

Sprachverarbeitung: Übung 12

DTW-Algorithmus

Die Ähnlichkeit zweier Sprachmuster kann z.B. anhand der mittleren cepstralen Distanz beurteilt werden. Vor dem Berechnen dieser Distanz sind jedoch allfällige Unterschiede der Sprechgeschwindigkeit auszugleichen, d.h. die Sprachmuster müssen zeitlich einander angepasst werden. Dazu wird der DTW-Algorithmus (*dynamic time warping*) eingesetzt.

In dieser Übung sind zwei Sprachmuster **X** und **Y** gegeben, d.h. zwei Sequenzen von Merkmalsvektoren (Mel-Cepstren), die aus den Sprachsignalen `u12_sig_x.wav` und `u12_sig_y.wav` berechnet werden. Auf diesen Sprachaufnahmen ist je einmal das Wort “vierzehn” mit unterschiedlicher Betonung zu hören.¹ Die Lautdauern sind deshalb in den beiden Mustern recht unterschiedlich² und eine zeitliche Anpassung (Zeitnormalisation) ist somit für einen Vergleich unabdingbar. Ein geeignetes Vergleichsmass ist die minimale normierte cepstrale Distanz aus dem DTW-Algorithmus (siehe Formel (184) im Buch).

Aufgabe 1: Zeitnormalisation (DTW)

Berechnen Sie die minimale normierte Distanz entlang der optimalen Warping-Kurve, welche den Pfaderweiterungen von Figur 1 entspricht. Die Warping-Kurve soll im Punkt (1,1) beginnen und im Punkt (L_x, L_y) enden, wobei L_x und L_y die Längen der Muster (Anzahl der Analyseabschnitte) sind.

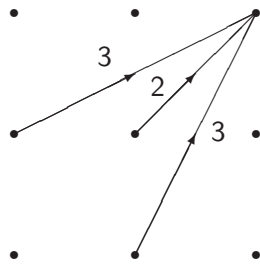
Schreiben Sie sich zuerst für die Pfaderweiterungen und Pfadgewichte der Figur 1 die Rekursionsformel für die akkumulierten Distanzen auf. Die erlaubten Startpunkte relativ zu einem bestimmten Gitterpunkt sind durch diese Pfaderweiterungen gegeben.

Im vorgegebenen Matlab-Rahmenprogramm `ueb12_frame.m` werden fünf Teilfiguren gezeichnet. Sie stellen Folgendes dar:

1. die beiden Merkmalssequenzen **X** und **Y** (nur der nullte Koeffizient des Mel-Cepstrums $\check{c}(0)$ in Funktion des Indexes des Analyseabschnittes wird gezeichnet)
2. die Warping-Kurve mit der berechneten Distanz
3. die einander dynamisch angepassten Merkmalssequenzen (wiederum $\check{c}(0)$ wie bei Punkt 1)

¹Diese Signale können mit den Matlab-Befehlen `[s,fs] = audioread('u12_sig_x.wav');` `sound(s,fs)` abgespielt werden.

²Dies ist aus der ersten Teilfigur aus Aufgabe 1 ersichtlich, die mit dem Matlab-Skript `ueb12_frame.m` erzeugt werden kann.



Pfaderweiterungen: $\text{pe}\{1\} = [-2 \ -1];$
 $\text{pe}\{2\} = [-1 \ -1];$
 $\text{pe}\{3\} = [-1 \ -2];$

Pfadgewichte: $\text{pg}\{1\} = 3;$
 $\text{pg}\{2\} = 2;$
 $\text{pg}\{3\} = 3;$

Figur 1: Pfaderweiterungen und Pfadgewichte für Aufgabe 1: Graphische Darstellung und entsprechende Spezifikation in Matlab

4. die lokalen Distanzen (auf der rechten Seite)
5. die akkumulierten Distanzen

Gehen Sie vom gegebenen Rahmenprogramm aus und ergänzen Sie die mit `%>>` markierten Stellen. Während der Programmentwicklung ist es zweckmässig, die Anzahl der Abtastwerte zu begrenzen, z.B. mit `ns = [1 700];`, damit das Testen speditiver geht.

Hinweise zum Matlab-Rahmenprogramm:

Distanzmatrix: In der Distanzmatrix `d` werden die lokalen Distanzen eingetragen. Die lokale Distanz `d(i,j)` gibt an, wie verschieden der Analyseabschnitt `i` des Signals `x` und der Analyseabschnitt `j` des Signals `y` sind. Als Distanzmass wird die euklidische Distanz der Mel-Cepstren (ohne $\check{c}(0)$) verwendet.

Akkumulierte Distanzen: Die akkumulierten Distanzen werden in die Matrix `Da` eingetragen. Sie wird am Anfang mit dem Wert `inf` initialisiert. Zudem wird `Da(1,1)`, also der Punkt, wo die Warping-Kurve startet, gleich `d(1,1)` gesetzt.

Pfaderweiterungen: Die Pfaderweiterungen werden in der Zellenmatrix `pe{i}(j)` angegeben (siehe Figur 1). Dabei unterscheidet der Index `i` die einzelnen Pfaderweiterungen und der Index `j` bezeichnet die x-Komponente (`j=1`) bzw. die y-Komponente (`j=2`) eines Punktes.

Pfadgewichte: Die Gewichte der einzelnen Pfade werden analog zu `pe{i}(j)` in der Zellenmatrix `pg{i}` spezifiziert.

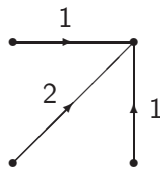
Zeigermatrix: Für die Speicherung des jeweils optimalen Teilpfades (Pfad `i` der Pfaderweiterungen, der zum Punkt `(x,y)` geführt hat) wird wiederum eine zweidimensionale Matrix `psi(x,y)` verwendet.

Optimaler Pfad: Der optimale Pfad (Warping-Kurve) `P(k,1:2)` mit `k=1...L` wird unter Verwendung der Pfaderweiterungen aus der Matrix `psi(x,y)` herausgelesen. Dabei ist selbstverständlich von hinten zu beginnen. Überlegen Sie zuerst, was die Abbruchbedingung beim Zurückverfolgen des Pfades ist!

Aufgabe 2: DTW mit weniger einschränkenden Pfaderweiterungen

In dieser Aufgabe geht es nun darum, die Auswirkung von Pfaderweiterungen zu untersuchen, die hinsichtlich der Steigung weniger einschränkend auf die Warping-Kurve wirken. Solche Pfaderweiterungen sind in Figur 2 dargestellt. Sie erlauben der Warping-Kurve eine beliebige positive Steigung.

Sie können entweder die Pfaderweiterungen von Figur 1 oder diejenigen von Figur 2 wählen, indem Sie im vorgegebenen Rahmenprogramm `ueb12_frame.m` die Variable `aufg` auf den Wert 1 bzw. 2 setzen.



Figur 2: Pfaderweiterungen und Pfadgewichte für Aufgabe 2

Überlegen Sie insbesondere, wieso sich die Pfaderweiterungen von Figur 2 schlechter für die Spracherkennung eignen. Wählen Sie auch anstelle des Signals `u12_sig_y.wav` das Signal `u12_sig_y_a.wav` bzw. das Signal `u12_sig_y_b.wav` aus und vergleichen Sie die Resultate, die Sie mit den beiden Pfaderweiterungen von Aufgabe 1 und 2 erhalten.