Transcoding compositionally: using attention to find more generalizable solutions

Kris Korrel, Dieuwke Hupkes, Verna Dankers, Elia Bruni
University of Amsterdam

Paper

Code

Objectives / Contributions

1. With specifically designed sequence-to-sequence tasks, the human-like compositional understanding is tested for standard seq2seq deep learning models.
2. A new design of the seq2seq model is created that should improve the compositional skills on the selected tasks by putting more emphasize on the attention module.
3. (i) is that all and (ii) are these objectives or contributions?

Introduction

Seq2seq models have become ubiquitous in the field of machine translation, language modeling, speech recognition and other tasks which can be casted into temporally-dependent sequences with possibly varying input and output lengths. Although their generalization capabilities in these fields might indicate an understanding of the rules and hierarchies that underlie the tasks, specialized tasks/synonym that test specifically for the compositional understanding of these systems have shown evidence that this is not the case. We take these specialized tasks(same synonym) as a basis for understanding where seq2seq models lack human-like generalization skills, and (i) extend the testing methodologies for compositional understanding by examining overgeneralization capabilities, and (ii) propose a new architecture with a specialized focus on its attention module, which we dub seq2attn, with the aim to improve compositional understanding.

Model

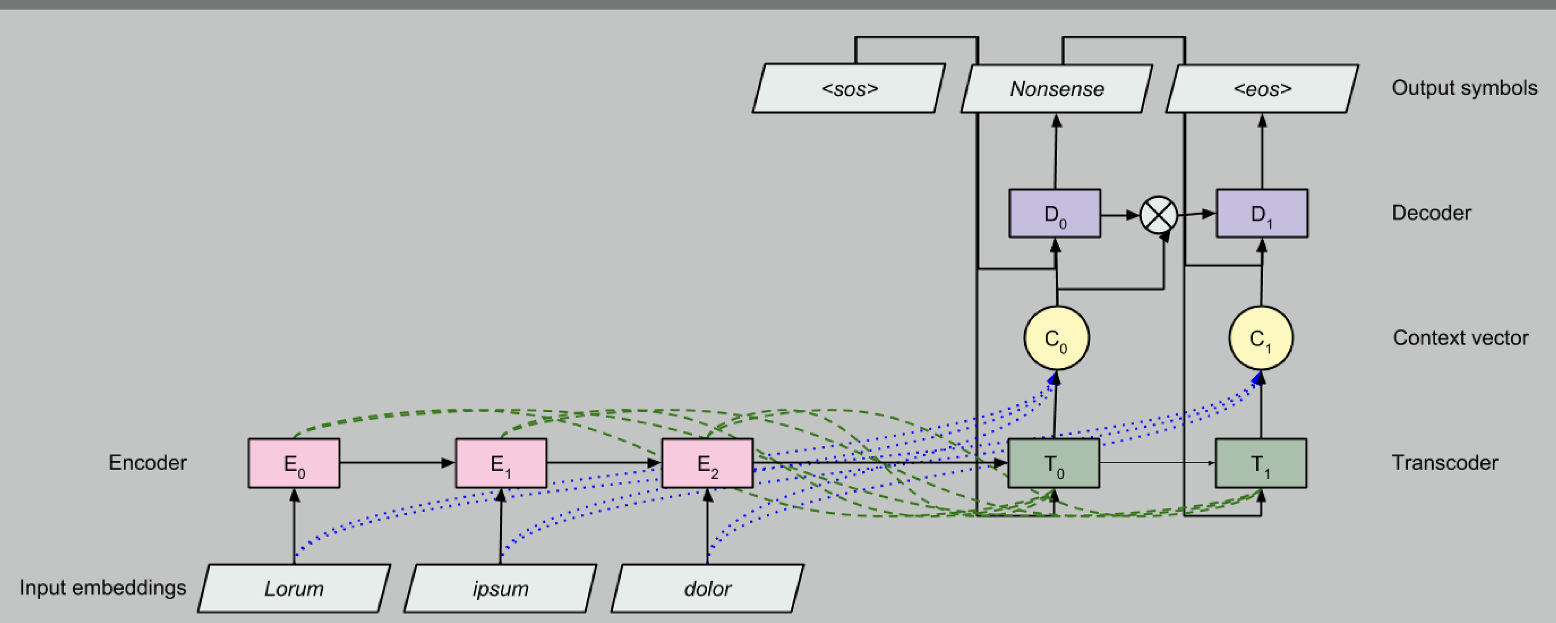


Figure: Schematic overview of the seq2attn architecture.

In summary, the proposed seq2attn model could be explained as a combination of four adaptations of the standard seq2seq architecture with attention.

1. The decoder receives information of the encoder solely through context vectors produced by a specialized attention module called the transcoder. This is to stimulate a focused processing of input tokens.
2. The context vectors produced by the decoder are weighted sums over the encoder's *embeddings* instead of its hidden states.
3. The weights with which the embeddings are combined into a context vector are the result not of a Softmax function, but of a Gumbel-Softmax function with the Straight-Through (ST) Estimator to enforce the use of a single embedding per decoder step.
4. The hidden states of the decoder are element-wise multiplied with the context vectors produced by the transcoder to further enforce the use of the attention module.

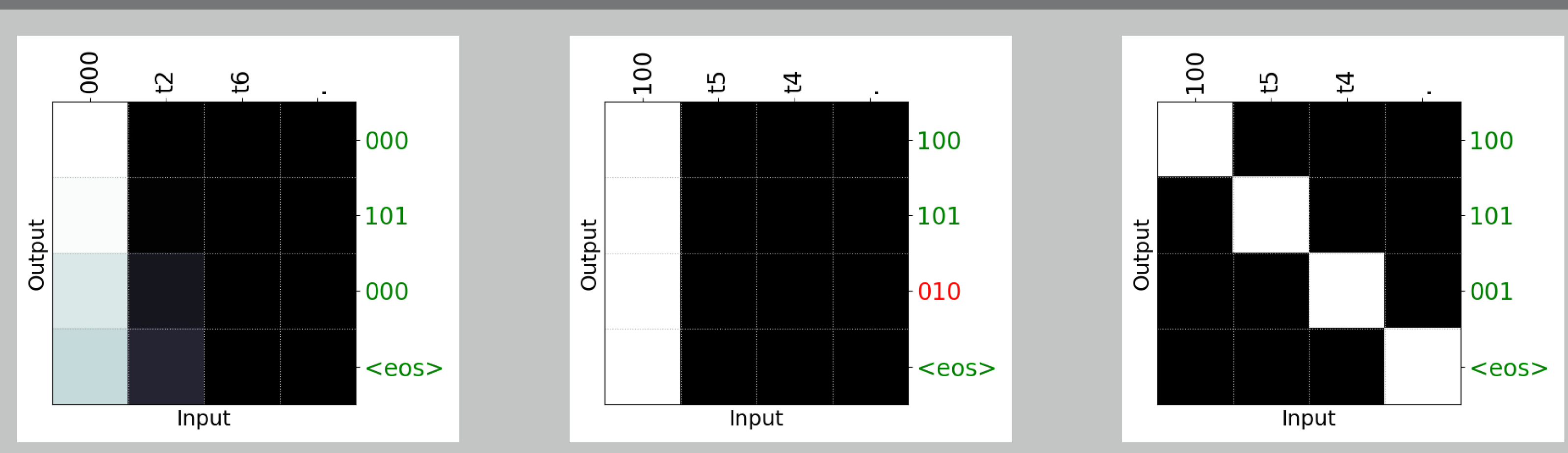
Results: Accuracies

	held-out inputs	held-out compositions	held-out tables
Baseline	38.25 ± 0.04	43.28 ± 0.09	7.86 ± 0.02
Seq2attn	**100 ± 0.00**	**100 ± 0.00**	**100 ± 0.00**

Table: Average sequence accuracies and standard deviations of the baseline seq2seq and seq2attn models on all lookup tables test sets.

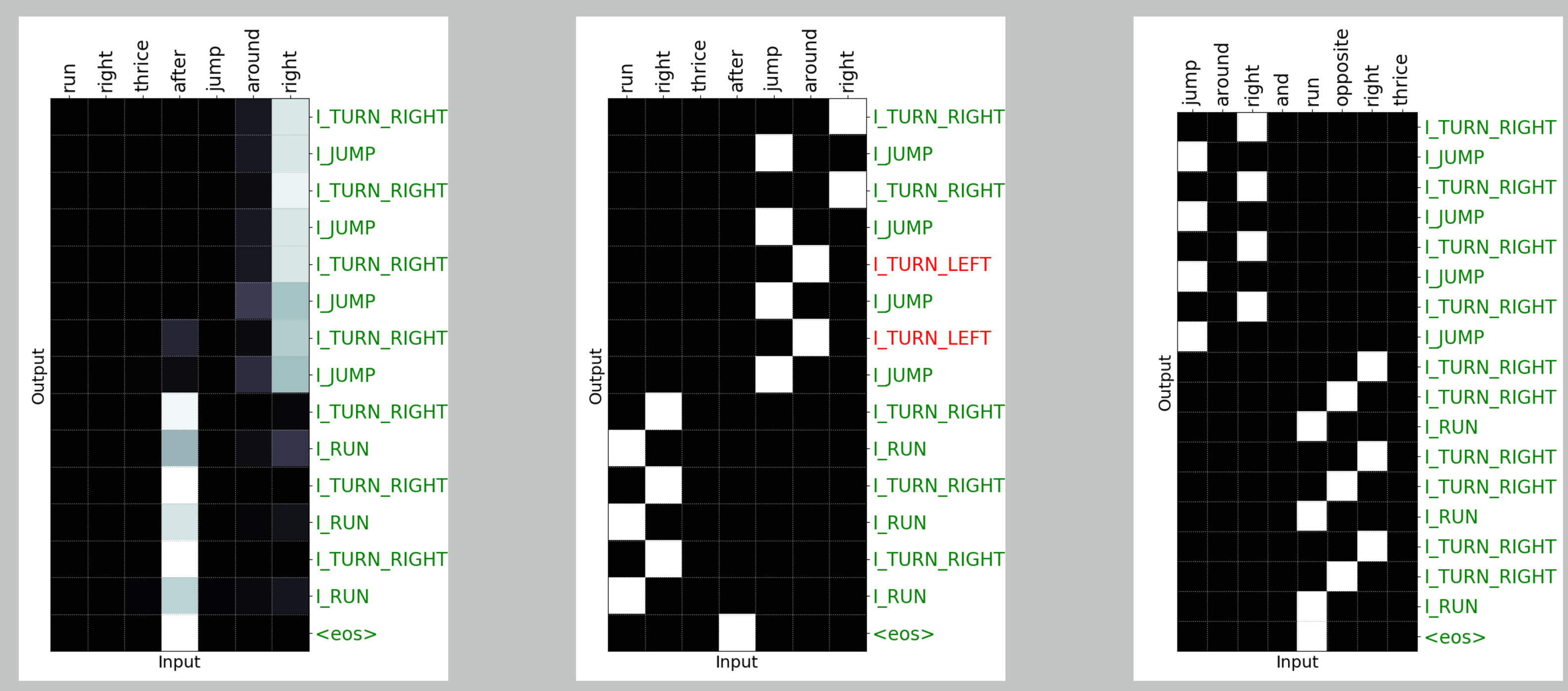
Figure: Mean sequence accuracies on experiment 3 of the SCAN task while testing on sequences containing `jump around right`.

Results: Attention patterns



(a) seq2seq (b) seq2seq (c) seq2attn

Figure: Examples of modeled attention patterns on held-out input examples of the lookup tables domain.

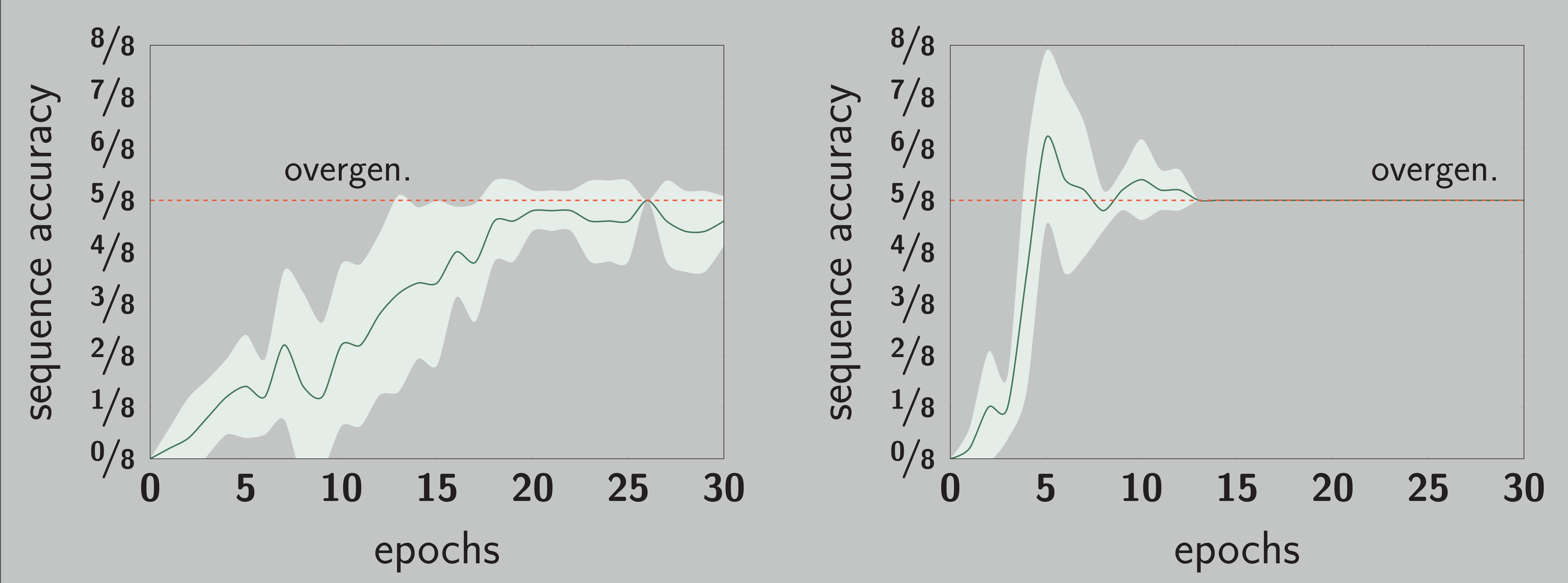


(a) seq2seq (b) seq2attn (c) seq2attn

Figure: Examples of attention patterns in the SCAN domain. Note how seq2attn models the output incorrectly when the attention patterns are incorrect.

Results: Overgeneralization

We suspect that compositional generalization leads to more *overgeneralization*. This is a phenomenon where a learner generalizes rules even to exceptions where these rules do not apply. For the lookup tables task, we test this by making the output of x `t1 t2` random for 3 of the 8 bit-strings.



(a) seq2seq (b) seq2attn

Figure: Average accuracies on the *original targets* for the eight inputs in composition x `t1 t2`. As three of these compositions are exceptions, we refer to accuracy higher than $5/8$ as overgeneralization.

Tasks

- Lookup Tables
  - The input sequences consist in a three-bit string followed by a series of function identifiers for lookup tables. The objective is to apply these bijective functions over the (intermediate) outputs successively
  - Example: `001 t1 t2` → `001 010 111`
- SCAN
  - The input consists in one or two sub-sequences which can be joined by the conjunctives `and` or `after`. Each sub-sequence contains a primitive command and possibly modifiers that act on this command. The learner must interpret the commands and mentally apply them in a 2D game grid.
  - Example: `jump after walk left twice` → `I_TURN_LEFT I_WALK I_TURN_LEFT I_WALK I_JUMP`

Conclusion

- increased general performance on specialized tasks indicate improved compositional generalization by seq2attn.
- The general ideas implemented could be extended to several variants of encoder-decoder architectures.
- Sparser attentional patterns improve interpretability of the found solutions,
- Preliminary results show neither increased or decreased performance on NMT (without the Sraight-Through estimator of Gumbel-Softmax)
- Strictly sparse attention vectors might limit the expressiveness of the model. Possible improvements may be made by research on alternatives methods.