Merkmalsextraktion aus Audiodaten Evolutionäre Aufzucht von Methodenbäumen

Ingo Mierswa · Katharina Morik

Musikstücke können als Zeitreihen variabler Länge aufgefasst werden. Im Falle der Musikdaten ist die typische Lernaufgabe die Klassifikation bzw. die Suche nach ähnlichen Reihen.

Gute Ergebnisse bei der Klassifikation von Audiodaten durch maschinelle Lernverfahren setzen die Extraktion aussagekräftiger Merkmale voraus. Die Suche nach der besten Menge extrahierter Merkmale ist jedoch äußerst komplex und muss für jede neue Klassifikationsaufgabe erneut durchgeführt werden.

In diesem Artikel wird der Rahmen für eine vollständig automatisierte Suche nach den besten Extraktionsmethoden vorgestellt. Methoden der statistischen Zeitreihenanalyse werden systematisiert und mit Hilfe genetischer Programmierung zu Methodenbäumen kombiniert. Die Performanz des Lernverfahrens auf der gegebenen Repräsentation weist den Weg durch den Raum aller Methodenbäume. Dabei muss zwischen der Vollständigkeit der Methoden und der Effizienz der Berechnungen abgewogen werden. Es stellt sich heraus, dass die Extraktion von Merkmalen mit Hilfe von dynamischen Methodenbäumen in polynomieller Zeit möglich ist.

1 Einleitung

Die Verbreitung von Musik in digitaler Form erhöht den Bedarf von Verwaltung und Indexierung dieser Daten. Ansätze für die automatische Indexierung [10] und die Anfrage nach Stücken (z. B. *query by humming*) [3]) entstammen dem Bereich des *Music Information Retrieval*. Maschinelles Lernen hat sich für die Klassifikation von Texten und das Dokumentretrieval hervorragend bewährt [6]. Ein ähnlicher Nutzen ist für die Klassifikation von Audiodaten sowie für personalisierte Benutzeranfragen zu erwarten. Hierbei stellt sich jedoch die Forderung nach Skalierbarkeit der Verfahren. Genau wie Dokumentenkollektionen können auch Musikdatenbanken mehrere Millionen Aufnahmen enthalten. Hinzu kommt, dass für eine *sampling rate* von 44 100 Hz ein Musikstück der Länge 3 Minuten insgesamt ca. $8 \cdot 10^6$ Werte enthält.

Darüber hinaus werden heutigen Ansätzen zur Zeitreihenindexierung und -ähnlichkeitsmaßen feste Zeitskalen zu Grunde gelegt [8]. Im Allgemeinen wird die Ähnlichkeit in Hinblick auf die Form und den Verlauf der Reihen ermittelt [7, 19]. Musikstücke variieren jedoch in ihrer Länge, und die *i*-ten Elongationen der Stücke weisen untereinander keine Korrelation auf. Die entscheidenden Aspekte für die Klassifikation sind daher nicht in dem konkreten Verlauf der Kurven zu suchen, sondern müssen aus den Originaldaten extrahiert werden. Die Merkmalsextraktion aus Audiodaten ist daher zu einem aktuellen Forschungsthema geworden [4, 11, 17, 20].

Viele verschiedene Extraktionsmethoden haben ihre Güte für unterschiedliche Lernaufgaben und Datensätze bewiesen. Das Problem ist nun die Suche nach dem besten Merkmalssatz für eine neue Lernaufgabe oder für einen neuen Datensatz. Ein einheitlicher Rahmen für Extraktionsmethoden fehlt bisher, so dass eine strukturierte Suche oder

DOI 10.1007/s00287-005-0015-2

Ingo Mierswa · Katharina Morik
Universität Dortmund,
Fachbereich Informatik,
Lehrstuhl für Künstliche Intelligenz
E-Mail: mierswa@ls8.cs.uni-dortmund.de

auch ein strukturierter Vergleich der Methoden kaum möglich ist. Hinzu kommt, dass jede neue Lernaufgabe einen neuen Merkmalssatz verlangt. Es ist unwahrscheinlich, dass ein Merkmalssatz, der eine hervorragende Trennung von Klassik und Pop ermöglicht, sich in gleicher Weise für die Klassifikation zwischen Techno und Hip Hop eignet. Dieses Problem wird durch eine Klassifikation nach Benutzerpräferenzen noch verschärft.

In diesem Artikel wird ein Rahmen für automatisierte Merkmalsextraktion vorgestellt, der zur Lösung der dargestellten Probleme beiträgt. In Abschnitt 2 werden elementare Basismethoden der statistischen Zeitreihenanalyse systematisch gruppiert. Diese Bibliothek von Basismethoden erlaubt uns, die Merkmalsextraktion als Sequenz von Datentransformationen zu betrachten. An deren Ende steht die Ausgabe der benötigten Merkmale. Dies führt zu der algorithmischen Struktur der Methodenbäume zur Extraktion von Merkmalen aus Wertereihen. Damit werden die bereits bekannten Merkmale aus Audiodaten abgedeckt, und einige neue Merkmale können mit ihrer Hilfe extrahiert werden. Gesucht ist also eine optimale Kombination der elementaren Basismethoden für eine gegebene Klassifikationsaufgabe. Diese Suche muss für jede neue Lernaufgabe und jeden neuen Datensatz erneut durchgeführt werden. Anstatt von Hand die möglichen Kombinationen durchzuprobieren, nutzen wir nun die Strukturierung der Methoden aus und suchen die optimale Extraktion mit Hilfe von genetischer Programmierung (siehe Abschnitt 3).

Die Suche durch den Raum aller denkbaren Methodenbäume ist geleitet durch eine Fitnessfunktion. In unserem Fall wird die Performanz eines eingebetteten Klassifikationslerners geschätzt. Die Fitness steigt monoton mit der Performanz des Lerners auf der gegebenen Repräsentation des Suchpunkts (Individuums). Das Ergebnis ist ein Methodenbaum, der angewendet auf die gegebenen Audiodaten die Performanz der aktuellen Lernaufgabe maximiert. Abb. 1 zeigt den kompletten Prozess der Suche nach Methodenbäumen. Er enthält zwei Lernschritte: zum einen das Lernen der besten Repräsentation, das heißt des besten Merkmalssatzes und zum anderen das Lernen der eigentlichen Klassifikationsaufgabe. Letzteres wurde hier mit der *support vector machine* mySVM [14] durchgeführt. Diese ist ebenfalls Teil der Fitnessbewertung während des Repräsentationslernens. Die Fitness braucht jedoch nur auf einen Teil der Daten abgeschätzt werden, wodurch weitere Rechenzeit eingespart wird. Weitere Details finden sich im Abschnitt Automatische Konstruktion von Methodenbäumen. Der hier beschriebene Ansatz wurde für mehrere Klassifikationen nach Genre sowie für Benutzerpräferenzen evaluiert (Abschnitt Klassifikation mit Hilfe gelernter Methodenbäume).

2 Methoden zur Merkmalsextraktion aus Audiodaten

Audiodaten sind Zeitreihen, bei denen die x-Achse den Verlauf der Zeit und die y-Achse die augenblickliche Auslenkung (Elongation) beschreibt. Im Folgenden generalisieren wir diese Reihe zu einer *Wertereihe*:

Definition 1. *Eine WERTEREIHE ist eine Abbildung* $x : \mathbb{N} \to \mathbb{R} \times \mathbb{C}^m$, *wobei wir* x_n *schreiben anstatt* $x(n)$ *und* x *für eine Reihe der Länge* n.

Jedes Element x_i der Reihe beinhaltet zwei Komponenten. Die erste ist die Indexkomponente, welche die Position auf einem Zahlenstrahl anzeigt (z. B. Zeit oder Frequenz). Die zweite Komponente ist ein m-dimensionaler Vektor von (komplexen) Werten. Durch die Indexdimension ist eine Ordnung innerhalb dieser Dimension nicht länger nötig, wodurch auch Transformationen in für Audiodaten eher ungewöhnliche Räume (wie z. B. den Phasenraum) ermöglicht wird.

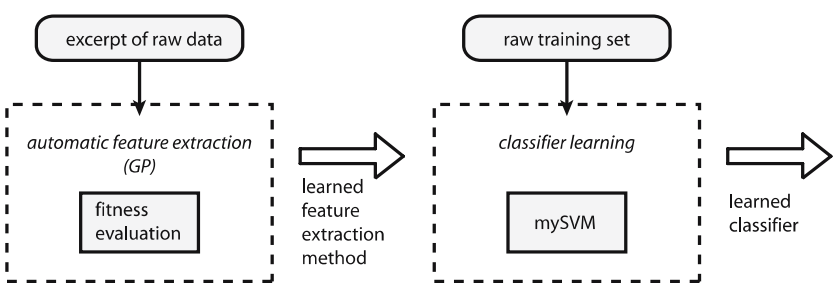


Abb. 1 Übersicht über den Prozess der automatisierten Merkmalsextraktion aus Audiodaten

2.1 *Systematisierung statistischer Basismethoden*

Eine Systematisierung der Methoden der Zeitreihenanalyse muss mächtig genug sein, um alle bekannten Methoden der Merkmalsextraktion abdecken zu können. Andererseits muss die Systematik präzise genug sein, um durch die Strukturierung eine automatisierte Suche nach den besten Kombinationen zu erlauben. Jede Methode arbeitet auf einer gegebenen Wertereihe und produziert dabei ein Ergebnis. Es stellt sich heraus, dass die Art dieses Ergebnisses ein gutes Kriterium für die Einteilung der Methoden darstellt. Wir unterscheiden:

Definition 2. *Alle Methoden, welche erneut eine Reihe als Ergebnis produzieren, d. h. eine Abbildung $t : F \rightarrow F'$ für Vektorräume F und F' von Wertereihen x, heißen TRANSFORMATIONEN.*

Definition 3. *Alle Methoden, welche einzelne Werte ohne Ordnung produzieren, d. h. eine Abbildung $f : F \rightarrow \mathbb{R}^m$ von einem Vektorraum F auf die reellen Zahlen, heißen FUNKTIONALE.*

Die Ergebnisse von Funktionalen werden schließlich als Merkmale eingesetzt. Vorher können jedoch noch beliebige Ketten von Transformationen durchgeführt werden. Transformationen, welche lediglich die Reihe ändern ohne direkt Merkmale zu produzieren, können wiederum in verschiedene Gruppen eingeteilt werden. *Basistransformationen* bilden von einem Vektorraum F in einen anderen Vektorraum F' ab (z. B. Fouriertransformationen). Durch die Verwendung einer anderen Basis werden oftmals Eigenschaften betont, die vorher nicht deutlich zu erkennen waren. So kann durch die Berechnung der Autokorrelation mit vorgegebenen Phasenverschiebungen das Tempo eines Stückes bestimmt werden (Abb. 2). Ein anderes und für Audiodaten weitestgehend unbekanntes Beispiel einer Basistransformation ist die Rekonstruktion des Zustandsraums [16], auch bekannt als Transformation in den Phasenraum. Die Winkel im Zustandsraum geben Aufschluss über die Perkussivität eines Stückes [12]. Abb. 3 zeigt den Phasenraum eines typischen klassischen Stückes im Vergleich zu einem Popstück. Der durchschnittliche Winkel und seine Varianz sind offensichtlich gute Merkmale für die Klassifikationsaufgabe Pop gegen Klassik. *Filter* hingegen ändern nicht den Raum selber, sondern nur die Position des Elementes (Differenzenfilter oder Glättung). Eine besondere Rolle spielen *Auszeichnungen in Wertereihen*. Diese versuchen, Intervalle in verschiedenen Dimensionen der Reihe zu finden,

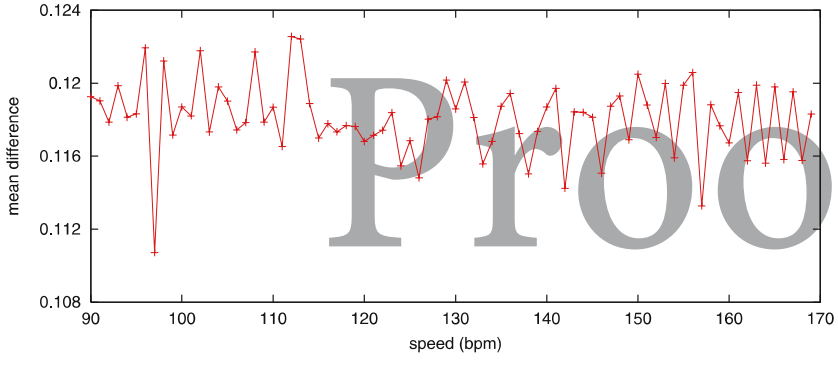
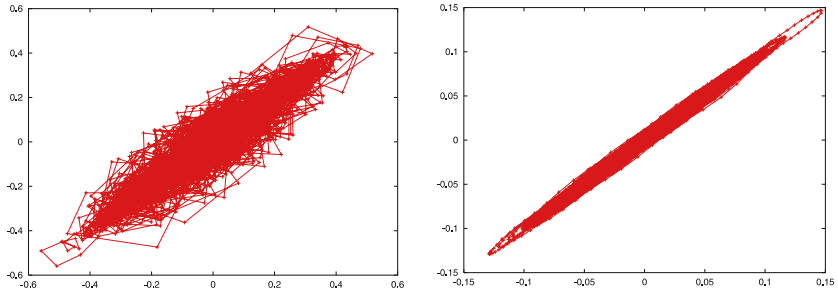


Abb. 2 Autokorrelation für Phasenverschiebungen möglicher Geschwindigkeiten. Das Minimum markiert das tatsächliche Tempo von 97 beats per minute (bpm)



(a) Pop Musik (b) Klassik

Abb. 3 Repräsentation im Phasenraum eines Populärstücks (links) und eines klassischen Stücks (rechts)

um damit die Extraktion zu beschleunigen bzw. sie nur in interessanten Regionen durchzuführen. Nachfolgende Transformationen können also von diesen Auszeichnungen profitieren [13].

2.2 *Fensterung erweitert den Methodenraum*

Um die bereits entwickelten Methoden der Wertereihenanalyse [1, 15] vollständig abdecken zu können, benötigt eine spezielle Transformation eine besondere Behandlung. Mit Hilfe einer *Fensterung* kann eine Vielzahl bekannter Merkmale und Transformationen gebildet werden:

Definition 4. *Sei eine Wertereihe x der Länge n gegeben. Eine Transformation heißt FENSTERUNG, wenn ein Fenster der Größe w mit Schrittweite s über die Reihe geschoben wird und für jedes Fenster das Funktional f berechnet wird:*

$$y_j = f(Axi(j \cdot s + 1 j \cdot s + w).$$

Die Werte y_j bilden erneut eine Reihe Ayj$0\lfloor (n-w)/s \rfloor$.

Durch die Verwendung aller denkbaren Funktionale f kann diese Definition bereits eine große Anzahl bereits bekannter Transformationen „simulieren". Ist f z. B. definiert als die Bildung des Durchschnitts der gegebenen Werte, so entspricht die Anwendung der Fensterung dem bekannten gleitenden Durchschnitt. Wir gehen jedoch noch einen Schritt weiter und erlauben vor Anwendung des Funktionals f noch eine beliebige Anzahl von zusätzlichen Transformationen. Diese Erweiterung der Fensterung nennen wir *verallgemeinerte Fensterung*. Die Forderung nach einem Funktional f für jedes Fenster verhindert ein Explodieren des Speicherbedarfs, was angesichts der großen Datenmengen dringend erforderlich ist. Es kann gezeigt werden, dass die Laufzeit der verallgemeinerten Fensterung von dem Grad der Fensterüberlappung abhängt. Im Vergleich zu der Laufzeit der auf die einzelnen Fenster angewendeten Transformationen und Funktionale kann sich die Gesamtlaufzeit durch Anwendung einer Fensterung mit realistischen Parametern für w und s jedoch nur verringern [13].

2.3 *Methodenbäume zur Merkmalsextraktion*

Die extrahierten Merkmale sind das Ergebnis einer Kette von Transformationen mit einem Endfunktional. Eine Fensterung ist ebenfalls eine Transformation, welche jedoch iterativ andere Methoden auf Teilen der Reihe anwendet. Eine mögliche Modellierung für diesen Zusammenhang ist die Betrachtung der angewendeten Methoden als Kinder der Fensterung, was zur algorithmischen Struktur der Methodenbäume zur Merkmalsextraktion führt.

Abb. 4 zeigt ein Beispiel eines solchen Methodenbaums. Dieser wird gemäß einer Tiefensuche durchlaufen, wobei die Kinder einer Fensterung im Gegensatz zur normalen Tiefensuche mehrfach angewendet werden. Gestrichelte Linien zeigen die Eltern-Kind-Relationen, durchgezogene Linien markieren den Datenfluss. Das letzte Kind des Wurzelknotens ist das Durchschnittsfunktional, welches die verlangten Merkmale „Durchschnitt und Varianz der maximalen Frequenz im Verlauf der Zeit" liefert.

In gleicher Weise können auch andere Merkmale definiert werden, welche lediglich für Audiodaten einen Sinn ergeben. Der *spectral crest factor* kann als einfache arithmetische Kombination des geometrischen Mittels und des Maximums eines Frequenzspektrums berechnet werden [5]. Die sogenannten *mel-frequency cepstral coefficients* ergeben sich ebenfalls mit Hilfe einer verallgemeinerten Fensterung. Hierzu wird zunächst das Frequenzspektrum jedes Fensters berechnet und dann eine psychoakustische Filterung gefolgt von der inversen Fouriertransformation durchgeführt. Abb. 5 zeigt, wie die Basismethoden kombiniert werden, um diese Merkmale zu erhalten.

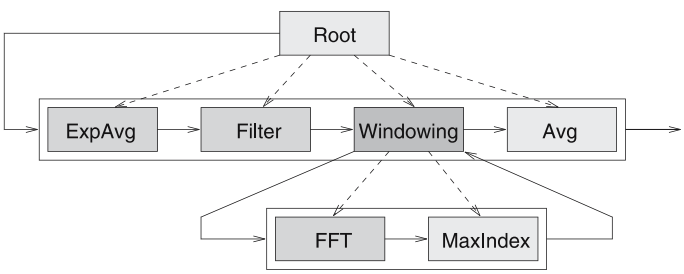


Abb. 4 Ein Methodenbaum, welcher die Merkmale „Durchschnitt und Varianz der maximalen Frequenz im Verlauf der Zeit" liefert

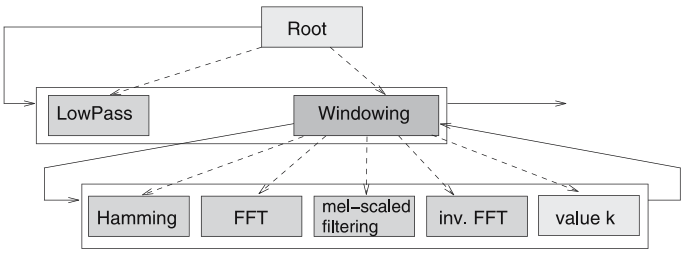


Abb. 5 Konstruktion der MFCC mit Hilfe von Methodenbäumen

2.4 *Dynamische Fensterung in Methodenbäumen*

Jeder Methodenbaum produziert ein oder mehrere Merkmale. Die Baumstruktur entsteht durch die Schachtelung der Fensterungen. Es ist daher unmöglich, zwei Fensterungen mit den gleichen Fensterweiten w ineinanderzuschachteln. Die Kindfensterung wäre auf Grund mangelnder Werte nicht mehr in der Lage, eine weitere Fensterung durchzuführen und eine neue Reihe zu produzieren.

Um eine effiziente Laufzeit der Fensterungen zu gewährleisten, sollte die Überlappung fest gewählt werden. Die Größe der Fenster muss bei fester Schrittweite dann dynamisch an die Tiefe im Methodenbaum angepasst werden (*dynamische Fensterung*). Sie ist $w = n/d$ für ein gegebenes d. Ein Methodenbaum hat dann eine maximale Tiefe von $\log_d n - 1$. Es gilt für den gesamten Methodenbaum – genau wie für die verallgemeinerte Fensterung –, dass die Laufzeit der Merkmalsextraktion niemals exponentiell wird, wenn die Laufzeit der Basismethoden nicht exponentiell war [13].

3 Automatische Konstruktion von Methodenbäumen

Elementare Methoden der statistischen Zeitreihenanalyse können zu Methodenbäumen kombiniert und zur Extraktion von Merkmalen verwendet werden. Die Forderung nach dem Funktional einer Fensterung stellt dabei sicher, dass der Speicherverbrauch nicht exorbitant ansteigt. Darüber hinaus konnte nachgewiesen werden, dass die Merkmalsextraktion mit Hilfe von Methodenbäumen effizient durchgeführt werden kann.

Es ist jedoch äußerst mühsam, solche Kombinationen von Hand zu erstellen und zu validieren. Da es sich bei Methodenbäumen um Bäume variabler Größe handelt, wenden wir *genetische Programmierung* [9] an, um den optimalen Methodenbaum für eine gegebene Anwendung zu züchten. Abb. 6 zeigt diese Suche nach der besten Repräsentation. Das Bild beschreibt den linken Teil von Abb. 1 im Detail. Der Suchraum wird Universum der Methodenbäume genannt; eine Population ist eine Menge von Methodenbäumen. Die Navigation durch das Universum aller Bäume ist ein Kreislauf aus Selektion der besten Individuen, Veränderung der Bäume durch Mutation und Kreuzungen, Anwendung der neuen Methodenbäume auf die Rohdaten und Evaluierung der Fitness durch Abschätzung der Güte eines Lernverfahrens.

Die lokalen Suchoperationen sind Mutationen und Kreuzungen. Mutationen fügen zufällig neue Methoden hinzu, löschen zufällig ausgewählte Methoden oder ersetzen eine Methode durch eine andere der gleichen Methodenklasse. Bei allen Änderungen wird gewährleistet, dass die strukturellen Bedingungen erfüllt bleiben.

Kreuzungen tauschen einen zufällig ausgewählten Teilbaum eines Elters mit einem Teilbaum eines anderen Methodenbaums. Auch hierbei werden alle Bedingungen berücksichtigt. Im Falle der Kreuzung

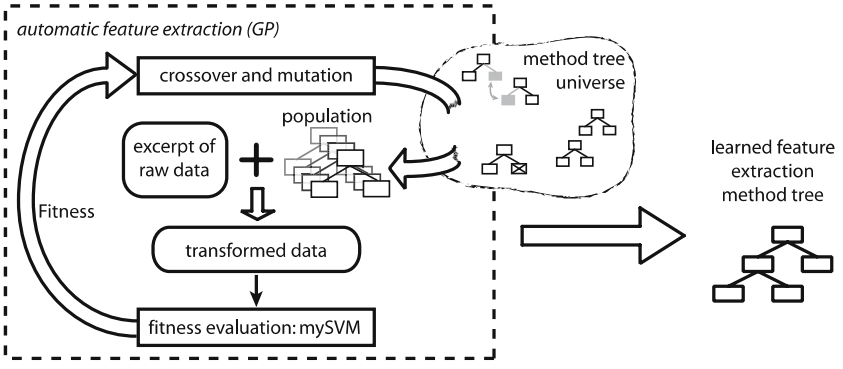


Abb. 6 Automatische Merkmalsextraktion mit Hilfe von genetischer Programmierung

ist dies gewährleistet, falls die Wurzeln der beiden Teilbäume vom gleichen Typ sind.

3.1 *Fitnessfunktion und Selektion*

Zur Selektion der besten Individuen wird die so genannte *Turnierselektion* eingesetzt. Hierbei werden t Methodenbäume zufällig aus der aktuellen Population ausgewählt und der Gewinner dieses „Turniers" der nächsten Population hinzugefügt. Dieses wird solange iteriert, bis die gewünschte Anzahl der Individuen erreicht ist.

Da Methodenbäume letztendlich einer verbesserten Klassifikationsgüte dienen sollen, wird diese auch als Kriterium zur Fitnessbewertung eingesetzt. Methodenbäume, deren Merkmale eine bessere Klassifikation erlauben, sollten auch mit einer höheren Wahrscheinlichkeit in die nächste Generation aufgenommen werden. Um die Fitness eines Methodenbaums zu bewerten, werden die folgenden Schritte durchgeführt:

1. Jedes Individuum, das heißt jeder Methodenbaum, wird auf einer Teilmenge der Rohdaten angewendet.
2. Das Ergebnis ist ein transformierter Datensatz bestehend aus den extrahierten Merkmalen, welcher für die Klassifikationsaufgabe verwendet wird.
3. Eine 10-fache Kreuzvalidierung schätzt die Performanz des Lernverfahrens auf den gegebenen Repräsentationen ab.
4. Die durchschnittliche Klassifikationsperformanz (z. B. *accuracy*) entspricht der zu maximierenden Fitness.

4 Klassifikationen mit Hilfe gelernter Methodenbäume

Das Ziel der automatisierten Merkmalsextraktion war eine Optimierung der Vorhersageperformanz des zweiten Lernschritts, welcher die extrahierten Merkmale verwendet. Abb. 1 aus der Einleitung verdeutlicht, dass die extrahierten Merkmale als Eingabe für einen zweiten Lernschritt verwendet werden. Abb. 7 beschreibt den rechten Teil der eingangs beschriebenen Übersicht.

Da die Merkmale nur auf Basis von Teilmengen der Daten extrahiert wurden, wird für diesen zweiten Lernschritt noch eine evolutionäre Merkmalsselektion mit Hilfe eines einfachen (1+1)EA durchgeführt. Erneut wird die Performanz des Klassifikationslerners als Fitnessfunktion verwendet. Die ausgewählten Methodenbäume werden dann auf dem kompletten Rohdatensatz angewendet und die mySVM hierauf angewendet. Alle Experimente wurden mit der maschinellen Lernumgebung YALE durchgeführt[1] [2].

4.1 *Klassifikation nach Genres*

Zunächst soll eine vorgegebene Klassifikation nach musikalischen Genres untersucht werden. Zu diesem Zweck werden drei Datensätze verwendet:

- Klassik/Pop: 100 Stücke jeder Klasse (Ogg Vorbis).
- Techno/Pop: 80 Stücke jeder Klasse (viele Alben, Ogg Vorbis).
- Hip Hop/Pop: 120 Stücke jeder Klasse (wenige Alben, MP3 mit 128 kbits/s).

Diese Lernaufgaben sind von ansteigender Schwierigkeit. Die Performanz der mySVM mit linearer Kernfunktion wurde mit einer 10-fach Kreuzvalidierung abgeschätzt. Die Ergebnisse finden sich in Tabelle 1. Für Klassik gegen Pop wurden vormals 93% Accuracy und für Hip Hop gegen Pop 66%

[1] YALE sowie ein Plugin zur automatisierten Merkmalsextraktion aus Wertereihen sind frei erhältlich unter http://yale.cs.uni-dortmund.de

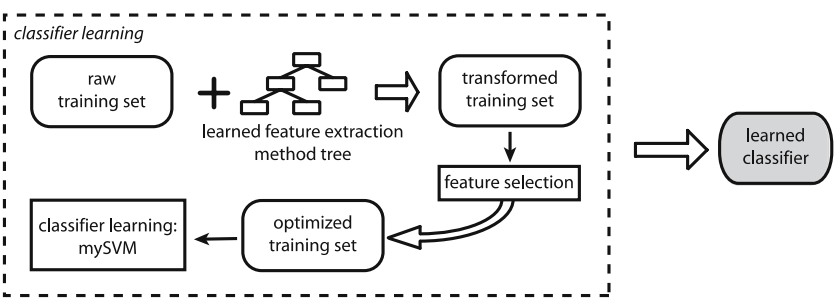


Abb. 7 Klassifikationslernen mit Hilfe des besten von der genetischen Programmierung „gezüchteten" Methodenbaums

Klassifikation nach Genres mit einer linearen SVM auf aufgabenspezifischen Merkmalssätzen

Tabelle 1

	Klassik/Pop	Techno/Pop	HipHop/Pop
Accuracy	100,00%	93,12%	82,50%
Precision	100,00%	94,80%	85,27%
Recall	100,00%	93,22%	79,41%
Fehler	0,00%	6,88%	17,50%

Accuracy veröffentlicht [17, 18]. Die mit dem hier beschriebenen Ansatz extrahierten Merkmale liefern also in jedem Fall vergleichbar gute Vorhersageergebnisse.

4.2 *Klassifikation nach Benutzerpräferenzen*

Empfehlungen von Liedern für potentielle Kunden basieren derzeit lediglich auf der Korrelation der zusammen verkauften Stücke. Dieser kollaborative Ansatz ignoriert jedoch vollständig inhaltliche Aspekte der Stücke. Insbesondere ist eine hohe Korrelation üblicherweise nur innerhalb des gleichen Genres gegeben, da Präferenzen über Genregrenzen hinaus seltener auftreten. Die Angabe der von einem Benutzer präferierten Stücke ermöglicht eine individuelle Klassifikation seiner Hörpräferenz. Dabei handelt es sich um eine äußerst anspruchsvolle Aufgabe, da aus nur wenigen Stücken zuverlässig und robust generalisiert werden muss. Das Benutzerverhalten kann dabei beliebig variieren und Genregrenzen ebenso beliebig überschreiten. Dies verdeutlicht nochmals den Bedarf für auf die Klassifikationsaufgabe zugeschnittene Merkmalssätze.

Vier Benutzer mit völlig unterschiedlichen Hörpräferenzen waren in der Lage, zwischen 50 und 80 Stücke ihrer Lieblingsmusik anzugeben sowie ebenso viele negative Beispiele. Benutzer 1 wählte positive Stücke mit einer dominierenden elektrischen Gitarre. Benutzer 2 wählte sowohl positive wie auch negative Stücke aus dem Bereich des Jazz. Benutzer 3 wählte genreübergreifend sowohl aus Klassik, Latin, Soul, Rock und Jazz. Benutzer 4 wählte ebenfalls Musik aus unterschiedlichen Genres, jedoch nur von wenigen verschiedenen Alben. Jeder Benutzer definierte also eine Lernaufgabe, für die ein Satz von Methodenbäumen gesucht war. Tabelle 2 zeigt die Ergebnisse.

Das exzellente Lernergebnis für Benutzer 1 entspricht den Erwartungen, da die positiven Beispiele alle aus einem leicht zu identifizierenden Genre stammen. Die Performanz für eine Lernaufgabe mit positiven und negativen Beispielen aus dem gleichen Genre führt zu vergleichbar guten Ergebnissen. Überraschend gut ist das Ergebnis für Benutzer 3, bei dem die Auswahl äußerst genreübergreifend stattfand. Dies ist ein Indikator dafür, dass die extrahierten Merkmale die Bildung von Präferenzclustern im Merkmalsraum eher bevorzugen als die Bildung von Genreclustern. Dies erklärt auch das im Vergleich schlechte Abschneiden für den vierten Benutzer, bei dem die Auswahl von Stücken nur einer kleinen Anzahl unterschiedlicher Alben mit ähnlichen Klangcharakteristiken entstammt.

Eine umfassende Diskussion der Ergebnisse sowie der Vergleich zu Standardmerkmalsmengen und unterschiedlichen Lernverfahren findet sich in [13].

Klassifikation gemäß der Hörpräferenz von Benutzern

Tabelle 2

	$Benutzer_1$	$Benutzer_2$	$Benutzer_3$	$Benutzer_4$
Accuracy	95,19%	92,14%	90,56%	84,55%
Precision	92,70%	98,33%	90,83%	85,87%
Recall	99,00%	84,67%	93,00%	83,74%
Fehler	4,81%	7,86%	9,44%	15,45%

5 Zusammenfassung

In diesem Artikel wurden Methoden für die Analyse großer Mengen von Audiodaten in einem strukturierten Rahmen vorgestellt. Einige neue Methoden – wie zum Beispiel die Rekonstruktion des Zustandsraums – wurden diskutiert und in diese Systematik eingeordnet. Von anderen Methoden wurde abstrahiert. Die verallgemeinerte dynamische Fensterung führt zum Konzept der Methodenbäume, welche effizient Merkmale aus Audiodaten extrahieren. Damit sind alle bekannten Methoden der Merkmalsextraktion aus Audiodaten abgedeckt, entweder direkt als Basismethode oder als Kombination elementarer Methoden. Durch diese Strukturierung können Methodenbäume automatisch generiert werden für neue Klassifikationsaufgaben. Hierzu wurde genetische Programmierung verwendet.

Die Verwendbarkeit dieses Ansatzes wurde durch die Anwendung auf zwei unterschiedliche Typen von Klassifikationsaufgaben demonstriert: die Klassifikation nach Genre und die nach Benutzerpräferenzen. Diese Experimente liefern äußerst vielversprechende Ergebnisse. Zukünftige Arbeiten sollten die Realzeitfähigkeit des vorgestellten Systems prüfen. Günstige Methodenbäume zur Genreklassifikation können bereits im Vorfeld erstellt und lediglich angewendet werden. Die Klassifikation nach Benutzerpräferenzen muss jedoch für jeden Anwender einzeln trainiert werden nach Angabe seiner bevorzugten Stücke.

6 Danksagung

Diese Arbeit wurde unterstützt von der *Deutschen Forschungsgemeinschaft (DFG)* im Rahmen des Sonderforschungsbereichs „Design und Management komplexer technischer Prozesse und Systeme mit Methoden der Computational Intelligence“.

Literatur

1. Bradley, E.: Time-Series Analysis. In: Intelligent Data Analysis: An Introduction, Springer 1999
2. Fischer, S., Klinkenberg, R., Mierswa, I., Ritthoff, O.: Yale: Yet Another Learning Environment – Tutorial. Technical Report CI-136/02, Collaborative Research Center 531, University of Dortmund, Dortmund, Germany, Juni 2002. ISSN 1433-3325
3. Ghias, A., Logan, J., Chamberlin, D., Smith, B.C.: Query by Humming: Musical Information Retrieval in an Audio Database. In: Proc. of ACM Multimedia, S. 231–236, 1995
4. Guo, G., Li, S.Z.: Content-Based Audio Classification and Retrieval by Support Vector Machines. IEEE Transaction on Neural Networks, 14(1):209–215, January 2003
5. Jayant, N.S., Noll, P.: Digital Coding of Waveforms: Principles and Applications to Speech and Video. Prentice Hall 1984
6. Joachims, T.: Optimizing search engines using clickthrough data. In: Proc. of Knowledge Discovery in Databases, 2002
7. Kahveci, T., Singh, A.K.: An efficient index structure for string databases. In: Proc. of the 27th VLDB, S. 352–360. Morgan Kaufmann 2001
8. Keogh, E., Pazzani, M.: An enhanced representation of time series which allows fast classification, clustering and relevance feedback. In: Proc. of the 4th Conference on Knowledge Discovery in Databases, S. 239–241, 1998
9. Koza, J.R.: Genetic Programming: On the programming of Computers by Means of Natural Selection. Cambridge, MA: MIT Press 1992
10. Kurth, F., Clausen, M.: Full-text indexing of very-large audio data bases. In: 110th Convention of the Audio Engineering Society, 2001
11. Liu, Z., Wang, Y., Chen, T.: Audio Feature Extraction and Analysis for Scene Segmentation and Classification. Journal of VLSI Signal Processing System, June 1998
12. Mierswa, I.: Beatles vs. Bach: Merkmalsextraktion im Phasenraum von Audiodaten. In: LLWA 03 – Tagungsband der GI-Workshop-Woche Lernen – Lehren – Wissen – Adaptivität, 2003
13. Mierswa, I., Morik, K.: Automatic Feature Extraction for Classifying Audio Data. Machine Learning Journal, 58:127–149 (2005)
14. Rüping, S.: mySVM Manual. Universität Dortmund, Lehrstuhl Informatik VIII, 2000. http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/
15. Schlittgen, R., Streitberg, B.H.J.: Zeitreihenanalyse. Oldenburg 2001
16. Takens, F.: Detecting strange attractors in turbulence. In: Rand, D.A., Young, L.S. (Hrsg.): Dynamical systems and turbulence, volume 898 of Lecture Notes in Mathematics, S. 366–381. Berlin: Springer 1980
17. Tzanetakis, G.: Manipulation, Analysis and Retrieval Systems for Audio Signals. PhD thesis, Computer Science Department, Princeton University, June 2002
18. Tzanetakis, G., Essl, G., Cook, P.: Automatic musical genre classification of audio signals. In: Proc. of the Int. Symposium on Music Information Retrieval (ISMIR), S. 205–210, 2001
19. Yi, B., Jagadish, H., Faloutsos, C.: Efficient retrieval of similar time series under time warping. In: Proc. of the 14th Conference on Data Engineering, S. 201–208, 1998
20. Zhang, T., Kuo, C.: Content-based Classification and Retrieval of Audio. In: SPIE's 43rd Annual Meeting – Conference on Advanced Signal Processing Algorithms, Architectures, and Implementations VIII, San Diego, July 1998