RNNs implicitly implement tensor-product representations:
Uncovering compositionality in neural network representations

R. Thomas McCoy,[1] Tal Linzen,[1] Ewan Dunbar,[2] and Paul Smolensky[3,1]
[1]Johns Hopkins University, [2]CNRS - Université Paris Diderot - Sorbonne Paris Cité, [3]Microsoft Research AI

1 Overview

- **The puzzle:** How do recurrent neural networks (RNNs) use vectors of continuous values to represent discrete symbolic structures?
- **Finding:** RNNs trained on structure-dependent tasks learn to implicitly implement tensor product representations.

2 Tensor Product Representations

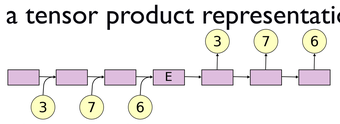
- A principled method for representing compositional symbolic structures in vector space (Smolensky 1990)
- Represent the input with pairs of fillers and roles:

  3,7,6 = 3:first + 7:second + 6:third

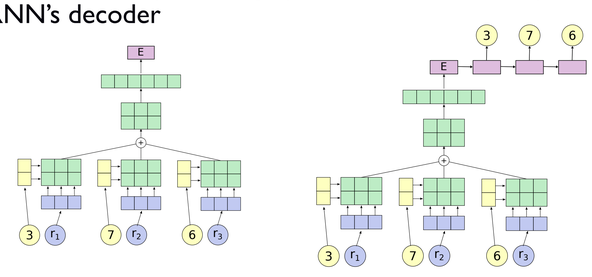
- Each filler f_i and role r_i has a vector embedding
- The representation of the input is the sum of the outer products of each f_i and r_i: $\sum f_i \otimes r_i$

3 Tensor Product Decomposition

- **Goal:** Approximate an RNN's learned encodings (such as E below) with a tensor product representation

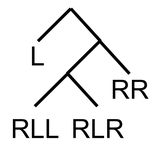


- **Approach:** (below, left) Train a model to generate tensor product representations that are close to the RNN's encodings
- **Evaluation:** (below, right) Pass this model's outputs to the RNN's decoder



4 Role Schemes

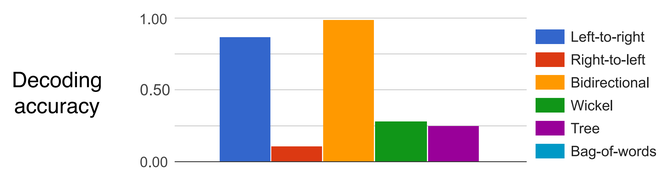
	3	1	1	6
Left-to-right	0	1	2	3
Right-to-left	3	2	1	0
Bidirectional	(0,3)	(1,2)	(2,1)	(3,0)
Wickelroles	#_1	3_1	1_6	1_#
Tree	L	RLL	RLR	RR
Bag-of-words	r_0	r_0	r_0	r_0



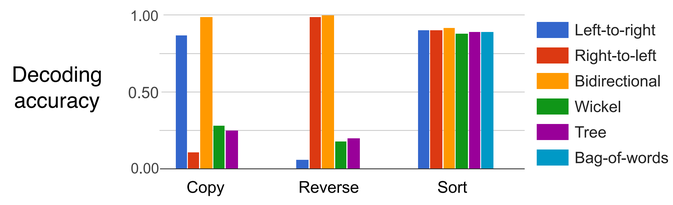
Tree used for tree roles

5 Digit Sequence Experiments

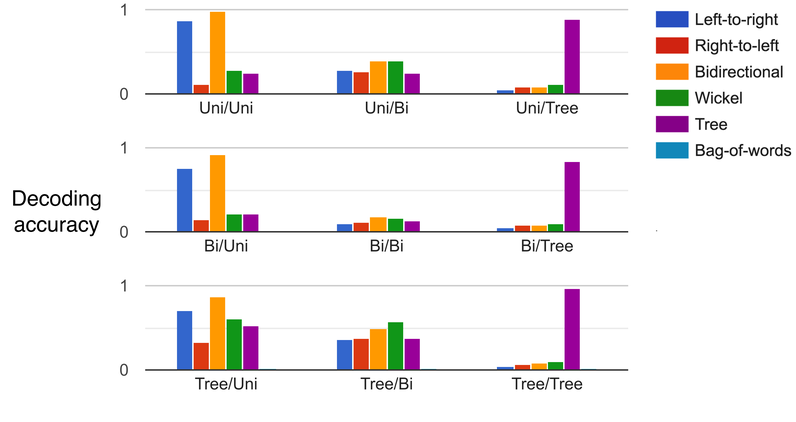
- **An RNN trained to copy can be approximated almost perfectly:**
  - Model being approximated: Unidirectional RNN trained to copy digit sequences
  - Best role scheme: bidirectional (orange bar)



- **Different tasks lead to different roles:**
  - Reversal favors right-to-left where copying favors left-to-right
  - Bag-of-words works for sorting, which requires no structure



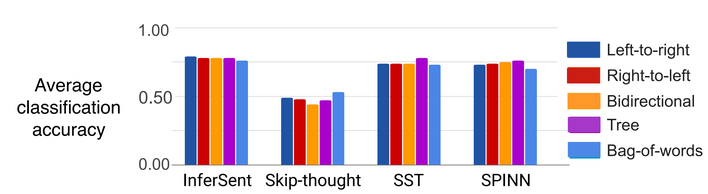
- **The decoder determines the learned role scheme:**
  - 3 architectures: unidirectional, bidirectional, tree-based.
  - We vary the encoder and decoder architecture. E.g., "Bi/Tree" has a bidirectional encoder and tree-based decoder.



6 Sentence Encoder Experiments

	Model Type	Training task
InferSent	BiLSTM	Natural Language Inference
Skip-thought	LSTM	Previous/next sentence prediction
SST	Tree	Sentiment prediction
SPINN	Tree	Natural Language Inference

- Compare outputs of classifiers applied to a sentence encoding model and its tensor product approximation
- All 4 models are reasonably well approximated with non-structure-sensitive bag-of-words roles, suggesting they do not have robust representations of structure:



7 Conclusion

- RNNs trained on directly structure-dependent tasks can be well-approximated by tensor-product representations, suggesting that some form of this representation is their solution for encoding compositional structure.
- 4 popular sentence encoding models did not display such clear evidence of compositional structure
- Overall, tensor product decomposition is a versatile technique for studying vector representations

Acknowledgments

- This material is based upon work supported by the NSF GRFP, an NSF INSPIRE grant, an ERC grant (BOOTPHON), and ANR grants IEC, PSL*, GEOMPHON, USPC, and EFL. All opinions are our own.

Link to paper

- https://openreview.net/pdf?id=BJx0sjC5FX