

Inhaltsbasierte Musikempfehlung mit Convolutional Neuronalen Netzwerken

WEIDHAS PHILIPP

Matr.nr: 123456

philipp.weidhas@st.oth-regensburg.de

WILDGRUBER MARKUS

Matr.nr: 123456

markus.wildgruber@stud.oth-regensburg.de

Zusammenfassung

Ein wachsender Streamingkonsum und die immer größer werdenden Musikauswahl der Streamingplattformen, erfordern bessere automatisierte Musikempfehlungssysteme. In dieser Arbeit werden Grundlagen der automatisierten Musikempfehlung und deren klassische Ansätze erläutert. Aufbauend auf diese wird, das Maschinellen Lernen, mit neuronalen Netzen, als nächster Schritt zu einem verbessertem Empfehlungsergebniss betrachtet und ein Vergleich zweier verschiedener Methoden gezogen.

1. EINLEITUNG

Im ersten Halbjahr des Jahres 2017 wurden 62% der Einnahmen der amerikanischen Musikindustrie durch Streaming Plattformen¹ erzielt. Im Vergleich zum Vorjahr erhöhten sich dadurch die Einnahmen um 48% auf 2.5\$ Milliarden [1]. Dieser Erfolg basiert nicht nur auf einer guten Verfügbarkeit der Lieder und einem günstigen Preis, sondern auch aufgrund automatischen Musikempfehlungsdiensten, welche dem Nutzer ein angenehmeres Konsumverhalten ermöglichen.

Obwohl Empfehlungsdienste in den letzten Jahren viel erforscht wurden, ist das Problem der Musikempfehlung sehr komplex. Neben einer großen Anzahl an verschiedenen Stilen und Genres, beeinflussen sowohl soziales und

geographisches Umfeld, sowie der aktuelle Gemütszustand die Vorliebe eines Hörers. Diese Aspekte müssen, in einem Musikvorschlag, berücksichtigt werden. [2]

Neben der Empfehlung bestimmter Lieder sollen Empfehlungssysteme zusätzlich noch das Cold-Start Problem, sowohl bei einem Benutzer², als auch bei einem Lied³ überwinden. Das Cold-Start Problem besteht darin, dass noch keine Bewertungen für ein Lied vorliegen, wodurch es auch nicht vorgeschlagen werden kann. Das selbe Problem gibt es bei einem neuen Benutzer: Diesem kann kein adäquater Vorschlag gemacht werden, da es an Information mangelt, welche Art von Musik ihm gefällt. [3]

Mit Hilfe von Convolutional Neuronalen Netzwerken (CNN), können noch nicht alle Probleme überwunden werden. Sowohl das NSP als auch die Empfehlungsrate werden, durch die Verwendung eines CNN als Empfehlungssystem, verbessert.

Der weitere Verlauf der wissenschaftlichen Arbeit ist wie im Folgenden organisiert. Im 2. Abschnitt werden die Grundlagen der Musikempfehlung vorgestellt. Im 3. Kapitel wird der Aufbau von Neuronale Netze, soweit zwei Neuronale Netze für inhaltsbasierte Musikempfehlung beschrieben. Diese Arbeit, schließt im letzten Abschnitt, mit einem Vergleich der

¹wie Spotify, Apple Music, Pandora etc.

²New User Problem (NUP)

³New Song Problem (NSP)

Ergebnisse der beiden Modelle.

2. GRUNDLAGEN DER MUSIKEMPFEHLUNG

In dem Gebiet der Musik Information Retrieval (MIR), gibt es vier Kategorien [4], die einen Einfluss, auf die Wahrnehmung, von ähnlicher Musik haben.

Musikmerkmale sind Eigenschaften, welche aus dem Audiosignal eines Liedes extrahiert werden. Dazu zählen Aspekte wie der Rhythmus, die Melodie, die Harmonie oder die Stimmung eines Stückes.

Als *Musikkontext* versteht man alle Aspekte, die nicht aus dem Audiosignal abgeleitet werden, sondern Informationen die über ein Musikstück bekannt sind. Beispielsweise Metadaten wie der Titel eines Lieds, das Genre, Name des Künstlers oder das Erscheinungsjahr.

Die *Benutzereigenschaften* beziehen sich auf Persönlichkeitsmerkmale, wie Geschmack, musikalisches Wissen und Erfahrung oder den demographischen Hintergrund.

Im Unterschied dazu steht der *Benutzerkontext*, der sich auf die aktuelle Situation des Hörers bezieht. Dabei wird er durch seine Umgebung, seiner Stimmung oder der aktuellen Aktivität beeinflusst. [5]

Es gibt verschiedene Methoden, die in Musikempfehlungssystemen verwendet werden: kollaboratives, merkmalsbasiertes, kontextbasiertes Filtern und die hybride Methode. Diese werden genutzt, um Informationen aus den genannten Eigenschaften zu gewinnen, um sie für Empfehlungen an den Nutzer zu verarbeiten. [6]

2.1 Mel Frequency Cepstral Coefficients

Die Mel Frequency Cepstral Koeffizienten werden zur Analyse von Musikstücken verwen-

det und um Metadaten zuordnen zu können. Durch dieses Verfahren können die Tonhöhen, getrennt von der Sprache betrachtet werden. TODO genauer erklären? Todo

2.2 Bag of Words

BOW stellt ein klassisches Modell in der MIR da. Es stammt aus dem Feld der Textanalyse und wird dort Beispielsweise verwendet, um Dokumente automatisiert klassifizieren zu können. In diesem Modell wird ein Text als Ansammlung von Wörtern gesehen. Diese Wörter werden gezählt und aufsummiert wie häufig das selbe Wort in einem Text auftritt. Über diese Häufung von Wörtern kann eine Klassifizierung dieses Textes erfolgen.[7] In MIR wird dieses Modell in abgewandelter Form ebenfalls verwendet. Musikstücke werden mit Audiofeatures beschrieben. Abhängig davon wie häufig ein bestimmter Feature auf ein Lied zutrifft wird dies summiert. Für eine Klasse wird definiert welche Features für diese von spezifischer Natur sind, danach können die Lieder zugeordnet werden [8].

2.3 Kollaborativer Filter

Kollaboratives Filtern prognostiziert Vorlieben eines Hörers, indem es aus unterschiedlichen Benutzer-Lied-Verhältnissen lernt. Es basiert auf der Annahme, dass Verhalten und Bewertungen andere Nutzer, auf eine vernünftige Vorhersage, für den aktiven Benutzer, schließen lassen [3]. Durch explizite⁴ und implizite⁵ Rückmeldung eines Hörers an das Empfehlungssystem, empfiehlt dieses neue Lieder, indem es Gemeinsamkeiten, auf Basis seiner Bewertungen, mit dem Nutzungsverhalten anderer Anwender, der gleichen Plattform, vergleicht [9].

In der praktischen Umsetzung bedeutet dies: hört ein Anwender ein bestimmtes Musikstück.

⁴ Bewertungen eines Nutzers

⁵ Beobachten des Konsumverhalten

Dann werden ihm, von der Empfehlungsplattform, Lieder vorgeschlagen welche andere Nutzer, die ebenfalls dieses Lied hörten, hören. Dieses Verfahren geht davon aus, dass durch die Verbindung der Lieder durch vorhergehende Aufrufe eine gute Aussage darüber getroffen werden kann wie gut diese Stücke zusammen passen. Werden Lieder häufig nacheinander gehört, wird diese Verbindung höher bewertet und die Empfehlung häufiger ausgesprochen. Auch wird das Verhalten und der Musikgeschmack des Kunden selbst durch ein System analysiert, um so über Ähnlichkeiten der Kundenpräferenzen mit derer anderer, diesen wiederum bessere Empfehlungen aussprechen zu können. So werden Lieder einem Musikstil zugeordnet und so zielgerichtet dem Nutzer nahegelegt.

Verschiedene Studien ([9][10]) zeigen, dass KF alternative Methoden in der Genauigkeit übertrifft, weshalb es nicht nur im Bereich der Musikempfehlung als die erfolgreichste gilt.

2.4 Merkmalbasierter Filter

Mittels des merkmalsbasierter Filtern werden Nutzern Musikstücke aufgrund aus Lieder gewonnener Informationen vorgeschlagen. Dies bedeutet im Detail dass aus den Musikstücken mittels verschiedenster Metriken die Audio Signale eines Liedes analysiert werden um Erkenntnisse über die Stimmung eines Musikstücks, die Frequenz oder Rhythmus zu erhalten. Auf Grund dieser Informationen können Stücke dem Konsumenten vorgeschlagen werden die einen gleichen oder sehr ähnlichen Inhalt bieten.

2.5 Kontextbasierter Filter

Kontextbasiertes Filtern verwendet Informationen um Lieder zu beschreiben und zu cha-

rakterisieren. Diese Informationen können Metainformationen wie etwa Genre, Emotionen, Tags und Labels sein. Auch Erscheinungsjahr, Künstler und Album werden hierfür zur Bewertung zu Rande gezogen. Es werden nach Ähnlichkeiten der gehörten Lieder des Nutzer mit noch ungehörten verglichen, auf Basis der dadurch entstandenen Bewertungen werden dem Nutzer Lieder empfohlen. [11]

2.6 Hybride Methoden

Bei hybriden Methoden werden kollaborative, merkmalsbasierte und kontextbasierter Filter miteinander verknüpft. Dadurch kann ein besseres Empfehlungsergebnis mit weniger Nachteilen der einzelnen Methode erzielt werden. Meistens wird ein kollaborativer Filter mit einem der beiden anderen kombiniert.

Als *gewichtet* wird eine hybride Methode bezeichnet, bei der Empfehlungsrate der einzelnen Methoden, durch eine Linearkombination zusammengerechnet wird. Das Ergebnis der Linearkombination stellt den Empfehlungswert eines Liedes dar. Durch unterschiedliche Gewichtung der Methoden, kann das Empfehlungsergebnis optimiert werden. Der *wechselnde* Ansatz benutzt ein bestimmtes Kriterium anhand dessen es, die Methode zur Vorschlagbestimmung, wechselt. Dies kann beispielsweise dann der Fall sein, wenn der erste Filter kein zuverlässiges Ergebnis⁶ liefert. Dann wechselt das System den Filter und kann ein besseres Empfehlungsergebnis bekommen. Bei *gemischten* hybriden Empfehlungen werden unterschiedliche Techniken⁷ miteinander vermischt. Dadurch kann für ein System mit inhaltsbasierten Filter, das Cold-Start Problem vermieden, werden.

Hybride Methoden können einige Nachteile von kollaborativen Filtern entfernen, allerdings stehen auch sie vor dem NUP. Dennoch sind hybride Methoden sehr beliebt, da Information

⁶semantische Unterschiede

⁷meist kollaborativ mit inhaltsbasiertem Filter

über einen neuen Benutzer, schnell herausgefunden⁸ werden oder durch Profilangaben, bereits nach der Registrierung, vorhanden sind. [12]

3. NEURONALE NETZE IN DER MIR

CNN sind durch das biologische Sehen inspiriert und konnten den ersten großen Erfolg, im Bereich der Bildklassifizierung [13], verzeichnen. Trotzdem werden CNN auch in verschiedenen Audibereich, wie der Spracherkennung [14], sowie in der MIR mehr genutzt und erforscht.

Erste Forschungen im Bereich der MIR nutzen CNNs, um die Aufgabe der Musikgenre-Klassifizierung [15] zu untersuchen. Die Ergebnisse⁹ zeigen, dass eine automatisierte Klassifizierung die herkömmliche Methode, MFCC, deutlich übertrifft. Das erste CNN für inhaltsbasierte Musikempfehlung [2], benutzt zunächst eine Matrix-Faktorisierung, um Merkmalsvektoren¹⁰, für alle Lieder, zu erhalten. Anschließend wird das Neuronale Netz für die Zuordnung, der Audio-Inhalte an die Merkmalsvektoren, genutzt. [6]

Im nachfolgenden Absatz werden die Schichten und das supervised¹¹ Training, eines CNN, beschrieben.

3.1 Convolutional Neuronale Netze

In einer CNN Architektur werden drei Haupttypen von Schichten/Ebenen verwendet: Convolutional Layer (CL), Pooling Layer (PL) und Fully-Connected Layer (FCL). Jede Schicht besteht aus einer Anzahl von Knoten, die die Eingabedaten der Ebene wieder spiegeln. Knoten einer Schicht sind nur mit Knoten der nächsten Ebene verbunden. Diese Verbindung wird

als Gewicht oder Parameter bezeichnet. Durch das Training von bekannten Daten und Ergebnissen werden diese Parameter automatisch angepasst. Anschließend ist das CNN fähig das Ergebnis unbekannter Daten zu errechnen.

3.1.1 Schichten eines Convolutional Neuronale Netzwerks

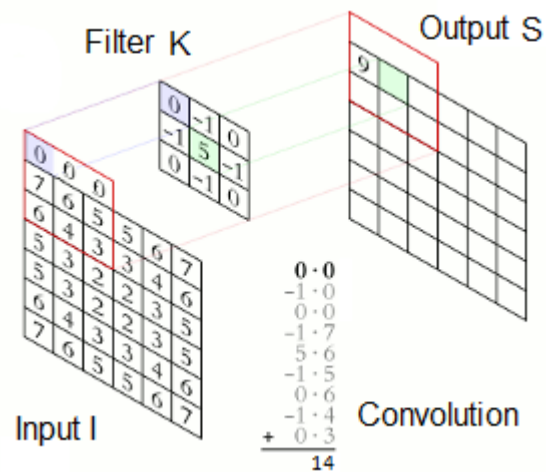


Abbildung 1: Faltung einer 6x6 Matrix (mit Zero-padding) und einem 3x3 Filter [16]

Convolutional Layer

In einem CL findet eine Faltung der Eingangsdaten, in Form einer Matrix und einem oder mehreren Filtern, statt. Ein Filter dient, beispielsweise zur Glättung oder zur Verkleinerung, der Daten. Eine Verkleinerung der Eingangsmatrix findet statt, wenn ein Filter ohne Zero-padding¹² verwendet wird. Die Parameter eines Filters werden zufällig initialisiert, können aber, mit Hilfe des Backpropagation Verfahrens (3.1.2) angepasst werden. Werden mehrere Filter auf die Eingangsdaten angewendet, ändert sich die Tiefe der gesamten

⁸Datamining

⁹richtige Klassifizierung

¹⁰Eigenschaften eines Musters in Vektordarstellung

¹¹Ausgabeergebnisse der Testdaten sind vorhanden

¹²Eine Matrix wird, bei einer Faltung, am Rand, um Nullen erweitert.

Bsp. aus einer 7x7 Matrix wird eine 9x9 Matrix

Ausgangsmatrix entsprechend der Anzahl der Filter. [17]

In Abbildung 1 ist die Eingabematrix I eine 6x6 Matrix und K ein 3x3 Filter. Die Ausgabematrix S wird an den Stellen (i,j) , durch die nachfolgende Gleichung, berechnet. Eine genauere Herleitung der Gleichung findet der Leser u. a. bei [18](328f).

$$S(i, j) = (I \star K)(i, j)$$

$$(I \star K)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n)$$

Pooling Layer

Ein PL wird zwischen zwei CL eingefügt. Ihre Funktion besteht darin, die Größe der Daten zu reduzieren und damit die Anzahl der Parameter für das nächste CL. Durch die Reduzierung wird die Berechnung des gesamten Netzwerkes beschleunigt. [17]

Ein PL wandelt die Ausgabe eines CL, durch eine statistische Zusammenfassung von nebeneinander liegenden Ausgängen, um. Verschiedene Methoden für ein PL sind: Max Pooling [19], eine Übergabe der größten Zahl in einem rechteckigen Umfeld; die Durchschnittsberechnung des Umfeldes oder ein gewichteter Durchschnitt, basierend auf der Entfernung eines zentralen Punktes [18](355).

Abbildung 2 zeigt einen 2x2 Max-Filter, der auf eine 4x4 Datenmatrix angewandt wird. Die Verschiebung oder Stride des Filters ist 2 dh. der Filter wird zunächst auf der y-Achse verschoben. Erreicht er dort das Ende, wird er um eine Stride auf der x-Achse verschoben und beginnt wieder mit der y-Verschiebung.

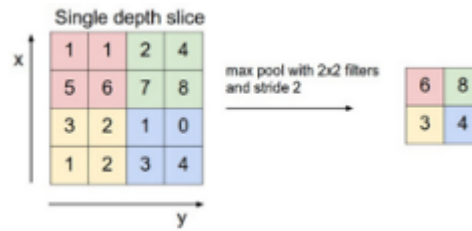


Abbildung 2: Maxpooling mit einem 2x2 Filter[17]

Fully-Connected Layer

Ein oder mehrere geschachtelte FCL dienen als Ausgabeschicht. Jeder Knoten in einer FCL, hat eine Verbindung, zu allen Knoten, der vorherigen Schicht. Abhängig von der Problemstellung, Klassifizierung oder Regression, ist die Anzahl der Ausgabeknoten unterschiedlich. Bei einer Klassifizierung entspricht die Anzahl, in der letzten FCL, der der Klassen, die das CNN unterscheiden soll. Die Ausgabe der FCL ist ein Vektor in der jeder Eintrag, die Wahrscheinlichkeit, der jeweiligen Klasse, spiegelt. Bei einer Regression gibt das FCL einen Realwert aus. [17]

3.1.2 Training

CNNs werden, durch die Backpropagation Methode, trainiert. Backpropagation basiert auf dem Gradientenverfahren, welches versucht für die Fehlerfunktion E , durch sukzessive Iteration der Parameter, ein globales Minimum zu finden, meistens aber nur ein lokales findet. Um das Minimum zu erreichen, werden die Werte der Gewichte w_{ij} , durch Verwendung der Kettenregel, der partiellen Ableitung, berechnet¹³:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} \frac{\partial net_j}{\partial w_{ij}}$$

Der neue Wert eines Parameters w , zum Zeit-

¹³Ausgabeparameter o , beliebige Anzahl an Knoten net zwischen Ausgabe und w

punkt t , lässt sich durch folgende Formel berechnen:

$$w(t) = w(t-1) + \eta * \frac{\partial E}{\partial w}$$

Die Lernrate η ist eine Konstante, die definiert werden muss.

Anhand der Kostenfunktion, kann die Parameteranpassung und damit der funktionale Wert eines CNN, mit reellen Zahlen, dargestellt und verglichen werden. Diese Funktion C lässt sich durch die Summe, über Trainingsbeispiele m und einer Fehlerfunktion E , hier die negativ conditional Log-Likelihood¹⁴, darstellen:

$$E(x, y) = -\log p(y|x)$$

$$C = \frac{1}{m} \sum_{i=1}^m E(x^i, y^i)$$

Das Training eines Netzes ist abgeschlossen, wenn die Kostenfunktion minimal ist bzw. in einem gegebenen Zeitraum, keine bessere gefunden wird. [18](80ff,129ff)

Anwendung eines CNN zur automatischen Musikempfehlung

Dieser Ansatz wird wie folgt umgesetzt: das Netz wird darauf trainiert latente Faktoren aus einzelnen Musikstücken zu generieren, welche für eine Empfehlung verwendet werden. Dieses Verfahren wird anschließend im Vergleich mit einem konventionellen Ansatz, welcher dem Bag-of-Words Prinzip folgt. Daraus folgend wird beurteilt, inwiefern mittel CNN Eigenschaften zur spezifizierung des Nutzergeschmacks generiert und ausgewertet werden können. In den folgenden Kapiteln wird auf diesen Ansatz nun genauer eingegangen.

3.1.3 Datenbasis

Als Datenbasis zur Umsetzung dieses neuen Ansatzes wurden Datensätze des Million Song

Dataset (MSD) Datensatzes verwendet. Dieser Datensatz verfügt über Metainformationen und bereits analysierter Audio Informationen von Liedern. Ebenso ist dieser öffentlich zugänglich und kann kostenlos heruntergeladen werden. Des weiteren stellt dieser derzeit die größte Forschungsdatenbasis im Gebiet der Musikanalyse dar. [20] Zwei Datensätze im Umfeld des MSD sind von besonderem Interesse, der Echo Nest Taste Profile Subset (ENTPS) Datensatz und der The Last.fm Datensatz. Die Problematiken mit diesen Datensätzen, waren zum einen eine schlechte Dokumentation der Generierung der Informationen und der Daten selbst, sowie das keine unverarbeiteten Musikquellen mitgeliefert werden. Das Problem der nicht vorhandenen Rohdaten konnte beseitigt werden, in dem für 99% des Datensatzes Musikschnipsel mit der Dauer von 29 Sekunden von 7digital.com bezogen wurden. Der ENTPS sticht durch seine Eigenschaft als größte, bereits mittels kollaborativen Filtern ausgewerteten, Informationsbasis hervor und bietet sich dadurch für eine Weiterverarbeitung mittels CNN an. Auch kann durch die Größe der Daten ein realitätsnaher Versuch unternommen werden. Durch die Verwendung dieses Datensatzes, ist zu jedem Lied, pro Nutzer, die genaue Anzahl gespeichert, wie häufig das Stück vom Nutzer angehört wurde. [2]

3.1.4 Weighted matrix factorization

Der ENTPS Datensatz enthält, die Abspielhäufigkeit eines Songs für jeden Nutzer, also eine implizite Bewertung des Nutzer für diese Lieder. Auch kann für ein Musikstück, für einen oder mehrere Nutzer, keine Bewertung vorliegen. Der Umstand dass ein Lied keine Wertung erhalten hat, kann mehrere Gründe haben, unter anderem dass ein Nutzer dieses Lied schlicht und ergreifen nicht kennt. Der Nutzer könn-

¹⁴Abgeleitet von Maximum Likelihood, Dichtefunktion für Maximafindung

te allerdings das Lied bereits kennen, aus anderen Quellen, es nicht mögen und deshalb diesen Titel nicht anhören. Dies führt zu zwei unterschiedlichen Szenarien auf Grundlage der gleichen Wertung. Um diesen Umstand richtig bewerten zu können, muss der Algorithmus flexibel sein. Deshalb kommt ein weighted matrix factorization (WMF) Algorithmus zum Einsatz, um die Informationen aus dem ENTPS trainieren zu können. Diese Idee ist angelehnt an den Versuch bei der Bewertung von Fernsehschows und deren automatisierter Empfehlung, dem Nutzer gegenüber, möglichst gute Ergebnisse zu erzielen. Es wurde gespeichert wie oft ein Nutzer ein Fernsehformat angesehen hat. Auch hier tritt der Fall auf, dass für Formate von einzelnen Nutzern keine Bewertungen vorlagen. Dies kann mehrere Gründe haben: der Nutzer kennt das Format nicht, eine Sendung, die er noch mehr schätzt, kommt zur gleichen Sendezeit oder er mag die Sendung nicht. Folglich gibt es auch hier eine Ausgangswertung und drei verschiedene Schlussfolgerungen sind möglich. [21]

Dieser, für implizite Bewertungen optimierte, Algorithmus ist wie folgend aufgebaut:

$$p_{ui} = I(r_{ui} > 0),$$

$$c_{ui} = 1 + \alpha \log(1 + e^{-1} r_{ui})$$

r_{ui} ist die Anzahl, wie häufig ein Lied i durch den Nutzer u gehört wurde. Für jedes dieser Datenpaare, wird eine Präferenz Variable p_{ui} definiert. Sie indiziert, ob der Nutzer das Lied jemals angehört hat, entspricht der Wert, der Variable, 1 dann wird ausgegangen, dass der Nutzer mag das Lied. Die zweite Variable c_{ui} , sie zeigt an, wie stark die Bewertung des Liedes in das Empfehlungssystem einfließen soll. Lieder mit höheren Spielraten, werden stärker berücksichtigt, der Wert von c_{ui} ist hoch, da man sich hier sicherer ist, dass der Nutzer, das Lied mag. Lieder mit geringen Spielraten wer-

den mit einem niedrigen Wert für c_{ui} bedacht. α und ϵ sind Hyperparameter.

Die Zielfunktion der WMF ist wie folgt gegeben:

$$\min_{x_*, y_*} \sum_{u, i} c_{ui} (p_{ui} - x_u^T y_i)^2 +$$

λ ist der Regularisations-Parameter. X_u stellt den Feature Vektor des Nutzers u dar. y_i Feature-Vektor für Musikstück i dar. Die Zielfunktion besteht aus einem L2 Regularisationsterm und einem Term über die mittlere Quadratische Abweichung für die bereits erläuterten Variablen. Die Summe im ersten Teil der Funktion beinhaltet jede mögliche Kombination von Nutzern und Liedern, deshalb kommt eine Methode zur Näherung der alternierenden kleinsten Quadrate zum Einsatz. Dieser Ansatz ist ebenfalls an das bereits genannte Verfahren, zur Empfehlung von Fernsehsendungen, [2]

3.1.5 Extraktion und Analyse möglicher Faktoren aus dem Audiosignal

Der Akt der Extraktion von Features auf Basis von Audiosignalen eines gegebenen Liedes, ist ein Regressionsproblem. Dies benötigt das Lernen einer Funktion, welche Zeitintervalle auf einen Vektor reeller Zahlen abbilden kann. Um dies zu lösen, wird ein neuer Ansatz verfolgt: die Nutzung eines CNNs. Die mittels des, bereits erläuterten, WMF Algorithmus gefundenen Vektoren auf Basis der vorhandenen Daten, werden eingesetzt, um das Empfehlungssystem zu trainieren. Es wurden insgesamt 3 definiert: Um das Netz gut trainieren zu können, ist es nötig, auf eine große Menge an Trainingsdaten zurückgreifen zu können. Dies wird mittels der MSD, welche bereits vorgestellt wurde, abgesichert. Um das Problem des Vanishing Gradient Problem zu verringern, kommen rektifi-

zierte Lineareinheiten zum Einsatz. Auch die Geschwindigkeit des Trainings an sich wird als kritisch betrachtet. Um ein schnelles Lernen zu ermöglichen, wurde der Lernvorgang parallelisiert und mittels GPU Unterstützung zur Berechnung optimiert. Eine GPU ist im Vergleich zu einer CPU in der Lage, in der gleichen Zeit, deutlich mehr Operationen durchzuführen, welche zum trainieren eines Neuronalen Netzes nötig sind.

Das Training des CNN erfolgte in folgenden Schritten: Als erstes wurde aus den Audiosignalen Zeit-Frequenz-Details extrahiert, um diese als Input für das Netz zu verwenden. Dies erfolgte durch logarithmisch komprimierten Mel-Spektrogramme welche aus 128 Komponenten bestehen. Das Netz wurde mit 3 Sekundenlangen Audioschnipseln trainiert, welche zufällig aus den Songs entnommen wurden, dies sorgte für eine zusätzliche Beschleunigung der Trainingsgeschwindigkeit. Um die Features für den gesamten Song zu ermitteln wurden die Faktoren von aufeinanderfolgenden 3-Sekunden-Musikschnipseln gemittelt.

Da als Ergebnis der Zielfunktion die Faktoren als reelle Werte benötigt werden, muss der Quadratische Mittlere Fehler der vorhersagen reduziert werden. Alternativ wurde auch eine Verbesserung der Genauigkeit der WMF Zielfunktion aufgezeigt. [2]

3.1.6 Versuchs und Durchführung vergleichender Tests

Um die Leistungsfähigkeit des neuen Ansatzes auf Basis eines neuronalen Netzes darlegen zu können, wurde ein Experiment mit diesem durchgeführt. Auch wurde ein klassisches Modell basierend auf dem bereits erläuterten BoW Ansatz umgesetzt, um die Ergebnisse vergleichen zu können. Um die Qualität der Empfehlungen aber auch die extrahierten Faktoren an sich untersuchen zu können, wurden folgende Schritte unternommen.

Um die Vielseitigkeit der extrahierten

Audio-Features zu untersuchen, wurden diese, mit Tags für Lieder aus einem Tag-Prediktion-Verfahren verglichen. Tags können Songs beschreiben, unter anderem Genre, Instrumentierung, Tempo, Stimmung und Erscheinungsjahr. Dieser Vergleich wurde auf Basis aller 9,330 Lieder des Last.fm Datensatzes erstellt und die 50 Beliebtesten Tags für jedes Lied extrahiert. Der Vergleich ergab dass die Vektoren einen erhöhte Empfehlungsgenauigkeit erbringen.

Um quantitativ zu beurteilen, wie gut die aus den Audioquellen der Lieder den Liedern extrahiert Faktoren werden können wurden die Vorhersagen verwendet um damit diesem für ein Empfehlungssystem umzusetzen. Für jeden Nutzer u und jeden Song i der Datenbasis wurde eine Wertung $x_{u,i}^T y_i$ errechnet und das Lied mit dem höchsten Wert als erstes vorgeschlagen. Zum Vergleich wurden ebenfalls eine Metrik auf Basis von Liederähnlichkeiten mittels Bag-of-Words System gelernt um Vorschlagsbewertung zu erhalten. In diesem Modell wurden alle Wertungen für einen gegebenen Nutzer in eine Durchschnitts-Ähnlichkeits-Wertung umgerechnet basierend auf alle Lieder welche dieser Nutzer bereits hörte.

Um die Vektoren zur Empfehlung generieren zu können, wurde jeweils folgende Ansätze verfolgt: Eine Lineare Regression welche auf den Bag-Of-Words Ansatz trainiert wurde. Ein mehrlagigen Perzeptron, ein vereinfachtes künstliches Netz, trainiert auf ebenfalls das gleiche BoW-Modell. [22] Ein CNN, welches auf log-skalierten Mel-Spektrogrammen trainiert wurde, um den den Mittleren quadratischen Fehler der Vorhersagen zu reduzieren. Sowie das gleiche CNN trainiert um den Empfehlungen aus dem WMF zu verbessern.

In ersten Experimenten wurden nur Teilmengen der vorhanden Datensätze genutzt. Betrachtet man die Durchschnittsgenauigkeit der Empfehlungssystem ergibt sich folgendes Bild: Die lernenden Ansätze sind Leistungsfähiger als die Lineare Regression. Selbst der Ansatz basieren auf dem Perzeptron erweist sich als

besser. Die beiden auf CNN basierenden Modelle sind deutlich besser im Vergleich zu den beiden erstgenannten Modellen. Die vergrößerung der Datenbasis für weitere Experimente lässt die Leistungsfähigkeit und den Leistungsvorsprung der beiden auf CNN basierenden Ansätze noch größer werden.

Eine Bewertung der Qualität der gefundenen Vektoren, kann nicht nur durch Metriken bestimmt werden. Ein Vergleich der mittels CNN vorhergesagten Vektoren empfohlenen Liedern einerseits und empfehlungen eines 50-dimensionalen WMFs andererseits ergaben folgendes: die empfohlenen Lieder sind zumeist unterschiedlich, nur wenige Teilmengen vorhanden. Allerdings erbringen beide Modelle ein gutes Ergebnis und die mittel CNN-Modell vorgeschlagenen Auswahl an Liedern ist etwas abwechslungsreicher was für ein Empfehlungssystem als Vorteil zu sehen ist.

3.2 Hybride Musikempfehlung mit einem Neuronalen Netzwerk

Im Unterschied, zu der davor dargestellten Forschung (3.2), wird nun ein Deep Belief Netzwerk (DBN) verwendet, um ein hybrides inhaltsbasiertes Musikempfehlungssystem zu entwickeln. Im Gegensatz zu einem CNN, nutzt ein DBN ein so genanntes Pretraining¹⁵ als Parameterinitialisierung. Bisherige inhaltsbasiertes Systeme verfolgen typischerweise einem zweistufigen Ansatz: Zunächst extrahieren sie aus Audioinhalte den MFCC Koeffizienten; anschließend prognostizieren sie Musikpräferenzen, eines Nutzers. Das nachfolgende Modell führt diese beiden Schritte simultan und automatisch aus. [6]

Das hybride Modell basiert auf einem hierarchisch linearen Modell mit einem Deep Belief Netzwerk (HLDBN), das zunächst erläutert wird, um anschließend die Funktions-

weise, des hybriden Systems, darzustellen.

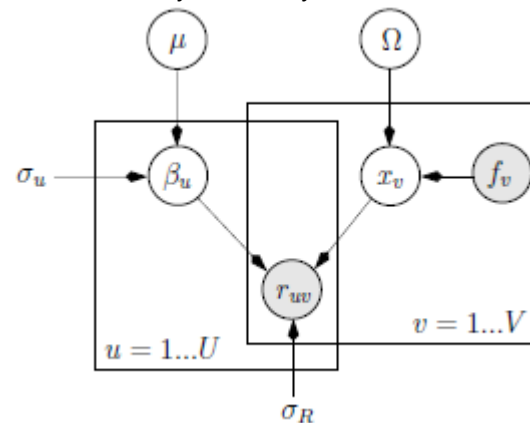


Abbildung 3: Hierarchisches lineares Modell eines Deep Belief Netzwerks [6]

3.2.1 Hierarchisches lineares Modell mit einem Deep Belief Netzwerk

Das in Abbildung 3 gezeigte Modell ist wie folgt definiert: f_v sind Musikmerkmale eines Liedes v , die durch den Merkmalsvektor x_v automatisch errechnet werden. Die bevorzugte Musik eines Benutzers u wird als Vektor β_u bezeichnet. Ω bezeichnet die Parameter, die das DBN lernt. Die Bewertung r_{uv} , die ein Nutzer einem Lied v gibt, ist ein Skalarprodukt von x_v und β_u . Durch σ_R wird die Varianz aller Bewertungen des Nutzers betrachtet. μ repräsentiert den allgemeinen Musikgeschmack aller Benutzer, wobei σ_u die Varianz des einzelnen Nutzers definiert. Alle Benutzer und Lieder Paare werden als I bezeichnet. Für eine Regularisierung der Werte wird die Gaußsche Normalverteilung \mathcal{N} verwendet.¹⁶ Das Modell ist wie folgt formuliert:

$$r_{uv} \sim \mathcal{N}(\beta_u' x_v, \sigma_R^2)$$

$$\beta_u \sim \mathcal{N}(\mu, \sigma_u^2 I)$$

$$x_v = \text{DBN}(f_v; \Omega)$$

¹⁵Parameter von Model A wird für das neue Model B verwendet

¹⁶ $\mathcal{N}(a,b)$ ist die Normalverteilung mit Mittelwert a und Varianz b . $x \sim p$ zeigt, dass x die Verteilung p erfüllt

Für das Training des Systems wird im Unterschied zu einem CNN zunächst ein unsupervised¹⁷ Training durchgeführt um die Knoten zu initialisieren. Anschließend findet ein supervised Training zur Optimierung der Parameter statt. Als Optimierungsmethode wird das stochastische Mini-Batch¹⁸ Verfahren mit Backpropagation genutzt, um ein Overfitting¹⁹ des Modells zu vermeiden. Nach der Lernphase kann die Bewertung r_{xv} eines Benutzers u über ein Lied v geschätzt werden, wodurch diesem neue Lieder empfohlen werden können. [6]

3.2.2 Hybrides Modell mit einem Deep Belief Netzwerk

Basiert auf dem HLDBM wird das in Abbildung 4 gezeigte Modell um einen kollaborative Filter erweitert, um eine noch bessere Empfehlungsrate zu erhalten.

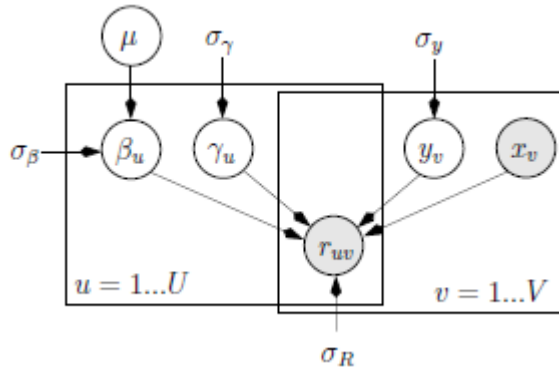


Abbildung 4: Hybrides Empfehlungs Modell [6]

Die Musikmerkmale eines Liedes x_v wird wie in (3.4.1) berechnet. Die bevorzugte Musik eines Nutzers wird als β_u bezeichnet. Der kollaborative Teil: γ_u stellt einen Vektor über alle Benutzer u dar; γ_v den Vektor über alle

Lieder v . β_u , γ_u und y_v werden gemeinsam, durch das Neuronale Netzwerk, anhand der Trainingsdaten gelernt. Die Empfehlungsrate r_{uv} ergibt sich aus der Summe der Skalarmultiplikationen $\gamma'_u y_v$ und $\beta'_u x_v$. Die A-priori-Wahrscheinlichkeit²⁰ werden durch die folgenden Formeln definiert:

$$r_{xv} | \beta_u, x_v, \gamma_u, y_v, \sigma_R \sim \mathcal{N}(\beta'_u x_v + \gamma'_u y_v, \sigma_R^2)$$

$$\beta_u | \sigma_\beta \sim \mathcal{N}(\mu, |\sigma_\beta^2 I)$$

$$\gamma_u | \sigma_\gamma \sim \mathcal{N}(0, |\sigma_\gamma^2 I)$$

$$y_v | \sigma_y \sim \mathcal{N}(0, |\sigma_y^2 I)$$

Die Fehler Funktion des Modells wird auch mit Hilfe der Backpropagation minimiert. Um Zeit zu sparen, können die Werte für γ_u und y_v durch eine PMF berechnet und als Initialwerte verwendet werden. [6]

Um Informationen aus einem Lied, als Eingabeparameter für das DBN, nutzen zu können, muss dieses in einen Vektor umgewandelt werden. Dies geschieht in verschiedenen Schritten. In [6] werden alle Lieder in ein .wav Dateiformat, mit einer 8 kHz Abtastrate und 16 Bit Tiefe, umgewandelt. Aus den einzelnen Liedern wird ein 5 Sekunden Abschnitt genutzt, der in ein 166x120 Spektrogramm umgewandelt wird. Anschließend wird eine Hauptkomponentenanalyse²¹ durchgeführt, wodurch sich ein Vektor für den Liedabschnitt ergibt.

4. VERGLEICH DER VORGESTELLTEN MODELLE

Abschließend wird nun ein Vergleich zwischen dem vorgestellten CNN Ansatz einerseits und dem danach folgendem DBN gezogen. In Versuchen [2] wurde festgestellt dass das CNN

¹⁷Ausgabeergebnisse der Testdaten sind nicht vorhanden

¹⁸nur ein kleiner Teil der vorhandenen Daten wird trainiert

¹⁹Spezialisierung

²⁰Anfangswahrscheinlichkeit

²¹Zusammenhang der Darstellung einer Menge an Variablen durch wenige Faktoren

Model einem BOW System überlegen ist und eine bessere Empfehlungsrate erreicht. Das anschließend erläuterte Versuchsmodell, welches mittels DBN einen hybriden Methode verfolgt, konnte im direkten Vergleich zu CNN eine nochmals verbesserte Empfehlungsgenauigkeit erreichen. Vergleichende Versuchsreihen [6] haben folgendes festgestellt: ein nicht hybrider Ansatz, welcher allein auf Training der beiden Netze basiert, ergab, dass das vorgestellte HLDBN Modell genauere Ergebnisse lieferte als das vorgestellte CNN. Die Integration der beiden Modelle in einen hybriden Aufbau ergab wiederum ebenso dass der Einsatz des HLDBN Netzes zu genaueren Empfehlungs-raten führte als die Verknüpfung von CNN und CF. Die Empfehlungs-raten für bereits bekannte Lieder, konnten somit im Vergleich zu klassischen Ansätzen verbessert werden. Auch das bereits eingeführte Problem des NSP konnte durch den Einsatz Neuronaler Netze gelöst werden. Sowohl der Einsatz des CNN als auch der des HLDBN Netzes führen hierbei zum Erfolg. Das ebenfalls bereits vorgestellte NUP konnte dagegen nicht gelöst werden, da in diesem Modellen derzeit keine Möglichkeit besteht aus einem, dem System unbekannten, Nutzer Merkmale für eine Vektordarstellung zu generieren. Während aus einem Lied, durch die vorgestellten Modelle Audiofeatures extrahiert und dem System zugeführt werden können. Sowohl der in Kapitel 3.2 vorgestellte Ansatz mittels eines CNNs, sowie der in Kapitel 3.3 erläuterte Ansatz mit Einsatz eines hybriden DBN Netzwerkes eine gute Möglichkeit zur automatisierten Musikempfehlung. Beide Verfahren haben in Rahmen von Versuchen bewiesen, dass sie sowohl zuverlässig sind und durch bessere Empfehlungs-raten einem klassischen Ansatz überlegen sind.

LITERATUR

- [1] Joshua P. Friedlander. News and notes on 2017 mid-year riaa revenue statistics. RIAA, 2017.
- [2] Aäron van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. *Advances in Neural Information Processing Systems* 26, 2013.
- [3] Òscar Celma. *Music Recommendation and Discovery - The Long Tail, Long Tail, and Long Play in the Digital Music Space*. Springer, 2010.
- [4] Markus Schedl, Arthur Flexer, and Julián Urbano. *The neglected user in music information retrieval research*, volume 36. Springer, 2013.
- [5] Peter Knees and Markus Schedl. *Music Similarity and Retrieval*, volume 41. Springer, 2016.
- [6] Xinixi Wang and Ye Wang. Improving content-based and hybrid music recommendation using deep learning. *Proceedings of the ACM International Conference on Multimedia*, pages 627–636, 2014.
- [7] Deepu S., Pethuru Raj, and S. Rajaraajewari. A framework for text analytics using the bag of words (bow) model for prediction. *International Journal of Advanced Networking and Applications (IJANA)*, 2013.
- [8] Chang-Hsing Lee, Hwai-San Lin, and Ling-Hwei Chen. Music classification using the bag of words model of modulation spectral features. *15th International Symposium on Communications and Information Technologies (ISCIT)*, 2015.
- [9] B. McFee, T. Bertin-Mahieux, D. P. W. Ellis, and G. R. G. Lanckriet. The million song dataset challenge. *21st International Conference Companion on World Wide Web*, pages 909–916, 2012.
- [10] Luke Barrington, Reid Oda, and Gert Lanckriet. Smarter than genius? human evaluation of music recommender systems. *Proceedings of the 10th International Society*

- for Music Information Retrieval Conference, pages 357–362, 2009.
- [11] Juuso Kaitila. A content-based music recommender system. Master thesis, University of Tampere, 2017.
- [12] Robin Burke. Hybrid recommender systems: Survey and experiments. *User Model. User-Adapt. Interact.*, pages 331–370, 2002.
- [13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [14] Alex Graves, Abdel-Tahman Mohamed, and Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks. *Acoustics, Speech and Signal Processing, IEEE International Conference on*, pages 6645–6649, 2013.
- [15] Honglak Lee, Yan Largman, Peter Pham, and Andrew Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. *Advances in Neural Information Processing Systems*, 2009.
- [16] wikipedia, 2017. https://de.wikipedia.org/wiki/Convolutional_Neural_Network#/media/File:3D_Convolution_Animation.gif.
- [17] Andrej Karpathy. *Convolutional Neural Networks for Visual Recognition*. Stanford University, 2017. <https://github.com/cs231n/cs231n.github.io/blob/master/convolutional-networks.md#conv>.
- [18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [19] Zhou Y. and Chellappa R. Computation of optical flow using a neural network. *IEEE International Conference*, 71–78, 1988.
- [20] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. *12th International Society for Music Information Retrieval Conference (ISMIR 2011)*.
- [21] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, 2008.
- [22] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review Vol. 65, No. 6*, 1958.