In D. Reitter & F. E. Ritter (Eds.), Proceedings of the 14th International Conference on Cognitive Modeling (ICCM 2016). University Park, PA: Penn State.

Concept Learning, Recall, and Blending with Regulated Activation Networks

Alexandre Miguel Pinto (ampinto@dei.uc.pt), **Rahul Sharma (rahul@dei.uc.pt)**
CISUC - Department of Informatics Engineering, University of Coimbra, Portugal

Herein we present the cognitive model Regulated Activation Networks (RANs), which aims at unifying the three perspectives (symbolic, connectionist, and geometric feature-space) of conceptual representations. It learns new concepts from input data, dynamically builds a hierarchy of abstract concepts, and learns the associations among them, both between different levels, and within the same level of the hierarchy. Its recall mechanism, the geometric backpropagation algorithm, allows the understanding of the meaning of higher level concepts in terms of input level features. The regulation mechanism we also introduce has a de-noising effect over the results obtained from the recall mechanism.

Keywords: Cognitive modeling, connectionism, dynamic systems, conceptual representations.

Design, Methodology and Approach

The Regulated Activation Networks (RANs) model is based upon the Principles laid out in (Pinto & Barroso, 2014), and its geometric interpretation is inspired from the theory of conceptual spaces(Gärdenfors, 2004) whereby concepts are regions in multidimensional spaces (dimension = feature). Topologically, a RAN is a connectionist model where each node represents one dimension/feature and its activation value represents the concept's value (in the interval $[0,1]$) along that dimension. An instance of the model is initialized with one layer of nodes – one node per input data feature – and dynamically builds new nodes and new layers solely driven by the complexity in the input data.

Inter-Layer Learning As the model's instance is exposed to input data, it resorts to some user-specified clustering algorithm to identify centroids of clusters in the data. The RANs model then creates one new node per centroid in a higher layer. The coordinates of each centroid are encoded as the inter-layer weights $w_{m,n}$ associated to the edges between the newly created centroid-node n and the nodes m in the lower input layer. After the creation of the second layer of nodes, each input datum (with values in the first layer of feature/node space) can be re-represented in the second layer of centroid/node space – we obtain this re-representation via our *upward activation propagation* algorithm.

Upward activation propagation This algorithm takes an activation pattern, i.e., the coordinate values, at layer L and calculates its normalized squared euclidean distance to each centroid in layer $L+1$. These distances are then passed through a non-linear radial basis function (in this paper we used $f(x) = (1-\sqrt[3]{x})^2$ but it can be replaced by any other similarly behaving function) that behaves as an activation/-transfer function – the smaller the distance, the higher the activation of the corresponding centroid. This results in an activation pattern in layer $L+1$ with one activation value for each of its centroid nodes. Figure 1 illustrates a RANs instance with two layers: L and $L+1$, where L has i nodes $(n_1, n_2, \ldots, n_i)$ with $(a_1, a_2, \ldots, a_i)$ corresponding activation values; and layer $L+1$ has j nodes $(N_1, N_2, \ldots, N_j)$ with $(a_1, a_2, \ldots, a_j)$ activations.

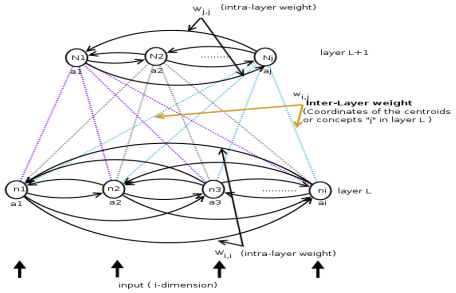

Figure 1: Learning in RANs

Intra-Layer Learning As lower layer L activation patterns get re-represented in the upper layer $L+1$ via upwards propagation, a pairwise correlation calculation takes place ate layer $L+1$: the intra-layer learning. These correlations are calculated via equation 1 and their values are stored as weights of the connections between the corresponding nodes.

$$W_{m\to n} = \frac{\sum_{k\in input\_set} [(1-|a_m^k - a_n^k|) - (1-a_m^k)*(1-a_n^k)]}{\sum_{k\in input-set} [1-(1-a_m^k)*(1-a_n^k)]} \quad (1)$$

In the numerator the part $(1-|a_m^k - a_n^k|)$ calculates the similarity of activations among nodes m and n, and the product $(1-a_m^k)*(1-a_n^k)$ is used to reduce the impact of the similarity when both activations are very close to 0 albeit similar.

Recall Concept recall amounts to obtaining, at the input feature space level, the representation of the selected higher concept node(s). Our geometric downward propagation algorithm works as follows: the user selects how strongly (s)he wishes to recall which higher layer concept(s) by injecting the corresponding activations Aj in their layer $L+1$; the algorithm generates a random activation pattern in layer L below, propagates it upward to obtain actual activation A'_j, and calculates the error $e_j = A'_j - A_j$; we use these individual errors to adjust the activation a_i of each node i in layer L below via $\Delta_{a_i} = (\sum_1^j \Delta_{a_i,A_j})/(\#j)$ where $\Delta_{a_i,A_j} = (W_{j,i} - a_i)*(e_j)$, with $W_{j,i}$ being the coordinate of centroid j in layer $L+1$ along dimension-node i in layer L. The overall impact of a_i on all A_j is summed together and normalized by dividing with

maximum possible impact i.e. $\#j$. Finally, the geometric error correction at node i of layer l is obtained by : if $\Delta_{a_i} >= 0$ then $a_i = a_i + \Delta_{a_i} * (1 - a_i)$; otherwise $a_i = a_i + \Delta_{a_i} * (a_i)$. The cycle <upwards propagation; error calculation; lower layer activation pattern correction via the error> is repeated until convergence of the lower layer activation pattern.

Regulation The recall results obtained are reasonable, but noisy. To denoise recall results a complementary Intra-Layer (IL) activation formula is developed which uses intra-layer weights to estimate each node's expected activation according to its same-layer companion nodes via

$$IL(a_n) = \frac{\sum_m \sigma_{m\to n}[(a_m * W_{m\to n}) + (1 - a_m) * (1 - W_{m\to n})]}{\sum_m \sigma_{m\to n}} \quad (2)$$

Here $W_{m\to n}$ is the intra-layer weight learned as in equation (1), and $\sigma_{m\to n} (= [2 * |W_{m\to n} - 0.5|]^2)$ is the impact factor of each correlation: $W_{m\to n=0.5}$ indicates high probability that node m has minimal (or no) impact over node n.

Our regulation mechanism uses IL activation producing a regulated activation $pr(a_n) = (1-\rho) * a_n + \rho * IL(a_n)$, where ρ is regulation-factor(a constant in $[0,1]$).

Experiments and Observations

First experiment We generated an artificial data set of 300 2-dimensional data points, c.f. Fig. 2.

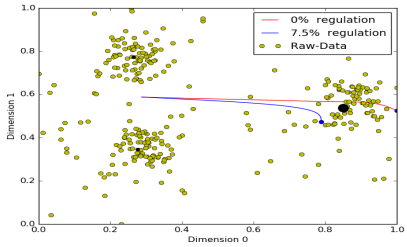

Figure 2: Observations on 2-D artificial data set

Setup The artificial data was generated such that it had 3 distinct clusters. We used K-means (MacQueen, 1967) to identify the clusters. The RANs model created 3 nodes in the first new layer (2nd layer), and 1 node in the second new layer (3rd layer), c.f., Fig. 3. To simulate recall we input the

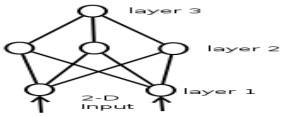

Figure 3: 3 layered model for artificial data

activation pattern $[1,0,0]$ in the second layer as expected activation *(in Fig. 2 black circles represents centers of clusters and their sizes depict expected activations)* and initiate the downwards propagation experiment.

Table 1: Observation of Artificial Data

Starting Layer 1 Act.	***Expected Layer 2 Act.***	**Regulation Factor (%)**	***Obtained Layer 2 Act.***
[0.28 0.58]	[1 0 0]	0	[**0.60** 0.1 0.12]
		7.5	[**0.68** 0.21 0.25]

Observations The algorithm randomly chooses a starting point ($[0.28, 0.58]$) and then repeats the ¡upwards activation propagation; error geometric downpropagation¿ cycle up to a maximum of 1000 iterations times; we did this for both with and without regulation. Fig. 2 shows the trajectories (each trajectory is a succession of points in the 2-D input feature space corresponding to the activations of the 2 bottom layer nodes) in 2-D. As per the expectation the trajectory obtained from regulation converges closer to the highly active center. Table 1 shows the activation at nodes in layer 2 corresponding to the converged points (without_regulation $[1, 0.52]$, with_regulation $[0.78, 0.47]$) in layer 1.

Blending Experiment with the MNIST data set We performed the experiment for blending (simultaneous recall of multiple concepts resulting in their fusion) using the MNIST (*The MNIST database of handwritten digits*, n.d.) data set with 250 images, and K-means which identified 31 clusters whose centroids are shown in Fig. 4. We create just two layers to show the concept blending operation – layer 1 has 784 nodes representing the pixels, layer 2 has 31 nodes.

Figure 4: Images represented by nodes in layer 2

Blends are obtained by injecting full activation (1) at nodes in layer 2 representing the concepts to blend, and by downwards geometric backpropagation. E.g., injecting 1 at nodes 6, 8, 25, 27, and 28 (these nodes correspond to images of digits 2 and 5, c.f., Fig. 4), and zeros in others we obtain the blend shown in the left-most image of Fig.5. In Fig. 5, the 2nd left image is a blend of 4's and 9's, the 3rd is a blend of 8's and 3's, and the last is a blend of 5's and 3's. These are not mere superpositions of the original clusters.

Figure 5: Blend of different centers

Acknowledgment

This work was supported by the project ConCreTe. The project ConCreTe acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 611733.

References

Gärdenfors, P. (2004). *Conceptual spaces: The geometry of thought*. MIT press.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth berkeley symposium on mathematical statistics and probability, volume 1: Statistics* (pp. 281–297). Berkeley, Calif.: University of California Press. Retrieved from http://projecteuclid.org/euclid.bsmsp/1200512992

The mnist database of handwritten digits. (n.d.). Retrieved from http://yann.lecun.com/exdb/mnist/

Pinto, A. M., & Barroso, L. (2014, July). Principles of regulated activation networks. In N. Hernandez, R. Jäschke, & M. Croitoru (Eds.), *Graph-based representation and reasoning: ICCS 2014* (Vol. 8577, pp. 231–244). Iaşi, Romania: Springer. doi: 10.1007/978-3-319-08389-6_19