

Seminararbeit : TimeWarpings

Dirk Schäfer - saed@informatik.uni-marburg.de

Bioinformatikseminar im Wintersemester 2005/06 ·
Fachbereich Mathematik und Informatik · Philipps Universität Marburg

Zusammenfassung

Für den Vergleich von zwei Sequenzen bietet das Dynamic Time Warping - Verfahren ein intuitiv korrektes Maß für Ähnlichkeit. Im Gegensatz zu dem gebräuchlichen euklidischen Distanzmaß kann DTW unterschiedlich lange Sequenzen auf nicht-lineare Weise ausmessen, indem lokale Verzerrungen in Mustern berücksichtigt werden.

Stichworte: DTW, DP, Zeitanpassung, Zeitreihen

1 Einleitung

1.1 Ein Beispiel aus der Kardiologie

Als Anwendungsgegenstand für das TimeWarping bieten sich Herzschläge an, die in Form von EKG-Signalen als Zeitreihen aufgezeichnet werden können. Abbildung 1 zeigt zwei Graphen mit jeweils zwei EKG-Signalreihen aus dem MIT-BIH Datensatz[1]. Die jeweils vorderen Reihen sind die Herzschläge einer 75-jährigen, während die hinteren Reihen denjenigen einer 24-jährigen Person entsprechen. Zu der ausgewählten Zeitspanne weisen beide Reihen neben geringen Differenzen in der Frequenz und im zeitlichen Offset einen normalen Sinusrhythmus auf. Abbildung 1a) zeigt ein 1 : 1 Alignment zwischen den beiden EKG-Signalen, wie es bei der Berechnung der euklidischen Distanz genutzt wird, indem paarweise Distanzen chronologisch verrechnet werden. Die Abb. 1b) zeigt

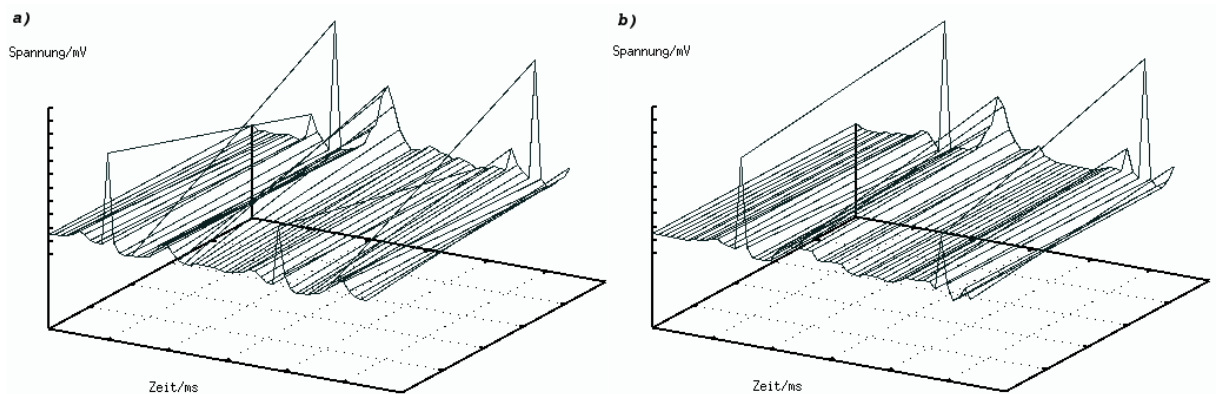


Abbildung 1: EKG-Signale von 2 unterschiedlichen Personen(vorne/hinten). Bestimmung der Ähnlichkeit mit den Methoden a) Euklid und b) TimeWarping

demgegenüber das Alignment, das durch DTW berechnet wird, welches die zeitlichen Differenzen ausgleicht. Am deutlichsten tritt der Unterschied in den Zuordnungen der QRS-Komplexe (hohen Spitzen) hervor. In der Eigenschaft, genauere Distanzen zwischen Sequenzen zu errechnen, eignet sich das DTW für Anomalie-Detektion und Clustering, z.B. im Zusammenhang mit der Langzeit-EKG-Analyse mittels PDA (Personal Digital Assistant) [2].

1.2 Hintergrund

Die dynamische Zeitanpassung (engl. **D**ynamic **T**ime **W**arping) ist zum ersten mal von Vintsyuk in Kibenetika im Jahr 1968 erwähnt worden [3]. Frühe Anwendung der DTW fand sich in der automatische Spracherkennung (ASR) in den 1970ern (Velichko and Zagoruyko 1970, Itakura 1975, Sakoe and Chiba 1978). In diesem Feld stellt sich das Problem, dass man Wörter in der schriftlichen Form zeichenweise auf eine recht einfache Weise vergleichen kann, jedoch die gesprochenen Sprache von Faktoren wie Betonung, Tonhöhe, emotionale Zustände und Sprechergeschwindigkeit beeinflusst wird. Eine Lösung für das Problem der zeitlichen Invarianz wird über Stauchungs- und Streckungs-operationen auf den Sprachsamples zugänglich. Im Zuge der Entwicklung in der ASR haben seit Mitte der 1980er Jahre die Hidden-Markov-Modelle(HMM) das DTW verdrängt, um verstärkt auch grammatikalische Aspekte zu berücksichtigen.

Eine Einordnung der DTW zu ähnlichen Techniken des Sequenzvergleiches ist in [4] dargestellt. Wenn anstatt numerischer Sequenzen für den Vergleich, Ketten von Zuständen verwendet werden, nutzt man den Viterbi-Algorithmus mit HMMs. Wenn andererseits die Sequenzen durch Symbole ersetzt werden, so arbeitet man bereits mit der Levenshtein-Distanz, die sich im Kostenmaß bezüglich der Operationen gegenüber DTW unterscheidet [5].

Die DTW-Technik wurde in den 1990ern über Berndt und Clifford [6] auch für die Data-Mining und Datenbank Community präsent und ist in jüngster Zeit im Zusammenhang mit Zeitreihen effektiv untersucht worden [7, 8, 9, 10]. Einsatz findet es praktisch überall dort, wo Informationen mit der Zeit anfallen oder eine Überführung von Daten in Zeitreihen möglich ist. Das DTW kann nach Kruskal/Liberman [5] auf drei unterschiedliche Weisen angewandt werden. Erstens ist es möglich, diejenigen Muster der Stauchung und der Streckung zu entdecken, die zwei Sequenzen miteinander verbinden. Zweitens, für die Gewinnung eines Differenzmaßes. Und drittens dazu, um ein gewichtetes Mittel von zwei Sequenzen zu bilden. Aus der Sicht des Data-Mining ist die erste Variante interessant für die Aufdeckung von Anomalien (anomaly detection) und die Varianten zwei und drei für Clusterung, Klassifikation [11] und Information Retrieval.

Der nachfolgende Text gliedert sich wie folgt : Zunächst erfolgt die Beschreibung des DTW-Algorithmus in der ursprünglichen Version, gefolgt von einer Diskussion von Varianten in Abschnitt 3 und einer Betrachtung des DTW im Zusammenhang mit der Bioinformatik im letzten Abschnitt.

2 Dynamic Time Warping

Im folgenden sei für die Erklärung der Funktionsweise das DTW im Zusammenhang mit 1-dimensionalen Zeitreihen (Folgen von Temperaturen, Winkeln, Spannungen,...) dargestellt.

Eine Zeitreihe $\vec{x} = \langle x_1 = (v_1, t_1), \dots, x_n = (v_n, t_n) \rangle$ ist ein n-Tupel von Elementen x_i , die aus einem Wert v_i und einer Zeitmarke t_j bestehen, die auch mit $t(x_i)$ referenziert

wird. Die Daten-Werte sind gewöhnlich reelle Zahlen, können jedoch auch aus qualitativen Daten bestehen und multivariat sein. Für Zeitmarken bezeichnet $\Delta = t(x_{i+1}) - t(x_i)$ die Samplingrate, wenn die Meßintervalle konstante Längen aufweisen. Unregelmäßige Intervalle sind ebenso möglich, zum Beispiel bei der Messung von RNA Expressionsdaten an ausgewählten Zeitpunkten. Oft werden Zeitreihen einfach nur als Sequenz ihrer Werte ohne Zeitmarken dargestellt.

Es sei nun S eine Zeitreihe der Länge n , die für ein neues, unbekanntes Muster, und T eine Zeitreihe der Länge m , die für ein Muster steht, welches sich in einem größeren Datenbestand befinden kann und gegen S abgeglichen werden soll.

$$S = s_1, s_2, \dots, s_i, \dots, s_n$$

$$T = t_1, t_2, \dots, t_j, \dots, t_m$$

Für beide Zeitreihen wird eine gemeinsame Zuordnung (alignment) in der Form eines sogenannten Warping-Pfades W realisiert, welcher eine Relation $W \subseteq S \times T$ darstellt. Dazu sei $W = w_1, w_2, \dots, w_k, \dots, w_K$, wobei jedes Element von w_k einem Paar (i, j) entspricht, mit dem Hintergrund, daß eine lokale Distanz $d_{S,T}(i, j)$ auf die Datenpunkte s_i aus S und t_j aus T angewandt wird. Das lokale Distanzmaß steht hierbei in Abgrenzung zur globalen DTW Distanz, welche die volle Länge der Zeitreihen berücksichtigt. Am meisten verbreitet für die lokale Distanz $d_{S,T}$ ist mit $d_{S,T}(i, j) = ||s_i - t_j||^2$ der euklidische Abstand (in quadrierter Form), aber auch der Absolutbetrag der Differenz [7, 6] und andere Distanzmaße sind je nach Anwendung möglich.

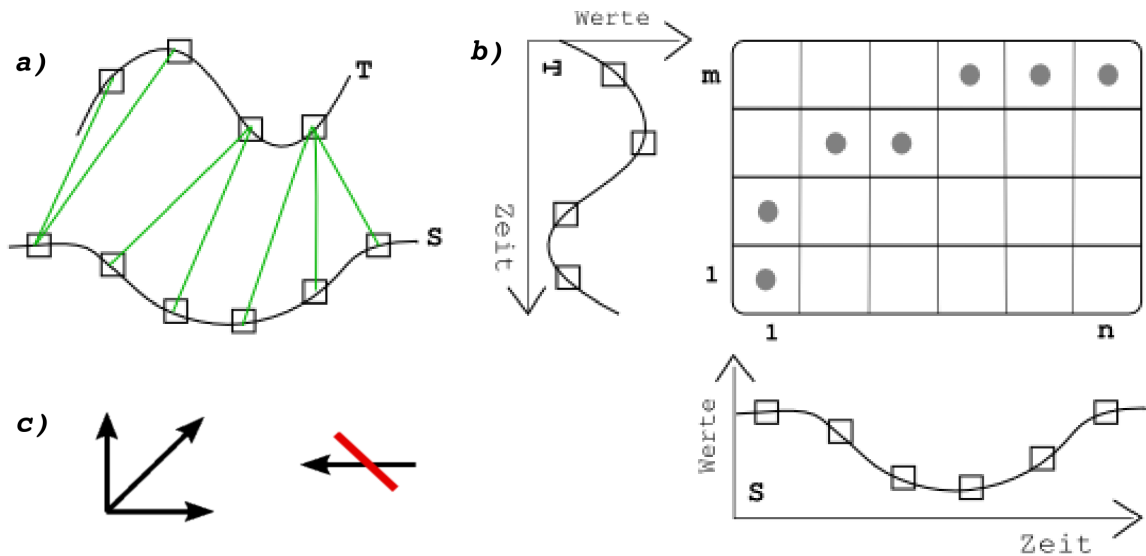


Abbildung 2: a) Zwei Sequenzen mit Alignment, b) DTW-Matrix mit Warping-Pfad und c) das innerhalb der Matrix verwendete Schrittmuster unter Vermeidung von Rücksprüngen

Über die Berechnung aller paarweisen Distanzen bildet sich eine $n \times m$ - Matrix aus, durch die mögliche Warping-Pfade führen können (siehe Abb 2b). Anhand dieser Matrix wird nun der optimalste aller Warping-Pfade gesucht, welcher für die beiden Zeitreihen das bestmögliche Alignment darstellt. Die nachfolgende dynamic time-warping Funktion DTW realisiert diesen Pfad.

$$DTW(S, T) = \min \left\{ \sqrt{\sum_{k=1}^K w_k / K} \right\} \quad (1)$$

Allen Warping-Pfaden ist gemeinsam, dass sie unten links, und oben rechts in der Matrix enden (siehe Abbildung 2b), dabei können sie über drei mögliche Schrittfolgen (horizontal, vertikal und diagonal) durch die Matrix ziehen. Horizontale Schritte bedeuten eine Kompression, und vertikale Schritte eine Streckung der Zeitreihe S, die auf der horizontalen Achse der Matrix aufgetragen wird. Bedingt durch diese Schritte variieren die Warping-Pfade in ihrer Länge, welche durch den Faktor K in der Formel (1) für Vergleiche normiert wird. Für die praktische Berechnung der DTW-Distanz wird das folgende Rekursions-Schema mit der Dynamischen Programmierung genutzt:

$$\begin{aligned}
 DTW(\varepsilon, \varepsilon) &= 0 \\
 DTW(S, \varepsilon) &= DTW(\varepsilon, T) = \infty \\
 DTW(S, T) &= d_{S,T}(First(S), First(T)) + \\
 &\min \begin{cases} DTW(S, Rest(T)) \\ DTW(Rest(S), Rest(T)) \\ DTW(Rest(S), T) \end{cases}
 \end{aligned}$$

Mit den vorangestellten Definitionen ist der Wert der DTW-Distanz stets positiv oder gleich Null, wenn ein perfektes Matching auftritt. Aufgrund möglicher Verletzung der Dreiecksungleichung ist die Distanz jedoch im allgemeinen keine Metrik.

2.1 Einschränkung an den Warping-Pfad

Über Einschränkungen möchte man zum einen die Effizienz erhöhen, und zum anderen unsinnigen, der Intuition widersprechenden Alignments begegnen. Ein Beispiel für eine solche pathologische Zuordnung, die man gerne vermeiden möchte, ist die sogenannte Singularität, bei der ein einzelner Daten-Punkt aus der Sequenz S einer größeren Menge an Daten-Punkten aus der Sequenz T zugeordnet wird [8].

Eine Verbesserung der Effizienz ist schon alleine deshalb notwendig, da sich mit der Länge der Sequenzen die Anzahl der zu berechnenden Warping-Pfade exponentiell erhöht. Daher haben sich relativ früh bezüglich des klassischen DTW gewisse Einschränkungen an den Warping-Pfad eingestellt, die im folgenden beschrieben werden:

2.1.1 Boundary Condition

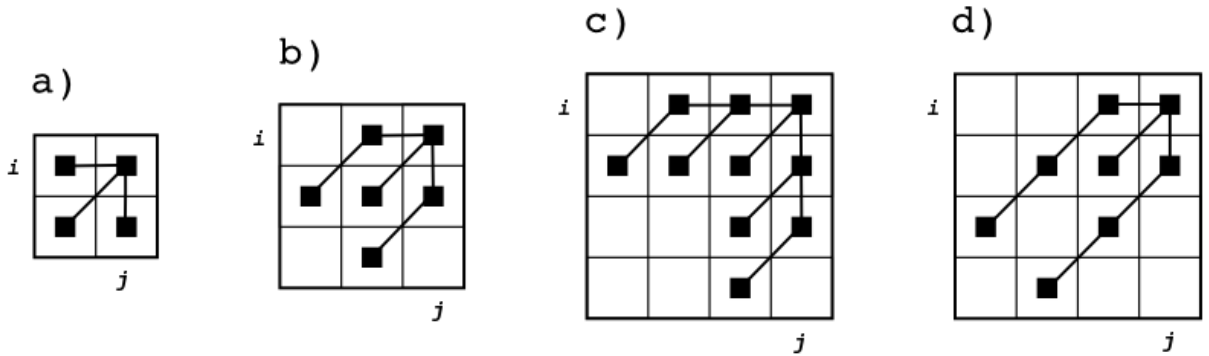
Diese Einschränkung bewirkt, dass der Warping-Pfad die jeweils ersten und beiden letzten Punkte der Sequenzen umfasst. Für die DTW-Matrix bedeutet dies, dass der Pfad unten links startet und oben rechts endet. In Formeln : $w_1 = (1, 1)$ und $w_k = (m, n)$.

2.1.2 Monotonie

Monotonie gewährleistet die Beibehaltung der relativen Ordnung der Zeitpunkte im Time-Warp. Innerhalb der Matrix kommt dies einem Verbot von Rücksprüngen gleich (siehe Abbildung 2 a.c). Konsequenterweise lassen sich damit überkreuzende Alignments verhindern und chronologische Abfolgen sichern. In Formeln : $a - a' \geq 0, b - b' \geq 0$ für $w_k = (a, b), w_{k-1} = (a', b')$.

2.1.3 Slope Constraint

Die Aufstiegsbedingung verhindert die Pathologie, dass sich zu flache oder steile Warping-Wege in der Matrix ergeben, d.h. Mappings von relativ kurzen auf recht lange Teilsequenzen. Ein Bruch p/q beschreibt, dass nach maximal q Schritten in der einen Richtung (horizontal oder vertikal), p Schritte in diagonale Richtung erfolgen müssen.



TYP	REF	REKURRENZ	SLOPE
I	a)	$\gamma(i, j) = \min \begin{cases} \gamma(i-1, j) + d(s_i, t_j) \\ \gamma(i-1, j-1) + 2d(s_i, t_j) \\ \gamma(i, j-1) + d(s_i, t_j) \end{cases}$	0
II	b)	$\gamma(i, j) = 2d(s_i, t_j) + \min \begin{cases} \gamma(i-2, j-1) + d(s_i, t_j) \\ \gamma(i-1, j-1) \\ \gamma(i-1, j-2) + d(s_i, t_j) \end{cases}$	1
V	c)	$\gamma(i, j) = \min \begin{cases} \gamma(i-3, j-1), \gamma(i-2, j-1), \gamma(i-1, j-1) \\ \gamma(i-1, j-2), \gamma(i-1, j-3) \end{cases}$	$\frac{1}{2}$
VI	d)	$\gamma(i, j) = d(s_i, t_j) + \min \begin{cases} \gamma(i-2, j-3) \\ \gamma(i-1, j-1) \\ \gamma(i-3, j-2) \end{cases}$	2

Tabelle 1: Auswahl an symmetrischen Schrittmustern

2.1.4 Continuity

Man versteht hierunter grob, dass keine 'relevanten' Zeitpunkte in den Sequenzen übersprungen werden sollen. Traditionsgemäß fasst man unter diesem Begriff alle lokalen Schrittmuster zusammen, obwohl im strengen Sinn nur der Typ I der lokalen Schrittmuster eine Stetigkeit bewirkt. Die Reglementierung der Schrittabfolgen innerhalb der DTW-Matrix wird auch unter dem Begriff 'Local Path Constraint' verstanden, deren korrekte Wahl vom Anwendungsfall abhängt (siehe Rabiner, Juang 1993).

2.1.5 Warping(/Adjustment) Window

Hierbei geht es um eine globale Einschränkung, welche primär nicht intuitive Alignments ausschließen und sekundär einen Geschwindigkeitsvorteil hervorbringen soll. Zwei der bekanntesten Vertreter für das Warping-Window sind das Sakoe-Chiba Band (siehe Abb 3a) und das Itakura-Parallelogramm. Die Breite des Fensters wird hierbei in Bezug auf die Matrixdiagonalen über den Parameter r ausgezeichnet. Wenn die Fenster eine Breite von 0% aufweisen, dann muss der Warping-Pfad diagonal verlaufen, was dem speziellen Fall des euklidischen Abstandes entspricht, also eine Linearisierung des Distanzmaßes bewirkt.

Einen weiteren Aspekt greifen Ratanamahatana und Keogh in [12] auf, so wird in vielen Publikationen aus Trägheit oder Unwissenheit einfach das Sakoe-Chiba Band und eine Fenster-Breite, sofern überhaupt angegeben, von 10% in Bezug auf die Sequenzlänge gewählt, obwohl eine Erhöhung der Accuracy für große Datenbestände bei eher noch schmalen Fenstern eintritt[12]. Feststeht, dass Fensterbreiten größer 10 % auf keinen

Fall einen positiven Zuwachs für die Exaktheit des Verfahrens bewirken können, wie in manchen Arbeiten auch schon fälschlicherweise ausgesagt wurde.

2.2 Dynamische Programmierung

Ein naiver Ansatz für DTW liegt in der Berechnung aller möglichen Berechnungswege des oben genannten Rekursions-schemas, was zu einem Aufwand von $O(3^{\max(m,n)})$ (Berechnungsbaum der Höhe $\max(m,n)$ mit jeweils 3 Kindknoten pro Stufe) führt. Eine Optimierung liegt in der Speicherung von Zwischenergebnissen in einer Tabelle, was gerade eines der Features der Dynamische Programmierung (DP) ist. Für die Berechnung des minimalen Warping-Pfades wird DP in Verbindung mit der nachfolgenden oder einer der Rekurrenz-Relationen aus Tabelle 1 verwendet, wobei $\gamma(i, j)$ die kummulative Distanz für jedes einzelne Feld der Warp-Matrix ist und $d(s_i, t_j)$ die lokale Distanzfunktion (z.B. Euklid) repräsentiert.

$$\gamma(i, j) = d(s_i, t_j) + \min[\gamma(i - 1, j), \gamma(i - 1, j - 1), \gamma(i, j - 1)] \quad (2)$$

Die diskrete Optimierungsaufgabe bei DTW ist über DP lösbar, in dem sukzessiv die Matrix der kumulierten Abstände errechnet wird, und im Nachhinein mit ihnen über Rückwärtsinduktion auf den minimalsten Warping-Pfad geschlossen werden kann. Der Vorteil liegt neben der Eleganz der Methode in der Vermeidung von lokalen Maxima. Nachteilig ist der immernoch recht hohe quadratische Rechenaufwand.

2.3 Symmetrie versus Asymmetrie

Die Form der Rekurrenz bestimmt die Symmetrie des Verfahrens. Eine Asymmetrie kann man beispielsweise herstellen, wenn man in Formel (2) nur eine der beiden kumulierten Distanzen $\gamma(i - 1, j)$ oder $\gamma(i, j - 1)$ verwendet. Es sei angemerkt, daß in der Praxis das asymmetrische Vorgehen eher selten auftritt. Ein Beispiel für eine Anwendung der Asymmetrie liegt in der Spracherkennungs-Domäne, in der man gelegentlich die bereits gespeicherten Referenzen in Bezug auf neue Sprachsamples anders gewichten möchte [5].

3 Varianten des DTW

In diesem Abschnitt geht es um die Erhöhung der Genauigkeit, um Verbesserung des Laufzeitverhaltens, um Indexing und eine Methode für Alignments in Echtzeit. Für die jeweiligen Aspekte seien im folgenden eine exemplarische Auswahl von Verfahren vorgestellt.

Eine Schwachstelle an DTW ist die quadratische Zeit- und Raum-komplexität. Sie kann nach Salvador und Chan mittels drei grundlegender Prinzipien behoben werden [10]. Erstens durch die beschriebenen Constraints, die über geschickte Einschränkung von Berechnungszellen der DTW-Matrix Kosten einsparen. Diese haben allerdings auch den Nachteil, das korrekte Alignments unterschlagen werden können, wenn extreme Zeitdifferenzen auftreten, z.B. bei Messungsbeginn durch Artefakte des Meßgerätes. Zweitens über das Mittel der Datenabstraktion, deren Idee darin besteht, DTW auf eine reduzierte Repräsentation der Daten anzuwenden und das optimale Ergebnis schrittweise zu approximieren (siehe [10]). Die größte Schwierigkeit liegt hierbei in der Rückprojektion auf die höhere Auflösungsstufe. Und Drittens über Indexing, welches darauf abzielt, über Lower-Bounding-Funktionen die teureren DTW-Berechnungen, die im Zusammenhang mit DB-Transaktionen bei Klassifikationen und Clusterungen mehrfach auftreten, einzusparen.

3.1 Daten-Abstraktion

3.1.1 Scaling up DTW : SDTW, PDTW (1999/2000)

Die Idee besteht darin, DTW auf eine höheren Abstraktionsschicht der Daten anzusetzen [7, 13]. Für die Repräsentation der Daten gibt es zum Beispiel folgende zwei Möglichkeiten. Die Zeitreihe in $N \ll n$ Fragmente, sogenannte Segmente oder Frames, zuteilen, und dann die dem Segment unterliegenden Daten linear zuapproximieren(SDTW). Die andere Möglichkeit besteht darin, für jedes Segment das arithmetische Mittel der Datenpunkte aus der vorigen Abstraktionsschicht zu verwenden(PDTW). Letztere Einteilung kursiert unter verschiedenen Namen, hier: PAA - Piecewise Aggregate Approximation. In beiden Versionen ist für die Detailstufe der Daten-Repräsentation ein Kompressionsfaktor $c = n/N$ (ratio) zuständig. Bei Whole-Matching und Subsequence-Matching-Aufgaben ist damit ein konstanter Speed-up von c^2 und c^3 zu verzeichnen. Die Schwierigkeit für den Anwender bei diesem Ansatz liegt in der Wahl des 'richtigen' Kompressionsfaktors, der wenn er zu grob gewählt wird, falsche Ergebnisse liefert und andererseits, wenn er zu fein gewählt wird, keinen nennenswerten Speed-up erzeugt.

3.1.2 Derivative DTW: DDTW (2001)

Ein anderes Problem an DTW ist, dass es bereits geringe lokale Unterschiede auf der Y-Achse zu stark über Warpings auf der Zeitachse interpretiert. Ferner ist auch dies eine der Ursachen für das Auftreten von die nicht-intuitiven Singularitäten. DDTW konzentriert sich daher verstärkt auf lokale Verzerrungen bei den Features-alignments. Unter Features werden hier bestimmte charakteristische Muster und Formen wie Spitzen, Täler, Biegungen, Plateaus, usw. verstanden. Eine Lösung [8] zum Problem der Feature-verzerrung besteht darin, die grobe Form der Features zu berücksichtigen, indem die Steigungen in den unmittelbaren Umgebungen der Datenpunkte miteinbezogen werden. Der Aufwand beläuft sich dabei auf $O(nm) + \text{eine Konstante}$ für die Berechnung der Ableitungen. Keogh betont mit der Methode die Einsparung der Offset-Translation - eine von mehreren globalen Zeitreihen-Transformationen (offset-translation, amplitude scaling, linear trends, exponential smoothing).

3.1.3 Iterative Deepening : IDDTW (2002)

Diese Variante läßt sich gut für das Whole Matching einsetzen, bei dem ein Anfragemuster mit tendenziell vielen Mustern aus einer DB verglichen werden muss. Die Idee [14] ist die folgende: In einem ersten, frühen Schritt erstellt man für jede Approximationsstufe ein Modell, welches die Verteilung der Approximationsfehler von Distanzen beschreibt. Später können für zwei beliebige Zeitreihen die DTW-Approximationen auf immer feineren Detail-Stufen durchgeführt werden, wobei die vorhandenen Verteilungsmodelle dazu genutzt werden können, die schlechten Matches rauszufiltern. Das Verfahren sieht desweiteren einen vom Anwender gewählten Tolleranzfaktor vor, der bereits bei kleinen Eingeständnissen einen großen Speed-up bewirkt.

3.1.4 FastDTW (2004)

Ziel hierbei war eine Verbesserung des DTW auf lineare Laufzeit und Raum-komplexität. Die Idee [10] liegt in der sukzessiven Übertragung des Warping-Pfades einer niederen auf die nächst feinere Detailstufe. Dabei wird auf der niedrigsten Detailstufe das Standard-DTW und in nachfolgenden Schritten eine kostengünstige Variante des DTW angewandt. Dem Verfahren liegen die folgenden drei Operationen zu Grunde: Coarsening, Projection

und Refinement. Coarsening bezeichnet den Prozess der Abbildung einer Zeitreihe auf eine Zeitreihe mit weniger Datenpunkten. Ein Beispiel für eine solche komprimierte Repräsentation der Daten ist das bereits erwähnte PAA-Format. Die zweite Operation ist die Projektion des Warping-Pfades der aktuellen Stufe auf die nächste Auflösungsstufe. Der projizierte Pfad dient dabei als Heuristik für das Refinement. Wegen der Heuristik als solcher kann jedoch die Korrektheit der Projektion nicht zu 100% garantiert werden. Die dritte Operation ist schließlich das Refinement, welche die Anpassung des noch groben Warping-Pfades auf die nächste Stufe realisiert. Dazu berechnet ein spezielle Variante des DTW anstelle der vollen Matrix lediglich die Kosten der unmittelbaren Umgebung. Der Anwender hat dabei die Möglichkeit über den Parameter r die Größe dieser Umgebung, und damit implizit den Trade-Off zwischen Genauigkeit und Geschwindigkeit festzulegen. Empirische Experimente in [10] suggerieren, dass Zeitreihen erst ab einem Umfang von ca. 1000 Datenpunkten für FastDTW interessant werden, da dann erst die Kosten des Standard-DTW stärker ins Gewicht fallen. Ferner ist eine $O(N)$ - Implementierung nicht völlig möglich, da es bei großen Zeitreihen (80000) zu Zeitverzögerung bei dem Zugriff auf Daten kommen kann, die sich außerhalb des Hauptspeichers befinden.

3.2 Indexing

In vielen DTW-Anwendung taucht immer wieder die Aufgabe auf, zu einem neuen Muster die k -ähnlichsten (k -NN) Muster aus einem großen Pool von Referenzen zu finden. Durch einen sequentiellen Scan durch den Datenbestand wächst der Aufwand auf $O(nmx)$ an, wenn x die Anzahl der Muster in einer Daten-Bank bezeichnet [6]. Dieser Aufwand ist zu minimieren, indem über eine möglichst einfache Abschätzung und eventuell unter Berücksichtigung von einem, von dem Anwender gewählten Schwellenwert, eine große Anzahl von unähnlichen Objekten aussortiert werden (pruning). Dabei sollten zu unrecht verworfene Objekte (false dismissals) vermieden und die Anzahl von nicht-erwünschten Objekten (false alarms) verringert werden.

3.2.1 Lower-Bounding-Funktion

Lower-Bounding-Funktionen realisieren kostengünstige Abschätzungen an das DTW. Nach [9] sollten sie schnell zu berechnen, d.h. möglichst linear in der Länge der Eingabe, und so nah wie nur möglich an die wahren DTW-Funktionswerte heranreichen.

Unter Verwendung eines der oben beschriebenen Warping-Windows, lässt sich eine Umhüllung der ursprünglichen Sequenz erzielen. In Abbildung 3 b) ist dies anhand des Sakoe-Chiba-Bands ausgeführt. Die obere und untere Sequenz der Hülle entsprechen den Werten an den Rändern des Warping-Windows. In der Grafik rechts oben in Abb. 3 ist die ursprüngliche Sequenz kräftig und die beiden einhüllenden Sequenzen $O(ben)$ und $U(nten)$ dünn eingezeichnet. Diese Hülle kann nun für die Definition der Lower-Bounding-Funktion, z.B. LB_Keogh [9], genutzt werden, um die Suche nach ähnlichen Sequenzen in der Datenbank zu beschleunigen.

3.3 Sonstige

3.3.1 Einsparung von Raum

Eine triviale Möglichkeit die Raumkomplexität von quadratisch auf lineare Größe zu reduzieren, besteht nach [10] darin, gar nicht erst die volle Distanzmatrix zu berechnen, sondern jeweils nur die aktuelle Spalte und wenige Vorgänger-Spalten im Speicher zu halten und den Rest zu verwerfen. Nachteil bei dieser Greedy-Methode ist, dass sie nur

für die Berechnung der Sequenzen-Distanz brauchbar sein kann, nicht jedoch für die Entdeckung von Features oder für das Merging von Sequenzen, da der Warping-Pfad aufgrund der verworfenen Matrixspalten nicht rekonstruiert werden kann.

BioInf-Link: Diese Vorgehensweise entspricht einem Teil der Hirschberg-Variante des Needleman-Wunsch Algorithmus für die Auffindung von Sequenzalignments. Auch dort liegt die Reduktion des Platzbedarfs von quadratisch auf linear bei der Berechnung der Distanz (Levenshtein-/Edit-) in der Verwerfung von Matrixspalten. Darüberhinaus ist bei Hirschberg die Rekonstruktion eines optimalen Alignments (Pfad durch die Matrix) über eine geschickte Divide-Conquer-Strategie realisiert.

3.3.2 Uniform Time Warping

UTW ist eine Generalisierung von TimeScaling, weil bei TS im Gegensatz zu UTW die eine Sequenz in der Länge ein Vielfaches der anderen sein muss. Andererseits ist es ein Spezialfall von DTW, weil die Warping-Pfade ausschließlich diagonal durch die Kostenmatrix führen dürfen. Dies geschieht über Upsampling beider Sequenzen auf $m * n$ oder das $kgV(s,t)$ [15].

3.3.3 Uniform Scaling

Uniform Scaling (US) gehört zu einer von mehreren Zeitreihentransformationen ¹, die in der Landmark-Indexierungstechnik [16] Verwendung findet. Das US ist als solches keine eigene, sondern lediglich in Verbindung mit SWM(siehe unten) eine Variante des DTW. Die Stärke des DTW liegt im Ausgleich lokaler Differenzen, aber es bringt schlechte Alignments bei globalen Unterschieden, wenn zum Beispiel Sequenz A wesentlich kürzer ist als Sequenz B. Dieser Schwachstelle kommt US entgegen, indem die Kürzere von beiden hochskaliert wird, und danach ein gebräuchliches Distanzmaß wie Euklid angewandt wird.

3.3.4 cDTW

Bei dem **constrained-DTW** wird gegenüber DTW nicht die komplette Kostenmatrix berechnet, sondern nur die Felder, die innerhalb eines Warping-Window (z.B. Sakoe-Chiba-Band) liegen. Daher wird dem cDTW als Parameter neben den Sequenzen noch die Fensterbreite r übergeben.

¹Shifting, Amplitude Scaling, Bi-Scaling, TimeWarping, Non-Uniform Amplitude Scaling

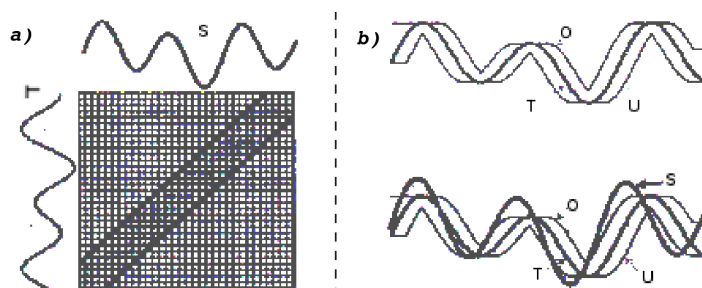


Abbildung 3: Links das Warpingwindow, rechts die beiden daraus abgeleiteten, welche die ursprüngliche Sequenz einhüllt.

3.3.5 DTW with Uniform Scaling: SWM (Scaled and Warped Matching, 2005)

Die Notwendigkeit für eine Kombination von DTW und US wird in [17] anhand von Beispielen aus dem Bereich des Motion-Capture und Animation begründet. Die Idee ist, alle möglichen Scalings im Verbund mit lokalen DTW's auszutesten und das optimalste Alignment anzunehmen.

3.4 TimeWarpings on the fly

Aus dem Audiobereich kommt der folgende Online-Algorithmus [18], bei dem eine noch unbekannte Zeitreihe mit einer bereits bekannten Zeitreihe in Echtzeit verglichen wird (z.B. für die Visualisierung von Live-Performances). Aufgrund der Unvollständigkeit der neuen Zeitreihe zum Zeitpunkt t muss die Boundary Condition geschätzt werden. Ferner können aufgrund der fehlenden Matrix-Diagonalen die globalen Constraints, z.B. das Sakoe-Chiba-Band oder Itakura-Parallelogramm, nicht auf direktem Weg realisiert werden. Die Idee ist hier schrittweise während dem Aufbau der Distanzen-Matrix zugleich den Warming-Pfad zu erstellen. Dies ist unter der Verwendung von zwei Hilfsmitteln möglich: Realisierung der Fensterbreite über den Toleranzparameter c für den Suchraum. Und ein weiterer Parameter, der die Steigung s des Pfades regelt (MaxRunCount-Parameter). Nachteilig sind die bei dem Vorgehen auftretenden Diskontinuitäten, für die der Autor eine 'Glättung' des Warming-Pfades über zwei Ansätze vorschlägt, die jedoch Latenzzeiten verursachen.

4 DTW in der Bioinformatik

Das Verhältnis von DTW gegenüber der Euklid-Distanz bei numerischen Daten entspricht in etwa dem der Levensthein-Distanz zur Hammond-Distanz bei symbolischen Daten (z.B. über ein Alphabet $\Sigma = \{A, C, G, T, -\}$), in dem Sinne, dass lokale Differenzen durch wenige, grundlegende Operationen ausgeglichen werden. Es existieren desweiteren Unterschiede in der genauen Berechnung und Darstellung zwischen Time-Warpings und den konventionellen String-Alignment-Methoden in der Bioinformatik[5]. So werden bei den Time-Warpings sogenannte Linking-, und bei den konventionellen String-Alignment-Methoden Trace-Diagramme zur Visualisierung eingesetzt. Ferner widerspiegeln sich bei den String-Alignments die Einfügungs- und Löschungskosten in den rekursiven Berechnungsschemata wieder. Demgegenüber stehen bei den Time-Warpings die sogenannten Substitutionskosten, die noch mit Time-Weights multipliziert werden.

Auch motiviert in [19] die Anwendung von DTW an RNA-Expressions-daten für biologische Prozesse, die sich über einen gewissen Zeitrahmen erstrecken. Es können Instanzen eines einzelnen biologischen Vorgangs je nach Organismus oder Zustand (z.B. Krankheit) auf unterschiedlichen Zeitraten ablaufen. Ein Problem dabei ist, dass das ursprüngliche DTW nicht direkt angewandt werden kann, da die Datenreihen aufgrund des hohen Aufwandes für die Extraktion der RNA-Expressions-Daten nur aus unregelmäßigen und seltenen Samples bestehen. Eine Lösung besteht in der (erstmaligen) Anwendung des Interpolationsalgorithmus von Kruskal und Liberman [5, Sec.8].

Literatur

- [1] A. L. Goldberger *et al.*, PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals, 2000 (June 13), circulation Electronic Pages: <http://circ.ahajournals.org/cgi/content/full/101/23/e215>.
- [2] A. I. Jimena Rodríguez, Alfredo Goñi, IEEE Transactions on Information Technology in Biomedicine **9**, 23 (2005).
- [3] T. Vintsyuk, Speech Discrimination by Dynamic Programming, 1968.
- [4] P. Somervuo and T. Kohonen, Self-Organizing Maps and Learning Vector Quantization for Feature Sequences, 2004.
- [5] J. B. Kruskal and M. Liberman, *The Symmetric Time Warping Problem : From Continuous to Discrete* (CSLI, Stanford, 1983), Vol. Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison, pp. 125–161.
- [6] D. J. Berndt and J. Clifford, in *Advances in Knowledge Discovery and Data Mining*, edited by U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (AAAI Press, Menlo Park, CA, 1996), pp. 229–248.
- [7] E. Keogh and M. Pazzani, Scaling up Dynamic Time Warping to Massive Datasets, 1999.
- [8] E. Keogh and M. Pazzani, Derivative Dynamic Time Warping, 2001.
- [9] E. Keogh, Exact indexing of dynamic time warping, 2002.
- [10] S. Salvador and P. Chan, FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space, 2004.
- [11] M. L. Melvyn J. Hunt and P. Mermelstein, Use of Dynamic Programming on a Syllable-Based Continuous Speech Recognition System, 1983.
- [12] C. A. Ratanamahatana and E. Keogh, Everything you know about Dynamic Time Warping is Wrong, 2004.
- [13] E. J. Keogh and M. J. Pazzani, Scaling up dynamic time warping for datamining applications, 2000.
- [14] S. Chu, E. J. Keogh, D. Hart, and M. J. Pazzani, Iterative Deepening Dynamic Time Warping for Time Series., 2002.
- [15] Y. Zhu and D. Shasha, Warping indexes with envelope transforms for query by humming, 2003.
- [16] M. Hetland, A Survey of Recent Methods for Efficient Retrieval of Similar Time Sequences.
- [17] A. W.-C. Fu, E. J. Keogh, L. Y. H. Lau, and C. A. Ratanamahatana, Scaling and Time Warping in Time Series Querying., 2005.
- [18] S. Dixon, Live Tracking Of Musical Performances Using On-Line Time Warping, 2005.
- [19] J. Aach and G. M. Church, Aligning gene expression time series with time warping algorithms, 2001.