Computational Neuroscience: Neuronal Dynamics of Cognition

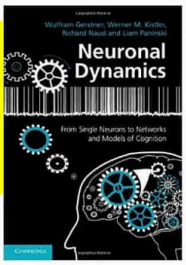
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

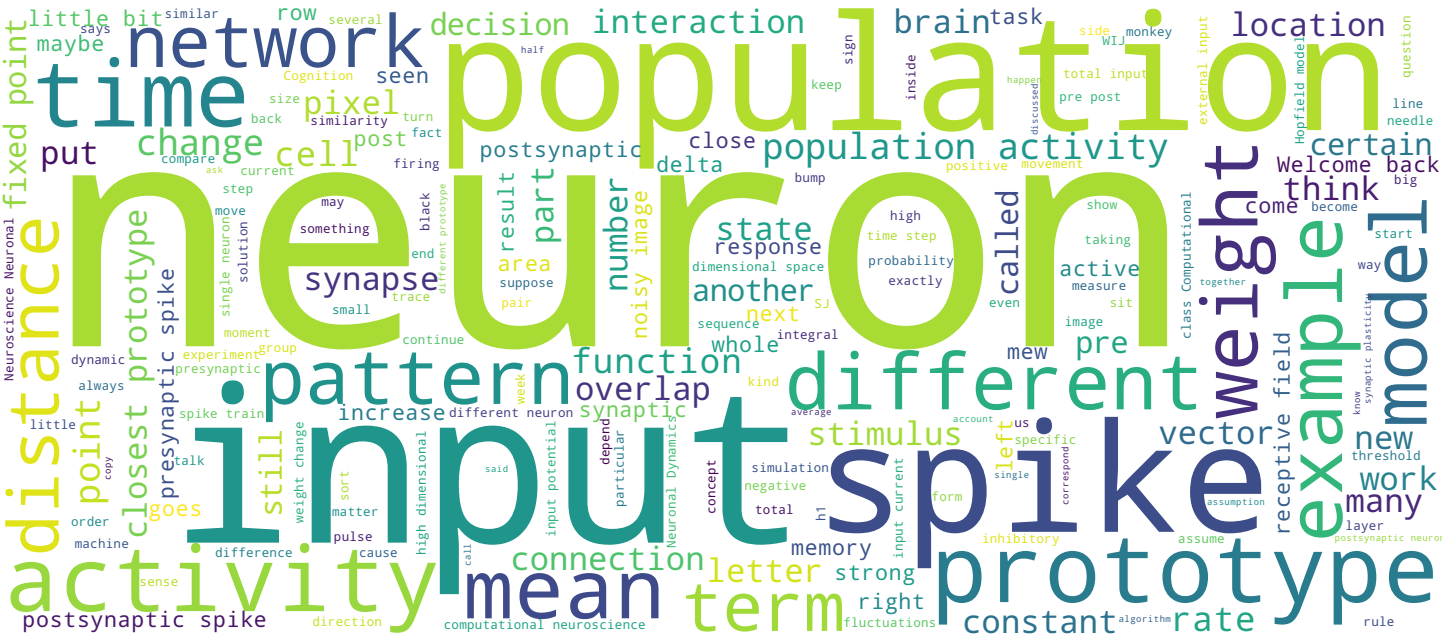
A: ASSOCIATIVE MEMORY in a Network of Neurons

Wulfram Gerstner
EPFL, Lausanne, Switzerland

Reading for this week:
NEURONAL DYNAMICS
- Ch. 17.1 - 17.2.4

Cambridge Univ. Press



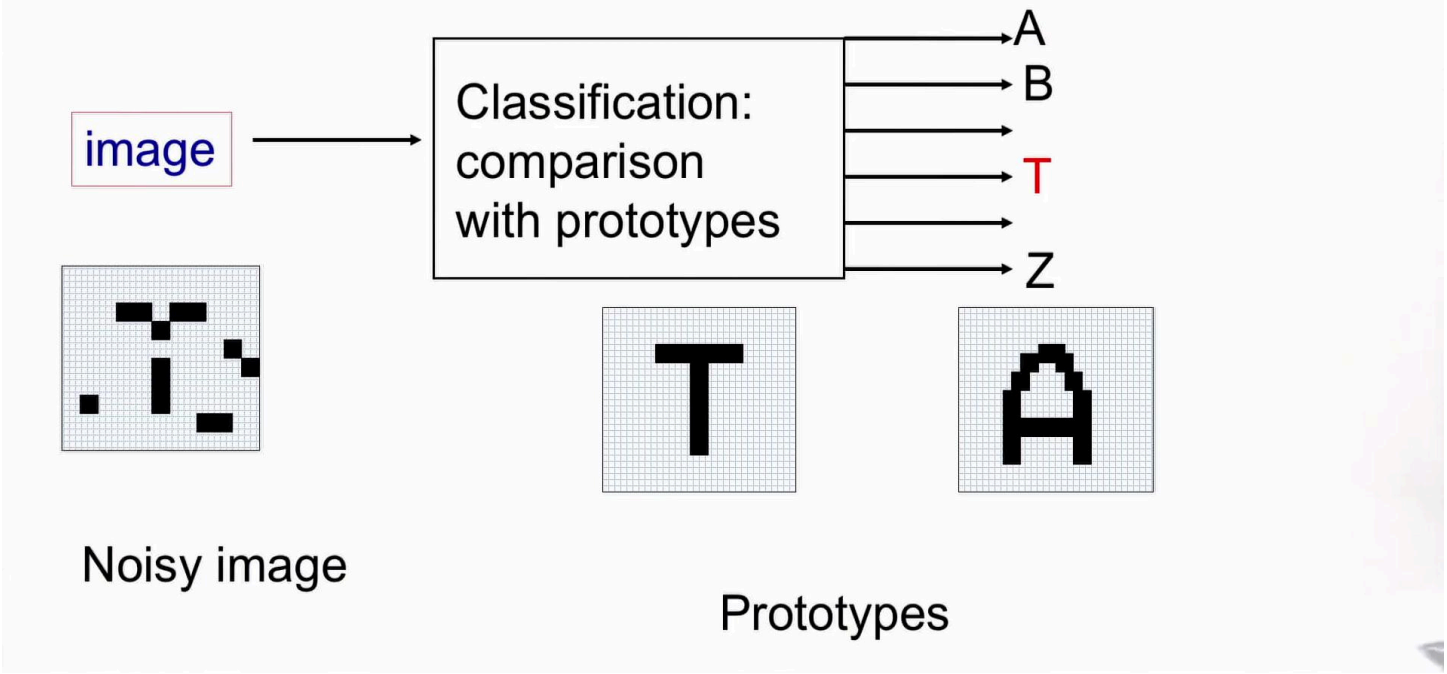


Search MOOC

Video

EPFL

2. Classification by similarity: **pattern recognition**



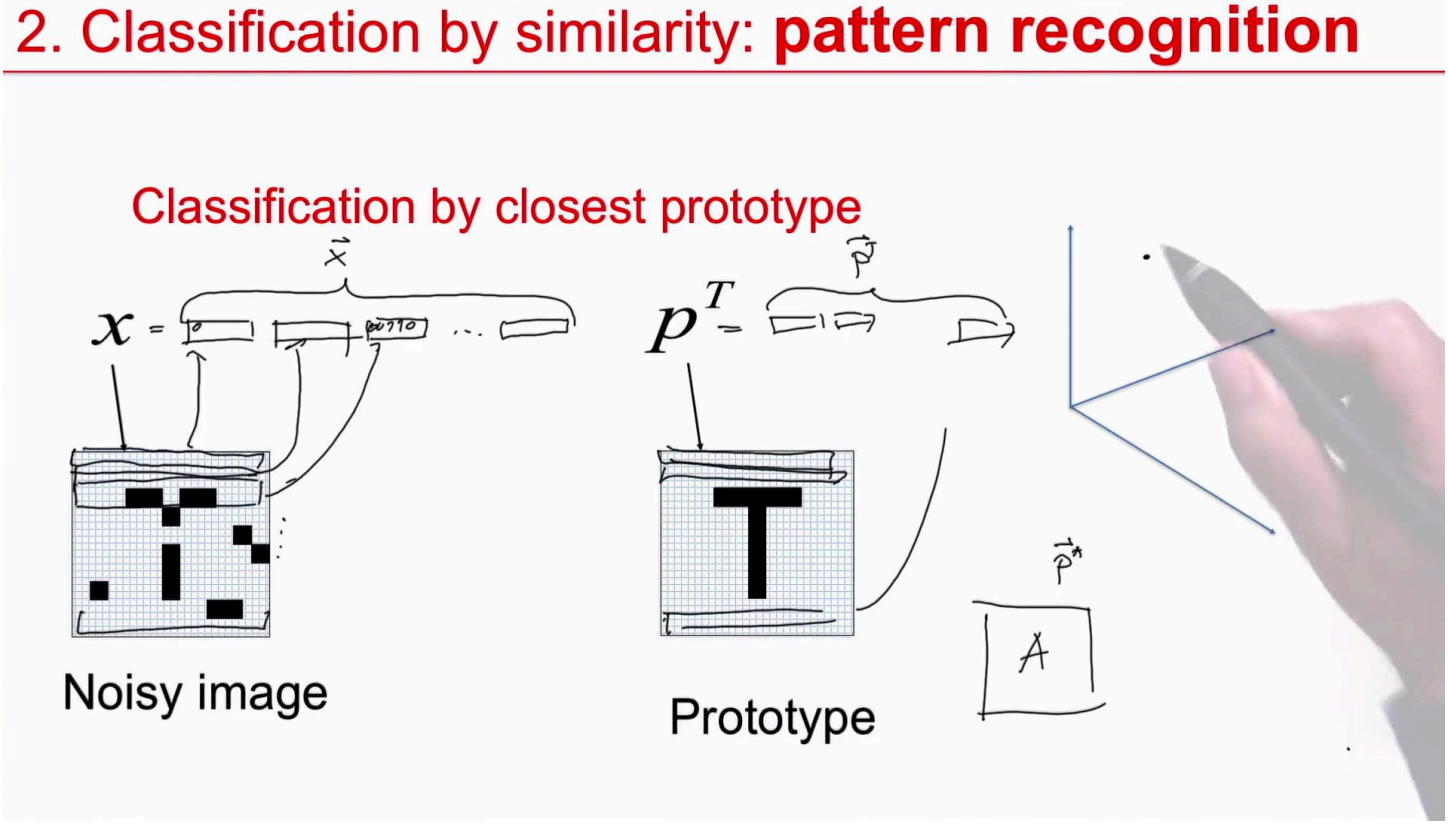
So welcome back. So what we have seen is that our brain fills in missing information. And instead of taking the example of a partial verb that's completed to the full word, let's now look at a partial image or a noisy image. And the task would be to find the purified image, the prototype, in this case, the prototype of a character. So you can imagine that we have a machine and inside the machine there, we have different prototypes, a prototype for A, B, C, D, and become a noisy image. And what comes out of the machine is the prototype that matches best in that sense, it's the closest prototype. So let's work on this a little bit.

Notes

Summary

0m 05s

2. Classification by similarity: **pattern recognition**



So I have here a noisy image. And even though I show it as a two-dimensional image that we use to think about it's just a vector. So think of taking the first row. So think of taking the first row and shifting it up here, that's the first row, then you take the second row, and paste it next to it. And then you take the third row and put it up there. And so you continue. And this is the final row that you put at the end. Now, each thing here is a sequence of say, zero for blank, and one for pixel, that is black. And the whole thing then is a vector. And this is my vector x. So this is the current state, this is the input. And I compare this with one of these prototype patterns. And here I do the same, I have here the different lines that I just put next to each other. And so this is my prototype for the letter T. And then I would have another prototype, say, a prototype for the letter A, I do the same thing. This is again, a vector prototype for the vector a. And now I can visualize these vectors which sit in a high dimensional space. So if this is 32 times 32 pixels, that would make thousand 24 dimensions, I cannot visualize 1024 dimensions, but again, just draw a point x.

Notes

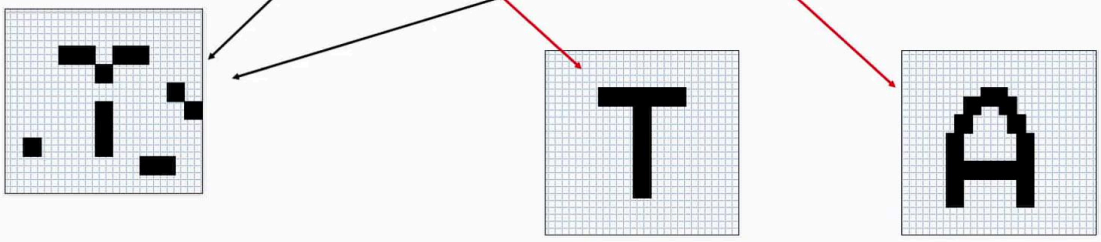
Summary

0m 48s

2. Classification by similarity: **pattern recognition**

Classification by closest prototype

$$\left|x - p^T\right| \leq \left|x - p^A\right|$$



Noisy image

Prototypes

So this point x here is now just a point in this high dimensional space, I have my prototype, this is the prototype for T. And then maybe I also have another point in the high dimensional space, which is the prototype for A. Now, in order to find the closest prototype, I would like I would look at the similarity or more precisely at the distance between my input x and the prototype P. And it will also look at the distance between the input x and some other prototypes, for example, the prototype of A and then let us compare this distance x minus P for the letter T, the prototype for the letter T, with the distance between x and the prototype for the letter A. And if this distance is more than this distance, then I would classify X as belonging to the prototype P for the letter T. Okay, so if this distance is smaller compared to the distance to a lot of potential prototypes, then the classification is that this is a T so X image at the input is mapped to the closest prototype, to classification by closest prototype means I measure distances, and then I have to rank and find the one with the smallest distance.

Notes

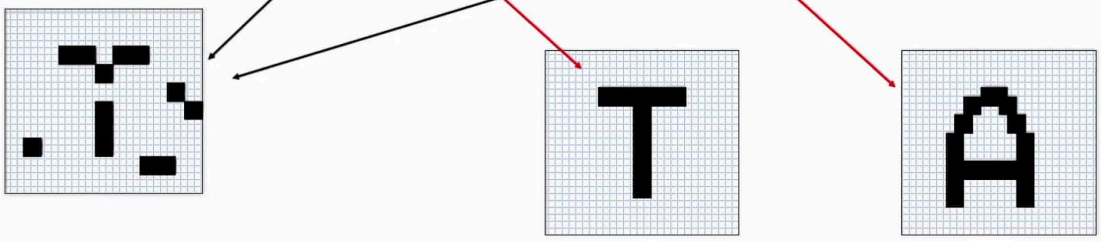
Summary

2m 23s

2. Classification by similarity: **pattern recognition**

Classification by closest prototype

$$\left|x - p^T\right| \leq \left|x - p^A\right|$$



Noisy image

Prototypes

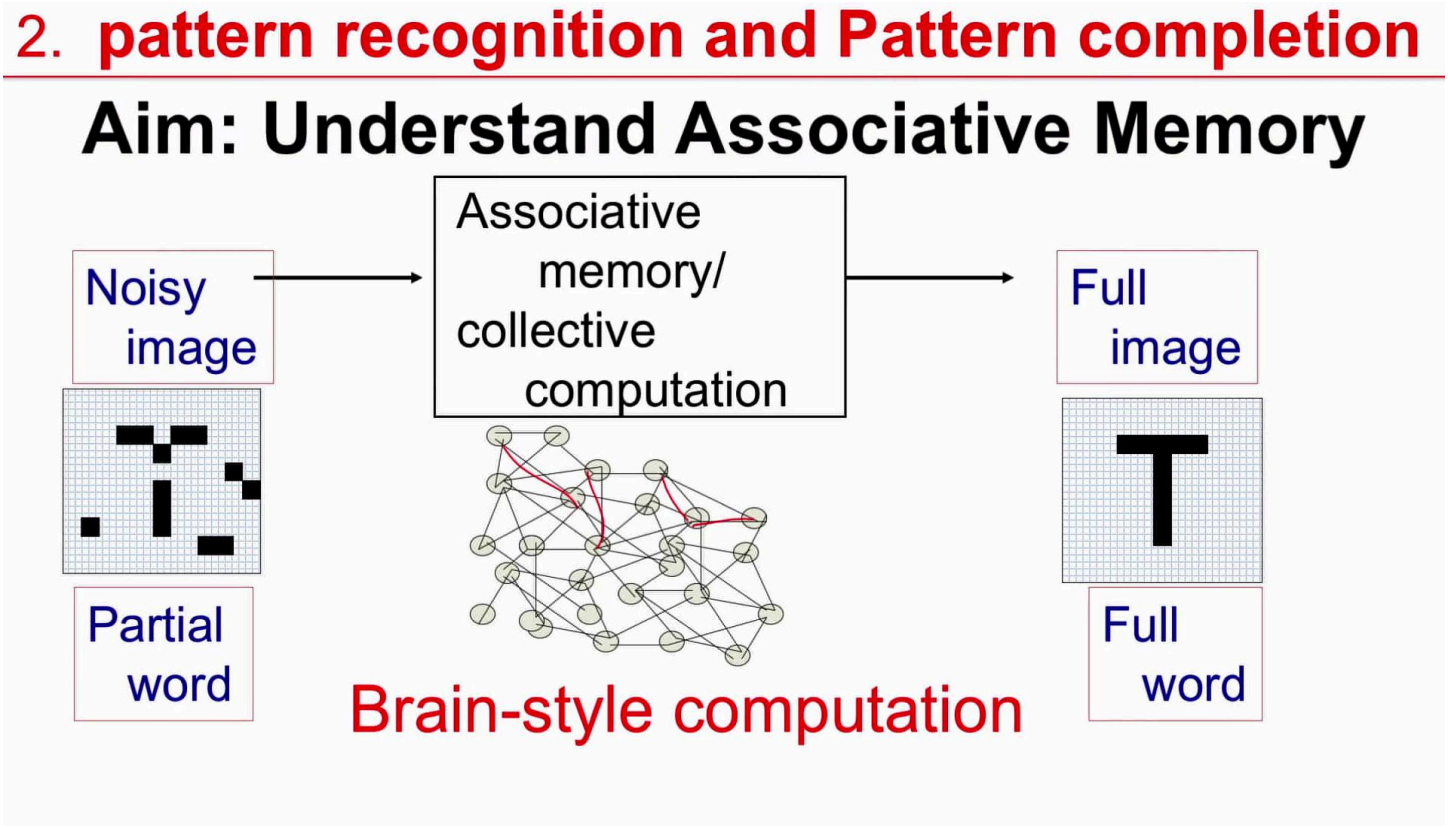
So, this is sort of algorithm that you could implement into a computer. And it's just a few lines of algorithm you measure the distances with all the different prototypes than your sword, you find the shortest distance, and for the gift bag is the closest prototype.

Notes

Summary

3m 53s

2. pattern recognition and Pattern completion



Now, we want to do the same processing, we have incomplete information, a noisy image or partial word. And we want to map it to the full version or the full image or the purified prototype. But we don't want to do it in this traditional algorithmic CPU based sequential manner, but we want to do it in a brain-style fashion. So in particular, we don't want to have a central controller that says So in particular, we don't want to have a central controller that says either this or neither that running through a sequence of comparisons, what we want is that it's just the interactions between the different neurons. That takes out the closest prototype, and how to do this. That's the task, the content of the following lectures. But before we go to the next section, let's just have a quick look at the next quizz.

Notes

Summary

4m 08s

Quiz 2: Closest prototype

Classification by closest prototype (tick one or several answers)
[ ] Needs a similarity measure
[ ] Needs a distance measure
[ ] Needs a method to find the maximum or minimum

Notes

Summary

4m 57s