TD 2 – Dynamic Programming

Abstract

The goal of this tutorial is to use an algorithm which allows to find the best path for sequence alignment: the dynamic programming algorithm. You will study the practical aspects of the method. In practical works, you will implement this algorithm and a voice recognition system based on isolated words.

1 Dynamic Programming Algorithm

1.1 Introduction

The algorithm allows to make an alignment between two sequences A and B. The optimal path ("warping function") is represented by the path C.

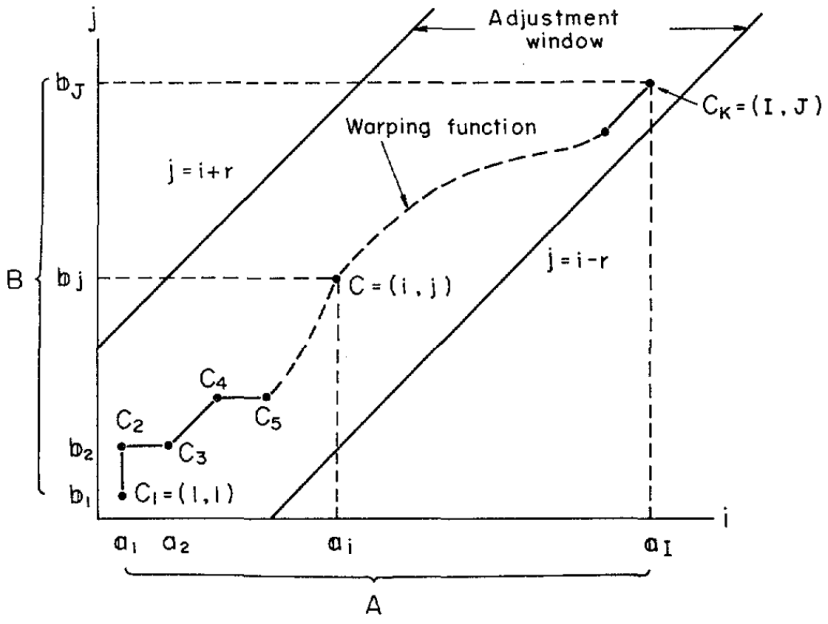

Figure 1: Global scheme of the method which aligns two sequences A and B

1.2 Algorithm

```
1  begin
2  |  g(0,0) ⟵ 0;
3  |  for i ⟵ 1 to J do
4  |  |  g(0,j) ⟵ +∞;
5  |  end
6  |  for i ⟵ 1 to I do
7  |  |  g(i,0) ⟵ +∞;
8  |  |  for j ⟵ 1 to J do
9  |  |  |  /* search of a minimal path                                   */
10 |  |  |  g(i,j) ⟵ min(g(i−1,j) + ω0 ∗ d(i,j), g(i−1,j−1) + ω1 ∗ d(i,j), g(i,j−1) + ω2 ∗ d(i,j));
11 |  |  end
12 |  end
13 |  D ⟵ g(I,J)/(I + J);
14 end
```

2 Applications

2.1 Comparing two numerical sequences

Compare the two following numerical sequences using dynamic programming:

- (-2, 10, -10, 15, -13, 20, -5, 14, 2)
- (3, -13, 14,-7, 9,-2)

The local constraints are $(\omega_0, \omega_1, \omega_2) = (1, 1, 1)$. In other terms, it remains computing $g(i, j)$ as:

$$g(i,j) = \min \begin{cases} g(i-1,j) + d(i,j) \\ g(i-1,j-1) + d(i,j) \\ g(i,j-1) + d(i,j) \end{cases}$$

The distance $d(i, j)$ is the classical Euclidean distance.

2.2 Comparing DNA sequences

Compare the following DNA sequences:

- ATGGTACGTC
- AAGTAGGC

To compute the matrix g, the global constraint will be used: a cell of the matrix situated at more than 4 cells from the diagonal should not be computed. Moreover, the local constraints are $(\omega_0, \omega_1, \omega_2) = (1, 1, 1)$.

Distance to be used:

$$d(B_i, B_j) = \begin{cases} 0 \text{ if } \ B_i = B_j \\ 1 \text{ otherwise} \end{cases}$$

This means that, the distance between two different symbols (DNA base: A, C, G or T) is equal to 1 and the one between two identical symbols is 0.
Indicate the final cost of the comparison of the sequences and the resulting alignment of the best path.

2.3 Comparing audio signals

We would do the speech recognition of the numbers "cinq", "vingt" and "cent" ("five", "twenty" and "hundred" in french). The acoustic signal is analyzed during time frames of 20ms every 10ms. This preprocessing consists in tagging every signal frames using an acoustic references dictionary:

$$D = X, C, U, U_X, V$$

The references represent:

- X: frame not voiced noise;
- C: silence frame;
- U: frame of a voiced consonantal zone;
- U_X: frame of a voiced and noised consonantal zone;
- and V: frame of a formantic zone.

We suppose that the numbers "cinq", "vingt" and "cent" are respectively coded by: "$XVVCX$", "U_XVV" and "XVV".

In Table 1, the distances between acoustic references are given. The dynamic programming algorithm used for temporal sequences is named DTW (Dynamic Time Warping) algorithm.

Distance	X	C	U	U_X	V
X	0	1	1	1	2
C	1	0	1	1	2
U	1	1	0	1	1
U_X	1	1	1	0	2
V	2	2	1	2	0

Table 1: (Symmetric) distances between acoustic references

2.3.1 Question 1

Using the distances in Table 1 and the local constraints (1, 2, 1), construct the best path and compute the score obtained for each reference word (each french number) using the DTW algorithm considering the observed audio signal "$XXVUXCX$". What is the french recognized number ?

2.3.2 Question 2

We suppose that we can know recognize "cinq", "vingt", "cent" as previously but also composition of some of these numbers: "cent cinq" and "vingt cinq" ("hundred five" and "twenty five"). We talk about connected words. Use the DTW algorithm on the connected words with the observed audio signal "$U_XU_XVXXVU_XXXV$". What is the french recognized number ?