

AI and Music

Seminararbeit

im Studiengang Wirtschaftsinformatik
in der Fakultät Angewandte Informatik
der Otto-Friedrich-Universität Bamberg

Lehrstuhl: Angewandte Informatik, insbesondere Kognitive Systeme
Inhaber: Prof. Dr. Ute Schmid
Betreuer/in: Prof. Dr. Ute Schmid

Verfasser/in: Ludwig Schallner
Matrikelnummer: 1850413
Adresse: Pestalozzistraße 9e, 96052 Bamberg
E-Mail: ludwig-felix-walter.schallner@stud.uni-bamberg.de
Studiengang: Master Wirtschaftsinformatik
2. Fachsemester / 9. Hochschulsesemester

Datum der Abgabe: 28.02.2017

Abstract

Die verfasste Seminararbeit soll einen Eindruck in die Aspekte der Musikererkennung mit Hilfe von sogenannter „Künstlicher Intelligenz“ vermitteln. In Punkt eins wird kurz drauf eingegangen, was die Motivation hinter der Idee Musik überhaupt erkennen und verstehen zu wollen steckt. In den nachfolgenden Punkten zwei bis vier werden verschiedene Anwendungsbeispiele vorgestellt, beispielsweise den der Akkorderkennung mit Hilfe von einem einfachen Templated-Based Ansatzes aber auch von einem ausgefeilteren Ansatzes der mit einem Hidden Makrov Modell arbeiten. Ein anderes Anwendungsbeispiel ist die Genreerkennung mithilfe verschiedenster Algorithmen von NaivesBayes bis hin zum J48 Algorithmus. Das zuletzt betrachtete Anwendungsbeispiel ist, die Erkennung des Sängers, dies gestaltet sich deutlich komplizierter, da viel mehr sogenannte Noise vorliegt, daher um die Ansätze besser vergleichen zu können werden die Daten einmal mit instrumentaler Begleitung analysiert und einmal wird der Gesang von jener separiert.

Schlüsselwörter: AI, Music, Klassifikation, Akkorde, Genre, Sänger

Inhaltsverzeichnis

Abstract	I
Inhaltsverzeichnis	II
Abkürzungsverzeichnis	III
Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
1 Motivation	1
2 Chord Classification	2
2.1 Theoretische Grundlagen	2
2.1.1 Oktave	2
2.1.2 Chords and Scales	2
2.2 Template-Based Chord Recognition	2
2.3 HMM-Based Chord Recognition	4
3 Classification of musical genre	6
3.1 Theoretische Grundlagen	6
3.1.1 MIDI	6
3.1.2 Algorithmen	7
3.2 Ergebnisse der Untersuchungen	7
4 Composer Classification	10
4.1 Theoretische Grundlagen	10
4.1.1 Lineare und Quadratische Diskriminanzfunktion.....	10
4.1.2 GGM basierender maximum likelihood classifier	10
4.1.3 Kullback-Leibler Divergenz	11
4.2 Die Ansätze im Vergleich	11
5 Fazit	13
Literaturverzeichnis	1
Eidesstattliche Erklärung	2

Abkürzungsverzeichnis

HMM	Hidden Markov Modell
MIDI	Musical Instrument Digital Interface
VFI	Voting Feature Intervals
NNge	Nearest-Neighbor ohne generalisierten Beispielen
JRip	Ripper
GMM	Gaussian Mixture Model
dB	Dezibel
SAR	singing-to-accompaniment ratio
LDF	Linear discriminant
QDF	Quadratic discriminant functions
GMM-A	Artist level GMM
GMM-KL-A	Artist level GMM mit KL Distanz
GMM-S	Song level GMM
GMM-KL-S-1NN	Song Level GMM mit KL Distanz und mit einem Nearest Neighbor
GMM-KL-S-3NN	Song Level GMM mit KL Distanz und den drei nearest Neighbors
G-KL-A	Artist level single Gaussian und KL Distanz
G-Mah	Artist-level Mahalaobis distance

Abbildungsverzeichnis

Abbildung 1 Grafischer Überblick über den Prozess des Template-Based-Ansatzes	3
Abbildung 2 Die verschiedenen Ansätze in der Gegenüberstellung	5
Abbildung 3 Anwendung der Multiclass Genre Klassifikation.....	9
Abbildung 4 Binäre Genre Klassifikation mit 66% Trainingsdaten.....	9
Abbildung 5 LDF als Basisvergleich und die zwei besten Klassifikatoren im Vergleich.....	12

Tabellenverzeichnis

Tabelle 1 Ergebnisse der F-Messung.....	8
Tabelle 2 Mehrstimmiger Musik	11
Tabelle 3 Leistung bei separierten Gesang	11

1 Motivation

Musik spielt für Milliarden von Menschen – auf unserem Planeten – eine enorm wichtige Rolle. Die digitale Revolution führte auch im Bereich der Künstlichen Intelligenz in Bezug auf die Musik zu Weiterentwicklungen. Vor allem der Nutzen des Endverbrauchers ist in diesem Punkt besonders Interessant, nämlich das dieser Musik ganz anderes betrachten und entdecken kann. Auch kann man mit Hilfe von den vorgestellten Techniken Musik auf verschiedenen Ebenen erkannt oder eventuell können gar Vergleiche gezogen werden (Meinard, 2015, S. Preface).

2 Chord Classification

Der Ansatz der „Chord Classification“ der in dieser Seminararbeit als relevantesten erachtet wurde stammt von Prof. Dr. Meinard Müller vom Lehrstuhl Semantische Audiosignalverarbeitung der Friedrich-Alexander-Universität Erlangen-Nürnberg.

2.1 Theoretische Grundlagen

Nachfolgend werden einige theoretische Grundlagen näher erläutert. Ein Intervall, beispielsweise, spielt eine enorm wichtige Rolle in der Musik(-erkennung), dieses Repräsentiert „den Abstand zwischen zweier Töne“ (Kaiser-Kaplaner).

2.1.1 Oktave

Eine Oktave ist ein, der grundlegendsten Intervalle. Definiert ist eine Oktave als eine Distanz zwischen zwei Tönen mit der halben oder doppelten Grundfrequenz (Meinard, 2015, S. 239)

2.1.2 Chords and Scales

Ein Akkord ist eine Gruppe von mehreren Noten, die gleichzeitig ertönen. Durch die Anzahl der – gleichzeitig gespielten – Töne werden Akkorde untereinander wie folgt kategorisiert (Meinard, 2015, S. 243):

- Dyad (Zwei Noten)
- Triad (Drei Noten)
- Tetrad (Vier Noten)
- Und so weiter

Bei einer Harmonieanalyse werden häufig Noten die sich um eine oder mehrere Oktaven unterscheiden derselben Klangqualität zugeordnet. Deshalb macht es Sinn in dem Zusammenhang der Musikererkennung von Tonhöhenklassen zu sprechen, anstatt von Noten (Meinard, 2015, S. 243).

2.2 Template-Based Chord Recognition

Das Ziel dieses Ansatzes – des Templated-Based-Ansatzes – der „Musikererkennung“ ist es herauszufinden, welcher Akkord zur welche Zeit gespielt wurde, anhand des Abgleiches von vordefinierten Templates (Meinard, 2015, S. 254).

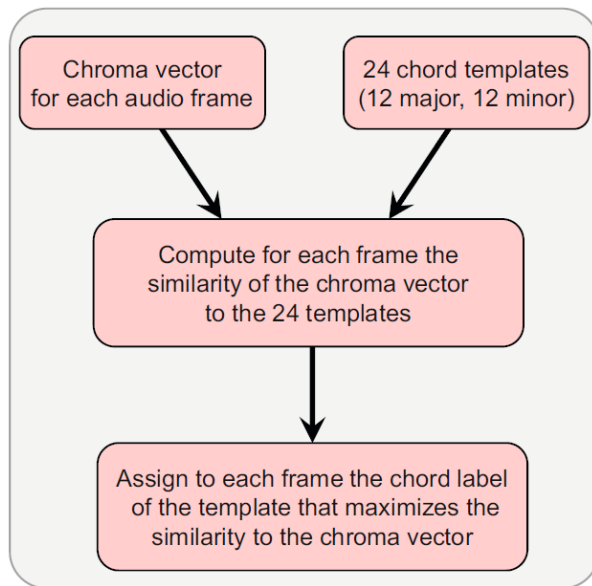


Abbildung 1 Grafischer Überblick über den Prozess des Template-Based-Ansatzes (Meinard, 2015, S. 255)

Der erste Schritt, bei dem Template-Based Chord Recognition-Ansatzes, ist es das aufgenommene Material in eine Sequenz ($X = (x_1, x_2, x_3 \dots x_N)$) von „feature“-vektoren ($x_N \in F$, $n \in [1 : N]$), dabei bildet F einen geeigneten „feature space“ ab, einzuteilen. Danach wird jeder „feature vektor“ (x_N) einem „Akkord-Label“ zugeordnet $\lambda_n \in \Lambda$, wobei Λ eine Sammlung von möglichen Akkorden ist (Meinard, 2015, S. 254).

Beispielhaft $\Lambda = \{C, C^\#, \dots, B, C_m, C^\#_m, \dots, B_m\}$ bestehend aus 12 Dreiergruppen mit Haupt- und Nebendreiklängen. In diesem Beispiel wird jedes Frame $n \in [1 : N]$ einem Hauptakkord oder Nebenakkord der von λ_n spezifiziert wird, zugewiesen (Meinard, 2015, S. 254).

Als nächstes folgt die Extraktion der Eigenschaften des Musikstückes. Die meisten Erkennungssysteme arbeiten mit sogenannten Chroma-Features. Diese gegeben die Eigenschaften – der harmonischen und melodischen Charakteristiken – von Musik wieder und sind dabei Robust gegenüber Änderungen im Takt und der Rhythmik.

In seinen Untersuchungen setzte Prof. Dr. Meinard Müller, auf folgendes Chroma-Set $\{C, C^\#, D, \dots, B\}$ also von $[0:11]$, wobei C durch 0, $C^\#$ durch 1 bis B durch 11 repräsentiert wurde und einer sogenannten „Window size“ von 200ms. Daraus resultierte, dass die „feature“ Sequenz $X = (x_1, x_2, \dots, x_N)$ eine „feature“-Rate von 10 Hz hat. Am Ende lag also eine Chroma-Feature-Repräsentation des betroffenen Musikstückes vor (Meinard, 2015, S. 254).

Für den „pattern-matching-step“ wurde eine simple Template-basierende Erkennungsstrategie eingesetzt. Es wird jeder Chroma-Vektor mit jedem einzelnen der 24 binären Akkord-Templates verglichen, herauskommt ein Gleichheitswert für jedes Template.

Anhand des Gleichheitswertes wird das Akkordlabel ausgewählt, welches den höchsten Gleichheitswert erzielt (Meinard, 2015, S. 254 ff.).

Zu guter Letzt werden die Resultate in Bezug auf der Euclidean-Norm normalisiert. Zusätzlich kann noch ein sogenanntes „prefiltering“ einsetzen, dieses führt zwar dazu das sogenannter „Noise“ herausgefiltert werden kann, dadurch gehen aber auch Übergänge zwischen den Akkorden verloren, was die Erkennung jener erschweren kann (Meinard, 2015, S. 257).

2.3 HMM-Based Chord Recognition

Beim bisherigen Ansatz war es jedes Frame einzeln zu betrachten, allerdings ohne Berücksichtigung der vorherige Frame oder der nachfolgenden Frames. In der Musik sind Akkord Änderungen nicht willkürlich, sondern folgen bestimmten Regeln (Meinard, 2015, S. 273). Im Umkehrschluss lässt sich dadurch viel besser drauf schließen, welche Akkorde gerade gespielt werden könnten, wenn die vorherigen bekannt sind.

Die Grundidee in diesem Ansatz, von Prof. Dr. ist es ein Modell (Hidden Markov Model) zu verwenden das die Wahrscheinlichkeit ausdrückt, dass ein Akkord zu einem anderen Akkord übergeht (Meinard, 2015, S. 273).

Im Gegensatz zum Pattern-Matching-Ansatz bleiben die Chroma-Features unberührt, dies führt dazu, dass der Ansatz mit Hilfe eines HMM auch „postfiltering“ bezeichnet wird (Meinard, 2015, S. 273).

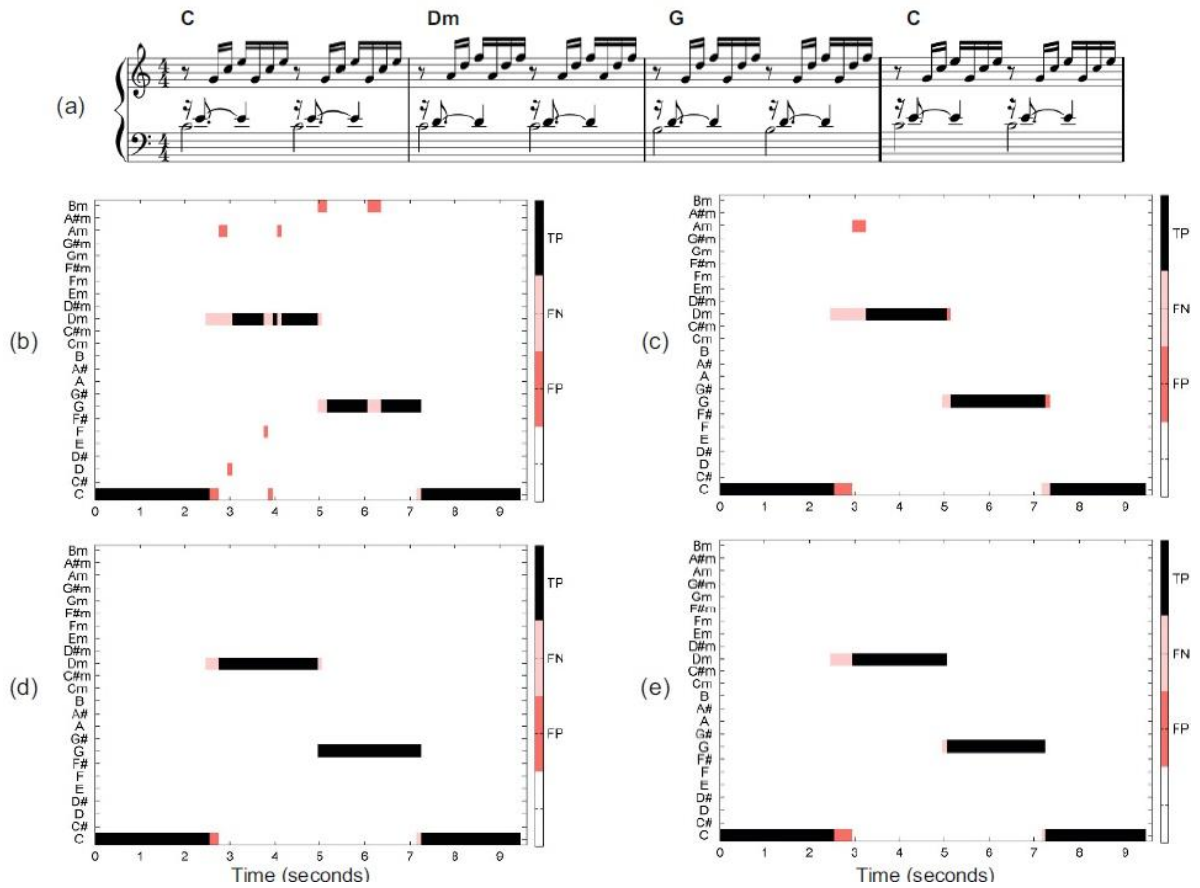


Abbildung 2 Die verschiedenen Ansätze in der Gegenüberstellung (Meinard, 2015, S. 292)

Die obige Abbildung stellt folgende Ansätze dar:

- b) Template-based-Ansatz
- c) Template-based-Ansatz mit Prefiltering
- d) HMM
- e) HMM mit prefiltering

Diese Grafik macht erkennbar, dass der Ansatz mithilfe eines HMM deutlich besser funktioniert – im Sinne von bessere Resultate erzielt – als der vorher vorgestellte Template-Based-Ansatz.

3 Classification of musical genre

Im Nachfolgenden wird eine weitere Einsatzmöglichkeit von „Machine Learning“ in musikalischen Aspekten betrachtet, die Genreerkennung des betrachteten Musikstückes.

Die Wissenschaftler der Italienischen Universität „Rome Tor Vergata“: Roberto Basili, Alfredo Serfini und Armando Stellato, beschäftigten sich intensiv mit diesem Aspekt, diese Seminararbeit soll ihre Ansätze dem Leser einen kleinen Überblick über Ihre Arbeit verschaffen.

Wenn versucht wird einer Person ein Ihr unbekanntes musikalisches Genre zu erläutern – im Sinne von, dieser zu erläutern wodurch sich dieses Genre auszeichnet – wird auf ein großes Problem gestoßen, welches die Genreerkennung betrifft, nämlich die Definition dieser. Diese existieren nur sehr vage, in anderen Worten eine eindeutige Definition existiert leider nicht (Basili, Serafini, & Stellato, 2004, S. 1 f.)

Die Untersuchung von Roberto Basili, Alfredo Serfini und Armando Stellato beschäftigen sich mit den symbolischen musikalischen Aspekten damit möglichst viele Informationen ohne sogenannter “Noise“ gesammelt werden. Wie die Autoren in ihrer eigenen Arbeit erwähnen ist der von Ihnen gewählte Ansatz bei weitem nicht perfekt um eine fein granulare Genrespezifikation zu ermöglichen. Aber es ist ein Anfang. (Basili, Serafini, & Stellato, 2004, S. 1 f.).

3.1 Theoretische Grundlagen

In Punkt 3.1 werden die theoretischen Grundlagen, welche benötigt werden die Untersuchungen der genannten Autoren zu verstehen, näher erläutert.

3.1.1 MIDI

MIDI ist das sogenannte Musical Instrument Digital Interface, dies kann als Äquivalent zum Notenpapier angesehen werden. Diese Dateien beinhalten also nicht direkte Musik, sondern geben nur Anweisungen beispielsweise wie laut, welche Note und welches Instrument gespielt werden soll und logischerweise auch, wann aufgehört werden soll eine Note zu spielen (Newmarch, 2017).

3.1.2 Algorithmen

Nachfolgend werden die Algorithmen die in der Arbeit der schon mehrfach genannten Autoren zum Einsatz kamen näher erläutert.

Der sogenannte „**Naive-Bayes-Klassifikator**“ führt statistische Analysen anhand der gegebenen Trainingsdaten durch. Ermittelt für jedes Objekt die bedingten Wahrscheinlichkeiten, um diese als Entscheidungskriterium verwenden zu können (Basili, Serafini, & Stellato, 2004, S. 2).

Der so genannte Nearest-Neighbor-like Algorithmus ohne den Gebrauch von verschachtelten generalisierten Beispielen vorgestellt werden. Der abgekürzt **NNge** Algorithmus zeichnet sich dadurch aus, dass dieser ein regelbasierter Algorithmus ist. Dieser baut eine Art von „hypergeometrisches“ Modell welches zusätzlich Wenn-Dann Regeln beinhaltet. Auch ein regelbasierter Algorithmus ist der RIPPER kurz **JRip** dieser nutzt allerdings auch ein so genannten „propositional rule learner“ (Basili, Serafini, & Stellato, 2004, S. 2).

Eine Implantation des Quinlan Algorithmus ist der **J48 Algorithmus**. Dieser Algorithmus arbeitet mit einem Entscheidungsbaum (Top-Down-Navigation), dessen Knotenpunkte Abgrenzungsregeln – basierend auf selektiven Eigenschaften – repräsentieren (Basili, Serafini, & Stellato, 2004, S. 2). Der **Part**-Algorithmus ist dem J48 Algorithmus sehr ähnlich, dieser benutzt aber eine separate-and-conqueror Strategie um das beste „Blatt“ des Entscheidungsbaumes bei jedweder Iteration auszuwählen. Dies führt zu einem optimierten Teil-Entscheidungsbaum (Basili, Serafini, & Stellato, 2004, S. 2).

Der „**Voting Feature Intervals**“-Algorithmus baut zu aller erst Eigenschaftsintervalle zur jeder Klasse und Attribut auf, dann erfolgt eine sogenannte Abstimmungsstrategie um das Modell zu bewerten (Basili, Serafini, & Stellato, 2004, S. 2).

3.2 Ergebnisse der Untersuchungen

Es wurden 300 Songs (im MIDI Format) aus sechs verschiedenen Genres betrachtet:

- Klassische Musik,
- Blues,
- Disco,
- Jazz,

- Pop und
- Rock

171 dieser Songs wurden von zwei sogenannten „Annotator“ dem jeweiligen Genre zu gewiesen. Danach wurde eine Standard F-Messung durchgeführt um die „inter-annotator“-Übereinstimmung festzustellen (Basili, Serafini, & Stellato, 2004, S. 2).

Tabelle 1 Ergebnisse der F-Messung (Basili, Serafini, & Stellato, 2004, S. 2)

MusicalGenres	Annotations		Common Annotations	F-Measure
	1st	2nd		
Blues	51%	40%	40%	89%
Classical	17%	17%	17%	100%
Disco	31%	24%	24%	89%
Jazz	24%	28%	23%	89%
Pop	26%	29%	20%	73%
Rock	22%	33%	22%	83%

Anhand der oben gezeigten Grafik bestätigt sich der Verdacht der Autoren, dass “Pop“ ein schwer erkennbares Genre ist. Da dieses ein Schmelztiegel verschiedener Songs ist, welche einfach nur der Masse der Hörer gefällt (Basili, Serafini, & Stellato, 2004, S. 2).

Für die verschiedenen Algorithmen untereinander besser vergleichen zu können, wurde nicht nur jeder einzelne Algorithmus untereinander verglichen, sondern wurden auch der Anteil der Trainingsdaten der beobachteten Daten verändert (66%, 75%, 90%). Außerdem wurden zwei verschiedene Verfahren zur Kategorisieren und Daten angewandt. Beim sogenannten „**Single Multiclass Class**“-Kategorisieren werden alle Instanzen betrachtet und einen der sechs Genres zugeordnet, beim „**Multiple Binary**“-Kategorisieren werden verschiedene Kategorizierer – einer pro Genre – durch unabhängige Lernprozesse abgeleitet (Basili, Serafini, & Stellato, 2004, S. 2 f.).

Durch die Abbildung 3 wird ersichtlich, dass zwar der **NNge** eine weitbessere Trefferquote bei reinen Trainingsdaten hat, aber **NaiveBayes** leistet bei weitaus realistischeren Umständen deutlich bessere Arbeit.

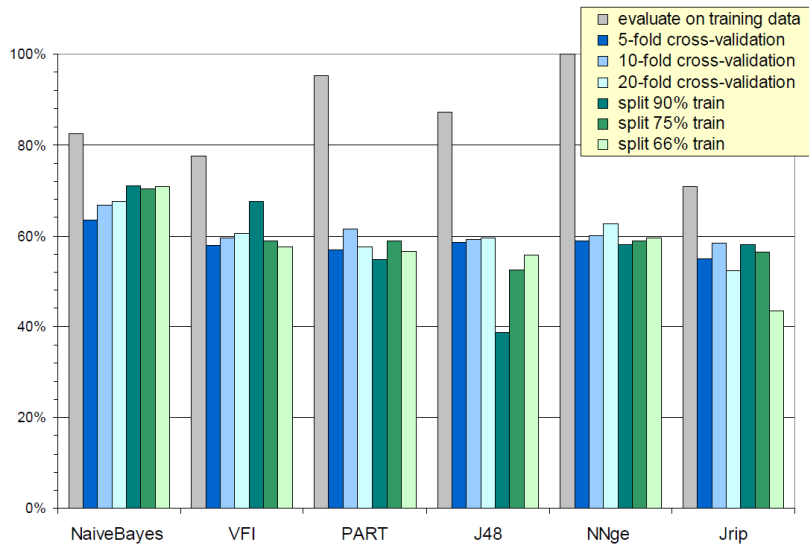


Abbildung 3 Anwendung der Multiclass Genre Klassifikation (Basili, Serafini, & Stellato, 2004, S. 3)

Um die Leistung der verschiedenen Algorithmen besser messen zu können wurde die „Binary Genre“-Klassifizierung verwendet und zusätzlich mit 66% Anteil an Trainingsdaten die Leistung der Algorithmen untersucht siehe Abbildung 4. Auch hier wird wieder ersichtlich das NaivesBayes eine sehr gute Leistung der Erkennung des Genres erzielt

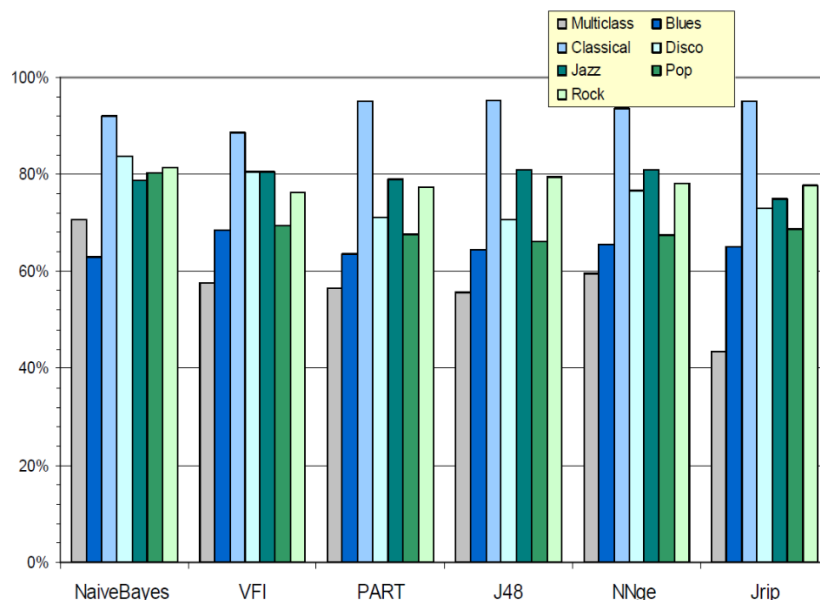


Abbildung 4 Binäre Genre Klassifikation mit 66% Trainingsdaten (Basili, Serafini, & Stellato, 2004, S. 4)

4 Composer Classification

Die Ansätze von Annamaria Mesaros, Tuomas Virtanen und Anssi Kalpuri haben zum Ziel den Sänger des betrachteten Musikstücks zu identifizieren. Die Erkennung des Interpreten ist deshalb so wichtig, da dieser in einem Musikstück mit Gesang meist die höchste Beachtung erhält. Die meisten Menschen erkennen anhand des Gesanges den Song, also den Titel oder auch den Interpreten et cetera. Ein System welches eben dies vollführen kann wäre für den Bereich des musikalischen „Information Retrieval“ sehr relevant und nützlich.

Das Problem der Erkennung des Sängers oder der Sängerin des gesungenen Liedes liegt der Tatsache zu Grunde, dass eben diese stimmliche Performance von musikalischen Instrumenten begleitet wird. Diese erschweren logischerweise die Erkennung des Sängers da diese „Geräusche“ nicht vom Sänger produziert werden (Mesaros, Virtanen, & Klapuri, 2007, S. 1).

4.1 Theoretische Grundlagen

Im nachfolgenden werden theoretische Grundlagen die zum Verständnis der Untersuchung benötigt werden näher erläutert.

4.1.1 Lineare und Quadratische Diskriminanzfunktion

Die Diskriminante Analyse ist eine einfaches Verfahren um eine Sammlung von Beobachtungen in vordefinierte Klassen zu Klassifizieren. Das Verfahren gestaltet basierend auf den vorliegenden Trainingsdaten eine so bezeichnete Diskriminanzfunktion (Mesaros, Virtanen, & Klapuri, 2007, S. 2 f.).

Wenn eine neue Beobachtung gegeben ist, wird diese Funktion ausgewertet und die Beobachtung wird der Klasse zugeordnet mit dem höchsten Wert der jeweiligen Diskriminanzfunktion. Nach Zuweisung der Klasse – mit dem höchsten Wert – für jedes Frames wird das gesamte Signal der Klasse zugewiesen, welche der Mehrheit der einzelnen Frames zugewiesen wurde. Durch das Anwenden von sogenannten “cross-terms“, erhält man eine quadratische Diskriminanzfunktion. Diese unterscheidet sich zur Linearen dahingehend, dass a_i ein Vektor und A_i ist eine Matrix ist (Mesaros, Virtanen, & Klapuri, 2007, S. 2 f.).

4.1.2 GGM basierender maximum likelihood classifier

Auch ein Gaussian Mixture Model (GMM) wurde eingesetzt, um dieses zu trainieren wurde wie gewöhnlich der Expectation-Maximization Algorithmus angewandt, die daraus resultierenden Parameter führen zu einem inhärenten Diskriminanzmodell der Sänger Klassen.

An sich befasst sich der Klassifikator damit die Klasse mit der höchsten Wahrscheinlichkeit für die beobachtete Menge zu finden (Mesaros, Virtanen, & Klapuri, 2007, S. 2f).

4.1.3 Kullback-Leibler Divergenz

Zusätzlich kam die Kullback-Leibler Divergenz zum Einsatz – welche einfach nur den Song der Sängerklasse zuzuweisen die den geringste Kullback-Leiber Divergenz vorwies (Mesaros, Virtanen, & Klapuri, 2007, S. 2 f.).

4.2 Die Ansätze im Vergleich

Um die Ansätze gut und faktenbasiert vergleichen zu können, wurden die Datenbank mit vier bis sechs Melodien – welche eine Länge von 20 bis 30 Sekunden vorwiesen – pro Sänger (von 13 Sänger insgesamt) gefüllt. Jeder der Sänger hatte die selbe musikalische Begleitung. Dies geschah damit die Musikalische Begleitung nicht Sänger spezifisch ist, um zu gewährleisten, dass der Sänger durch seinen Gesang – also seine Stimme und nicht der musikalischen Begleitung – erkannt wurde. Außerdem wurde verschiedene Dezibel Verhältnisse von Gesang und instrumentaler Begleitung mit einbezogen. In den nachfolgenden Tabellen als SAR Erkenntlich (Mesaros, Virtanen, & Klapuri, 2007, S. 3 f.).

Tabelle 2 Mehrstimmiger Musik (Mesaros, Virtanen, & Klapuri, 2007, S. 3 f.)

SAR [dB]	-5	0	5	10	30
LDF	28	42	55	61	63
QDF	42	53	57	69	75
GMM-A	38	36	53	65	71
GMM-KL-A	26	51	63	73	78
GMM-S	25	28	44	50	57
GMM-KL-S-1NN	21	32	32	55	59
GMM-KL-S-3NN	26	42	48	61	73
G-KL-A	13	25	36	40	38
G-Mah	25	34	48	57	65

Tabelle 3 Leistung bei separierten Gesang (Mesaros, Virtanen, & Klapuri, 2007, S. 3 f.)

SAR [dB]	-5	0	5	10	30
LDF	44	46	50	59	46
QDF	63	61	67	77	67
GMM-A	67	75	79	80	84
GMM-KL-A	63	69	82	78	75
GMM-S	51	59	71	73	76
GMM-KL-S-1NN	50	61	65	65	67
GMM-KL-S-3NN	51	61	59	65	69
G-KL-A	46	51	50	51	48
G-Mah	53	51	53	51	48

Nachfolgend werden die Abkürzungen die in Tabelle 2 und 3 verwendet wurden erläutert.

- LDF = Lineare Diskriminanzfunktion;
- QDF = Quadratische Diskriminanzfunktion
- GMM-A = Artist level GMM;

- GMM-KL-A = Artist level GMM mit KL Distanz
- GMM-S = Song level GMM
- GMM-KL-S-1NN = Song Level GMM mit KL Distanz und mit einem Nearest Neighbor
- GMM-KL-S-3NN = Song Level GMM mit KL Distanz und den drei Nearest Neighbors
- G-KL-A = Artist level single Gaussian und KL Distanz
- G-Mah = Artist-level Mahalaobis distance

(Mesaros, Virtanen, & Klapuri, 2007, S. 3 f.)

Logisch nachvollziehbar steigt die Performance der verschiedenen Ansätze dadurch, wenn der Gesang vorher von der instrumentalen Begleitung separiert wurde. Auch wird erkenntlich, dass der GMM-KL-A Klassifikator am robustesten bei nicht separierten Daten ist, vor allem bei realistischen Zuständen, bei denen der Gesang nicht weitaus lauter als die instrumentale Begleitung ist. Also ein SAR von -5dB bis 0dB vorliegt. Dies wird nochmal eindeutiger bei der Betrachtung von Abbildung 6 der zuwachsenden Performance der am besten performanten Ansätze (Mesaros, Virtanen, & Klapuri, 2007, S. 3 f.).

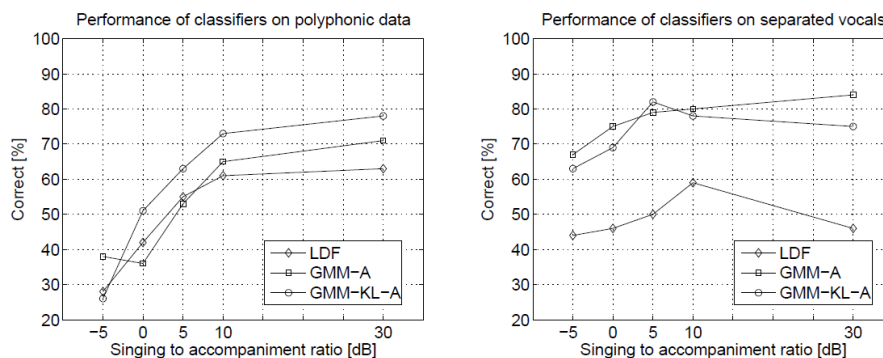


Abbildung 5 LDF als Basisvergleich und die zwei besten Klassifikatoren im Vergleich (Mesaros, Virtanen, & Klapuri, 2007, S. 4)

5 Fazit

Die in dieser Seminararbeit vorgestellten Ansätze zur Erkennung von Eigenschaften in einem Musikstück, sei es die Erkennung von Akkorden, des Genres oder gar des Sängers, zeigen teilweise ganz verschiedene Herangehensweisen. Vor allem bereiten Sie den Weg für den nächsten Schritt in der Künstlichen Intelligenz und Musik vor. Nämlich die Musikerschaffung, um beispielsweise Musik zu generieren die einem gewissen Genre entsprechen soll, muss natürlich bekannt sein muss bekannt sein wodurch sich jenes auszeichnet um genau für diesen ein neue Musikstück zu generieren

Literaturverzeichnis

- Basili, R., Serafini, A., & Stellato, A. (2004). *Classification of Musical Genre: a machine learning approach*. Rom: Universität Pompeu Fabra.
- Kaiser-Kaplaner, J. (kein Datum). *Musiklehre.at*. Abgerufen am 20. 02 2017 von http://www.musiklehre.at/7_001.htm
- Meinard, M. (2015). *Fundamentals of Music Processing*. Springer .
- Mesaros, A., Virtanen, T., & Klapuri, A. (2007). *SINGER IDENTIFICATION IN POLYPHONIC MUSIC USING VOCAL SEPARATION AND PATTERN RECOGNITION METHODS*. Austrian Computer Society.
- Newmarch, J. (2017). *Linux Sound Programming*. Apress.

Eidesstattliche Erklärung

Ich erkläre hiermit, dass ich die vorstehende Seminararbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ort/Datum

Unterschrift