Spatial Pooler

nashamri edited this page on Nov 12, 2014 · [4 revisions](https://github.com/numenta/nupic/wiki/Spatial-Pooler/_history)

**Pages 161**



* [**Home**](https://github.com/numenta/nupic/wiki)
* [**2013 Hackathon NLP Planning Meeting Notes**](https://github.com/numenta/nupic/wiki/2013-Hackathon-NLP-Planning-Meeting-Notes)
* [**2014 Fall Hackathon**](https://github.com/numenta/nupic/wiki/2014-Fall-Hackathon)
* [**2014 Goals For NuPIC**](https://github.com/numenta/nupic/wiki/2014-Goals-For-NuPIC)
* [**2014 January Core API Meeting Notes**](https://github.com/numenta/nupic/wiki/2014-January-Core-API-Meeting-Notes)
* [**2014 Spring Hackathon Planning Meeting**](https://github.com/numenta/nupic/wiki/2014-Spring-Hackathon-Planning-Meeting)
* [**Agile Planning**](https://github.com/numenta/nupic/wiki/Agile-Planning)
* [**Anomalies**](https://github.com/numenta/nupic/wiki/Anomalies)
* [**Anomaly Detection and Anomaly Scores**](https://github.com/numenta/nupic/wiki/Anomaly-Detection-and-Anomaly-Scores)
* [**AppVeyor CI**](https://github.com/numenta/nupic/wiki/AppVeyor-CI)
* [**Build and Installation FAQ**](https://github.com/numenta/nupic/wiki/Build-and-Installation-FAQ)
* [**C 11 14 References**](https://github.com/numenta/nupic/wiki/C--11-14-References)
* [**C Coding Guide**](https://github.com/numenta/nupic/wiki/C-Coding-Guide)
* [**CHANGELOG Guidelines**](https://github.com/numenta/nupic/wiki/CHANGELOG-Guidelines)
* [**CLA and HTM Theory**](https://github.com/numenta/nupic/wiki/CLA-and-HTM-Theory)
* [**CLA Classifier**](https://github.com/numenta/nupic/wiki/CLA-Classifier)
* [**CLA for ML AI Researchers**](https://github.com/numenta/nupic/wiki/CLA-for-ML-AI-Researchers)
* [**CLA Glossary for ML**](https://github.com/numenta/nupic/wiki/CLA-Glossary-for-ML)
* [**Classifier algorithms**](https://github.com/numenta/nupic/wiki/Classifier-algorithms)
* [**Classifiers**](https://github.com/numenta/nupic/wiki/Classifiers)
* [**CMake Style Guide**](https://github.com/numenta/nupic/wiki/CMake-Style-Guide)
* [**Community Content**](https://github.com/numenta/nupic/wiki/Community-Content)
* [**Community IRC Handles**](https://github.com/numenta/nupic/wiki/Community-IRC-Handles)
* [**Comparing NuPIC to other ML techniques**](https://github.com/numenta/nupic/wiki/Comparing-NuPIC-to-other-ML-techniques)
* [**Configuration and Build System**](https://github.com/numenta/nupic/wiki/Configuration-and-Build-System)
* [**Contributing to NuPIC**](https://github.com/numenta/nupic/wiki/Contributing-to-NuPIC)
* [**Contributing to the Wiki**](https://github.com/numenta/nupic/wiki/Contributing-to-the-Wiki)
* [**Contribution standards**](https://github.com/numenta/nupic/wiki/Contribution-standards)
* [**Contributor Model**](https://github.com/numenta/nupic/wiki/Contributor-Model)
* [**Cortical Learning Algorithm**](https://github.com/numenta/nupic/wiki/Cortical-Learning-Algorithm)
* [**CPP Unit Tests**](https://github.com/numenta/nupic/wiki/CPP-Unit-Tests)
* [**Creating Applications**](https://github.com/numenta/nupic/wiki/Creating-Applications)
* [**Data Sets for NuPIC**](https://github.com/numenta/nupic/wiki/Data-Sets-for-NuPIC)
* [**Developer workflow**](https://github.com/numenta/nupic/wiki/Developer-workflow)
* [**Development Process**](https://github.com/numenta/nupic/wiki/Development-Process)
* [**Development Tips**](https://github.com/numenta/nupic/wiki/Development-Tips)
* [**Documentation Priority Listing**](https://github.com/numenta/nupic/wiki/Documentation-Priority-Listing)
* [**Documenting NuPIC with Doxygen**](https://github.com/numenta/nupic/wiki/Documenting-NuPIC-with-Doxygen)
* [**Editing Wiki Offline**](https://github.com/numenta/nupic/wiki/Editing-Wiki-Offline)
* [**Encoders**](https://github.com/numenta/nupic/wiki/Encoders)
* [**Epilepsy Seizure Prediction**](https://github.com/numenta/nupic/wiki/Epilepsy-Seizure-Prediction)
* [**Experiments To Run**](https://github.com/numenta/nupic/wiki/Experiments-To-Run)
* [**External Libraries**](https://github.com/numenta/nupic/wiki/External-Libraries)
* [**FAQ**](https://github.com/numenta/nupic/wiki/FAQ)
* [**Future Plans**](https://github.com/numenta/nupic/wiki/Future-Plans)
* [**Google Summer of Code 2014 Ideas List for NuPIC**](https://github.com/numenta/nupic/wiki/Google-Summer-of-Code-2014-Ideas-List-for-NuPIC)
* [**Hackathon Planning Checklist**](https://github.com/numenta/nupic/wiki/Hackathon-Planning-Checklist)
* [**Hackathon Resources 2014 Spring**](https://github.com/numenta/nupic/wiki/Hackathon-Resources-2014-Spring)
* [**Hackathon Resources 2015 Spring**](https://github.com/numenta/nupic/wiki/Hackathon-Resources-2015-Spring)
* [**Hierarchical Temporal Memory**](https://github.com/numenta/nupic/wiki/Hierarchical-Temporal-Memory)
* [**Hierarchical Temporal Memory Theory**](https://github.com/numenta/nupic/wiki/Hierarchical-Temporal-Memory-Theory)
* [**HTM Blogs**](https://github.com/numenta/nupic/wiki/HTM-Blogs)
* [**HTM Challenge**](https://github.com/numenta/nupic/wiki/HTM-Challenge)
* [**HTM Challenge Community Meetup Planning**](https://github.com/numenta/nupic/wiki/HTM-Challenge-Community-Meetup-Planning)
* [**HTM Challenge FAQ**](https://github.com/numenta/nupic/wiki/HTM-Challenge-FAQ)
* [**HTM Challenge Resources**](https://github.com/numenta/nupic/wiki/HTM-Challenge-Resources)
* [**HTM History**](https://github.com/numenta/nupic/wiki/HTM-History)
* [**HTM Ports**](https://github.com/numenta/nupic/wiki/HTM-Ports)
* [**Inference Types**](https://github.com/numenta/nupic/wiki/Inference-Types)
* [**Initial Season of NuPIC 2014 Planning**](https://github.com/numenta/nupic/wiki/Initial-Season-of-NuPIC-2014-Planning)
* [**Install NuPIC using Docker**](https://github.com/numenta/nupic/wiki/Install-NuPIC-using-Docker)
* [**Installing and Building NuPIC**](https://github.com/numenta/nupic/wiki/Installing-and-Building-NuPIC)
* [**Installing NuPIC on Ubuntu**](https://github.com/numenta/nupic/wiki/Installing-NuPIC-on-Ubuntu)
* [**Installing NuPIC on Ubuntu 32bit**](https://github.com/numenta/nupic/wiki/Installing-NuPIC-on-Ubuntu-32bit)
* [**Installing With Wheels**](https://github.com/numenta/nupic/wiki/Installing-With-Wheels)
* [**Intermittently Failing Tests**](https://github.com/numenta/nupic/wiki/Intermittently-Failing-Tests)
* [**Introduction to the Algorithms**](https://github.com/numenta/nupic/wiki/Introduction-to-the-Algorithms)
* [**Issue Trackers**](https://github.com/numenta/nupic/wiki/Issue-Trackers)
* [**June 2013 Hackathon Participant Info**](https://github.com/numenta/nupic/wiki/June-2013-Hackathon-Participant-Info)
* [**Learning NuPIC**](https://github.com/numenta/nupic/wiki/Learning-NuPIC)
* [**Machine Learning Competitions**](https://github.com/numenta/nupic/wiki/Machine-Learning-Competitions)
* [**Mailing List Etiquette**](https://github.com/numenta/nupic/wiki/Mailing-List-Etiquette)
* [**Mailing List Status**](https://github.com/numenta/nupic/wiki/Mailing-List-Status)
* [**Mailman Spam Filtering**](https://github.com/numenta/nupic/wiki/Mailman-Spam-Filtering)
* [**Marked for Deletion**](https://github.com/numenta/nupic/wiki/Marked-for-Deletion)
* [**Milestone View**](https://github.com/numenta/nupic/wiki/Milestone-View)
* [**ML AI Working Page**](https://github.com/numenta/nupic/wiki/ML-AI-Working-Page)
* [**Models**](https://github.com/numenta/nupic/wiki/Models)
* [**MySQL Settings**](https://github.com/numenta/nupic/wiki/MySQL-Settings)
* [**Natural Language Processing**](https://github.com/numenta/nupic/wiki/Natural-Language-Processing)
* [**NDPR Jan 26, 2015**](https://github.com/numenta/nupic/wiki/NDPR-Jan-26,-2015)
* [**Network API**](https://github.com/numenta/nupic/wiki/Network-API)
* [**New Ideas About Temporal Pooling**](https://github.com/numenta/nupic/wiki/New-Ideas-About-Temporal-Pooling)
* [**NLP Projects**](https://github.com/numenta/nupic/wiki/NLP-Projects)
* [**NLP Tools**](https://github.com/numenta/nupic/wiki/NLP-Tools)
* [**Numenta Developer Setup**](https://github.com/numenta/nupic/wiki/Numenta-Developer-Setup)
* [**NuPIC 2014 Fall Hackathon Jeff Whiteboard Chat Questions**](https://github.com/numenta/nupic/wiki/NuPIC-2014-Fall-Hackathon-Jeff-Whiteboard-Chat-Questions)
* [**NuPIC API A bird's eye view**](https://github.com/numenta/nupic/wiki/NuPIC-API---A-bird's-eye-view)
* [**NuPIC Architecture**](https://github.com/numenta/nupic/wiki/NuPIC-Architecture)
* [**NuPIC Binary Installation on Ubuntu 14**](https://github.com/numenta/nupic/wiki/NuPIC-Binary-Installation-on-Ubuntu-14)
* [**NuPIC Community**](https://github.com/numenta/nupic/wiki/NuPIC-Community)
* [**NuPIC Community Roles**](https://github.com/numenta/nupic/wiki/NuPIC-Community-Roles)
* [**NuPIC Consumer Engagement Strategy**](https://github.com/numenta/nupic/wiki/NuPIC-Consumer-Engagement-Strategy)
* [**NuPIC Core Network API**](https://github.com/numenta/nupic/wiki/NuPIC-Core-Network-API)
* [**NuPIC Directory Structure Design**](https://github.com/numenta/nupic/wiki/NuPIC-Directory-Structure-Design)
* [**NuPIC Input Data File Format**](https://github.com/numenta/nupic/wiki/NuPIC-Input-Data-File-Format)
* [**NuPIC Office Hour Dec 2014**](https://github.com/numenta/nupic/wiki/NuPIC-Office-Hour-Dec-2014)
* [**NuPIC Office Hour Feb 2015**](https://github.com/numenta/nupic/wiki/NuPIC-Office-Hour-Feb-2015)
* [**NuPIC Office Hour Nov 2014**](https://github.com/numenta/nupic/wiki/NuPIC-Office-Hour-Nov-2014)
* [**NuPIC Office Hour Sep 2014**](https://github.com/numenta/nupic/wiki/NuPIC-Office-Hour-Sep-2014)
* [**NuPIC Repositories**](https://github.com/numenta/nupic/wiki/NuPIC-Repositories)
* [**NuPIC Usage FAQ**](https://github.com/numenta/nupic/wiki/NuPIC-Usage-FAQ)
* [**NuPIC's Dependency on nupic.core**](https://github.com/numenta/nupic/wiki/NuPIC's-Dependency-on-nupic.core)
* [**nupic.core Extraction Plan**](https://github.com/numenta/nupic/wiki/nupic.core-Extraction-Plan)
* [**Office Hours**](https://github.com/numenta/nupic/wiki/Office-Hours)
* [**Online Learning**](https://github.com/numenta/nupic/wiki/Online-Learning)
* [**Online Prediction Framework**](https://github.com/numenta/nupic/wiki/Online-Prediction-Framework)
* [**OPF Code Review (29 Aug 2011)**](https://github.com/numenta/nupic/wiki/OPF-Code-Review-(29-Aug-2011))
* [**Optimization & Profiling, Speed**](https://github.com/numenta/nupic/wiki/Optimization-&-Profiling,-Speed)
* [**Orphans**](https://github.com/numenta/nupic/wiki/Orphans)
* [**Other HTM CLA projects**](https://github.com/numenta/nupic/wiki/Other-HTM-CLA-projects)
* [**Pearls from the ML**](https://github.com/numenta/nupic/wiki/Pearls-from-the-ML)
* [**Pre Sprint Planning Tasks**](https://github.com/numenta/nupic/wiki/Pre-Sprint-Planning-Tasks)
* [**Projects**](https://github.com/numenta/nupic/wiki/Projects)
* [**Python Style Guide**](https://github.com/numenta/nupic/wiki/Python-Style-Guide)
* [**Reading List**](https://github.com/numenta/nupic/wiki/Reading-List)
* [**Reconstruction**](https://github.com/numenta/nupic/wiki/Reconstruction)
* [**Release Process**](https://github.com/numenta/nupic/wiki/Release-Process)
* [**River View and HTM**](https://github.com/numenta/nupic/wiki/River-View-and-HTM)
* [**Road Map**](https://github.com/numenta/nupic/wiki/Road-Map)
* [**Roadmap**](https://github.com/numenta/nupic/wiki/Roadmap)
* [**Roles of the Flag Bearer**](https://github.com/numenta/nupic/wiki/Roles-of-the-Flag-Bearer)
* [**Running Nupic in a Virtual Machine**](https://github.com/numenta/nupic/wiki/Running-Nupic-in-a-Virtual-Machine)
* [**Running NuPIC on Cloud Amazon EC2, Google GCE, GROK**](https://github.com/numenta/nupic/wiki/Running-NuPIC-on-Cloud---Amazon-EC2,-Google-GCE,-GROK)
* [**Running NuPIC on Crouton Ubuntu Chromebook**](https://github.com/numenta/nupic/wiki/Running-NuPIC-on-Crouton-Ubuntu-Chromebook)
* [**Running NuPIC on OS X**](https://github.com/numenta/nupic/wiki/Running-NuPIC-on-OS-X)
* [**Running NuPIC on Raspberry Pi**](https://github.com/numenta/nupic/wiki/Running-NuPIC-on-Raspberry-Pi)
* [**Running NuPIC on Windows**](https://github.com/numenta/nupic/wiki/Running-NuPIC-on-Windows)
* [**Running Swarms**](https://github.com/numenta/nupic/wiki/Running-Swarms)
* [**Season of NuPIC 2014 Idea List**](https://github.com/numenta/nupic/wiki/Season-of-NuPIC-2014-Idea-List)
* [**Season of NuPIC FAQ**](https://github.com/numenta/nupic/wiki/Season-of-NuPIC-FAQ)
* [**Sensorimotor Integration**](https://github.com/numenta/nupic/wiki/Sensorimotor-Integration)
* [**Serialization**](https://github.com/numenta/nupic/wiki/Serialization)
* [**SoN 2014 Final Report**](https://github.com/numenta/nupic/wiki/SoN-2014-Final-Report)
* [**SoN 2014 Mid Term Reports**](https://github.com/numenta/nupic/wiki/SoN-2014-Mid-Term-Reports)
* [**SoN 2014 Projects**](https://github.com/numenta/nupic/wiki/SoN-2014-Projects)
* [**Sparse Distributed Representations**](https://github.com/numenta/nupic/wiki/Sparse-Distributed-Representations)
* [**Sparse Distributed Representations Quiz**](https://github.com/numenta/nupic/wiki/Sparse-Distributed-Representations-Quiz)
* [**Spatial Classification**](https://github.com/numenta/nupic/wiki/Spatial-Classification)
* [**Spatial Pooler**](https://github.com/numenta/nupic/wiki/Spatial-Pooler)
* [**Spatial Pooling Quiz**](https://github.com/numenta/nupic/wiki/Spatial-Pooling-Quiz)
* [**Sprint 21 Plan**](https://github.com/numenta/nupic/wiki/Sprint-21-Plan)
* [**Sprint 26 Planning**](https://github.com/numenta/nupic/wiki/Sprint-26-Planning)
* [**Sprint 27 Planning**](https://github.com/numenta/nupic/wiki/Sprint-27-Planning)
* [**Sprint 28 Planning**](https://github.com/numenta/nupic/wiki/Sprint-28-Planning)
* [**Sprint 29 Planning**](https://github.com/numenta/nupic/wiki/Sprint-29-Planning)
* [**Sprint Planning Meetings**](https://github.com/numenta/nupic/wiki/Sprint-Planning-Meetings)
* [**Strategic Roadmap**](https://github.com/numenta/nupic/wiki/Strategic-Roadmap)
* [**Swarming Algorithm**](https://github.com/numenta/nupic/wiki/Swarming-Algorithm)
* [**Temporal classification**](https://github.com/numenta/nupic/wiki/Temporal-classification)
* [**Temporal Memory**](https://github.com/numenta/nupic/wiki/Temporal-Memory)
* [**Temporal Memory Quiz**](https://github.com/numenta/nupic/wiki/Temporal-Memory-Quiz)
* [**Temporal Pooler**](https://github.com/numenta/nupic/wiki/Temporal-Pooler)
* [**Temporal Pooling**](https://github.com/numenta/nupic/wiki/Temporal-Pooling)
* [**Travis CI**](https://github.com/numenta/nupic/wiki/Travis-CI)
* [**Using NuPIC**](https://github.com/numenta/nupic/wiki/Using-NuPIC)
* [**Vision Object Recognition Using NuPIC**](https://github.com/numenta/nupic/wiki/Vision-Object-Recognition-Using-NuPIC)
* [**Wiki Administration**](https://github.com/numenta/nupic/wiki/Wiki-Administration)
* [**Windows Support Group**](https://github.com/numenta/nupic/wiki/Windows-Support-Group)
* [**Zeta 1**](https://github.com/numenta/nupic/wiki/Zeta-1)
* [**Zeta 2**](https://github.com/numenta/nupic/wiki/Zeta-2)
* [Show 146 more pages…](https://github.com/numenta/nupic/wiki/Spatial-Pooler)
* [Home](https://github.com/numenta/nupic/wiki/Home)
  + [HTM Theory](https://github.com/numenta/nupic/wiki/Hierarchical-Temporal-Memory-Theory)
  + [Installing and Building](https://github.com/numenta/nupic/wiki/Installing-and-Building-NuPIC)
  + [Using NuPIC](https://github.com/numenta/nupic/wiki/Using-NuPIC)
  + [Learning NuPIC](https://github.com/numenta/nupic/wiki/Learning-NuPIC)
  + [Creating Applications](https://github.com/numenta/nupic/wiki/Creating-Applications)
  + [Contributing to NuPIC](https://github.com/numenta/nupic/wiki/Contributing-to-NuPIC)
* [HTM Challenge](https://github.com/numenta/nupic/wiki/HTM-Challenge)
* [NuPIC Community](https://github.com/numenta/nupic/wiki/NuPIC-Community)
* [NuPIC Projects](https://github.com/numenta/nupic/wiki/Projects)
* [Future Plans](https://github.com/numenta/nupic/wiki/Future-Plans)
* [Reading List](https://github.com/numenta/nupic/wiki/Reading-List)
* [NuPIC Code Repositories](https://github.com/numenta/nupic/wiki/NuPIC-Repositories)
* [Issue Trackers](https://github.com/numenta/nupic/wiki/Issue-Trackers)
* [FAQ](https://github.com/numenta/nupic/wiki/FAQ)

**Please read** [**Contributing to the Wiki**](https://github.com/numenta/nupic/wiki/Contributing-to-the-Wiki) **before editing any wiki pages.**

**Clone this wiki locally**



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

See the [CLA white paper](http://numenta.org/resources/HTM_CorticalLearningAlgorithms.pdf) for information on spatial pooling.

**Differences Between White Paper and Implementation**

**Global Inhibition**

In the white paper, active columns are determined based on their neighboring columns. Generally a column becomes active if it has a relatively high overlap score compared to his fellow neighbors. Computing inhibition in this manner is very computationally expensive, and represents a significant bottleneck in the implementation of the CLA algorithm. Global inhibition tries to tackle this problem by picking the columns with the most active columns from the region as a whole. This basically amounts to sorting the columns and selecting the top N most columns to become active. This is particularly useful when there is no topological information present in the data (i.e. the order in which columns are laid out is irrelevant).

**Permanence Trimming**

Theoretically, every column has a permanence value associated with every input bit in its potential pool. Practically though, many of the permanence values are extremely small. To reduce the memory footprint of the spatial pooler, permanence values below a certain threshold are zero'ed out. This is referred to as trimming.

**Raising Permanences To Overlap Threshold**

Columns must have a certain overlap score in order to be considered for activation. This is called the stimulus threshold. If the number of synapses in a connected state drop below this threshold, then it would be impossible for that column to be activated. While the activation duty cycle of such a column will eventually decrease enough such that it's permanence values will be boosted, we also take extra care to ensure that every column has at least the minimum number of connections required to exceed the stimulus threshold.

**Concepts**

**Orphan Columns**

Orphan columns are defined as columns which, for a given input pattern, all of their connected synapses were connected to input bits which are turned on, i.e. have a 100% overlap with the input. Yet despite this good match to the input, these columns did not become active after inhibition has been applied. Such columns are referred to as orphans since they have learned to represent a particular input pattern, but another column or columns have learned to represent the input patterns more accurately.

**Input Border**

For vision problems, the columns are laid out regularly in a 2D grid and each column as a square radius of inputs that it is connected to. Specifying an input border would result in the input bits on the exterior of the grid to not be included in the potential pools (which in the domain of vision is also referred to as receptive fields). For example, specifying an input border of '2' would result in the bits on the 2 leftmost columns, the 2 rightmost columns, the 2 top rows, and 2 bottom rows to be excluded from the potential pools of all columns

**Shared Inputs**

Shared inputs are inputs that are connected, via connected synapses, to more than one active columns. Remember that our goal is to have each column learn a unique spatial pattern. Shared inputs, by definition, do not lend themselves to such a goal. As such, the permanence values associated with shared inputs used to be incremented slightly less than non-shared inputs.

**Cloning**

In vision related problems, it is typical that lower level regions learn to represent nearly identical patterns, namely edges. This makes it possible to optimize the learning process by replicating a set of learned permanences from a small subset of columns throughout the entire CLA region. This is also known as 'weight sharing' in other machine learning domains. Cloning has been removed from the code base since its implementation has resulted in a lot of complexity.

**High Tier Columns**

High Tier columns were added to support datasets with very little input patterns. It is important to note that this scenario is only relevant for artificial data sets as real, sensory inputs do not lack input patterns (quite the contrary). In scarce input pattern scenarios, only a small subset of the columns would learn to represent the entire range of input patterns, leaving the remainder of the columns to 'starve', i.e. never become active. As these 'starved' columns will have their boost values raised (see boosting), they will eventually displace the columns that have learned to represent the input patterns, and will starve a new set of columns. This will lead to perpetual oscillation among the columns, with a different set of columns learning to represent the same input spatial patterns. High Tier is a method that was conceived to prevent this from happening. With High Tiering, we first look at the overlap percent (i.e. what percentage of connected synapses for a column are connected to input bits which are turned on). If this percentage is sufficiently high, we exclude these columns from participating in the inhibition round, securing their spot in the set of active columns.