Informatics 1 Cognitive Science – Tutorial 2

Frank Keller, Carina Silberer, Frank Mollica

Week 3

The goal of this tutorial is to discuss models of past tense verbs and to build an understanding of how perceptrons work. Before working on the exercises, you should make sure that you have revised the slides for lectures 4 and 5.

1 Modeling Past Tense Verbs

We've discussed how to explain the facts of regular and irregular past tense verbs in English and introduced the **words-and-rules model** in the lectures.

Exercise 1 List some salient features of this model, in particular what it has to say about how adult speakers produce regular and irregular verb forms in English.

We can evaluate a model by determining the extent to which it accounts for the empirical data and experimental observations we are trying to explain. Please refer back to the experimental facts and empirical observations about the past tense in English that we discussed lecture 4.

Exercise 2

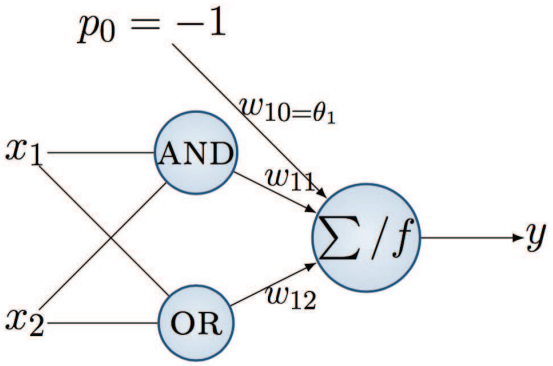
1. Evaluate the words-and-rules model based on whether or not it accounts for these facts and observations.

2. What would we have to add to the words-and-rules theory, to make it a better theory of regular and irregular verbs?

2 Perceptrons

In the lecture you learned that the primitive Boolean functions AND and OR can be represented by a single perceptron each. You also saw that a single perceptron does not have the representational power to represent the Boolean function XOR. A network of perceptrons two levels deep can in fact represent every Boolean function.

Exercise 3 Consider the two-level network illustrated below. It is composed of three perceptrons. The two perceptrons of the first level implement the AND and OR functions, respectively. We assume that their weights are fixed. However, we assume that the weights of the second layer can be adjusted.



Determine the weights w_{11}, w_{12} and threshold $\theta_1 (= w_{10})$ such that the network implements the XOR function. The initial weights are set to zero, i.e., $w_{11} = w_{12} = w_{10} = 0$, and the learning rate η is set to 0.1 (Feel free to choose other initial values for $w_{1.}$ and η).

Notes

- The input function for the perceptron on level j is the weighted sum $\sum$ of its input.
- The activation function f for a perceptron is a step function:

$$f = \begin{cases} 1 & if \sum > 0 \\ 0 & otherwise \end{cases}$$

- The threshold θ is considered as a weight, with input $p_0 = -1$.
- Assume that the weights for the perceptrons of the first level are given, meaning that these perceptrons already compute AND and OR correctly and their weights do not change.
- You don't need to do the computations manually. Use the spreadsheet `xor-nosols.xlsx` that comes with the tutorial, or write your own code.