Recursive Neural Networks Can Learn Logical Semantics

Samuel R. Bowman, Christopher Potts, Christopher D. Manning

Compares tree RNNs and tree RNTNs with four experiments.

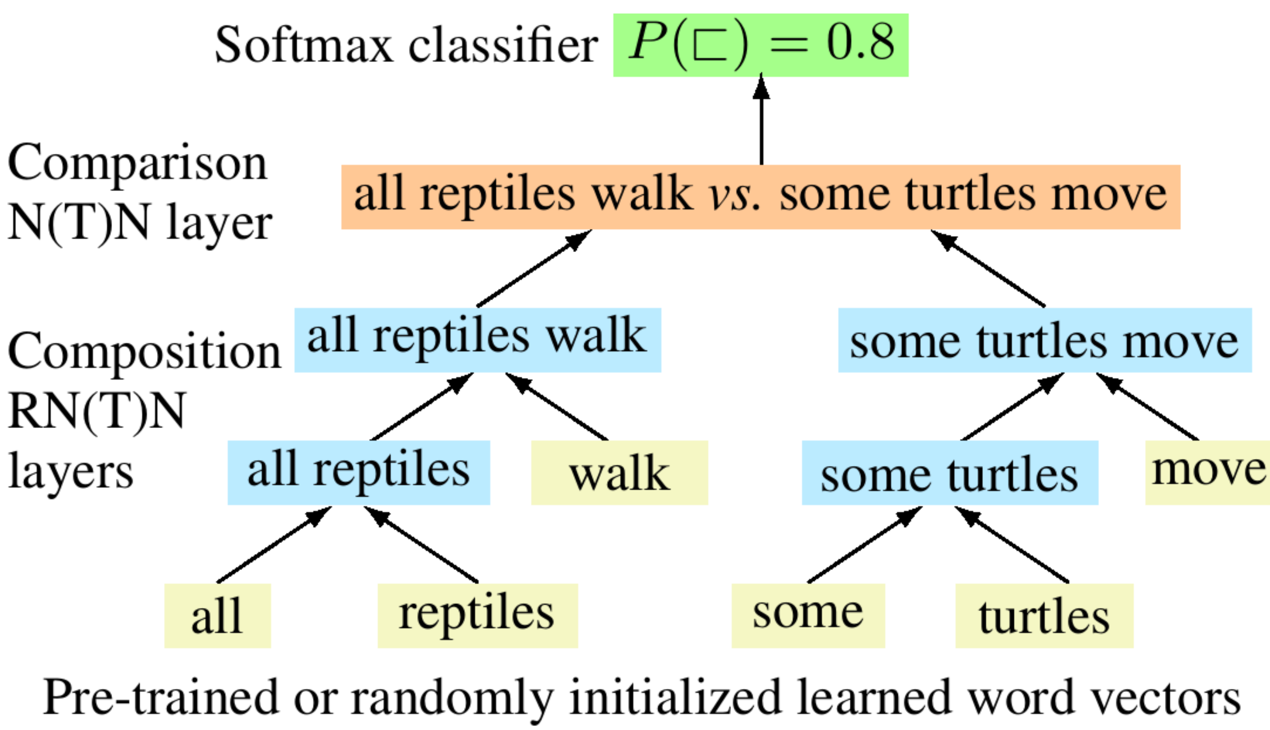
1. Reasoning about semantic relations
2. Reasoning with recursive logical sentences
3. Reasoning with quantifiers & negation
4. SICK textual entailment

TreeRNN and TreeRNTN:

Input: sentence pair
Output: classification
Assume we are given a parsed sentence (or we parse it ourselves)

There are two types of layers:

- compositional layer
- comparison layer



Compositional layer

- for RNN: $y_{\text{RNN}} = f(W_1 x_{\text{left}} + W_2 x_{\text{right}} + b)$ where $f = tanh$
- for RNTN:

- $h_i = x_{\text{left}}^T W_i x_{\text{right}}$
- $y_{\text{RNTN}} = y_{\text{RNN}} + f(h)$

Comparison layers are the same as compositional layers, but with separate weights and leaky relu.

Notation

Name	Symbol
(strict) entailment	$x \sqsubset y$
(strict) reverse entailment	$x \sqsupset y$
equivalence	$x \equiv y$
alternation	$x \mid y$
negation	$x \wedge y$
cover	$x \smile y$
independence	$x \# y$

Important distinction:

- $x \sqsubset y$ is different from $x \to y$.
- $x|y$ is different from $x \vee y$

The former is outside of the formal system whereas the latter is inside of the system.

- $x \to y$ is a proposition (it is a mathematical object)
- x entails y is a claim, made in english, about two mathematical objects

---

Semantic Relations

This experiment only tests the final comparison layer and not the composition layers, so it does not take full advantage of the tree-structure.

We have some boolean variables a, b, c and some propositions $p_1, p_2, \cdots, p_8$.

Training set:

- $p_1 \equiv p_2$

- $p_1 | p_7$

Test set:

- p_2, p_7 (answer: $|$)
- $p_5 \equiv p_6$ (answer: #)

Experimental procedure:

1. initialize an trainable embedding vector for each proposition
2. train just the composition layer

tested with 80 propositions, 7 boolean variables.

Results:

	Train	Test
# only	53.8 (10.5)	53.8 (10.5)
15d NN	99.8 (99.0)	94.0 (87.0)
15d NTN	**100 (100)**	**99.6 (95.5)**

Recursive Logical Sentences

Instead of having just a single proposition on each side of the comparison operator, we can have a complex expression, such as $p_4 \vee p_5$:

Training set:

- $\neg p_3 \wedge (p_4 \vee p_5)$
- $\neg p_1 \equiv \neg p_3$

Test set:

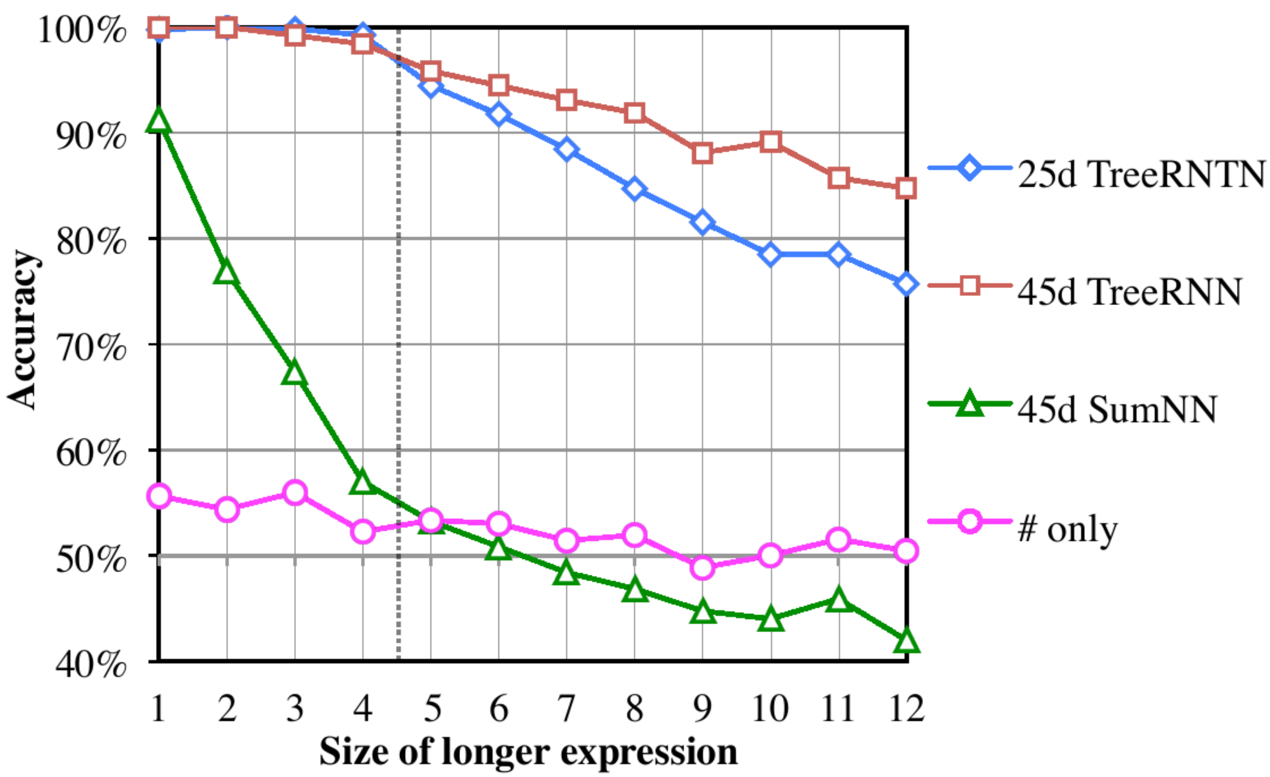
- $\neg p_1 \wedge (p_4 \vee p_5)$ (true)

Importantly, the symbols $\neg, \vee, \cdots$ also need to be embedded, since they are valid words in our language. Since we have more than one symbol on each side of the expression now, the compositional layers come into play.

Experimental procedure:

The formula on each side of the expression has up to 12 operators. There are 6 variables. Training was done with formulas with at most 4 operators, but tested on longer formulas.
Split the paired sentences into test and train.

Results:



SumNN: no weights in composition layer, just sum embedding vectors.

Quantifiers and Negation

Example of quantifiers:

- some, most, all, two, three

Negation of quantifiers:

- no, not-all, not-most, less-than-two, less-than-three

Other words:

- 5 nounts
- 4 verbs
- “not”

(3) (most turtle) swim | (no turtle) move

(4) (all lizard) reptile $\sqsubset$ (some lizard) animal

(5) (most turtle) reptile | (all turtle) (not animal)

Train on 3 and 4, test on 5.
Split the unpaired sentences before training.

Results: accuracy (F1)

	Train	Test
# only	35.4 (7.5)	35.4 (7.5)
25d SumNN	96.9 (97.7)	93.9 (95.0)
25d TreeRNN	99.6 (99.6)	99.2 (99.3)
25d TreeRNTN	**100** **(100)**	**99.7** **(99.5)**

SICK

Testing on real data.

The patient is being helped by the doctor	*entailment*	The doctor is helping the patient (PASSIVE)
A little girl is playing the violin on a beach	*contradiction*	There is no girl playing the violin on a beach (NEG)
The yellow dog is drinking water from a bottle	*contradiction*	The yellow dog is drinking water from a pot (SUBST)
A woman is breaking two eggs in a bowl	*neutral*	A man is mixing a few ingredients in a bowl (MULTIED)
Dough is being spread by a man	*neutral*	A woman is slicing meat with a knife (DIFF)

Minor changes: start with 200-D pretrained embeddings, and then pass through one layer to reduce dimension to 30/50.
Pretrained with DG data (a noisy dataset).

Results:

	neutral only	30d SumNN	30d TrRNN	50d TrRNTN
DG Train	50.0	68.0	67.0	**74.0**
SICK Train	56.7	96.6	95.4	**97.8**
SICK Test	56.7	73.4	74.9	**76.9**
PASSIVE (4%)	0	76	68	**88**
NEG (7%)	0	96	**100**	**100**
SUBST (24%)	28	**72**	64	**72**
MULTIED (39%)	**68**	61	66	64
DIFF (26%)	**96**	68	79	**96**
SHORT (47%)	50.0	73.9	73.5	**77.3**