Workshop on Artificial Intelligence in Practice

Part 2: AI Details and Applications in Navigation and Public Service

Sebastian Iwanowski
FH Wedel (D)
Erasmus Workshop at Fontys University, Eindhoven (NL)

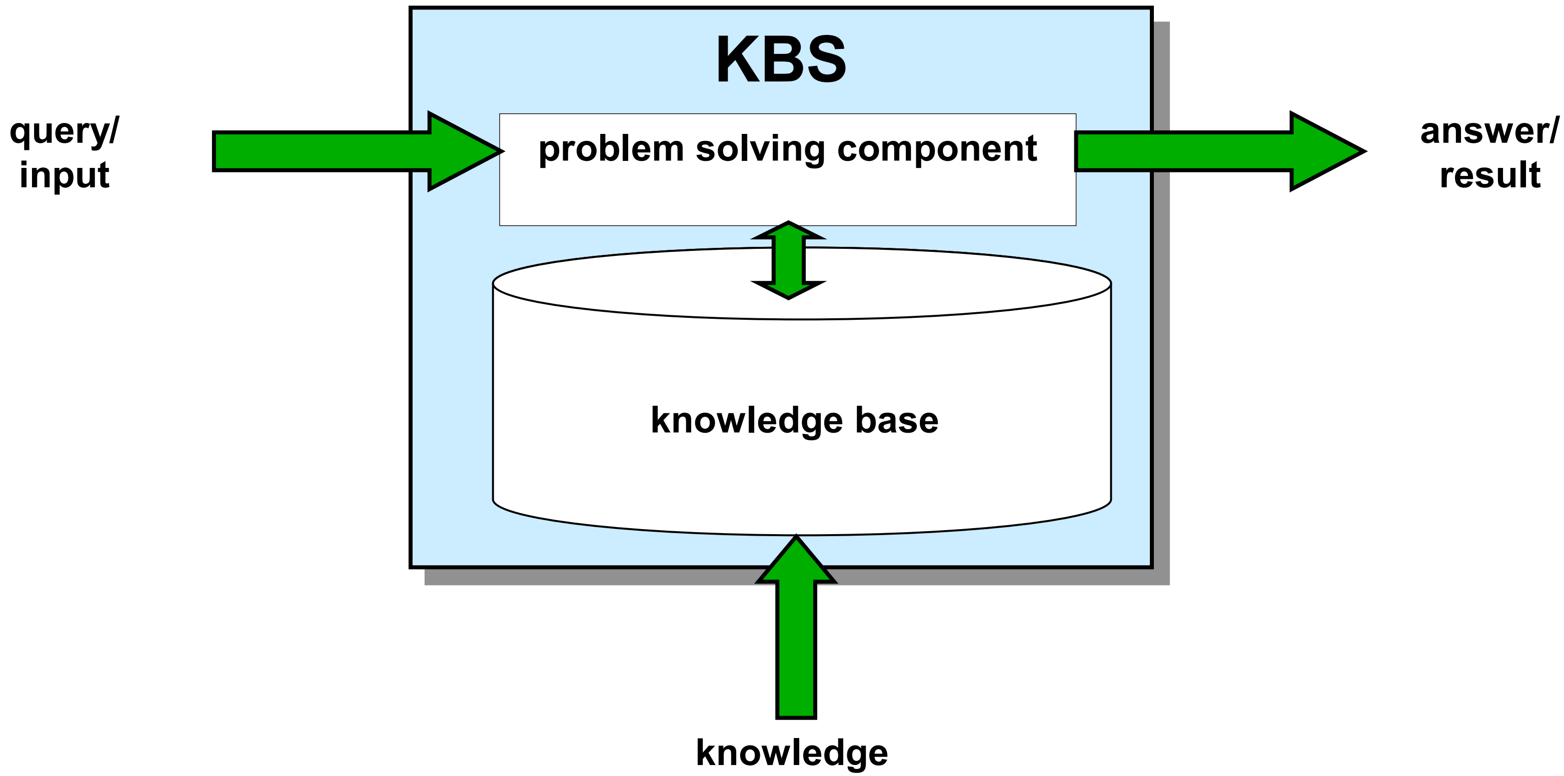
Section 3: Neural Networks

Basic Technology: Knowedge-Based System

knowledge + **problem solving component** = **KBS**

data **processing rules**

Architecture KBS (in general)



Case-Based Reasoning (CBR)

Underlying Principle

- Represent values which are related to each other as vectors.
  A given set of vectors describes a certain relation class.

Application example: Distinguish normal from faulty behaviour

- Give a set of complete vectors (where all coordinates are known):
  These vectors will be "learnt" and represent the knowledge base.
- At run time, give an incomplete vector (where some coordinates are missing):
  It is the task of the problem solver to classifiy this vector:
  Assign it to a class among the learnt examples.
- One way is to assign values to the coordinates missing and to use the newly assigned values in order to classify the vector.

Problem solver's work (simple way):

- For each unknown vector, find the most "similar" one in the knowledge base.
- This corresponds to the assignment of all coordinate values of the chosen reference vector to the new vector.

This procedure only makes sense if all values come from a noncontinuous (better: finite) domain!

Case-Based Reasoning (CBR)

Adaptation to continuous value domains:

Problem solver‘s work (more sophisticated variant):

- For each unknown coordinate of a new vector, assign values that are “in between” the values of reference vectors of the knowledge base which are “in the neigborhood” of the new vector.

Functional formulation of this problem:

- Consider the unknown coordinate values to be function values of the known coordinate values: Find (piecewise) linear functions such that all reference vectors of the knowledge base are solutions of these functions.
- Assign the unknown coordinates the values obtained by these functions.

Task: **How do we find the function parameters for a given set of reference vectors?**

Case-Based Reasoning (CBR)

Regression analysis:

Linear Regression:

- Find the weights W_i of a linear function of the form: $f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} W_i x_i$

Generalisation:

- Find the weights in a linear equation system:

$$f_1(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} W_{1,i} x_i$$

$$f_2(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} W_{2,i} x_i$$

$$.$$

$$.$$

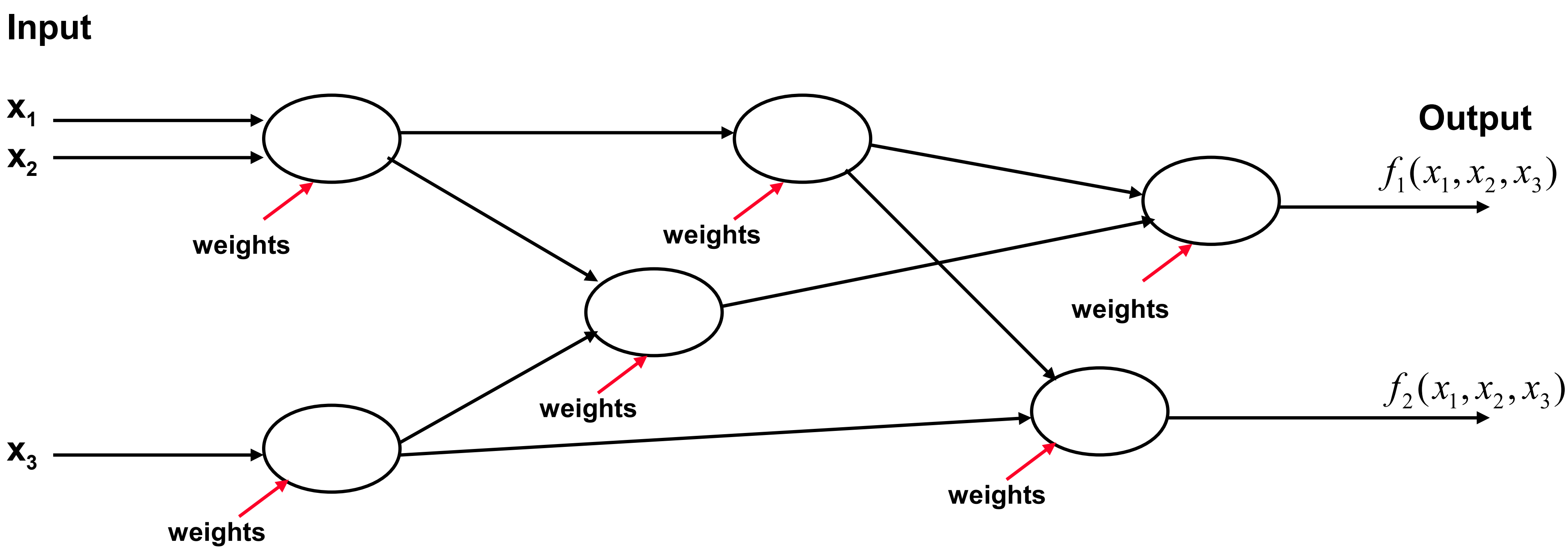
$$f_j(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} W_{j,i} x_i$$

Idea: Instead of a monolithic system consisting of heavy equations, work rather with a concatenated set of light-weight equations

Case-Based Reasoning (CBR)

Principle of neural networks:

Given a multi-valued linear function f **(defined by:** $f_i(x_1, x_2, \ldots, x_n)$**)**

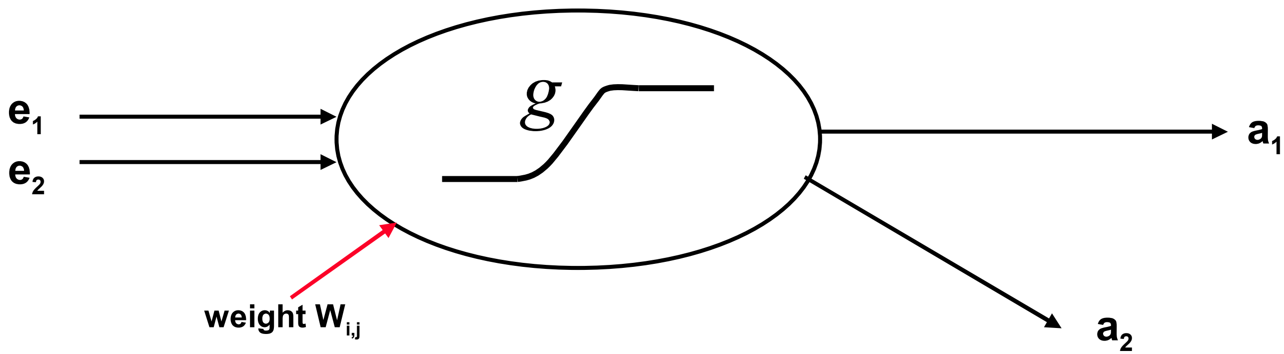


- Learning phase:
  The weights are consequently adapted to new reference examples (with given input and output)

- Run time phase:
  If a new input is given, compute the output applying the network with the weights computed.

Case-Based Reasoning (CBR)

Functionality of a single neuron:



$$a_i(e_1, e_2, \dots, e_n) = g\left(\sum_{j=1}^{n} W_{i,j} \cdot e_j\right)$$

- g is a generalised threshold function which is the same to all inputs.

Case-Based Reasoning (CBR)

Different types of neural networks:

Neural networks without intermediate layers:

- Two layers of neurons: The first is linked to the input, the second is linked to the output.

Neural networks with intermediate layers:

- Input and output layers are not connected directly but only via “hidden” layers.

Neural networks with feedback:

- This is how “memory” is formed.

Case-Based Reasoning (CBR)

Neural networks are history now.

Modern case-based engines work with

Support vector machines

Diploma thesis at FH Wedel (2008):

Helga Karafiat: Gender recognition at faces

Supervisor: Prof. Dr. Wolfgang Ülzmann

web: http://www.fh-wedel.de/~kar, http://www.fh-wedel.de/~ue **all information in German**