[Music]

Welcome back to the lecture on Hopfield networks and Boltzmann machine part 2. What we now want to leave Hopfield networks as such and look at some related approaches.

So Boltzmann machines are also recurrent neural networks that can be seen as the stochastic generative counterparts of Hopfield networks. The non determinism allows to come out of local minima, so I would say that is the main feature here. Boltzman machines were invented in 1985 by Geoffrey Hinton and Terry Sejnowski, And they are also called stochastic Hopfield networks with hidden units. So as you remember all units in Hopfield network were mutual input-output nodes, so one could say Hopfield is a one layer neural network while the Boltzmann machines may have hidden units. But as for Hopfield networks we have symmetric connections between units. Finding optimal States involves relaxation, so now we talk about optimal, we don't I don't talk about necessarily local optima, as many of the hill climbing or steepest descent approaches that we looked at during the course, here we talk about a method that has some probability of getting out of this local optima to find the global one. So in the Boltzman machine setup we have introduced some kind of relaxation procedure where the student can see to say jump out or potentially above the local minimum and this technology technique is called annealing, simulated annealing. So Boltzman machine after the Boltzman distribution in statistical mechanics which is used in their sampling function, they are also called energy based models.

So actually the state changes are occurred can either credited mystically if the Delta energy value is less than zero or Stochastically with a certain probability and that probability normally is defined as one divided when one plus e raised to minus Delta energy divided by a parameter T and T is so-called temperature of the model, and one important idea we have in annealing we'll come to that is that you start with the high temperature and decrease the temperature during the process which of course and will affect the probability of making state moves.

Looking at the process in a Boltzman machine on this slide there is a simple scheme. So actually we look at one example a time and for every sample we run the network in my two phases. So the positive phase is defined by you innovate clamp or lock the visible units with the pattern specified in the example and then let the network settle using this annealing technique and then you record the opposite of units. And you can iterate that a number of times and then you do it

again but then you left the all the units free and you follow the same procedure you do in annealing etc and then you compare the outcome of the two different phases and then one compute the probably odds that both units the two units in each pair of unit is Co active and the negative phase, so this gives us a set of probabilities and actually then what we do when we update the weights we base it on some statistics, so we update them with respect to the differences between the probabilities calculated from the two phases k is just learning if you've seen it many times now is just the damping factor. So this is the procedure we will next go and talk a little about the simulated annealing process.

So the source of inspiration for simulated annealing and actually annealing as it happens in metals, which means that you initially heat the solid state metal to a very high temperature which makes it possible for atoms to move around relatively free, but then you slowly cool the metal down according to some schedule and so if you start very high and cool it down very slowly, then you can achieve the atoms will place themselves in a pattern of response to the global energy minimum of a perfect crystal. So of course you can see the analogy here what we want is a optimized solution to our problem depending on how is modular or problem that means that we want to find a global minimum. So that's the analogy we strive for. So one way of describing some well assimilated annealing the steps in such an algorithm, would be the following so first you initialize, you start with a random initial configuration, you initialize a very high temperature value of that parameter and then you propose a change move or change of the configuration and then you calculate some score in your case, in our case it's the Delta energy if we still keep to the to that property of the Hopfield network that we inherit and so we calculate that Delta due to the proposed move and then depending on the Delta, the move is accepted or not according to the earlier given formula. And of course as you can understand depending on which step in the iteration we have in each step we have a certain temperature in a beginning we have a high temperature which may give a higher probability for making changes but when we then for the next we update the temperature and we do it by lowering it in a in a certain pace and of course when we lower the temperature by a for each iteration the likelihood that actually a move will be taken diminish. So we continue then this until we came to what could be called a freezing point and of course a freezing point need to be defined for each domain where we apply this in. This slide just shows you the phenomena that in contrast to a Greedy algorithm the possibility to take still take moves even if you if you reach the local minimum you can still take this tentative

moves that may bring you out of the minimum, and hopefully then in the end towards the global minima weight you can see in the end at the bottom. One issue with Hopfield networks and restrictive Boltzmann machine in practice is the connectivity, an unlimited connectivity of this network. So therefore there are a number of approaches where one in a way different ways limits the connectivity. So one such architecture is called restricted Boltzmann machine RBM and where you decide that the neurons as a system must form a bipartite graph. So bipartite graph is a graph where there cannot be any connections between the neurons or nodes within one of the one of these parts, so the graph is divided into part and there is a group of neurons in each and between neurons in one group there cannot be any connection, so it can be only be connection between yours that are in two groups. So and actually the two groups are typically the visible the visible neurons and units or the hidden units. This method is rather old but actually it get more popular just ten years ago when the researchers together with Hinton developed some better algorithms for this method and after that is come into more might use. Another related approach is called bi-directional associative memory abbreviated BAM and the interesting difference here is while Hopfield networks are auto associative which means if you remember that in Auto associative memory you can intrinsically only retrieve objects or patterns that are more or less of the same exactly the same form. So you enter a partial a partial description and you can get the full part or format wise the same kind of object. While in hetero associative memory and BAM is an instance of that, you can actually retrieve totally different patterns and the reason for that is possible in BAM is that, BAM work with actually two separate layers and in one layer you can have items on one format and then in the other layer you can have items with a totally different format. The two layers are fully connected so this means that there is an unlimited way of really connecting the two classes of patterns. So by having these layers where there are free format in both, you could possibly live up to the demands of what is termed hetero associative memory. We don't have much time to go into the details of BAM but on this side you can find a simple example that illustrates somehow how things work, so actually what you do is your store association between is pattern in this memory and you can also observe it up to the left that the two patterns that we associate cannot totally different a different form. So actually what is possible to do is to create a matrix from the sum of the products of these associated vectors and then that matrix can be used so when you want to retrieve a pattern you can then in this case the B the second part you can multiply that the vector and get the other part back, and if you do onto

the opposite you can use the transpose of this matrix. So there are no details here sorry for that but it's just to give you a feeling for how this retrieval can be done and also the fact that you in this case can have the association between totally different structures. Another interesting variant of associative memory is what's called the Kohonen network or self-organizing maps SOM. So SOM implements a form of unsupervised learning and in you know it takes essentially complex nonlinear statistical relationships between our dimensional data and maps it onto a low dimensional display, so actually one of the results here is to do dimensionality reduction. As this lever compresses information while preserving the most important topological and metric relationships of the primary data items on the display. It also may also be thought of give some kind of abstractions. It was invented in the 1980s but by Teuvo Kohonen and the map is predefined and usually relates to finite two-dimensional region where no nodes are arranged in a regular grid and you'll get the idea from the right. And then you can see you have input vectors below and then one can say that this model is fully connected in the sense that all inputs are actually connected not to themselves among themselves but all the inputs are connected to all the elements in this two-dimensional grid.

So actually the SOM network in the SOM network if one look at the nodes in in this in this flat space of nodes apart from being connected to every input it's also that it node keeps a weight factor where one weight for each connection and so that is built up during the training phase but in the testing phase actually what the system does, is it for each test case tries to find the node where with the weight vector that gives the smallest magical distance to the to the test case.

So by this we finish this lecture on Hopfield networks and related work, thank you very much for your attention the next lecture six point eight will be on the topic of convolutional networks.