

Künstliche Intelligenz und Musik

SaxEx

Proseminar Künstliche Intelligenz

Patricia Weikert

6.6.2006

Abstrakt

Meine Arbeit „Künstliche Intelligenz und Musik“ behandelt die Funktionen und die Arbeitsweise des Programms SaxEx. Zuerst wird auf das Fallbasierte Schließen (Case Based Reasoning) eingegangen, welches zur Problemanalyse und zum Auffinden eines geeigneten Lösungswegs benötigt wird. Um diesen durchführen zu können, wird die danach behandelte Spektralanalyse erklärt. Das Programm arbeitet auf der Basis von Noos, das sich sehr gut für CBR-Systeme eignet. Nachdem die grundlegenden Bestandteile von SaxEx erläutert wurden, wird auf die Arbeitsweise des Programms selbst eingegangen. Abschließend werden die Grenzen von SaxEx aufgezeigt.

Inhaltsverzeichnis

1	Musikalische Künstliche Intelligenz	3
2	SaxEx	3
3	Das Fallbasierte Lösungssystem-CBR	4
4	Spectral Modeling Synthesis-SMS	6
5	NOOS	7
6	Die Arbeitsweise von SaxEx	7

6.1	Vorbedingung	7
6.2	Die Bearbeitung eines neuen Problems	8
7	Grenzen des Programms	9
8	Zusammenfassung	9

1 Musikalische Künstliche Intelligenz

Die Entwicklung der musikalischen Künstlichen Intelligenz begann in den 50er Jahren. Ihr Ziel war es mit Hilfe eines Computers Musik zu erschaffen, bei der der Zuhörer nicht entscheiden kann, ob sie von einem Menschen oder von einem Programm stammt.

Das Hauptproblem dabei ist es, die Affekte eines Stücks richtig herauszuarbeiten. Menschen haben dies durch Beobachtung und Imitation gelernt, sie können den Zuhörern, durch unterschiedliche Art und Weise des Vortrags eines Stücks, bestimmte Gefühle vermitteln.

Aus diesem Grund lag es nahe, das menschliche Wissen um diese Interpretationen in die Computer-Musik mit einfließen zu lassen und nicht solches Wissen selbstständig zu kreieren. Daraus entstand die Idee, menschliche Aufnahmen als Beispiele zu verwenden, an denen sich der Lösungsweg der Programme orientieren kann.

Doch daraus resultiert ein weiteres Problem, es ist für Computer-Musik-Systeme sehr kompliziert mit Musikdateien zu arbeiten, die nicht als MIDI Datei vorliegen, da nicht nur strikte musikalische Parameter vorhanden sind, sondern auch solche für den Ausdruck.

Ältere Systeme konnten damit nur schwer umgehen, sie kannten normalerweise nur zwei Ausdrucksmittel. Director Musices von Anders Friberg, Vittorio Colombo, Lars Fryden und Johan Sundberg konnte bereits drei Ausdrucksmittel handhaben. Mit SaxEx sind heutzutage fünf Parameter möglich, was dem Programm einen großen Vorteil gegenüber den anderen Systemen verschafft. Die bearbeiteten Stücke kommen schon viel näher an menschliche Interpretationen heran.

2 SaxEx

In Zusammenarbeit mit dem Institut Universitari de l'Audiovisual der Universität Pompeu Fabra haben Josep Lluís Arcos, Ramon Lopez de Mantaras und Xavier Serra im Jahr 2002 das Projekt SaxEx ins Leben gerufen.

Ihr Ziel war es, ein Programm zu entwickeln, das eine gefühlvolle Darstellung von Jazzballaden für Tenorsaxophon basierend auf den Beispielen menschlicher Musiker erstellen kann.

Realisiert haben sie dieses Programm mit zwei Techniken, die zunächst kurz vorgestellt werden sollen, „Case Based Reasoning“ und „Spectral Modeling Synthesis“.

3 Das Fallbasierte Lösungssystem-CBR

SaxEx benutzt ein Fallbasiertes Lösungssystem (Case Based Reasoning) genannt CBR, bei dem man Probleme löst, indem man bereits gelöste Probleme wiederverwendet (meist durch Adaption).

Diesem liegt die Idee zu Grunde, dass ähnliche Probleme auch ähnliche Lösungen besitzen. CBR ist passend für

1. Probleme, zu denen es viele Beispiele von bereits gelösten Problemen gibt
2. oder bei Problemen, deren Lösung schwer zu formulieren ist

Zwei Hauptmechanismen werden hierbei benutzt:

1. Das Finden von ähnlichen bereits gelösten Fällen nach bestimmten Kriterien
2. Das Anwenden der Lösung des alten Problems auf das Neue

Die Arbeit des CBR kann in mehrere Phasen untergliedert werden:

Retrieve: In dieser Phase werden bekannte, ähnlich, bereits gelöste Probleme gesucht. Dies Aufgabe kann wiederum in drei Teile untergliedert werden, Identify, Search und Select.

Identify: Es werden die hervorstechenden Eigenschaften des aktuellen Problems herausgearbeitet. So erhält man die Kriterien, nach welchen ähnliche Fälle gesucht werden sollen.

Search: Die Datenbank mit den bereits gelösten Problemen wird nach den Fällen durchsucht, die den Suchkriterien entsprechen.

Select: Das Suchergebnis wird nach den größtmöglichen Übereinstimmungen sortiert.

Reuse: Hier wird eine mögliche Lösung für das aktuelle Problem erstellt, indem die Lösungen der früheren Probleme auf das neue angewendet werden. Die vorangegangene Sortierung spielt hierbei eine nicht unerhebliche Rolle. In der Select-Maske kann eine obere Grenze festgelegt werden, wie viele Probleme dabei in Betracht gezogen werden sollen.

Revise: Diese Phase ist dazu da, eventuelle Fehler zu entdecken und diese mit Hilfe einer Repariertechnik zu beheben. Hier kann der Benutzer selbst mitentscheiden wie die endgültige Lösung des Problems aussehen soll. Auch die Kombination mehrerer Lösungen ist möglich. Die Änderungen können ebenfalls mit in den Datenbestand aufgenommen werden, so dass das Programm aus Fehlern lernt.

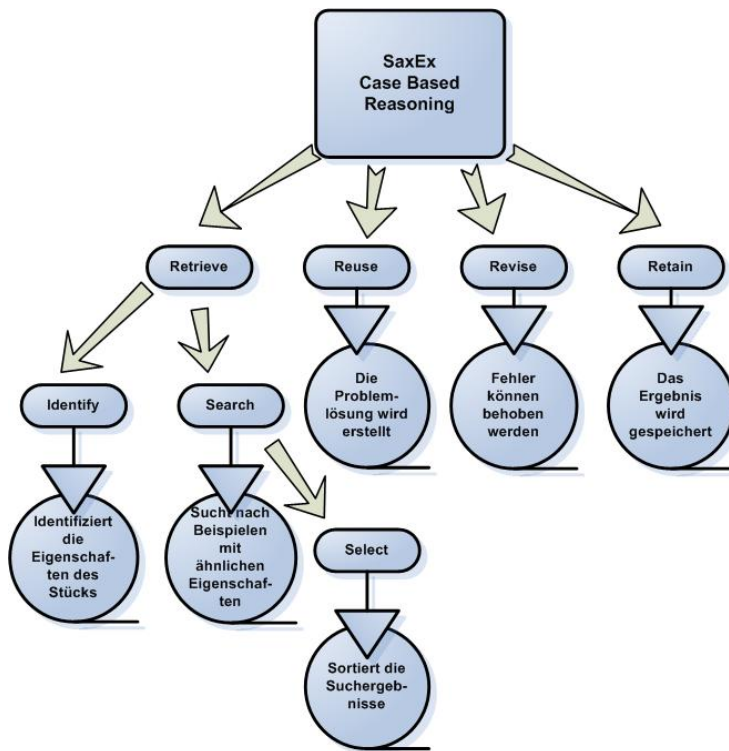


Abbildung 1: Dieses Schaubild verdeutlicht nochmals die Aufgaben der einzelnen Phasen

Retain: Es wird entschieden, welche Informationen des Falls wichtig genug sind um in das System mit aufgenommen zu werden und das gelöste Problem wird für spätere Fälle in den Programmdateien gespeichert.

Die Verwendung des „Case Based Reasoning Systems“ hat einen entscheidenden Vorteil: Dadurch dass die so gelösten Probleme in die Datenbank eingespeist werden können, „lernt“ das Programm automatisch. Und nicht nur durch die Fälle, sondern ebenso wenn vom Benutzer manuell Änderungen an der erstellten Lösung vorgenommen werden, denn diese Korrekturen werden von dem Programm beim Bearbeiten späterer ähnlicher Fälle berücksichtigt. Das selbstständige Lernen ist ein wichtiger Aspekt der Künstlichen Intelligenz, da nur so menschliches Verhalten realistisch nachgebildet werden kann.

4 Spectral Modeling Synthesis-SMS

Zu der „Spectral Modeling Synthesis“, kurz SMS, zählen mehrere Techniken zur Analyse, Transformation und Synthese von Geräuschen und Tönen. Sie haben das Ziel die Parameter zu finden, die echte Musik ausmachen, sie auf neue Stücke anzuwenden und die modifizierte Version des Originals in den bereits existierenden Datenbestand einzugliedern.

Die Spektralanalyse liefert alle wichtigen Schlüsselinformationen eines Stücks so dass eine Art musikalische Beschreibung entsteht. Genauer gesagt werden hierbei originale Sounddateien in ihre Sinuskurven und die spektralen Überreste (Störfrequenzen) umgewandelt. Technisch betrachtet geschieht das, indem die zeitabhängigen Spektral-Kenndaten eines Klangs durch zeitabhängige Sinuskurven dargestellt werden und diese aus dem Originalklang herausgefiltert werden. So bleibt nur die Störfrequenz übrig, die dann frequenziell approximiert werden kann. Dieser Schritt kann automatisch oder vom Benutzer selbst durchgeführt werden, abhängig von der Art der Klänge. Die Darstellung von monophonen (einstimmigen) Instrumentalklängen durch ihre Sinuskurven und zusätzliche Störfrequenzen lässt zu die Attribute herauszufiltern, die bei der Entschlüsselung des gefühlsbetonten Instrumentalvortrags eine wichtige Rolle spielen.

Die Attribute (oder auch Parameter) hierbei sind:

- Dynamik (Lautstärke)
- Rubato (Notenlänge)
- Vibrato (Wiederholung von Frequenz und Amplitude)
- Artikulation (Übergangszeit von Noten)
- Notenanschlag

Die extrahierten Attribute können verändert werden und auch wieder zu der Spektralen Darstellung der Töne hinzugefügt werden, ohne einen Qualitätsverlust zu erleiden. So wird aus einem normalen impressiven Ton ein gefühlvoller, bewegter Klang.

Durch die „Spectral Modeling Synthesis“ erhält man eine allgemeingültige Interpretation der Affekte beim expressiven Instrumentalvortrag. Die extrahierten Parameter können gespeichert, verändert und wiederverwendet werden.

Diese Technik wird von dem Programm SaxEx dazu benutzt, die grundlegenden Informationen der Stücke zu extrahieren, die die Ausdrucksstärke ausmachen und diese später wieder auf neue Stücke anzuwenden.

5 NOOS

SaxEx wurde in NOOS implementiert, eine reflektive objektorientierte Sprache zur Datendarstellung die programmiert wurde um Wissen zur Problemlösung und zum Lernen zu modellieren. NOOS wurde in LISP geschrieben und ist mit verschiedenen Plattformen kompatibel.

Die Modellierung eines Problems in Noos erfordert drei unterschiedliche Wissensgrundlagen:

Domain Knowledge: Eine Reihe von Konzepten, Relationen zwischen diesen Konzepten und Problem Daten die für die Anwendung relevant sind. Die Konzepte und ihre Relationen definieren die „Domain Ontology“ der Anwendung, diese besteht bei SaxEx zum Beispiel aus: Noten, Akkorden, der Realisationsstruktur und den Ausdrucksparametern. Die Problem Daten sind spezielle Probleme die von der Anwendung gelöst werden müssen.

Problem Solving Knowledge: Dieses Wissen bestimmt die Aufgaben die von der Anwendung gelöst werden sollen genauer. Die Hauptaufgabe von SaxEx zum Beispiel ist es, ausdruckslose musikalische Phrasen, expressiv zu gestalten.

Metalevel Knowledge ist das Wissen über die „Domain Knowledge“ und „Problem Solving Knowledge“. Dieses Wissen kann dazu verwendet werden, um Kriterien zu entwerfen, welche Lösung bei welchem Problem gewählt werden muss, um ein möglichst gutes Ergebnis zu erzielen.

Wurde ein Problem gelöst, so wird es von NOOS automatisch gespeichert und katalogisiert. Es ist von diesem Zeitpunkt an abrufbar und wiederverwendbar. Die Sammlung von gelösten Fällen wird „Episodic Memory“ genannt. Diese spezielle Fähigkeit von NOOS ist die Grundlage für das integrierende Lernen und das „Case Based Reasoning“ das von SaxEx genutzt wird.

6 Die Arbeitsweise von SaxEx

6.1 Vorbedingung

Verschiedene Aufnahmen von Jazzballaden wurden von einem Tenor Saxophon mit unterschiedlicher Ausdrucksstärke, inklusive einer ausdruckslosen Version, eingespielt. Zu jedem Stück wurde das sogenannte „score“ in die Datenbank eingespeist, eine MIDI Datei, die alle melodischen und harmonischen Informationen des Stücks enthält wie zum Beispiel die Noten und Akkorde. Diese Aufnahmen wurden dann mit Hilfe der „Spectral Modeling Technique“, kurz SMS, analysiert um die Ausdrucksparameter der Stücke zu erhalten. Die Noten und die

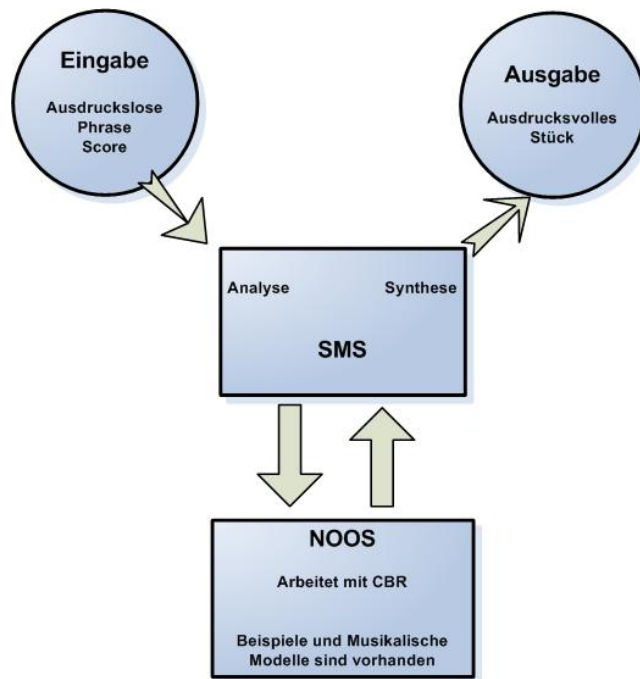


Abbildung 2: Schematische Darstellung der Arbeitsschritte

dazugehörigen Parameter ergeben zusammen die Fälle des Case Based Reasoning System (CBR). Wenn ein neues Stück bearbeitet werden soll wird auf diese Basisfälle zurückgegriffen, sie sind die Grundlage des ganzen Programms.

6.2 Die Bearbeitung eines neuen Problems

Als Eingabe benötigt das Programm eine Sounddatei, die Ballade von einem Musiker in neutraler Weise eingespielt und das „score“ (s.o.) mit allen wichtigen Daten des Stücks. Das Programm durchsucht daraufhin die bereits existierenden Fälle nach ähnlichen Stellen um die Bedeutung der einzelnen Töne im Zusammenhang herauszufinden. Diese Aufgabe ist technisch betrachtet die Retrieve Phase des Case Based Reasoning. In der Reuse Phase wird die originale Sounddatei mit Hilfe der „Spectral Modeling Technique“ den Attributen der gefundenen Stellen entsprechend verändert. Dazu werden die extrahierten Störfrequenzen zu den Sinuskurven der Töne der Sounddatei hinzugefügt. Anschließend kann das Ergebnis noch nach eigenen Wünschen verändert werden, sei es durch Kombination mehrerer Lösungen oder durch explizite Anpassung einzelner Stellen und Töne. Letztendlich wird der Fall mit der Lösung und den Änderungen in den Datenbestand aufgenommen.

Wie in Abbildung 2 zu sehen ist, kann die Arbeit in 3 Phasen eingeteilt werden:

1. Die Analyse: Die als Input erhaltene Sounddatei wird mit Hilfe der „Spectral Modeling Technique“ analysiert.
2. Die Schlussfolgerung: Durch das CBR System wird eine mögliche Lösung ermittelt.
3. Die Synthese: Die Lösung wird wiederum mit der „Spectral Modeling Technique“ durchgeführt und liefert als Output eine expressive Phrase.

7 Grenzen des Programms

SaxEx hat gegenüber den meisten anderen Programmen, die künstliche Intelligenz einsetzen, deutliche Vorteile, schon allein dadurch, dass es die fünf wichtigsten Parameter für expressive Musik extrahieren und rekonstruieren kann. Dennoch hat es auch seine Grenzen.

Bis jetzt wurde das Konzept von SaxEx nur speziell auf das Tenorsaxophon angepasst, so dass ein Stück, das auf einem anderen Instrument eingespielt wurde, überhaupt nicht bearbeitet werden kann. Dies liegt daran, dass die Frequenzspektren bei allen Instrumenten unterschiedlich ausgeprägt sind und daher neu analysiert werden müssten. Ebenso verhält es sich mit Stücken, die von mehreren Instrumenten eingespielt wurden. Hier ist es nicht möglich einzelne Stimmen extrahieren und diese bearbeiten. Die eingespeiste Sounddatei muss monophon (= einstimmig) sein. Außerdem kann das Programm nur Jazz Balladen analysieren, da andere Musikstile auch anderen Ausdrucksregeln folgen.

8 Zusammenfassung

Im Anwendungsbereich der Künstlichen Intelligenz versucht man Computerprogrammen beizubringen selbstständig so zu agieren wie ein Mensch tun würde. In der musikalischen KI bedeutet dies, dass Musikstücke so erstellt und vorgetragen werden können, so dass der Zuhörer keinen Unterschied zu einem menschlichen Vortrag erkennen kann. Durch SaxEx ist man diesem Ziel schon einen großen Schritt näher gekommen.

Die Parameter, die den gefühlvollen Vortrag ausmachen, können mit Hilfe der SMS Technologie gefunden und auf neue Stücke angewendet werden. Durch das „Case Based Reasoning“ ergeben sich die Regeln, nach denen diese Parameter auf neue Stücke angewendet werden müssen und das Programm erhält die Fähigkeit zu lernen, indem die bearbeiteten Stücke der Datenbank hinzugefügt

werden und manuelle Änderungen an der Lösung später ebenfalls berücksichtigt werden.

Bis jetzt wurde die Technologie von SaxEx jedoch nur für monophone Jazz Balladen für das Tenorsaxophon realisiert, es ist aber vermutlich nur eine Frage der Zeit, bis es weitere Programme für andere Instrumente entwickelt werden, die vielleicht auch mit mehr als einer Stimme umgehen können. Doch den Charme eines echten Konzerts wird auch ein noch so perfektes Programm niemals erreichen können.

Literatur

- [1] SaxEx: A Case-Based Reasoning system for generating expressive performances (1997)
- [2] Ramon Lopez de Mataras, Josep Lluís Arcos und Xavier Serra Veröffentlicht bei „Proceedings of the International Computer Music Conference“ (ICMC '97)
- [3] AI and Music from Composition to Expressive Performance (2002) Ramon Lopez de Mataras und Josep Lluís Arcos Artikel der American Association for Artificial Intelligence
- [4] A Case-Based Reasoning system for generating expressive performances Internetartikel von Josep Lluís Arcos (1997)
- [5] <http://www.iiia.csic.es/Projects/music/Saxex.html> (5.6.2006)