Prototype of Stability Mechanism for Viewpoint Invariance
Written by David McDougall, 2018

ABSTRACT
The cortex is able to recognize distinct objects from a continuous stream of sensory input. Existing theory describes how sensory features are processed and associated with objects, but does not explain how the brain determines where one object ends and the next begins. This paper proposes an unsupervised method of learning to recognize objects and their boundaries. The method is implemented, analyzed and tested on artificially generated text, where it appears to work.

INTRODUCTION
This paper is an extension to the cortical model described in (Hawkins J, Ahmad S and Cui Y (2017) "A Theory of How Columns in the Neocortex Enable Learning the Structure of the World. "), which is a biologically constrained model of cortical layers 4 and 2/3. The model explains how the cortex integrates information over time in order to recognize objects. The cortical model contains two instances of Hierarchal Temporal Memories (HTMs) which are also a biologically constrained model. The HTM model explains how a group of neurons can function as a Markov Chain of indeterminate length. The first HTM in the cortical model transforms sensory input into good sensory features and passes the result to second HTM. This first HTM is called the input layer and corresponds to cortical layer 4. The second HTM groups sensory features into a single representation of an object. The second HTM is called the output layer and corresponds to cortical layers 2/3. As the sensory organ moves across an object, the output layer accumulates information about the objects appearance while maintaining a static representation of the object. The result is that the output-layer responds to objects in a viewpoint invariant manner.

The 2017 model however requires supervision and can not run on-line; it needs to know when it crosses the boundary between different objects. The output layers activity is held constant until the sensory organ crosses from one object to the next, at which point the output-layer is assigned a new set of activations for this new object. This paper proposes an unsupervised method of learning objects and their boundaries.

The method used to learn objects and their boundaries was incidentally discovered while studying the "grid cell" phenomena in the Entorhinal Cortex. Grid cells are cells which respond to physical locations, such as when you stand in a certain area of a room. The areas which grid cells respond to form a repeating hexagonal grid pattern. It's thought that the brain uses these grid patterns to navigate. Kropff and Treves theorize that grid cells form as you explore areas (Kropff and Treves, 2008). Their solution does three interesting things: it learns representations of large and contiguous areas, it shapes these areas into spheres, and it packs the spheres into the environment. In a two dimensional environment spheres naturally pack into a hexagonal grid. This and other observed properties of grid cells have been reproduced by the (Kropff and Treves, 2008) model of grid cells. The method by which grid cells learn to represent large contiguous areas of the input space is used here to find objects.

MODEL

This model is an extension of the cortical model described in (Hawkins J, Ahmad S and Cui Y (2017) "A Theory of How Columns in the Neocortex Enable Learning the Structure of the World. "). That model and the one presented here differ primarily in the output layer. The excitatory input to the output layer's spatial pooler is modified such that it exponentially approaches the true excitatory input (Kropff and Treves, 2008). It is an exponential rolling average, and I name it the stability mechanism. The equation for it is:

```
r(t) = r(t-1) + alpha * (InputOverlap - r(t-1))
```

Where `t` is time,
Where `InputOverlap` is the amount of excitatory input,
Where `r(t)` is the response, it's used in the competition to activate,
Where `alpha` controls how fast mini-columns respond to changes in their input overlap.

Another significant change to the HTM model is that synapses only learn when either side changes its activation (Kropff and Treves, 2008). This filters out duplicate updates on sequential time steps. The effect is the agent only learns when it is moving around, a stationary agent does not learn.

The output layer spatial pooler is given additional segments. Instead of each mini-column having a single proximal segment, each mini-column has several segments. The proximal segments in a mini-column compete to activate and the winning segment represents the whole mini-column when the mini-columns compete to activate. When a mini-column activates, only the winning segment learns. The proximal segments are dealt with before the stability mechanism is applied. Giving the output layer mini-columns more segments allows them to learn a greater number of different input patterns, which is important because the input layer may have no overlap between areas of the same object, leading to a very large number of distinct patterns which the output layer mini-columns must learn.

METHODS OF ANALYSIS

I tasked the model with reading and analyzing books. First the book is processed: all non-alphabetic characters are removed, all white space is removed, all letters are capitalized. Each character is prepared as sensory input as follows: the input character is used as a seed for a pseudo-random number generator which generates the indecies of active input neurons for this character. The model reads the processed book one character per time step. The book is divided into a training section consisting of the first 4/5ths of the book, and a testing section consisting of the remainder of the book. While the model is reading the testing section learning is disabled and the models properties are analyzed.

Here are three ways to inspect the model: by its stability, category overlap, and classification accuracy.

The stability of the output layer is the overlap in its mini-column activations between consecutive time steps. Since we are more interested in activations which persist for longer durations of time, the overlap is weighted such that each activations contribution to the overlap is directly proportional to the duration of the activation. A high stability means that many mini-columns are activating on sequential time steps; a low stability means that many mini-columns activations are moving around to different mini-columns. Plotting the stability is a crude way of finding objects and their boundaries as areas of high stability bounded by low stability areas.

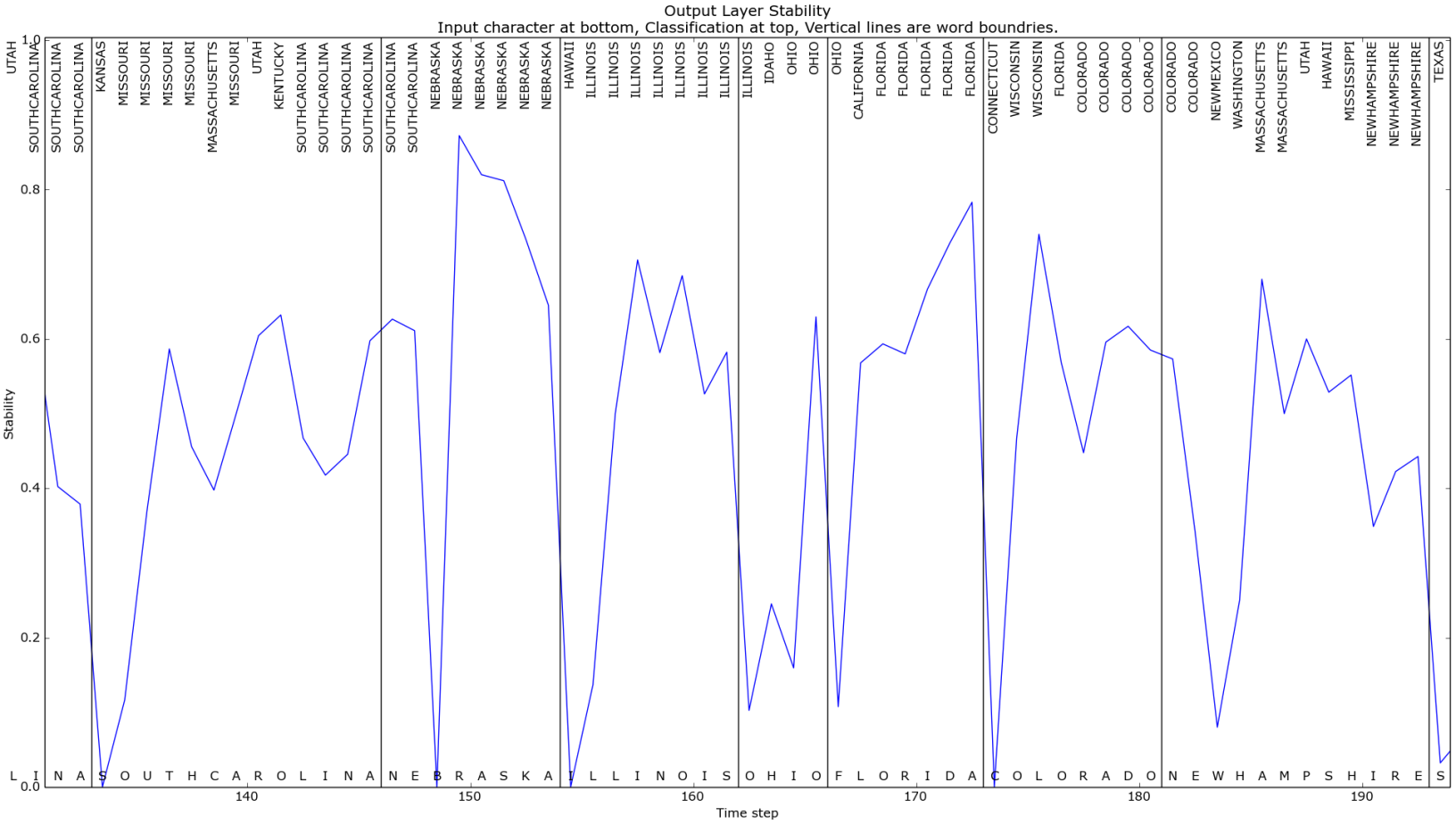

Figure 1: Example stability plot. The input is shown along the bottom by the X axis. The classification inferred from the output layer is shown along the top of the plot. The vertical lines indicate the true boundaries between words. The time steps shown are offset by the length of the training data. The book used in this example consisted of the names of the 50 states, repeated 50 times and shuffled. Notice that the stability decreases after crossing word boundaries, and that this decrease happens at the same time as the classification changes.

The category overlap measures the stability of each objects representations. There are two things measured: the intra-category overlap and the inter-category overlap. The category overlap is the average output-layer mini-column overlap between sensations from the same (intra) category of object versus different (inter) categories of objects. The average intra-category overlap is a measure of stability within a category; the average inter-category overlap is a measure of how distinct the categories are. These values not exhaustively calculated, instead 1 million overlaps are randomly sampled. In this scenario there is one category for each word.

The classification accuracy of both the input layer and output layer can verify that they both contain useful representations of the word. Two statistical classifier are trained, one for each layer. The classifiers are only used on the final time step in each category, in this scenario it's used on the final character in each word.

RESULTS

I implemented the model and posted the source code on-line at "https://github.com/ctrl-z-9000-times/sdr_algorithms/". Table 1 contains a summary of the models parameters. Most of these values were determined by a parameter meta-search, using the particle swarm optimization algorithm. The value function used for the optimization was the product of the intra-category overlap and the two classification accuracies.

Parameter	Value
Input layer mini-columns	1032
Input layer mini-columns sparsity	9.2 %
Input layer cells per mini-column	20
Input layer distal predictive threshold	9
Input layer synapses per distal segment	50
Input layer distal segments	22
Output layer mini-columns	3000
Output layer sparsity	0.733 %
Output layer proximal segments	4
Output layer potential percent	31.9961 %
Output layer stability rate (alpha)	0.27023961606581576

Table 1: Summary of parameters.

The dataset consists of 500 randomly selected words from the English dictionary. The words are repeated 25 times each and then shuffled into a random word order, with the constraint that the testing portion of the dataset contains exactly 5 examples of each word. Table 2 contains the results of reading this dataset, and figure 2 is a stability plot showing the performance of the trained model.

Metric	Measurement
Input Layer Average Anomaly	27 %
Intra Category Overlap	40.6641 %
Inter Category Overlap	6.24327 %
Input Layer Classification Accuracy	91.84 %
Output Layer Classification Accuracy	84.28 %

Table 2: Results of the dictionary dataset.

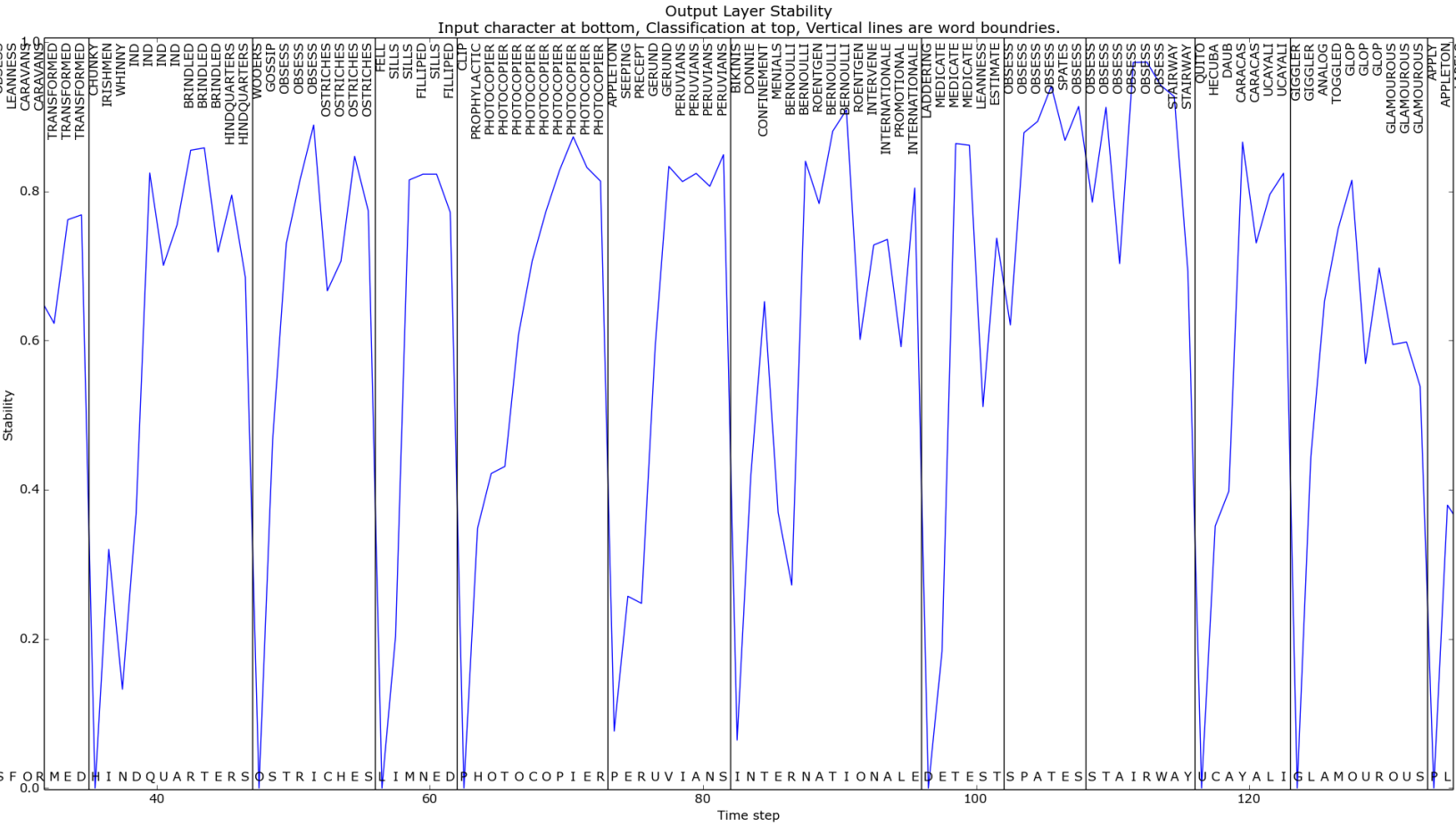

Figure 2: Stability plot for dictionary dataset.

REFERENCES

1) Hawkins J, Ahmad S and Cui Y (2017) "A Theory of How Columns in the Neocortex Enable Learning the Structure of the World. " Front. Neural Circuits 11:81. doi: 10.3389/fncir.2017.00081

2) "The emergence of grid cells: intelligent design or just adaptation?" Emilio Kropff and Alessandro Treves, 2008