Temporal Pooler Overview

Ryan McCall edited this page on Jun 17 · [5 revisions](https://github.com/numenta/nupic.research/wiki/Temporal-Pooler-Overview/_history)

**Pages 22**



* [**Home**](https://github.com/numenta/nupic.research/wiki)
* [**AI & ML Reference Material**](https://github.com/numenta/nupic.research/wiki/AI-&-ML-Reference-Material)
* [**AI & ML Talks**](https://github.com/numenta/nupic.research/wiki/AI-&-ML-Talks)
* [**Behavior and Pooling Requirements**](https://github.com/numenta/nupic.research/wiki/Behavior-and-Pooling-Requirements)
* [**Capacity experiment: setup and validation**](https://github.com/numenta/nupic.research/wiki/Capacity-experiment:-setup-and-validation)
* [**Capacity Test Results**](https://github.com/numenta/nupic.research/wiki/Capacity-Test-Results)
* [**Capacity Test Results (obsolete)**](https://github.com/numenta/nupic.research/wiki/Capacity-Test-Results-(obsolete))
* [**Current Research Tasks**](https://github.com/numenta/nupic.research/wiki/Current-Research-Tasks)
* [**Extraneous Predictions**](https://github.com/numenta/nupic.research/wiki/Extraneous-Predictions)
* [**Installation on EC2**](https://github.com/numenta/nupic.research/wiki/Installation-on-EC2)
* [**Meeting and Progress Notes**](https://github.com/numenta/nupic.research/wiki/Meeting-and-Progress-Notes)
* [**Meeting Notes from September 22**](https://github.com/numenta/nupic.research/wiki/Meeting-Notes-from-September-22)
* [**Modification to sensory motor temporal memory (Layer 4) algorithm**](https://github.com/numenta/nupic.research/wiki/Modification-to-sensory-motor-temporal-memory-(Layer-4)-algorithm)
* [**Open Questions**](https://github.com/numenta/nupic.research/wiki/Open-Questions)
* [**Overview of the Temporal Pooler**](https://github.com/numenta/nupic.research/wiki/Overview-of-the-Temporal-Pooler)
* [**Research Glossary**](https://github.com/numenta/nupic.research/wiki/Research-Glossary)
* [**Sensorimotor Inference Algorithm**](https://github.com/numenta/nupic.research/wiki/Sensorimotor-Inference-Algorithm)
* [**Sensorimotor Inference Algorithm Discussion**](https://github.com/numenta/nupic.research/wiki/Sensorimotor-Inference-Algorithm-Discussion)
* [**Temporal Pooler Overview**](https://github.com/numenta/nupic.research/wiki/Temporal-Pooler-Overview)
* [**Temporal Pooler Simultaneous Tandem Learning Experiments**](https://github.com/numenta/nupic.research/wiki/Temporal-Pooler-Simultaneous-Tandem-Learning-Experiments)
* [**Union Pooler Experiment Results**](https://github.com/numenta/nupic.research/wiki/Union-Pooler-Experiment-Results)
* [**Union Pooler Psuedocode**](https://github.com/numenta/nupic.research/wiki/Union-Pooler-Psuedocode)
* [Show 7 more pages…](https://github.com/numenta/nupic.research/wiki/Temporal-Pooler-Overview)

**Clone this wiki locally**



[**Clone in Desktop**](https://windows.github.com/)

[temporal\_pooler.py](https://github.com/numenta/nupic.research/blob/master/sensorimotor/sensorimotor/temporal_pooler.py) pseudocode

The temporal pooler attempts to form stable and unique representations of input sequences of active cells from Temporal Memory, but only if those cells were correctly predicted by Temporal Memory. A competition, similar to that of the spatial pooler's, is used to select active cells. In fact the first implementation inherits from the spatial pooler.

If a column is sufficiently activated by inputs that were predicted in the Temporal Memory, it enters a "pooling mode." A pooling column can, for a limited period, maintain active column status even without receiving any bottom-up input activity. This is implemented using a kind of "activation" for each column. When the activation drops to 0 it loses its pooling status. Whenever significant predicted input returns to a column's synapses, its activation is reset. The Temporal Pooler can be thought of as a particular kind of Spatial Pooler. One terminology note is that instead of referring to active columns we speak of active cells in the Temporal Pooler.

Pseudocode:

1) Calculate overlap:

Between correctly predicted input cells and pooling columns’ synapses

if learning

Use pooling columns' potential pool of synapses

else

Use pooling columns' connected synapses

(This type of overlap is heavily weighted compared to the others)

Between active input cells and all columns

Based on connected synapses (as in SP)

Between correctly predicted input cells and all columns

Based on connected synapses (as in SP)

# Unused, experimental idea

Add (large) parametric amount of overlap from bursting columns (false negatives) to all columns

2) Perform Spatial Pooler’s inhibition process returning a set of active columns

3) Perform Spatial-Pooler-like learning based on active columns

Stronger permanence change for synapses receiving correctly predicted (true positive) inputs.

(No change for false positive connections)

4) Update columns’ pooling activity and compute current pooling columns

if unpredicted inputs exceed threshold

Set all pooling activations to 0.0

else

for all pooling columns

Decay pooling activation

for all active columns receiving any true positive inputs

Set pooling activation to parameter value

return columns having non-zero pooling activation