The Neural Data Router: Adaptive Control Flow in Transformers Improves Systematic Generalization

Scuola universitaria professionale della Svizzera italiana SUPSI · Università della Svizzera italiana · IDSIA

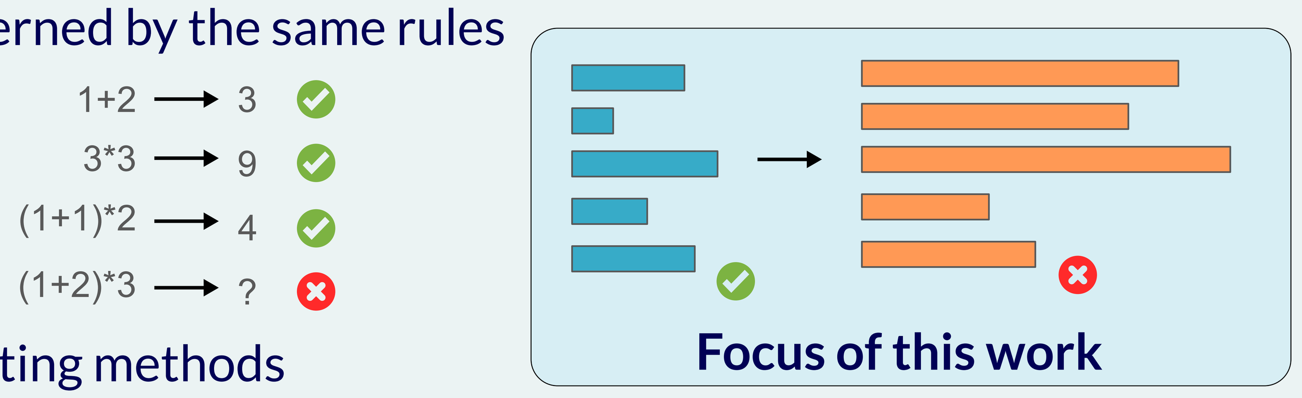
Róbert Csordás robert@idsia.ch

Kazuki Irie kazuki@idsia.ch

Jürgen Schmidhuber juergen@idsia.ch

Systematic generalization

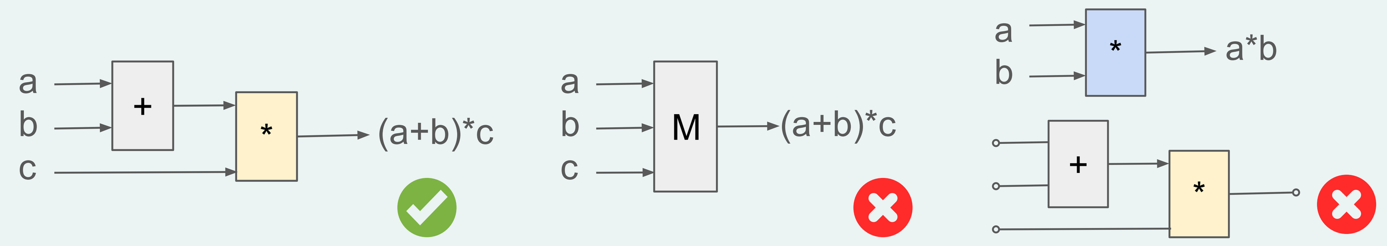
- Ability to perform well well on **systematically different** inputs, governed by the same rules



- Existing methods
  1. Neural networks with supervised learning - usually fail
  2. Meta-learning: helps a bit, but far from ideal
  3. Neuro-symbolic hybrids: work well, but task-specific
- Ideally, we would want a learning-based method that works well
- Generalization is difficult
  - There is no optimization pressure to generalize
    - Any solution, including memorization, is good enough from the perspective of optimization
  - Only algorithmic solutions will generalize
  - We need non-restrictive algorithmic biases

Hypotheses

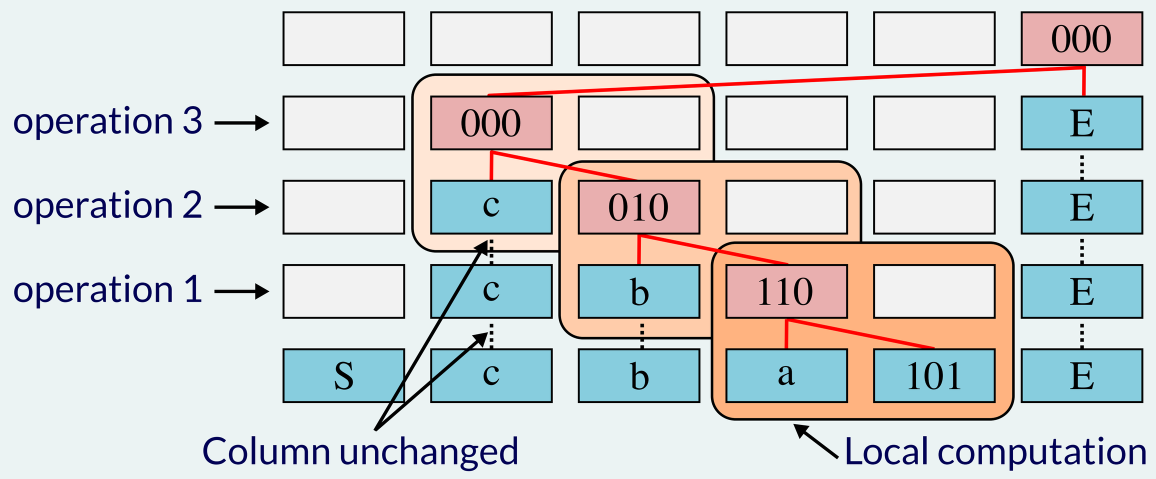
1. The basis of generalization should be compositionality



- Decompose problems to elementary operations
- In Transformers, the output of an operation is available only to the successive layers.
- Since operations should be composable in any order, layers should be shared.
- There should be at least as many layers as the depth of the computation graph of the underlying problem

2. Computation should be performed only when necessary
   - Columns should be kept unchanged until it is their turn to be processed
   - Keeping columns unchanged should be easy
3. Long computations are often made of multiple local computations
   - Bias, but not restrict to local computation

Free control flow: each column can decide what to do based on its own state and its neighborhood. This is a dataflow architecture.



Method

1. Copy gate
   - **Allows to skip the whole transformation**
   - Similar to Transformers

$$\boldsymbol{a}^{(i,t+1)} = \mathrm{LayerNorm}(\mathrm{MultiHeadAttention}(\boldsymbol{h}^{(i,t)}, \mathbf{H}_t, \mathbf{H}_t) + \boldsymbol{h}^{(i,t)})$$

$$\hat{\boldsymbol{h}}^{(i,t+1)} = \mathrm{LayerNorm}(\mathrm{FFN}^{\mathrm{data}}(\boldsymbol{a}^{(i,t+1)}))$$

No residual connection

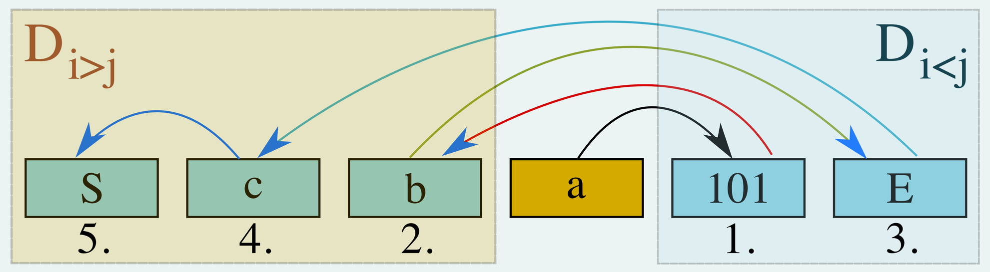
   - New: copy gate

Parallel branches

$$\boldsymbol{g}^{(i,t+1)} = \sigma(\mathrm{FFN}^{\mathrm{gate}}(\boldsymbol{a}^{(i,t+1)}))$$

$$\boldsymbol{h}^{(i,t+1)} = \boldsymbol{g}^{(i,t+1)} \odot \hat{\boldsymbol{h}}^{(i,t+1)} + (1 - \boldsymbol{g}^{(i,t+1)}) \odot \boldsymbol{h}^{(i,t)}$$

2. Geometric attention
   - **Bias towards attending to the nearest match**
   - Define an order of preference of nodes



   - Use sigmoid instead of softmax

$$\boldsymbol{P}_{i,j} = \sigma(\boldsymbol{k}^{(j)\top}\boldsymbol{q}^{(i)})$$

   - The final attention score is then the probability of attending to the node, multiplied by the probability of not attending to any closer ones

$$\boldsymbol{A}_{i,j} = \boldsymbol{P}_{i,j} \prod_{k \in \mathbb{S}_{i,j}} (1 - \boldsymbol{P}_{i,k})$$

   - No positional information, just direction

$$\boldsymbol{D}_{i,j} = \begin{cases} \boldsymbol{W}_{\mathrm{LR}}\boldsymbol{h}^{(i)} + b_{\mathrm{LR}}, & \text{if } i \leq j \\ \boldsymbol{W}_{\mathrm{RL}}\boldsymbol{h}^{(i)} + b_{\mathrm{RL}}, & \text{if } i > j \end{cases}$$

Neural Data Router (NDR): copy gate + geometric attention + shared layers + sufficient depth

Results

- Compositional Table Lookup (CTL) - diagnostic dataset
  - Input symbols: 3-bit binary strings (single symbol)
  - Single argument bijective functions: letters
  - Direction: *101 a b c* vs *c b a 101*
  - Models should generalize to longer *length independent of the presentation direction of the input*

	IID		Longer	
Model	Forward	Backward	Forward	Backward
LSTM	**1.00 ± 0.00**	0.59 ± 0.03	**1.00 ± 0.00**	0.22 ± 0.03
Bidirectional LSTM	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
DNC	**1.00 ± 0.00**	0.57 ± 0.06	**1.00 ± 0.00**	0.18 ± 0.02
Transformer	**1.00 ± 0.00**	0.82 ± 0.39	0.13 ± 0.01	0.12 ± 0.01
+ rel	**1.00 ± 0.00**	**1.00 ± 0.00**	0.23 ± 0.05	0.13 ± 0.01
+ rel + gate	**1.00 ± 0.00**	**1.00 ± 0.00**	**0.99 ± 0.01**	0.19 ± 0.04
+ abs/rel + gate	**1.00 ± 0.00**	**1.00 ± 0.00**	**0.98 ± 0.02**	**0.98 ± 0.03**
+ geom. att.	0.96 ± 0.04	0.93 ± 0.06	0.16 ± 0.02	0.15 ± 0.02
+ geom. att. + gate (NDR)	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**

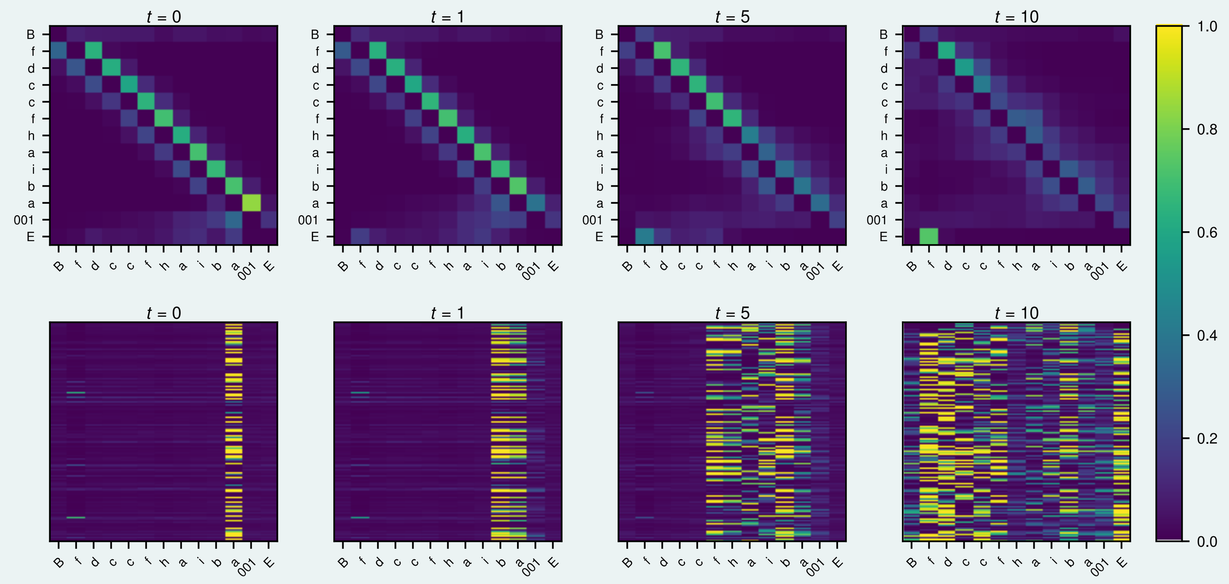
- Simple Arithmetics (modulo 10) Example: *(((3+2)*5)+(8*4)) = 7*
- ListOPS Example: *[MAX 2 9 [MIN 4 7] 0] = 9*

	Simple Arithmetics		ListOPS	
	IID (1..5)	Test (7..8)	IID (1..5)	Test (7..8)
LSTM	**0.99 ± 0.00**	0.74 ± 0.02	**0.99 ± 0.00**	0.71 ± 0.03
Bidirectional LSTM	**0.98 ± 0.01**	0.82 ± 0.06	**1.00 ± 0.00**	0.57 ± 0.04
Transformer	**0.98 ± 0.01**	0.47 ± 0.01	0.98 ± 0.00	0.74 ± 0.03
+ rel	**1.00 ± 0.00**	0.77 ± 0.04	**0.98 ± 0.01**	0.79 ± 0.04
+ abs/rel + gate	**1.00 ± 0.01**	0.80 ± 0.16	**1.00 ± 0.01**	0.90 ± 0.06
+ geom. att. + gate (NDR)	**1.00 ± 0.00**	**0.98 ± 0.01**	**1.00 ± 0.00**	**0.99 ± 0.01**

- **NDR shows near-perfect length generalization**

Analysis

- Analysis shows that the gates open when the operation of a given column should be performed (shown below for CTL task)



- Ablation study on CTL task shows that as soon as the number of layers falls below the computation depth, the model fails to generalize

	IID		Test	
n_{layers}	Forward	Backward	Forward	Backward
14	1.00 ± 0.00	1.00 ± 0.00	**1.00 ± 0.00**	**1.00 ± 0.00**
12	1.00 ± 0.00	1.00 ± 0.00	**1.00 ± 0.00**	**0.99 ± 0.02**
10	1.00 ± 0.00	1.00 ± 0.00	0.75 ± 0.04	0.62 ± 0.05 ←
8	1.00 ± 0.00	1.00 ± 0.00	0.23 ± 0.02	0.24 ± 0.03
6	1.00 ± 0.00	0.96 ± 0.03	0.22 ± 0.05	0.15 ± 0.01
4	0.96 ± 0.04	0.68 ± 0.11	0.14 ± 0.01	0.13 ± 0.01