Stochastic Searching Networks

J.M. Bishop
Reading University, UK

Proc. 1st IEE Conf. Artificial Neural Networks, 1989

Abstract

A fundamental difficulty when using neural networks applied to problems of pattern recognition is that of *stimulus equivalence*: the ability to classify a pattern, independent of position, rotation or scale, within the search space. This paper will describe how a network of stochastic cells can be used to find an efficient solution to such problems.

1 Hinton Mapping

The problem of stimulus equivalence is a specific example of a best fit constraint satisfaction search, where the model being searched for has been mapped into the search space, using either a translational, rotational or scale transformation. The goal of a system to solve such problems is thus to assertain the inverse mapping from the search space back to the model. For example, in a problem where the model has been rotated in the search space, the solution would be the angle of rotation; where the model has been translated, its XY coordinates etc.

One solution to the problem, suggested by Hinton [1], is to define a set of functional units corresponding to different mappings into the search space. These are connected to two sets of feature detectors, *canonical* and *retinocentric*, by mutually excitory connections. *Interactive activation* and *competition* [2] between cells ensures that the mapping which yields the best fit of the model into the search space will receive the most activation. Hinton shows that such a network will rapidly converge to the best mapping of the model in the search space.

A problem with Hinton Mapping is that it requires one mapping cell for each potential mapping into the search space. Searching for a model of size N in a search space of size M requires M mapping cells and this value increases polynomially with the number of degrees of freedom of the search. Stochastic Searching uses N cells independent of the size of the search space [3].

2 Stochastic Searching

Stochastic Searching is a search technique which uses a diffusion process to find the best fit of a Model within the Search Space. The Model and Search Space comprise of 'Feature Cells' between which comparisons can be made. For example, when searching for the best fit of a string of numbers within a larger string, the features may be the digits that comprise the strings.

Stochastic Searching uses a network of Dynamic Mapping Cells to probe possible mappings of the features into the Search Space. There is one Mapping Cell for each Feature Cell of the Model. A diffusion process ensures that Mapping Cells which yield succesful comparisons between features of the Model and features from the Search Space, spread their mappings across to other cells.

Stochastic Searching is inherently parallel in nature, where the algorithm is probabilistically bounded by $O(log(Tr))$, where Tr (the Time Ratio) is the ratio of the size of the model N to the size of the search space M. However, if the algorithm is implemented sequentially, then this order is linear - $O(Tr)$.

This behaviour has been established empirically, however work is ongoing to derive a formal description of algorithm behaviour.

2.1 Network Topology

A Stochastic Search Network consists of a two dimensional array of Dynamic Mapping Cells, where each Mapping Cell is linked to other Mapping Cells by Q bidirectional communication links. Each Mapping Cell represents a potential mapping of a Feature Cell into the search space. These mappings are notionally updated synchronously by a 'diffusion' process until the network has converged. Convergence is reached when the number of Mapping Cells with the same mapping exceeds an activation threshold (tH) and remains approximately constant for a settling time (tS).

2.2 Mapping Diffusion

At every time increment (iteration), each Mapping Cell tests its current mapping into the Search Space by comparing a randomly selected Model Feature Cell with a feature selected from the Search Space corresponding to the Model Feature Cell translated by a given mapping. A Mapping Cell remains active if this comparison is succesful.

Following the above 'Test Phase', active Mapping Cells diffuse their mappings to inactive cells. This is accomplished by each inactive Mapping Cell randomly choosing one of its Q links. If the selected link is connected to an active cell, then that cell's mapping is assigned to the inactive cell. If the selected link is an inactive cell, then a *new* mapping is chosen into the Search Space at random. An examply search is given in Appendix A.

3 Network Performance

If new mappings are chosen at random, then it is simple to calculate the number of iterations required before at least one of the Mapping Cells contains the correct mapping [3]. However, work is still ongoing to define the time required to diffuse a succesful mapping across to other cells. Hence the performance of Stochastic Search systems has to date been derived experimentally. It has been shown that for a given Time Ratio, the number of iterations required for convergence increases logarithmically with the size of the search space M [3].

3.1 Performance Degradation

The performance of Stochastic Search Networks is affected by two main factors:

The Time Ratio (Tr): This determines the number of iterations required before at least one of the Mapping Cells has located the best mapping into the search space. Convergence time increases exponentially as Tr approached zero.

The Probability of a Misleading Comparison: A misleading comparison occurs when an incorrect Mapping Cell remains active or a correct Mapping Cell goes inactive, after a comparison of feature cells. Convergence degrades exponentially, as this probability rises.

4 Conclusion

Stochastic Search Networks offer potentially high performance solutions to problems of stimulus equivalence. When locating a Model of size N in a Search Space of size M, the number of iterations required for convergence as been experimentally found to be probabilitically upper bounded by $O(log(Tr))$, where Tr, the Time Ratio, is the ratio of N/M.

Theoretical work is continuing with the aim of obtaining a full description of diffusion behaviour and proof of convergence. Current practical applications include the CONNEX test problem V3 [4] and the parsing, into labelled bracketed strings, of English sentences.

A An Example Search

A.1 Problem Definition

This is a simple example, where the Model has not been distorted within the Search Space, to illustrate the behaviour of a small Stochastic Search Network. The problem is to determine the best fit position of a 4 digit Model in a Search Space of 16 digits. The Search Space (SS) is defined by 16 Feature Cells (FCs) and the Model by 4 FCs. Hence the Model requires 4 Dynamic Mapping Cells DMCs to map it into the Search Space.

In this example, with $Q = 4$, as the Model is so small each DMC is directly linked to all others.

Model FC Number	0	1	2	3
Model Data	0	2	2	7
SS FC Number	1	2	3	4
SS Data	1	4	6	3
SS FC Number	5	6	7	8
SS Data	5	6	8	3
SS FC Number	9	10	11	12
SS Data	2	5	0	2
SS FC Number	13	14	15	16
SS Data	2	7	0	0

A.2 Initialisation

The initial mappings are set randomly.

DMC Number	0	1	2	3
Mapping	9	5	3	5
Connected to Model FC	1	3	0	2

A.3 Iteration 1 - Test Phase

The aim of the Test Phase is to check the mappings currently defined by the Dynamic Mapping Cells (DMC).

DMC Number	0	1	2	3
Mapping	9	5	3	5
Connected to Model FC	1	3	0	2
Result of Test	I	I	I	I

For the above, DMC[0] is INACTIVE (I), because the feature from the Search Space at location [10];

$$= [MAPPING + Model\ FC] = [9+1] = [10] \quad (1)$$

is not the same as the $[Model\ FC]$th element of the Model;

$$(Model\ FC[1] = 2) <> (SS\ FC[10] = 5) \quad (2)$$

DMC[1] is INACTIVE, because

$$(Model\ FC[3] = 7) <> (SS\ FC[5+3] = 3) \quad (3)$$

DMC[2] is INACTIVE, because

$$(Model\ FC[0] = 0) <> (SS\ FC[3+0] = 6) \quad (4)$$

DMC[3] is INACTIVE, because

$$(Model\ FC[2] = 2) <> (SS\ FC[5+2] = 8) \quad (5)$$

A.4 Iteration 1 - Diffusion Phase

The aim of the Diffusion Phase is to spread potentially correct mappings across all cells.

DMC Number	0	1	2	3
Try DMC Number	2	3	1	0
DMC ACTIVE ?	No	No	No	No
Assign New Mapping	2	11	9	14

For the above, DMC[0] checked the link to DMC[2], found it to be INACTIVE and so generated a *new mapping* into the Search Space ([2]).
DMC[1] checked DMC[3]; then selected a new mapping ([11]).
DMC[2] checked DMC[1]; then selected a new mapping ([9]).
DMC[3] checked DMC[0]; then selected a new mapping ([14]).

A.5 Iteration 2 - Test Phase

DMC Number	0	1	2	3
Mapping	2	11	9	14
Connected to Model FC	3	0	2	1
Result of Test	I	A	I	I

DMC[0] is INACTIVE, because

$$(Model\ FC[3] = 7) <> (SS\ FC[2+3] = 5) \quad (6)$$

DMC[1] is ACTIVE, because

$$(Model\ FC[0] = 0) = (SS\ FC[11+0] = 0) \quad (7)$$

DMC[2] is INACTIVE, because

$$(Model\ FC[2] = 2) <> (SS\ FC[9+2] = 0) \quad (8)$$

DMC[3] is INACTIVE, because

$$(Model\ FC[1] = 2) <> (SS\ FC[14+1] = 0) \quad (9)$$

A.6 Iteration 2 - Diffusion Phase

DMC Number	0	1	2	3
Try DMC Number	3	*	0	1
DMC ACTIVE ?	No	*	No	Yes
Assign New Mapping	13	*	3	[11]

DMC[0] checked DMC[3]; then selected a new mapping ([13]).
DMC[1] remained constant as it is ACTIVE.
DMC[2] checked DMC[0]; then selected a new mapping ([3]).
DMC[3] checked DMC[1]; then since DMC[1] is ACTIVE, $DMC[3] := DMC[1]$.

A.7 Iteration 3 - Test Phase

DMC Number	0	1	2	3
Mapping	13	11	3	11
Connected to Model FC	0	1	2	3
Result of Test	I	A	I	A

DMC[0] is INACTIVE, because

$$(Model\ FC[0] = 0) <> (SS\ FC[13+0] = 2) \quad (10)$$

DMC[1] is ACTIVE, because

$$(Model\ FC[1] = 2) = (SS\ FC[11+1] = 2) \quad (11)$$

DMC[2] is INACTIVE, because

$$(Model\ FC[2] = 2) <> (SS\ FC[3+2] = 5) \quad (12)$$

DMC[3] is ACTIVE, because

$$(Model\ FC[3] = 7) = (SS\ FC[11+3] = 7) \quad (13)$$

A.8 Iteration 3 - Diffusion Phase

DMC Number	0	1	2	3
Try DMC Number	1	*	3	*
DMC ACTIVE ?	Yes	*	Yes	*
Assign New Mapping	[11]	*	[11]	*

DMC[0] checked DMC[1]; then since DMC[1] is ACTIVE, $DMC[0] := DMC[1]$
DMC[1] remained constant as it is ACTIVE.
DMC[2] checked DMC[3]; then since DMC[3] is ACTIVE, $DMC[2] := DMC[3]$
DMC[3] remained constant as it is ACTIVE.

A.9 Iteration 4 - Test Phase

DMC Number	0	1	2	3
Mapping	11	11	11	11
Connected to Model FC	3	2	1	0
Result of Test	A	A	A	A

DMC[0] is ACTIVE, because

$$(Model\ FC[3] = 7) = (SS\ FC[11+3] = 7) \quad (14)$$

DMC[1] is ACTIVE, because

$$(Model\ FC[2] = 2) = (SS\ FC[11+2] = 2) \quad (15)$$

DMC[2] is ACTIVE, because

$$(Model\ FC[1] = 2) = (SS\ FC[11+1] = 2) \quad (16)$$

DMC[3] is ACTIVE, because

$$(Model\ FC[0] = 0) = (SS\ FC[11+0] = 0) \quad (17)$$

Thus after four iterations, all the Mapping Cells have converged onto the correct (best fit) mapping ([11]).

References

[1] Hinton, G., 1981, "A Parallel Computation that Assigns Canonical Object-Based Frames of Reference", Proc. 7th Int. Jnt. Conf. AI.

[2] Rhumelhart, D.E., and McClelland, J.E., the PDP Research Group, 1986, "Parallel Distributed Processing: Exploration in the Microstructure of Cognition", Vol. 1, the MIT Press.

[3] Bishop, J.M., 1989, "Anarchic Techniques for Pattern Classification", PhD Thesis, Chap. 5, Reading University.

[4] Welsh, W.J., Woodland, P.C. Myers, D.J., 1989, "A set of Test Problems for Assessing Neural Net Algorithms", BTRL Martlesham Heath, Ipswich.