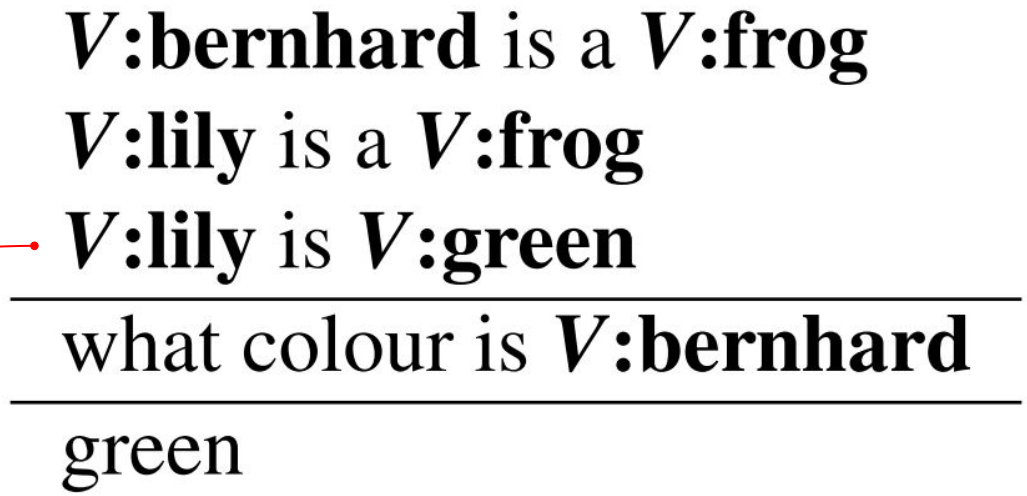
---

what colour is bernhard

---

green

Learning Invariants through Soft Unification



Variable with default symbol **lily**

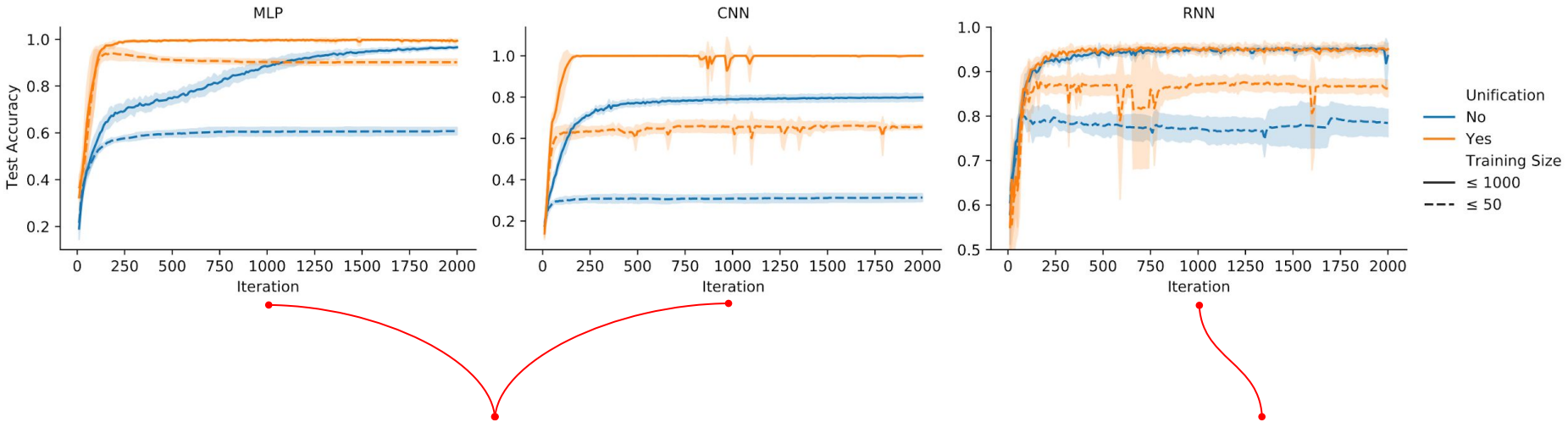
bAbI task 16. Single invariant can solve all of training, validation and test examples correctly by unifying variables with new values.

Dataset	Context	Query	Answer	Training Size
Sequence	8384	duplicate	8	$\leq$ 1k, $\leq$ 50
Grid	0 0 3 0 1 6 8 5 7	corner	7	$\leq$ 1k, $\leq$ 50
bAbI	Mary went to the kitchen. Sandra journeyed to the garden.	Where is Mary?	kitchen	1k, 50
Logic	p(X) $\leftarrow$ q(X). q(a).	p(a).	True	2k, 100
Sentiment A.	easily one of the best films	Sentiment	Positive	1k, 50

Sequences

Grid

Sentiment Analysis



Larger improvements in data efficiency in datasets with a clear pattern

Does not degrade performance in a real-world dataset

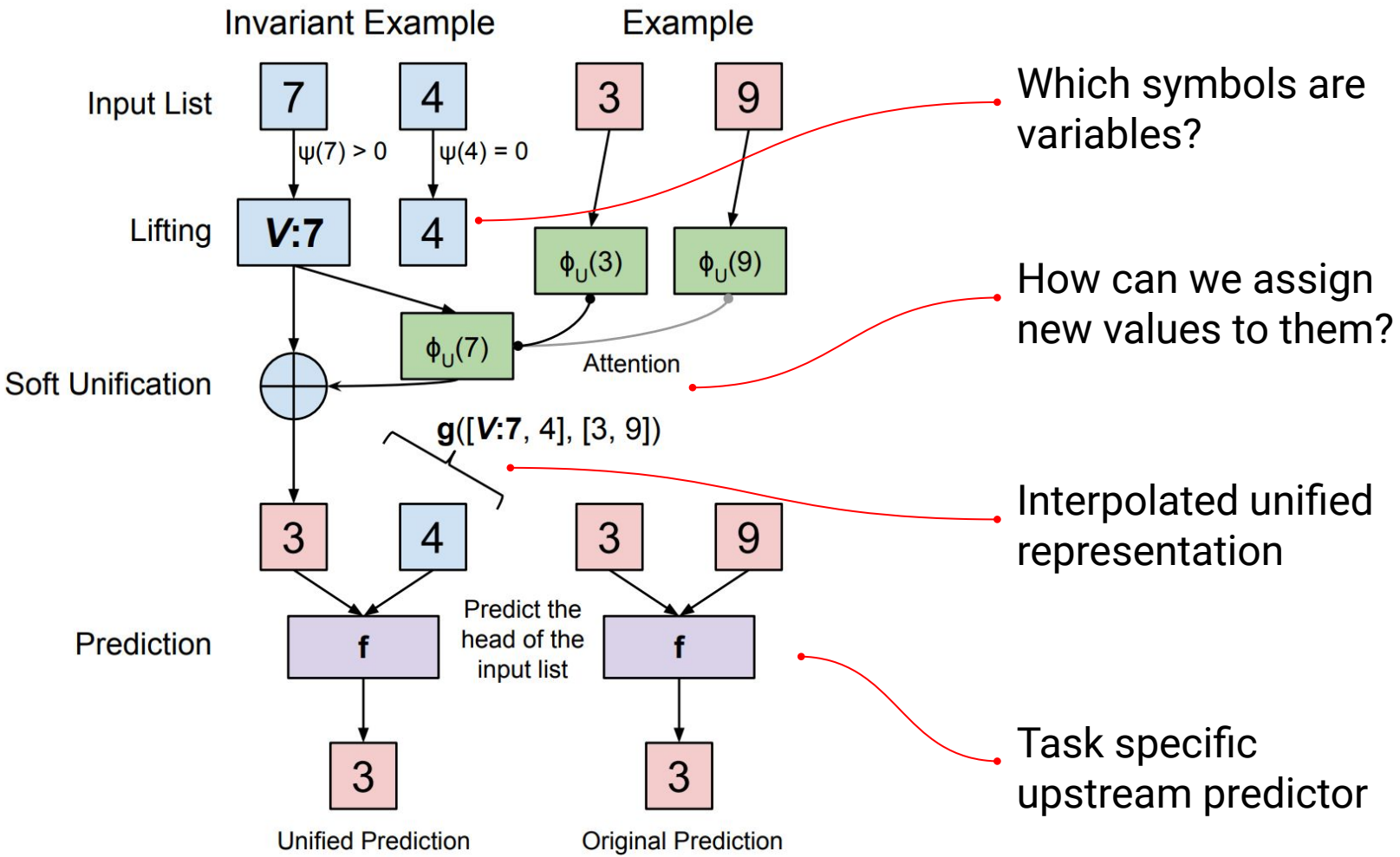
bAbI Dataset

Training Size	1k				50	1k				
Supervision	Weak		Strong			Weak				Strong
# Invs / Model	1	3	1	3	3	N2N	GN2N	EntNet	QRN	MemNN
Mean	18.8	19.0	**5.1**	6.6	28.7	13.9	12.7	29.6	11.3	6.7
# > 5%	10	9	3	3	17	11	10	15	5	4

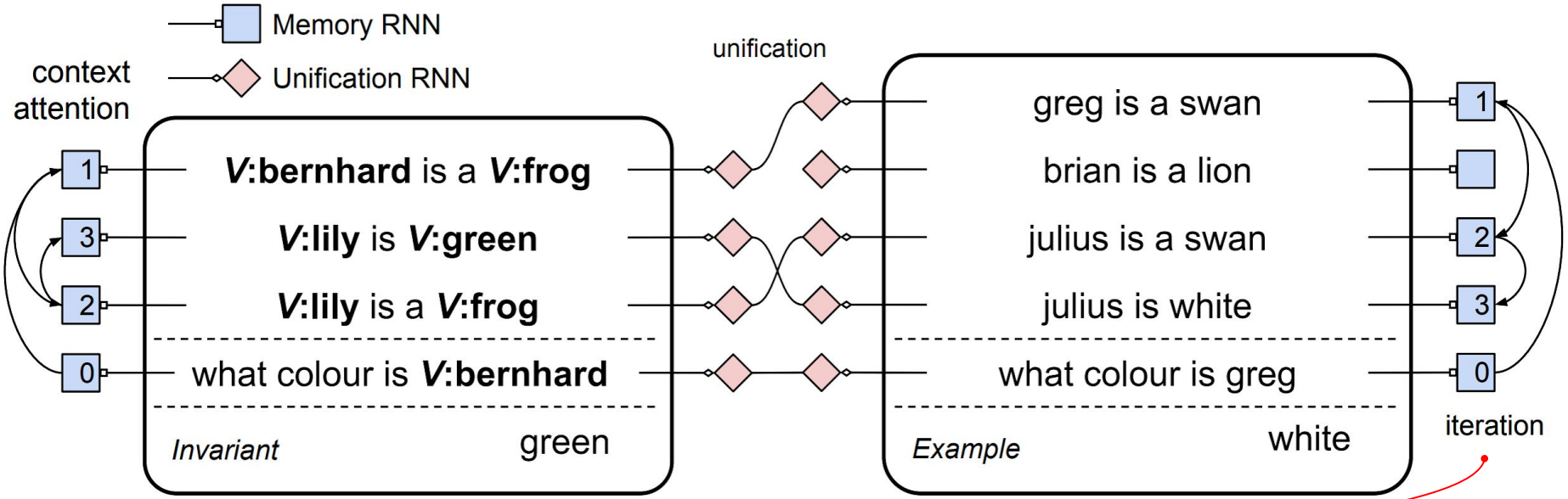
Unification Memory Networks perform competitively against baselines with minor improvements in performance

Logical Reasoning Dataset

Model	UMN					IMA	
Training Size	2k				100	2k	
Supervision	Weak		Strong			Weak	Strong
# Invs	1	3	1	3	3	-	
Mean	37.7	37.6	**27.4**	29.0	47.1	38.8	31.5
# > 5%	10	10	10	11	12	11	11


Invariant Example
Example
Input List
7
4
3
9
ψ(7) > 0
ψ(4) = 0
Lifting
V:7
4
ϕU(3)
ϕU(9)
ϕU(7)
Attention
Soft Unification
g([V:7, 4], [3, 9])
3
4
3
9
Prediction
f
Predict the head of the input list
f
3
3
Unified Prediction
Original Prediction
Which symbols are variables?
How can we assign new values to them?
Interpolated unified representation
Task specific upstream predictor

Unify two sentences at each iteration, interleave **f** and **g**

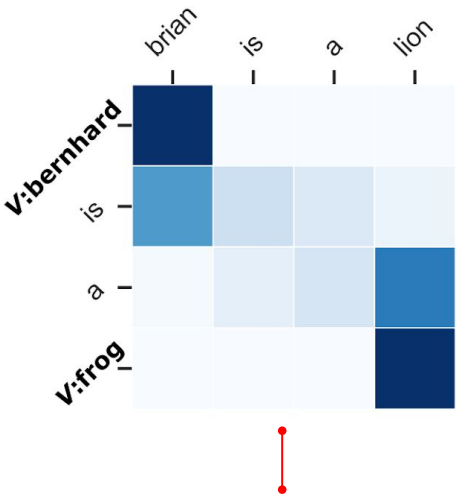


Pre-train **f** so that the memory network selects the correct sentences to unify

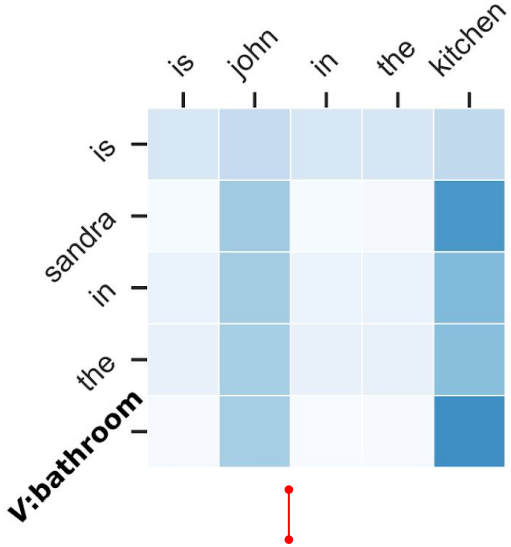
Add sparsity constraint to avoid the trivial solution: everything is a variable and invariant gets completely replaced

$$J = \overbrace{\lambda_K \mathcal{L}_{\text{nll}}(f)}^{\text{Original output}} + \lambda_I [\ \overbrace{\mathcal{L}_{\text{nll}}(f \circ g)}^{\text{Unification output}} + \overbrace{\tau \sum_{s \in \mathbb{S}} \psi(s)}^{\text{Sparsity}}\ ]$$

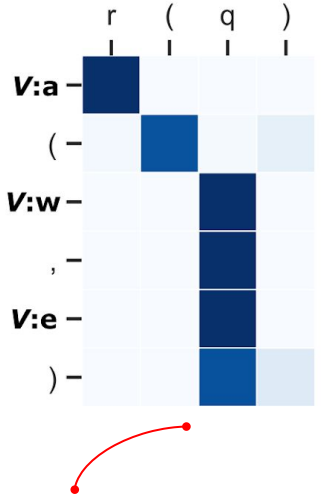
Adjust lambda K and I to pre-train or serve as a baseline



One-to-one mapping, bernhard interpolated towards brian



One-to-many, the model may *squeeze* the information to achieve sparsity



Many-to-one, upstream f learns to differentiate p(q,q) and p(q)

Invariants capture nothing about how the model actually solves the tasks or utilise the interpolated unified representations

Thank you

https://github.com/nuric/softuni