Ling 5701: Lecture Notes 5
A Model of Ambiguity in Sentence Processing

We have seen how complex ideas can be encoded and decoded into sentences.

This lecture will describe how this process controls for ambiguity.

Contents

5.1 Ambiguity as superposition

If multiple targets are associated with the same cue vector:

target 1 (lender)   cue (e.g. the word '*bank*')   target 2 (shore)   cue (e.g. the word '*bank*')
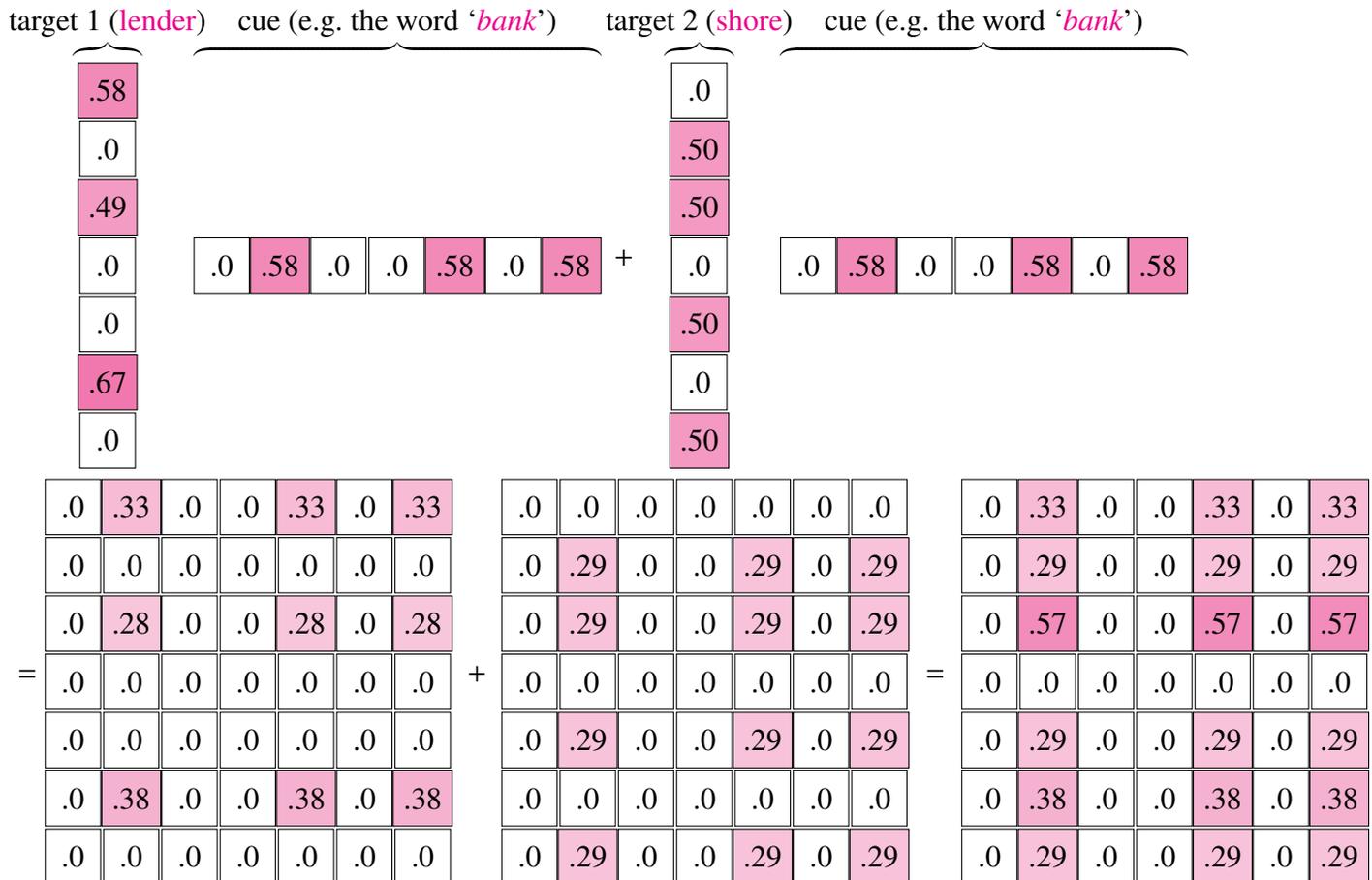
when associative memory is cued using that vector, the result is both vectors, **superposed**:

target 1 (lender)   target 2 (shore)   combined target   synaptic weights   cue ('*bank*')

.58		.0		.58		.0	.33	.0	.0	.33	.0	.33		.0
.0		.50		.50		.0	.29	.0	.0	.29	.0	.29		.58
.49		.50		.99		.0	.57	.0	.0	.57	.0	.57		.0
.0	+	.0	=	.0	=	.0	.0	.0	.0	.0	.0	.0		.0
.0		.50		.50		.0	.29	.0	.0	.29	.0	.29		.58
.67		.0		.67		.0	.38	.0	.0	.38	.0	.38		.0
.0		.50		.50		.0	.29	.0	.0	.29	.0	.29		.58
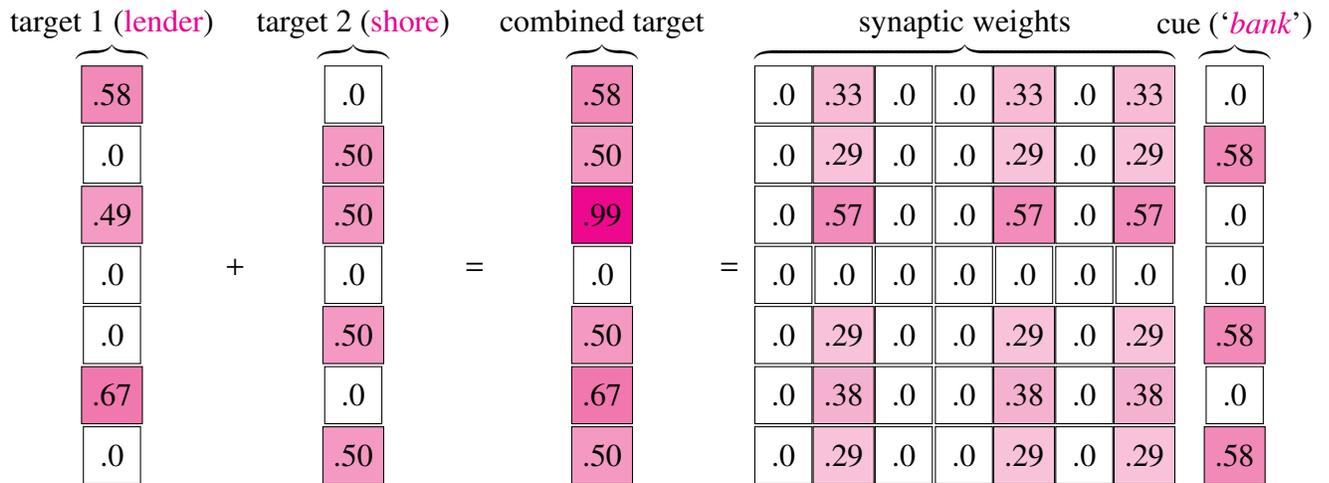
<u>PRACTICE:</u> If associative memory M is made from one cue u and two targets v_1 and v_2, as below:

target v_1 (insect)   cue u ('*bug*')   target v_2 (microphone)   cue u ('*bug*')

$M =$

.0		.0	.50	.50	.50	.0	.50	.0			.58		.0	.50	.50	.50	.0	.50	.0
.0											.58								
.80											.0								
.0									+		.0								
.0											.58								
.0											.0								
.60											.0								
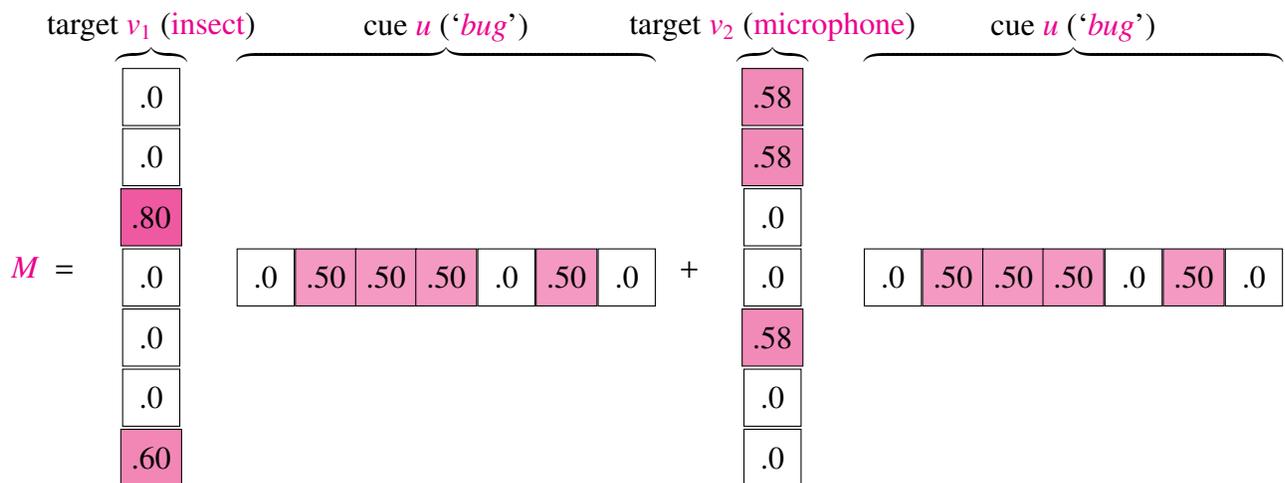
what is the result of cueing M with u?   (HINT: you don't have to calculate the matrix!)

result   synaptic weights M cue u ('*bug*')

|	=									.0
										.50
										.50
										.50
										.0
										.50
										.0
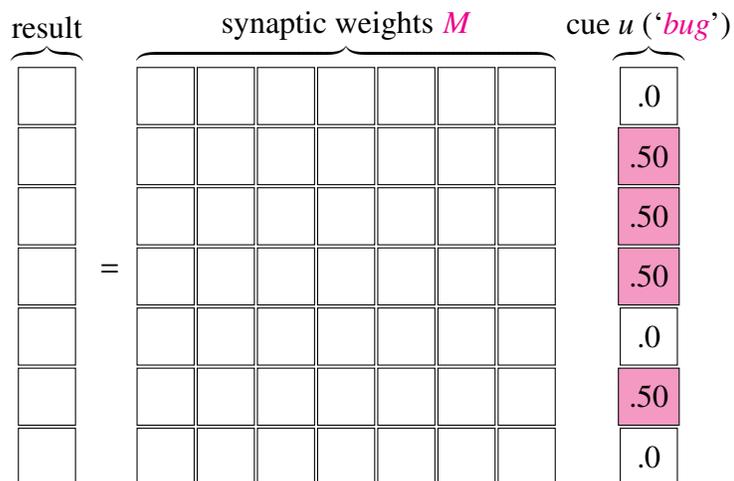
Describe the result in terms of v_1 and v_2.

5.2 Propagation of ambiguity

If we take a (non-interfering) set of associations:

target 1	cue 1								target 2	cue 2						
.58									.0								
.0									.50								
.58									.50								
.0	.0	.58	.0	.0	.58	.0	.58	+	.0	.0	.0	.58	.58	.0	.58	.0
.0									.50								
.58									.0								
.0									.50								

First matrix:

.0	.33	.0	.0	.33	.0	.33
.0	.0	.0	.0	.0	.0	.0
.0	.33	.0	.0	.33	.0	.33
.0	.0	.0	.0	.0	.0	.0
.0	.0	.0	.0	.0	.0	.0
.0	.33	.0	.0	.33	.0	.33
.0	.0	.0	.0	.0	.0	.0

$+$

.0	.0	.0	.0	.0	.0	.0
.0	.0	.29	.29	.0	.29	.0
.0	.0	.29	.29	.0	.29	.0
.0	.0	.0	.0	.0	.0	.0
.0	.0	.29	.29	.0	.29	.0
.0	.0	.0	.0	.0	.0	.0
.0	.0	.29	.29	.0	.29	.0

$=$

.0	.33	.0	.0	.33	.0	.33
.0	.0	.29	.29	.0	.29	.0
.0	.33	.29	.29	.33	.29	.33
.0	.0	.0	.0	.0	.0	.0
.0	.0	.29	.29	.0	.29	.0
.0	.33	.0	.0	.33	.0	.33
.0	.0	.29	.29	.0	.29	.0
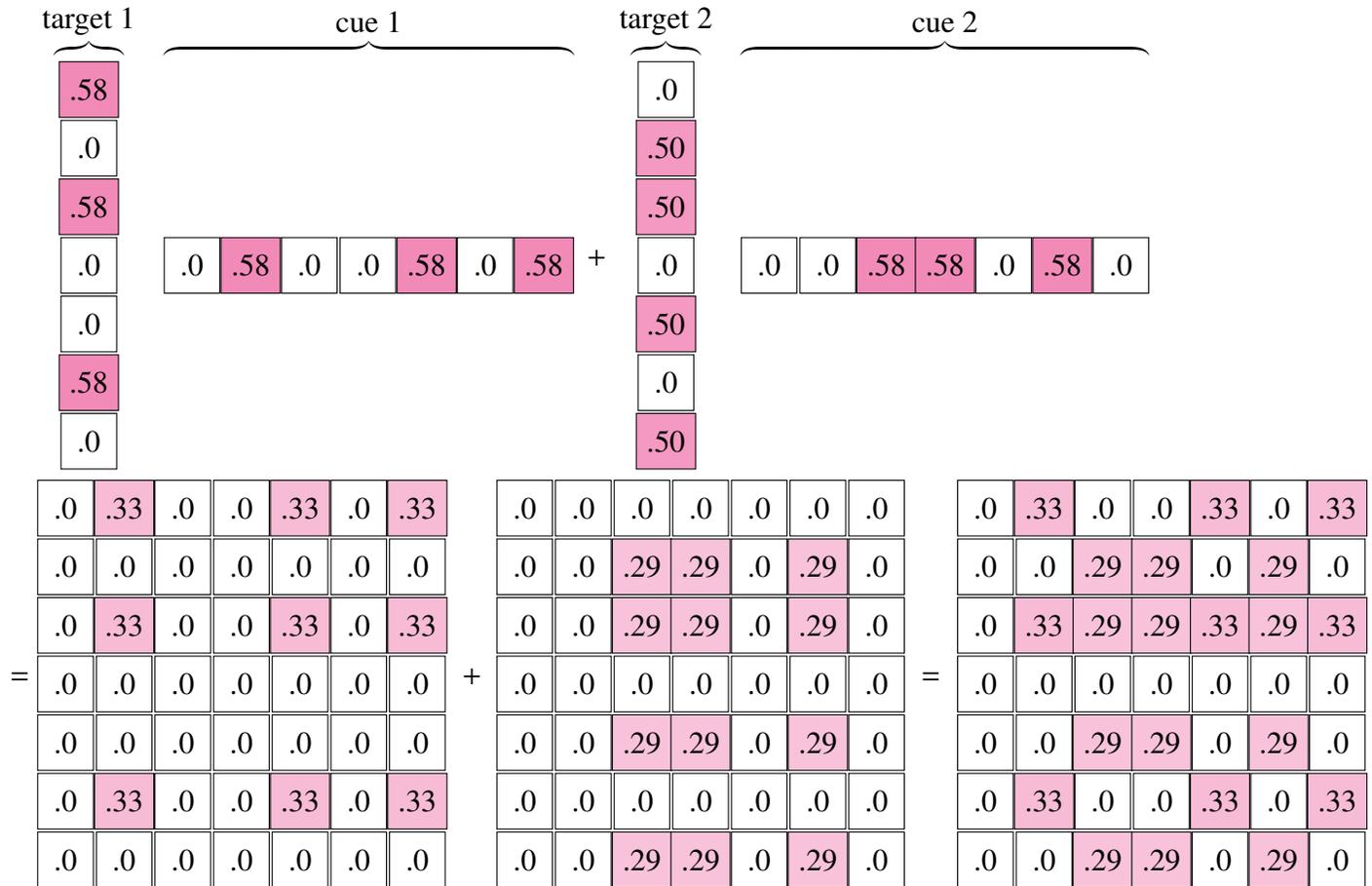
and cue them with a combination of states, we get a proportional combination of targets:

| target 1	target 2		combined target		synaptic weights							combined cue
.29	.0		.29		.0	.33	.0	.0	.33	.0	.33	.0
.0	.50	+	.50	=	.0	.0	.29	.29	.0	.29	.0	.29
.29	.50		.79		.0	.33	.29	.29	.33	.29	.33	.58
.0	.0		.0	=	.0	.0	.0	.0	.0	.0	.0	.58
.0	.50		.50		.0	.0	.29	.29	.0	.29	.0	.29
.29	.0		.29		.0	.33	.0	.0	.33	.0	.33	.58
.0	.50		.50		.0	.0	.29	.29	.0	.29	.0	.29
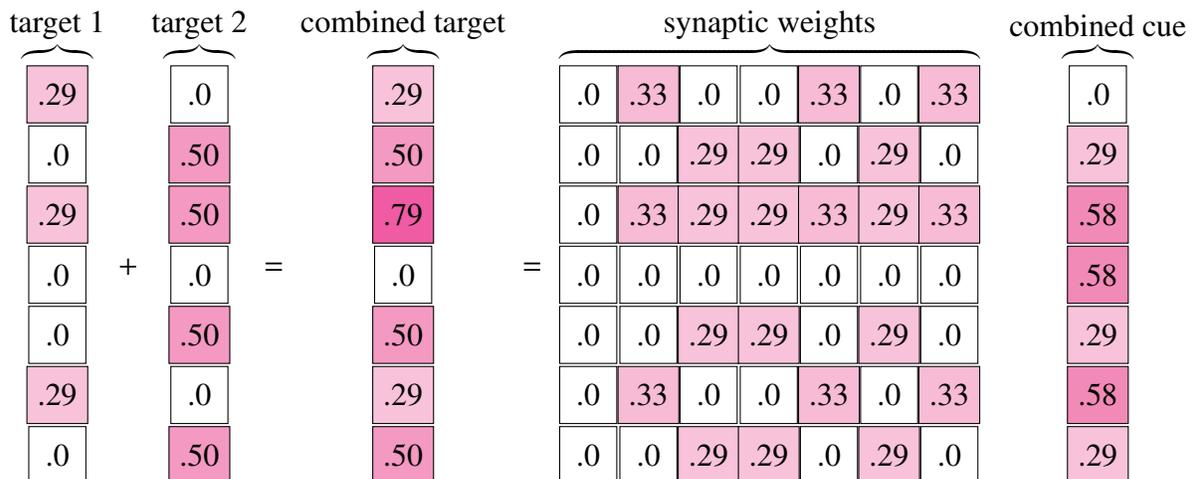
This is important because it allows ambiguity to propagate through a mental process.

PRACTICE: If associative memory M is made from cues u_1 and u_2 and targets v_1 and v_2, as below:

target v_1 (sting)   cue u_1 (insect)   target v_2 (record)   cue u_2 (microphone)

$$M = \begin{bmatrix} .0 \\ .50 \\ .50 \\ .0 \\ .0 \\ .50 \\ .50 \end{bmatrix} \begin{bmatrix} .0 & .0 & .0 & .80 & .0 & .0 & .60 \end{bmatrix} + \begin{bmatrix} .80 \\ .0 \\ .0 \\ .0 \\ .60 \\ .0 \\ .0 \end{bmatrix} \begin{bmatrix} .50 & .50 & .50 & .0 & .30 & .40 & .0 \end{bmatrix}$$
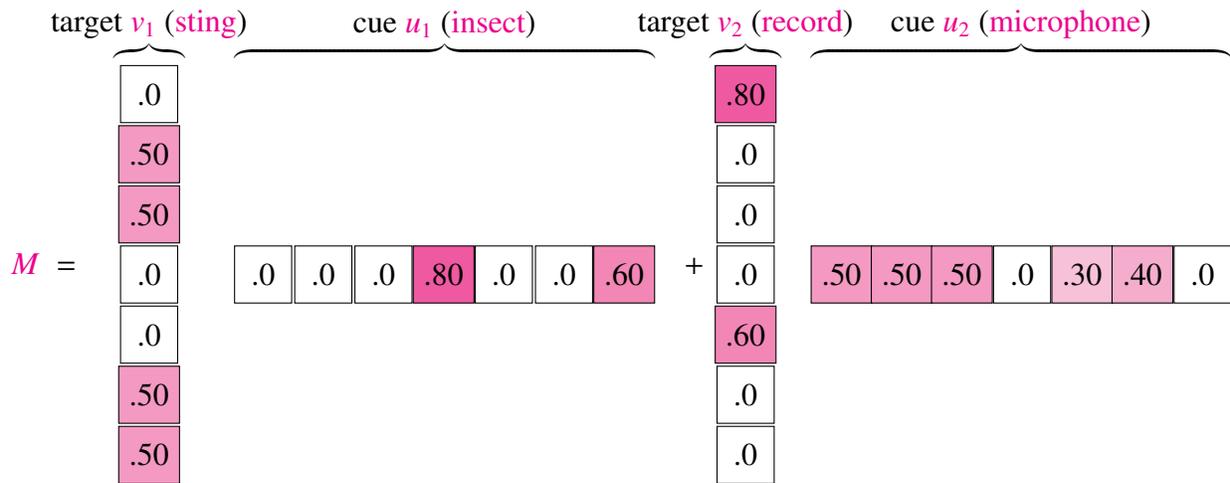
what is the result of cueing M with a mixture of $.7u_1$ and $.3u_2$? (HINT: don't calculate the matrix!)

result   synaptic weights M $.7u_1 + .3u_2$ combined cue

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \end{bmatrix} = \begin{bmatrix} & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \end{bmatrix} \begin{bmatrix} .15 \\ .15 \\ .15 \\ .56 \\ .09 \\ .12 \\ .42 \end{bmatrix}$$
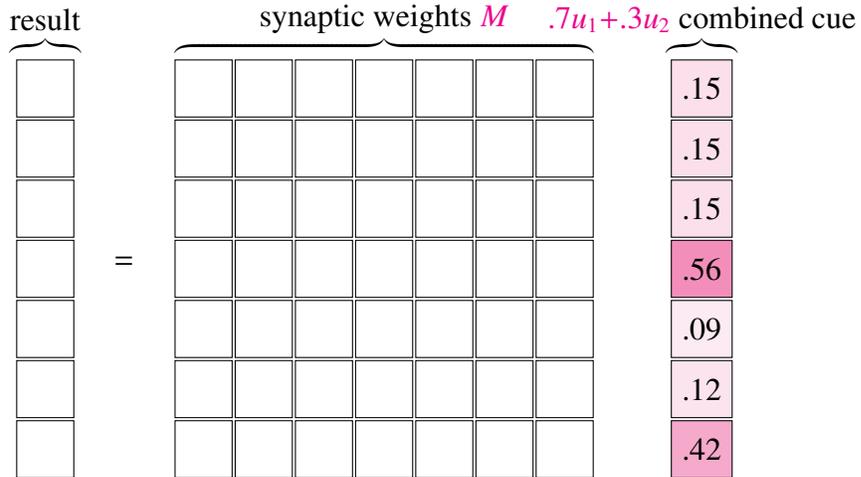
Describe the result in terms of v_1 and v_2.

5.3 Resolution of ambiguity

Recall the outer product of two vectors produces a matrix with pointwise products:

target v_1 cue u_1 associations M_1

$$\begin{bmatrix} .0 \\ .71 \\ .0 \\ .0 \\ .71 \end{bmatrix} \begin{bmatrix} .0 & .71 & .0 & .0 & .71 \end{bmatrix} = \begin{bmatrix} .0 & .0 & .0 & .0 & .0 \\ .0 & .50 & .0 & .0 & .50 \\ .0 & .0 & .0 & .0 & .0 \\ .0 & .0 & .0 & .0 & .0 \\ .0 & .50 & .0 & .0 & .50 \end{bmatrix} \qquad (1)$$

4
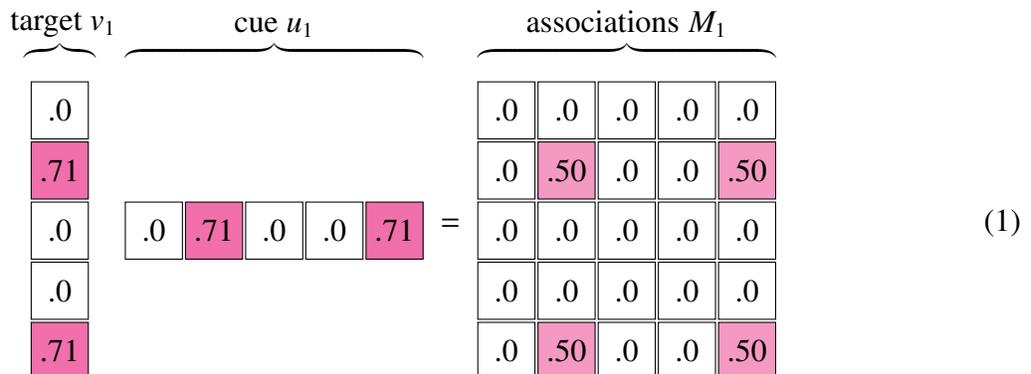
This generalizes to triples of vectors as a **tensor product**:
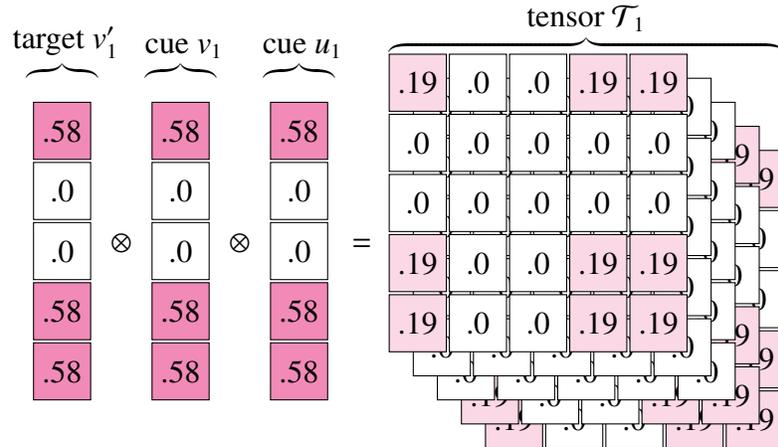
Targets are then cued by multiplication (left-associative):
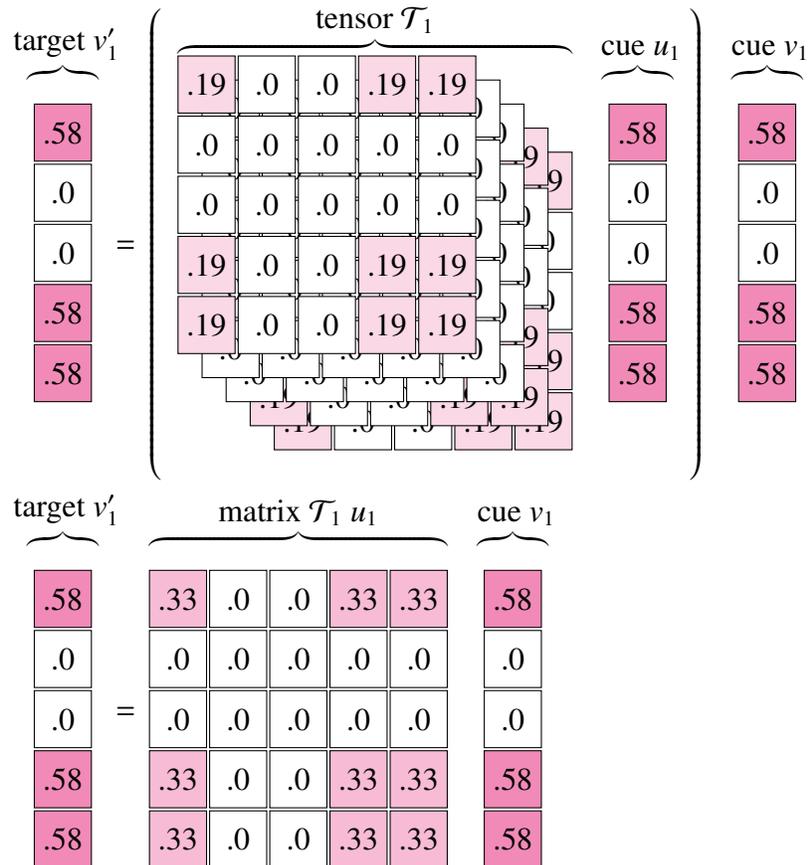
This can be implemented with 'switched' connections.

This gives us a means to combine 'top-down' predictions with 'bottom-up' observations...

Build auto-associations of all states:

target v_1' source v_1 observed u_1 target v_2' source v_2 observed u_2 tensor $\mathcal{T}$

$$
\begin{bmatrix} .58 \\ .0 \\ .58 \\ .58 \\ .0 \end{bmatrix} \otimes \begin{bmatrix} .58 \\ .0 \\ .58 \\ .58 \\ .0 \end{bmatrix} \otimes \begin{bmatrix} .58 \\ .0 \\ .58 \\ .58 \\ .0 \end{bmatrix} + \begin{bmatrix} .71 \\ .0 \\ .0 \\ .0 \\ .71 \end{bmatrix} \otimes \begin{bmatrix} .71 \\ .0 \\ .0 \\ .0 \\ .71 \end{bmatrix} \otimes \begin{bmatrix} .71 \\ .0 \\ .0 \\ .0 \\ .71 \end{bmatrix} =
$$

tensor $\mathcal{T}$:

.55	.0	.19	.19	.35
.0	.0	.0	.0	.0
.19	.0	.19	.19	.0
.19	.0	.19	.19	.0
.35	.0	.0	.0	.35
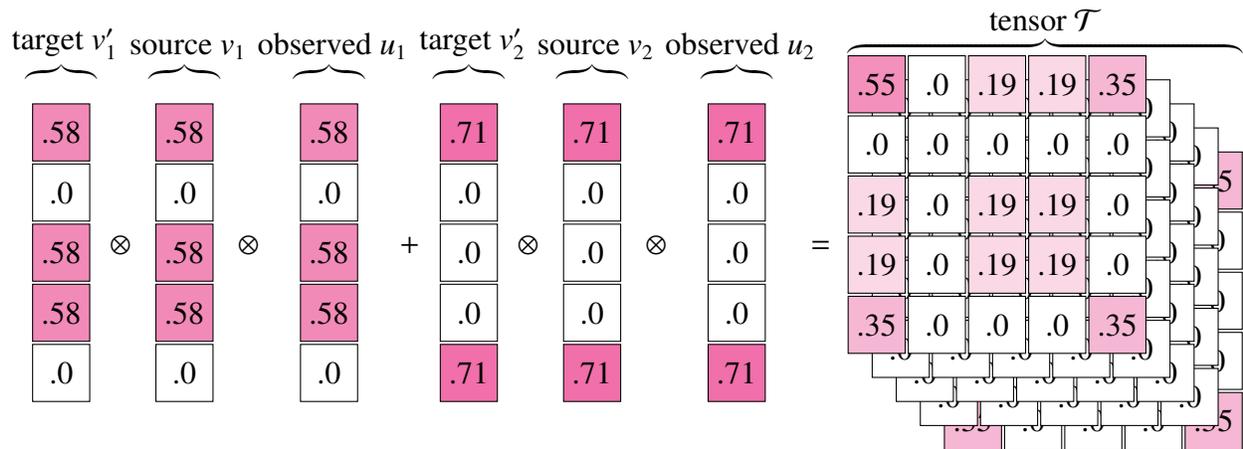
Then cue on observed state to pick out compatible component of mixed source state:

target v_2' tensor $\mathcal{T}$ observed u_2 source state $v_1 + v_2$

$$
\begin{bmatrix} .66 \\ .0 \\ .16 \\ .16 \\ .50 \end{bmatrix} = \left(\begin{bmatrix} .55 & .0 & .19 & .19 & .35 \\ .0 & .0 & .0 & .0 & .0 \\ .19 & .0 & .19 & .19 & .0 \\ .19 & .0 & .19 & .19 & .0 \\ .35 & .0 & .0 & .0 & .35 \end{bmatrix} \begin{bmatrix} .71 \\ .0 \\ .0 \\ .0 \\ .71 \end{bmatrix} \right) \begin{bmatrix} .64 \\ .0 \\ .29 \\ .29 \\ .36 \end{bmatrix}
$$
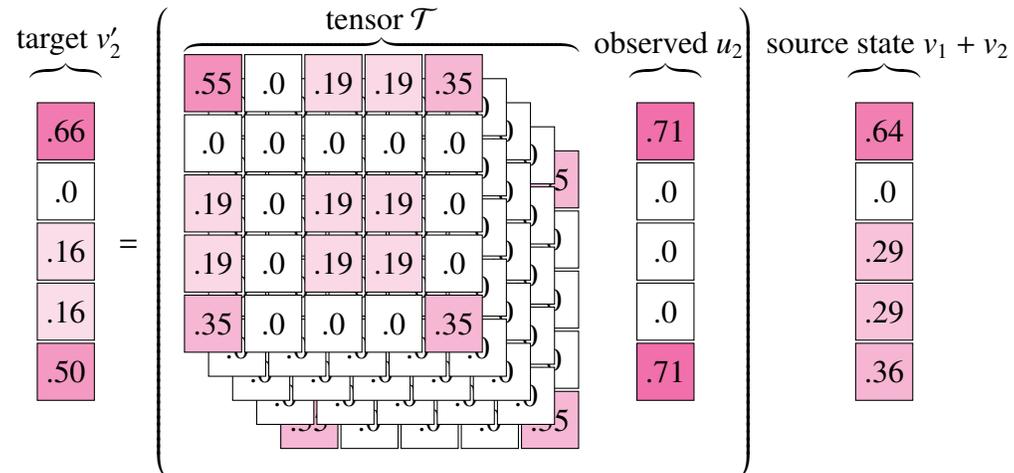
Note the magnitude of the target is reduced compared to the source.

This reduction correlates with reading time delays ('surprisal') on encountering unpredicted words.

(It may take time proportional to the reduction to 'amp up' this state to unit magnitude.)

If a filter F is made from auto-associated vectors v_1 and v_2 (say, from noisy speech):

v_1 (sting) v_1 (sting) v_2 (sing) v_2 (sing)

$$F = \begin{bmatrix} .0 \\ .58 \\ .58 \\ .0 \\ .0 \\ .0 \\ .58 \end{bmatrix} \begin{bmatrix} .0 & .58 & .58 & .0 & .0 & .0 & .58 \end{bmatrix} + \begin{bmatrix} .80 \\ .0 \\ .0 \\ .0 \\ .60 \\ .0 \\ .0 \end{bmatrix} \begin{bmatrix} .80 & .0 & .0 & .0 & .60 & .0 & .0 \end{bmatrix}$$
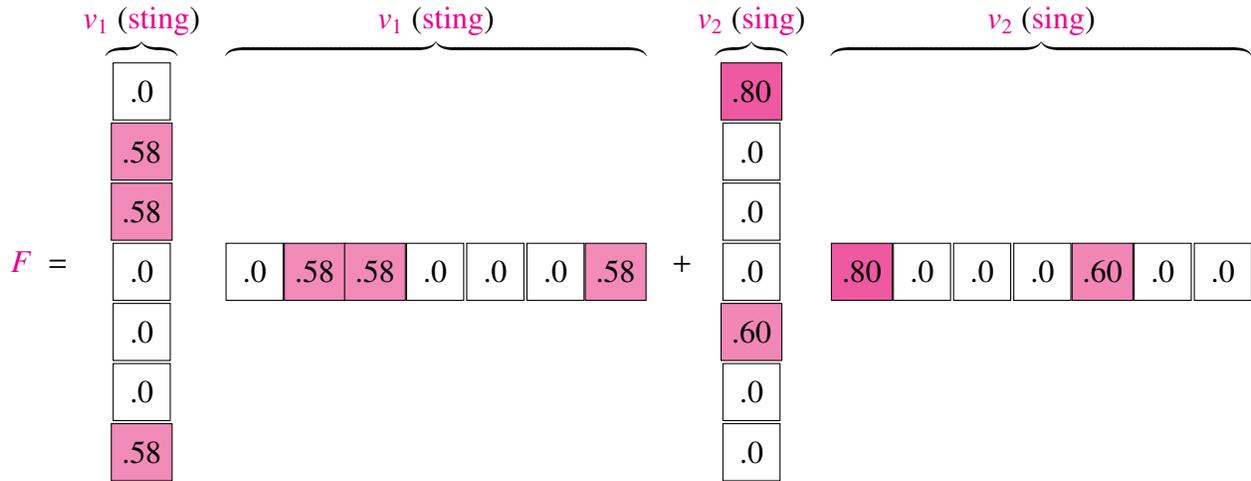
what is the result of cueing F with a mixture of $.2v_2$ and $.8v_3$?   (HINT: don't calculate the matrix!)

result   filter F $.2v_2+.8v_3$ combined cue

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \end{bmatrix} = \begin{bmatrix} & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \end{bmatrix} \begin{bmatrix} .16 \\ .0 \\ .0 \\ .48 \\ .12 \\ .64 \\ .0 \end{bmatrix}$$

References

7