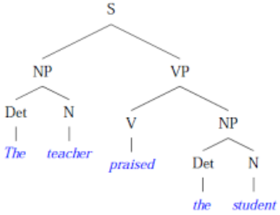
Learning compositionally through attentive guidance

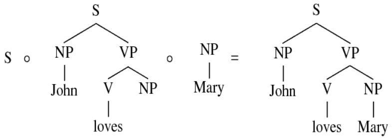
Dieuwke Hupkes

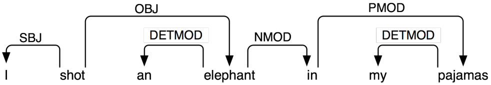
Institute for Logic, Language and Computation
University of Amsterdam

June 12, 2018

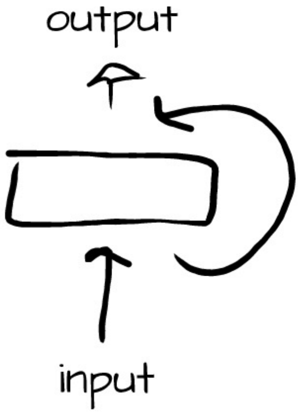
Structures in language







Neural networks

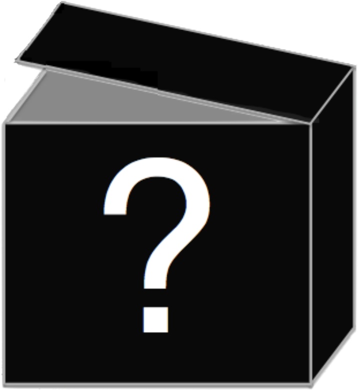


The successes of neural networks

They work very well:

- Machine translation
- Syntactic parsing
- Semantic role labelling
- Language modelling

The downside of neural networks



The downside of neural networks

- They are not useful as explanatory models of language
- We don't know how they relate to linguistic theories of language
- We don't know how to improve them (other than by applying engineering tricks)

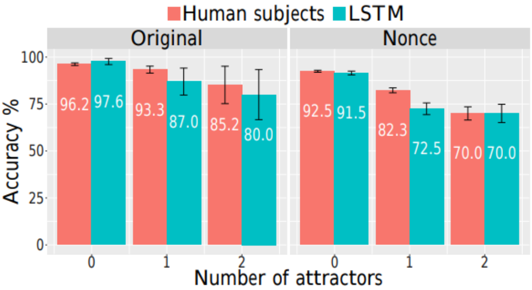
The downside of neural networks

- They are not useful as explanatory models of language
- We don't know how they relate to linguistic theories of language
- We don't know how to improve them (other than by applying engineering tricks)
- **Actually, we don't even have any idea what they encode**

What do we do?

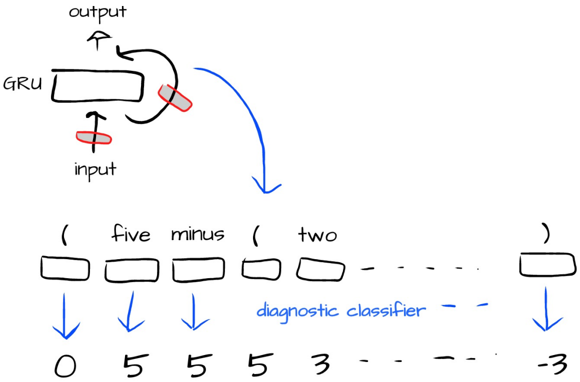
- We wait for the engineers to solve it
- We try to increase our understanding of what these networks are encoding
- We try to find new ways to make them behave more human-like

Linguistically



Gulordava et al. (2018)

Structurally



Hupkes et al. (2018b)

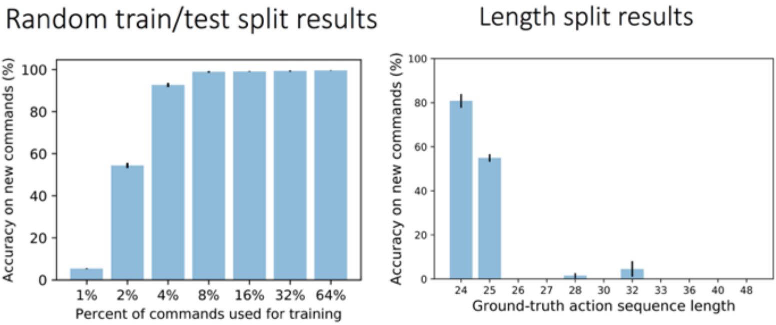
On a behaviour level

jump	⇒	JUMP
jump left	⇒	LTURN JUMP
jump around right	⇒	RTURN JUMP RTURN JUMP RTURN JUMP RTURN JUMP
turn left twice	⇒	LTURN LTURN
jump thrice	⇒	JUMP JUMP JUMP
jump opposite left and walk thrice	⇒	LTURN LTURN JUMP WALK WALK WALK
jump opposite left after walk around left	⇒	LTURN WALK LTURN WALK LTURN WALK LTURN WALK LTURN LTURN JUMP

Figure 1: Examples of SCAN commands (left) and the corresponding action sequences (right).

Lake and Baroni (2017)

Behaviourally



Thus?

- Networks can pick up on interesting (hierarchical) patterns
- We have some methods to look inside networks
- Networks are powerfull generalisation machines
- But: they don't do this in a human understandable way

Thus?

Pattern matching goes a long way

t_1	t_2	...	t_N
00 → 10	00 → 00	... → ...	... → ...
01 → 11	01 → 10	... → ...	... → ...
10 → 00	10 → 01	... → ...	... → ...
11 → 01	11 → 11	... → ...	... → ...

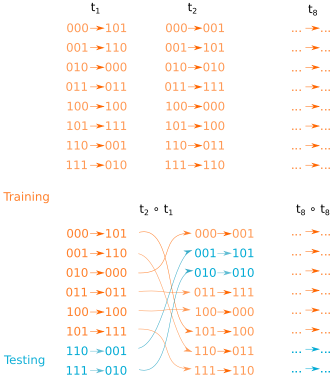
$t_2 \circ t_1$

t_1	t_2
00 → 10	00 → 00
01 → 11	01 → 10
10 → 00	10 → 01
11 → 01	11 → 11

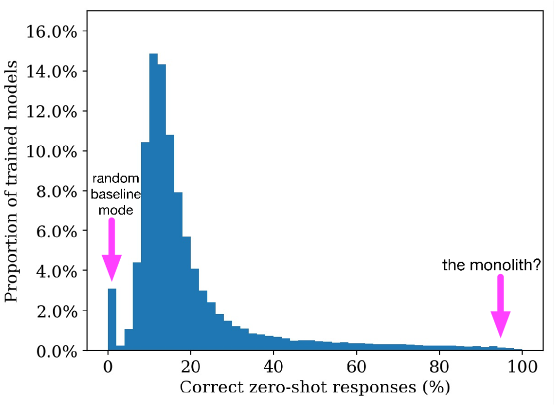
Liška et al. (2018)

Experimental setup

- Data: 8 randomly generated 3-bit atomic tasks and corresponding 64 composed tasks.
- Training on all atomic tasks and 6 out 8 inputs of composed tasks, and test on 2 held-out inputs (totaling 128 test compositions).



How do neural networks do?



Conclusion

1. Some RNNs find a generalising solution
2. Most networks do not exhibit systematic compositionality

Attentive Guidance

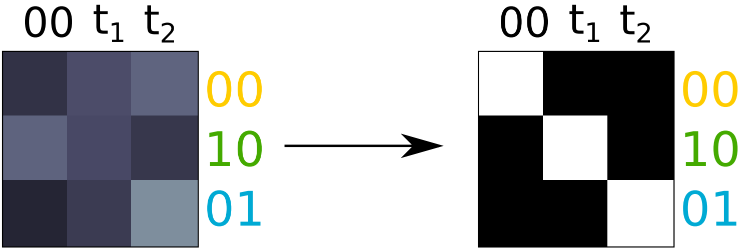
Hupkes et al. (2018a)

Attentive Guidance

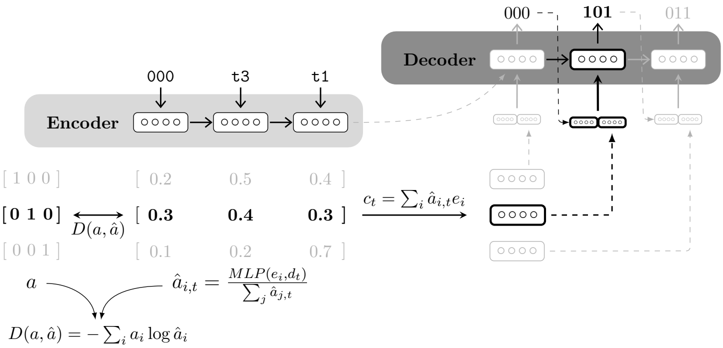
Hupkes et al. (2018a)

An important part of the training will consist in the teacher's pointing to the objects, directing the child's attention to them, and at the same time uttering a word; for instance, the word "slab" as he points to that shape.
Philosophical Investigations
L. Wittgenstein

Intuition



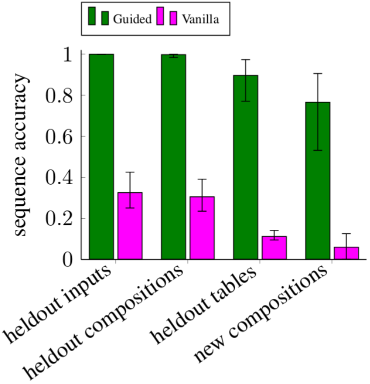
Supervise the attention mask of the network to match a compositional readout of the input.


Decoder
000
101
011
Encoder
000
t3
t1
[1 0 0]
[0.2 0.5 0.4]
[0 1 0]
[0.3 0.4 0.3]
[0 0 1]
[0.1 0.2 0.7]
$D(a, \hat{a})$
$c_t = \sum_i \hat{a}_{i,t} e_i$
a
$\hat{a}_{i,t} = \frac{MLP(e_i, d_t)}{\sum_j \hat{a}_{j,t}}$
$D(a, \hat{a}) = -\sum_i a_i \log \hat{a}_i$

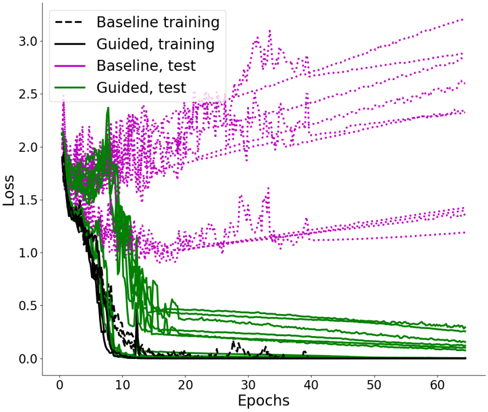
Data

- **Training** 6 out of 8 inputs in 28 compositions unseen: t_1 t_2 110
- **Heldout inputs** 2 out of 8 inputs in 28 compositions unseen: e.g. t_1 t_2 010
- **Heldout compositions** 8 entirely unseen compositions: t_1 t_3
- **Heldout tables** compositions with one of the two heldout tables: e.g. t_7 t_1 000
- **New compositions** compositions between the two heldout tables: e.g. t_7 t_8 000

Accuracies


Guided
Vanilla
sequence accuracy
1
0.8
0.6
0.4
0.2
0
heldout inputs
heldout compositions
heldout tables
new compositions

Overfitting

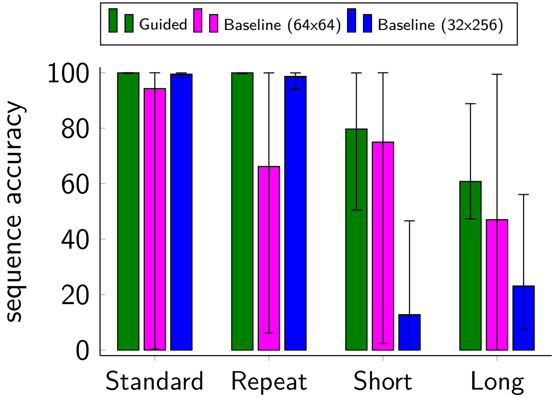


$$\mathcal{L}\colon X = \{A, B\},$$
$$Y_A = \{a_1, a_2, a_3\}, Y_B = \{b_1, b_2, b_3\}.$$
$$a_1 \to a_{11}|a_{12},\ \ a_2 \to a_{21}|a_{22},\ \ a_3 \to a_{31}|a_{32}$$
$$b_1 \to b_{11}|b_{12},\ \ b_2 \to b_{21}|b_{22},\ \ b_3 \to b_{31}|b_{32}$$

Input	**Valid output for $\mathcal{L}$**
AAB	$a_{21}a_{32}a_{12}a_{11}a_{22}a_{32}b_{13}b_{21}b_{32}$

Weber et al (2018)

Results

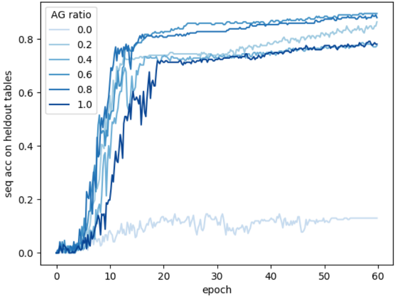

Guided
Baseline (64x64)
Baseline (32x256)
sequence accuracy
0
20
40
60
80
100
Standard
Repeat
Short
Long

What's next?

- Relaxing the need of guidance

What's next?

- Relaxing the need of guidance



Picture of Mathijs Mul

What's next?

- Relaxing the need of guidance
- Designing architectures that have compositional biases built in

What's next?

- Relaxing the need of guidance
- Designing architectures that have compositional biases built in

- Finding other tasks: What would be a convincing proof of compositionality?

Acknowledgments

Anand Kumar Singh

Kris Korrel

Elia Bruni

Germàn Kruszewski

Bibliography

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless green recurrent networks dream hierarchically. In *Proceedings of NAACL*, volume 1, pages 1195–1205, 2018.

Dieuwke Hupkes, Anand Singh, Kris Korrel, German Kruszewski, and Elia Bruni. Learning compositionally through attentive guidance, 2018a.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926, 2018b.

Brenden M. Lake and Marco Baroni. Still not systematic after all these years: On the compositional skills of sequence-to-sequence recurrent networks. *CoRR*, abs/1711.00350, 2017.

Adam Liška, Germán Kruszewski, and Marco Baroni. Memorize or generalize? searching for a compositional rnn in a haystack. *arXiv preprint arXiv:1802.06467*, 2018.