Department of Computer Science, University of Otago

Technical Report OUCS-2014-03

A revised neural network model of episode representations in working memory

Authors:

Martin Takac and Alistair Knott
Department of Computer Science, University of Otago, New Zealand

---

Department of Computer Science,
University of Otago, PO Box 56, Dunedin, Otago, New Zealand

http://www.cs.otago.ac.nz/research/techreports.php

A revised neural network model of episode representations in working memory

Martin Takac[1,2] **and Alistair Knott**[1]

[1]Department of Computer Science, University of Otago, New Zealand

[2]Centre for Cognitive Science, Comenius University, Slovakia

August 19, 2014

Abstract

This report describes a revised version of the model of working memory episode representations described in Takac and Knott (2014). The new model is adapted in several ways to enable distributed representations instead of localist ones used in the original model.

1 Introduction

In Takac and Knott (2014) we described a model of the storage of episode representations in working memory (WM). Our key idea was that episodes are stored as prepared sequences of attentional and motor operations involved in experiencing an episode. The architecture of the model is shown in Figure 1. The model was trained on sequences of SM signals representing episodes of different types—intransitive, transitive, intransitive with prepositional phrase (PP) complement, simple causative, and causative with PP. The set of all possible episodes was generated by the transcription rules listed in Table 1. In our original model, individual SM signals were represented by localist coding (one-hot, i.e. one neuron on and all the others off); this paper describes the modification of the model exploring distributed representations in various components of the model. For details of the original model, please refer to Takac and Knott (2014); here we only describe the differences between the original model and the new one (Section 2) and the results of experiments with the distributed model (Section 3).

2 A modified network using distributed representations

In this section we describe modifications to the network enabling it to use distributed representations.

Distributed representations should improve the network's ability to generalise away from training examples. For instance, if the distributed representation of a cat includes representations of properties also shared by dogs, then the network should be able to use its experience of episodes involving cats to make (partial) predictions about episodes involving dogs, and vice versa. The ability to represent generalisations is particularly important in the candidate episodes buffer. Here, distinct episodes are represented as separate localist units. Clearly, for reasons of computational complexity,

```
Episode : Intransitive | IntrWithPPComplement | Transitive |
   SimpleCausative | CausativeWithPP

Intransitive : Agent IntrVerb .
IntrWithPPComplement : Agent IntrVerb2 PP .
Transitive : Agent Target TransVerb .
SimpleCausative : Agent Target CausativeVerb ResultVerb .
CausativeWithPP : Agent Target CausativeVerb ResultVerb2 PP .

Agent : AnimateObj
Target : AnimateObj | InanimateObj
PP: Preposition Landmark
Landmark : Target

AnimateObj : man | dog | cat
InanimateObj : cup | ball | chair
Preposition : under | behind | near
IntrVerb : die | walk | lie | sneeze | sit | sleep | smell |
   run | snore | breathe
IntrVerb2 : sneeze | sit | sleep | smell | run
TransVerb :  grab  | hit | push | shove | see | bite | hold |
   squeeze | kick | hug
CausativeVerb : caused
ResultVerb : break | stop | go
ResultVerb2 : go | hide
```

Table 1: Transcription rules for episodes of different types. The colon separates the head and tail of each rule, alternative tails are separated with |. Words starting with capital letters are non-terminal symbols. There are 35 terminal symbols, corresponding to individual sensorimotor signals/operations and an end-of-episode (.).

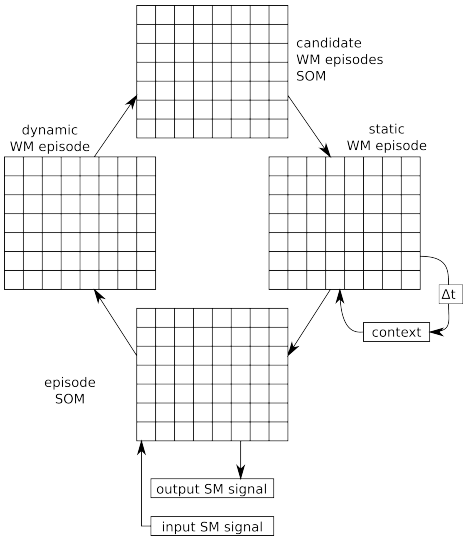

Figure 1: Architecture of the model (Takac and Knott, 2014).

there cannot be one unit in the buffer for each *possible* episode representation. In fact the candidate episodes buffer is not intended to function as a medium that can store all possible episodes in fine detail: in our model the medium with that capacity is the dynamic episodic buffer, which does indeed represent episodes very productively, as shown in the main paper. The role of the candidate episodes buffer is rather to guide the agent's experience of episodes, by offering a constantly updating distribution of the most likely episodes, or in action contexts, of the most desirable episodes. This distribution does not need to include representations of every possible episode—just of those which are in contention for being the most likely or desirable. And it does not need to represent *specific* episodes: in fact it would be useful if units in this medium could represent *types* of episode, each covering many possible specific episodes. In this section we describe a revised version of our network that operates on distributed input representations.

2.1 Distributed representation of concepts in the input layer

In the localist model with 1-hot coding in the input layer (i.e. one input neuron for each of the 35 possible sensorimotor elements), representations of all sensorimotor operations were equally dissimilar from each other. To allow for modelling of semantic errors and better generalization, we replaced 1-hot coding with distributed/featural representation of concepts.

We chose a simple distributed scheme in which each concept is represented by a set of generic semantic features (see Table 2). This scheme enables a graded notion of similarity between concepts, in which, for instance, the concept DOG is increasingly dissimilar to the concepts CAT, MAN, BALL, CHAIR and HIT. As well as the generic features used to implement this similarity metric, each concept had a unique feature distinguishing it from all other concepts. The network's input layer contains one unit for each feature (18 units for generic features, 35 for unique features). A concept is represented as a distributed pattern of activation over these units, with 1 coding

presence of a feature and 0 coding its absence.

The same coding is used both in the input SM signal and the aggregate SM signal layers. While the sparse distributed code in the input layer is always binary, the aggregate layer is fed top-down, i.e. with noisy information reconstructed from the weights of the neurons in the signal encoding SOM. In the localist model with 1-hot coding, this information could be directly interpreted as a probability distribution or degree of presence of individual SM signals; in the distributed model the noisy signal can represent any combination of features, which allows for generalizations, e.g. by representing generic features and no unique feature (or more than one unique feature). In order to interpret a noisy signal in terms of the original SM signals, we record cosine-similarities between the noisy signal and binary feature based sparse codes of each SM signal and then norm them dividing by their sum to obtain a probability distribution of interpreted SM signals.

2.2 Distributed representation of episode traces in the dynamic episodic buffer

An episode trace in the dynamic episodic buffer represents a sequence of SM signals in the signal-encoding SOM. In the localist model, the trace representing a sequence of n signals comprised exactly n units, corresponding to the winning units in the signal-encoding SOM at each step of the sequence. In the distributed model, we choose to represent each signal in an episode trace with a *group* of units, namely the k most active units in the signal-encoding SOM (henceforth called the k winners). We do this to ensure that the candidate episodes buffer learns to represent similar episodes in neighbouring units. While the dynamic episodic buffer has the same topographical organisation as the signal-encoding SOM, this organisation is not visible to the candidate episodes buffer. Like any SOM, this network learns to represent commonly occurring patterns in its input; whether these patterns involve adjacent units in the input vector or not is of no importance. Consequently, we must ensure that representations of similar signals in episode traces involve *overlapping groups* of units, rather than just adjacent units. For this reason, the k winners in the signal-encoding SOM are copied to the dynamic episodic buffer at each time step. As before, at each time step the activities of all units in the dynamic episodic buffer are proportionally downscaled at each time step before the new units are added, so the buffer continues to code the order of input signals by their level of activity. As we will discuss in the following section, the k should not be too large, so that the k units encoding each signal are reasonably close to each other, ideally forming a connected area with the most active unit in the centre. We experimented with different values of parameters and $k = 3$ yielded the best results.

Recall that the signal-encoding SOM uses a special constraint to ensure that repeated input signals are represented separately. In the localist model the constraint requires that units already used to represent a signal in the current episode trace are excluded from competition, so that a new unit will be selected to represent a repeated signal. In the distributed model, we replaced this fixed rule with a soft one: during the episode recording in the signal encoding SOM, the activity of each neuron in the competition phase is computed as the similarity of its weights with the input and context *minus* the activity of the corresponding neuron of the trace in the dynamic episodic buffer. Hence the trace in the dynamic episodic buffer serves as a top-down inhibitor for the competition in the signal encoding SOM.

SM signal	Features
.	EndOfSequence, F1
MAN	Object, Human, Animate, F2
DOG	Object, Animate, F3
CAT	Object, Animate, F4
CUP	Object, Inanimate, F5
CHAIR	Object, Inanimate, F6
BALL	Object, Inanimate, F7
UNDER	SpatialRel, F8
NEAR	SpatialRel, F9
BEHIND	SpatialRel, F10
CAUSE	Action, Causative, F11
GRAB	Action, Manual, F12
HIT	Action, Manual, F13
PUSH	Action, Manual, F14
SHOVE	Action, Manual, F15
SQUEEZE	Action, Manual, F16
WALK	Action, Self-Movement, F17
RUN	Action, Self-Movement, F18
GO	Action, Self-Movement, F19
LIE	Action, Self-position, F20
SIT	Action, Self-position, F21
SMELL	Action, Sensory, F22
SEE	Action, Sensory, F23
SNORE	Action, Physiological, F24
BREATHE	Action, Physiological, F25
SNEEZE	Action, Physiological, F26
SLEEP	Action, Physiological, F27
HOLD	Action, Arms, F28
HUG	Action, Arms, F29
BITE	Action, Mouth, F30
KICK	Action, Leg, F31
BREAK	Action, Result, F32
STOP	Action, Result, F33
HIDE	Action, Result, F34
DIE	Action, StateChange, F35

Table 2: Featural representation of sensorimotor signals. Features F1–F35 represent unique individual properties of each SM signal.

Fragment length	0-25%	25-50%	50-75%	75-100%	100%	total
Grammatical	75.2% (22.3)	78.0% (8.0)	85.7% (7.3)	93.0% (6.2)	94.3% (6.2)	85.0% (5.5)
Compatible	100.0% (0.0)	96.6% (3.4)	94.8% (4.4)	90.2% (5.8)	92.4% (6.7)	94.8% (3.5)
Matches	0.0% (0.0)	4.4% (1.7)	41.5% (6.7)	88.0% (6.3)	91.3% (6.9)	42.1% (4.1)
Rank	4.08 (3.48)	2.25 (0.44)	1.38 (0.14)	1.01 (0.02)	1.00 (0.00)	1.83 (0.80)
Rankable	75.2% (22.3)	76.1% (7.7)	83.8% (7.5)	88.1% (6.6)	91.3% (6.9)	82.6% (5.4)

Table 3: Prediction performance of the model using distributed representations, on initial fragments of different lengths of 100 episodes from the training set. Results are averaged over ten different simulation runs. Numbers in parentheses represent standard deviations.

2.3 Modified inhibition-of-return during episode replay

With episode traces now consisting of winners ordered by activities plus some areas around them, we need to also modify the mechanism of episode replay. When replaying the episode trace in the dynamic episodic buffer, the winner is determined in normal competition, and the winner activates its isomorphic neuron in the signal encoding buffer, as before. But in the inhibition-of-return phase (the removal of this element), not only the winner, but all the neurons within some neighbourhood radius r around it are inhibited. If the removal of the whole sparse distributed representation of the current element is successful, the next element can be again determined by normal competition and so on. The value of r should be large enough to cover all the k winners representing an element, but small enough not to cover neurons representing other elements of the sequence. We experimented with different combinations of k and r and $r = 9$ (expressed as squared Euclidean distance of neuron positions) worked the best.

3 Results

We trained the network with randomly generated sequences of sensorimotor operations, generated in the same way as those used to train the localist model (Takac and Knott, 2014), but now represented using the distributed scheme described above. In this section we summarise the performance of the trained network.

3.0.1 Immediate serial recall

After training we presented the modified network with the same three sets of sequences as the localist model. The model correctly recalled 94.6% (SD 3.8%) of seen sequences, 93.3% (SD 3.7%) of unseen sequences and 93.3% (SD 6.3%) of sequences containing repetitions. These results are worse by approximately 5% than those of the localist model. The drop is mainly due to the performance of the radius-based inhibition-of-return mechanism described above. Sometimes not all the units in a group were inhibited, leading to signals being repeated; sometimes units in other groups were inhibited, leading to signals being omitted.

3.0.2 Predicting complete episodes from fragments

We again exposed the trained network to 100 sequences randomly selected from the training set element by element, and examined its predictions about completions after each element. The results are shown in Table 3. Across the board, the distributed model's performance is slightly better than that of the localist model. While distributed

			CAT RUN BALL UNDER
CAT BALL CAUSE HIDE DOG UNDER CAT BALL CAUSE HIDE DOG BEHIND	CAT BALL CAUSE HIDE MAN BEHIND	CAT BALL SQUEEZE	CAT BALL HOLD
CAT BALL CAUSE HIDE CAT UNDER	CAT BALL CAUSE HIDE CHAIR BEHIND	CAT BALL CAUSE HIDE CHAIR NEAR	CAT BALL HUG
CAT BALL CAUSE GO CHAIR BEHIND	CAT BALL CAUSE HIDE BALL NEAR		
CAT BALL CAUSE GO DOG BEHIND CAT BALL CAUSE GO CAT BEHIND	CAT BALL CAUSE GO BALL NEAR		

Figure 2: Extract from a 'hit map' of the candidate episodes buffer, showing for each unit which input sequences it was the winner for.

representations make immediate recall of episodes slightly harder, they slightly improve[1] its ability to generate predictions about episode completions.

3.0.3 Representation of generalisations

The main reason for using distributed representations in the network was to allow it to generalise away from training episodes, and to allow explicit representation of generalisations in the candidate episodes buffer. To test the network's ability to generalise, we added a new object concept RABBIT to the network, which overlaps extensively with the animal concepts CAT and DOG, and also to a lesser extent with all animate concepts, and to a still lesser extent with all object concepts. We trained the signal-encoding SOM on a training set of 1000 input signals selected at random from the set of input signals excluding the signal RABBIT. Then we presented the trained SOM with the unseen signal RABBIT and used the winning unit's weights to reconstruct a pattern of activity in the input layer. The signals closest to the unseen input RABBIT were indeed CAT and DOG, followed by other animate concepts, and then other object concepts. This indicates that the signal-encoding SOM has some ability to generalise.

To examine the potential for generalisations in the candidate episodes buffer, we created a 'hit map' representing for each unit which input episodes it became the winner for. A fragment of this map is shown in Figure 2. As can be seen, the buffer has clear topographic organisation, with neighbouring units encoding episodes with similar structure and content. There are also some units that encode more than one episode, which again have similar structure and content. So both local regions and individual units have the potential to encode generalisations over episodes. However, our current method of reconstructing signals from units in the candidate episodes buffer does not allow these generalisations to be transmitted to lower layers. When a signal is reconstructed from a unit representing a mixture of specific episodes, either one of the episodes dominates the other and a single specific episode is produced as output, or a mixture of signals from both episode representations is reconstructed. In this latter case, the activity levels of units in the reconstructed representation no longer reliably indicate the temporal sequence in which signals should occur, and it is likely that an ungrammatical sequence of signals is produced. So while the candidate episodes buffer shows some potential as a medium for representing generalisations over episodes, it cannot yet translate these into coherent top-down expectations about sensorimotor sequences.

[1] The distributed model performed better by 3.7% in total grammaticality, 0.2% in compatibility, 0.8% in matches, 0.09 in rank, and 5.1% in rankability of predicted episodes.

4 Conclusion

In this report we described a revised version of the model of working memory for episodes using distributed inputs, and distributed encodings of signals and episodes in the SOM layers. The comparison between the localist and distributed versions of the model shows that the distributed model is worse in immediate serial recall and slightly better in prediction of episodes from their fragments. While there is still room for improvement, the results indicate that our proposed architecture can be adapted to operate with distributed representations.

References

Takac, M. and Knott, A. (2014). A neural network model of episode representations in working memory. Paper submitted to *Cognitive Computation*.