Distanzberechnung zur Bestimmung von Ähnlichkeiten im Filmkontext

Markus Deutschl
Dept. of MultimediaTechnology
Fachhochschule Salzburg
Puch bei Hallein, Austria
mdeutschl.mmt-m2012@fh-salzburg.ac.at

Abstract — Es gibt bereits zahlreiche Ansätze, wie Filmvorschläge von Recommender Systems (RS) berechnet werden können. Sie alle erfordern jedoch Benutzer- und Trainingsdaten. Diese Arbeit stellt eine Distanzfunktion vor, die die Ähnlichkeiten von Filmen fast ausschließlich anhand ihrer inhärenten Eigenschaften bestimmt und somit Vorschläge ohne Benutzerinteraktion ermöglicht.

Keywords — Distanzberechnung, Recommender, Dimensionsreduktion, Film, Ähnlichkeit.

I. Einführung

Die Präsenz eines Recommender Systems (RS) gehört im modernen Web bereits zum Standardrepertoir von Verkaufs- und Informationswebsites. Dieses Werkzeug erlaubt es, den Benutzern jener Websites, Vorschläge aufgrund ihrer Präferenzen anzubieten. Durch diese Vorschläge erwächst den Benutzerinnen und Benutzern ein Mehrwert, da sie auf Produkte bzw. Informationen aufmerksam werden, die sie vorher noch nicht kannten, was sich in längerer Aufenthaltszeit auf Websites und in höheren Verkaufszahlen für die Websitebetreiber niederschlägt. Gerade im Bereich von Filmen und Serien ist das Interesse an RS sehr hoch, da das riesige Angebot von Benutzerinnen und Benutzern nicht überblickt werden kann und diese auch gerne auf Neuerscheinungen bzw. populäre Werke ihres Interesses hingewiesen werden wollen. Dieses Interesse der Industrie wurde durch den *Netflix prize*[1] unterstrichen, welcher dem Team, das den *Netflix*-eigenen Vorschlagsalgorithmus um 10% verbessern konnte, eine Million Dollar Preisgeld bot.

Die Berechnung von Filmvorschlägen auf Basis von Benutzerinnen- und Benutzerinteressen ist ein sehr komplexes Fachgebiet, auf dem bereits einiges an Forschung betrieben wurde.

Neben den weithin bekannten Ansätzen *Content-based* RS [z.B. (Pazzani und Billsus 2007)], welche Vorschläge anhand des Benutzerprofils errechnen, sowie *Collaborative filtering* RS [z.B. (Herlocker, Konstan und Riedl 2000)], die die Vorschläge aufgrund von ähnlichen Benutzerinnen und Benutzern generieren, gibt es mittlerweile auch zahlreiche andere Ansätze zu diesem Problem.

In dieser Arbeit stelle ich eine Distanzfunktion vor, die Ähnlichkeiten zwischen Filmen aufgrund der Filmeigenschaften bestimmt. Aus diesen Distanzen kann in weiterer Folge ein Graph konstruiert werden, aus dem mit einer Breitensuche Vorschläge generiert werden können. Dieser Ansatz wurde gewählt, um Vorschläge für das RS von *MovLib*[2] generieren zu können, dessen Filmdaten in einer graphenorientierten Datenbank abgelegt werden, um solche Breitensuchen möglichst effizient durchführen zu können. Durch die Benutzerunabhängigkeit der Distanzberechnung können auch Vorschläge ohne Nutzerdaten generiert werden, was wiederum das „Kaltstartproblem“ anderer RS und auch die Notwendigkeit von Trainingsdaten eliminiert.

In der Sektion V werden Überlegungen geschildert, wie die Distanzfunktion verbessert und deren Ergebnisse verifiziert werden können.

[1] http://www.netflixprize.com/

[2] *MovLib* ist eine freie und quelloffene Filminformationsseite. http://movlib.org

II. Verwandte Arbeiten

In der Vergangenheit haben sich bereits einige Forscherinnen und Forscher mit dem Themengebiet von Filmvorschlägen auseinandergesetzt. Es sollen hier nun einige wenige Ansätze aufgelistet werden, die in direktem oder indirektem Bezug auf diese Arbeit stehen, was aber keinesfalls wertend aufzufassen ist.

Die Arbeiten von Perny & Zucker (Perny und Zucker 2001), Li & Yamada (Li und Yamada 2004), sowie Das & ter Horst (Das und ter Horst 1998) gehen darauf ein, wie Vorschläge anhand von Nutzerpräferenzen generiert werden können. Dies umfasst Bewertungen, sowie latente Interessen von Benutzerinnen und Benutzern.

Symeonidis, Nanopoulos & Manolopoulos (Symeonidis, Nanopoulos und Manolopoulos 2009) und Adomavicius, Sankaranarayanan, Sen & Tuzhilin (Adomavicius, et al. 2005) wiederum beziehen Kontextinformationen über Userdaten und Items in ihre Vorschlagsberechnungen mit ein.

Des Weiteren sind noch die Arbeiten von Mak, Koprinska & Poon (Mak, Koprinska und Poon 2003), sowie Golbeck & Hendler (Golbeck und Hendler 2006) zu nennen, welche Textkategorisierung von Filmsynopsen bzw. vertrauensbasierte Daten von social networks in ihren RS verwenden.

Die Ähnlichkeitsberechnung dieser Arbeit unterscheidet sich jedoch grundlegend von zuvor genannten Ansätzen, da rein nur die Eigenschaften und Domänenwissen von Filmen in sie miteinbezogen werden.

III. Ähnlichkeitsbestimmung

Für die Formulierung einer paarweisen Distanzfunktion mussten zu allererst die wichtigsten Eigenschaften von Filmen ermittelt werden, die einen Ähnlichkeitsvergleich ermöglichen. Dazu wurde auf domänenspezifisches Wissen zurückgegriffen, um diese Faktoren zu identifizieren.

Als grundlegende Unterscheidungsbasis von Filmen wurden die Filmgenres definiert, da diese eine grobe Kategorisierung der Inhalte zulassen. Da die Definition der Genres z.T. stark in Art und Anzahl variiert, wurden die 20 Basisgenres des *MovieLens*-Datasets[3] verwendet.

Einen weiteren einflussreichen Faktor stellt die Gesamtbewertung eines Films dar. Werden Filme ähnlich bewertet, kann davon ausgegangen werden, dass sie sich ähneln, da Benutzerinnen und Benutzer diese ähnlich gut einstufen. Diese Gesamtbewertung wurde vom *MovieLens*-Dataset extrahiert, welches diese wiederum von der Website *Rotten tomatoes*[4] entnahm. Die Bewertungsskala liegt hier bei 1-5, wobei 5 die beste Bewertung darstellt.

Der Regisseur eines Films bestimmt zu einem großen Teil den Stil desselbigen. Deshalb müssen Filme, die vom selben Regisseur stammen, auch als ähnlich angesehen werden. Natürlich existieren auch Filme die sehr unterschiedlich sind, obwohl sie vom gleichen Regisseur gedreht wurden. Diese Spezialfälle sollen aber durch die Verwendung aller anderen Ähnlichkeitsfaktoren abgefangen werden. Weiteren Einfluss auf den Stil von Filmen üben Herkunftsland und Erscheinungsjahr aus. Diese beiden Faktoren haben aber nur begrenzte Wirkung im Vergleich zu den anderen und sollten die Ähnlichkeit deshalb nicht zu stark beeinflussen.

Das genaue Zusammenspiel und der Einfluss der einzelnen Faktoren werden gemeinsam mit einer Beschreibung der finalen Distanzfunktion im nächsten Abschnitt dargelegt.

IV. Distanzfunktion

Die Ausgangsbasis der Distanzberechnung stellen die Genres eines Films dar. Dafür wurden die Genres in einen 20-dimensionalen Vektor gebracht, welcher den Wert 100 beinhaltet, wenn ein Film einem Genre zugeordnet werden kann, bzw. 0 wenn das nicht der Fall ist. Die paarweise Genredistanz (GD) wird danach mittels der euklidischen Distanzfunktion berechnet.

Der Einfluss der Gesamtbewertung wurde durch den Bewertungsfaktor (BF) abgebildet, der die normalisierte Bewertungsdifferenz abbildet. Die Variablen f_1 und f_2 stehen, wie bei allen Formeldarstellungen für die beiden Filme, die für die Distanzberechnung herangezogen werden.

[3] http://www.grouplens.org/node/462
[4] http://www.rottentomatoes.com/

$$BF(f_1, f_2) = 1 + \frac{|Bewertung(f_1) - Bewertung(f_2)|}{5}$$

Formel 1: Bewertungsfaktor

Der Faktor für das Erscheinungsjahr (JF) bedient sich der Differenz der Erscheinungsjahre. Diese wurde allerdings abgeschwächt, um den Einfluss auf die Gesamtdistanz zu verkleinern, was vor allem für Remakes von Filmen wichtig ist.

$$JF(f_1, f_2) = 1 + \frac{|Jahr(f_1) - Jahr(f_2)|}{200}$$

Formel 2: Erscheinungsjahrfaktor

Des Weiteren wurden Faktoren bestimmt, die die Distanz bei übereinstimmenden Regisseuren (RF) bzw. Produktionsländern (LF) mindern. Der RF wurde bei Übereinstimmung mit 0.75 und der LF mit 0.25 definiert. Beide erhalten den Wert 0 wenn es keine Übereinstimmung gibt.

Die finale Distanzfunktion mit all ihren Faktoren wurde zum Test auf die ersten 1.000 Filme des *MovieLens*-Datasets angewandt und dieser resultierende Graph anschließend mit dem Programm *Cytoscape* visualisiert. Bei einer maximalen Kantenlänge von 150 ergaben sich einige gut voneinander abgegrenzte Cluster, was auf eine vernünftige Partitionierung hinwies. Um die Ergebnisse noch genauer verifizieren zu können, müsste allerdings eine Dimensionsreduktion mit *PCA* oder *MVE* (Shaw 2011) durchgeführt werden, um diese Daten im 2 oder 3-dimensionalen Raum unverzerrt betrachten zu können.

$$Distanz(f_1, f_2) = \frac{100 * \left(1 + \frac{GF(f_1, f_2)}{100}\right) * JF(f_1, f_2)}{\left(1 + RF(f_1, f_2) + LF(f_1, f_2)\right) * \frac{1}{BF(f_1, f_2)}}$$

Formel 3: Finale Distanzfunktion

V. Weiterführende Arbeiten

Der Einfluss der einzelnen Faktoren auf die Gesamtdistanz ist in der gegenwärtigen Formel nicht einfach ersichtlich. Aus diesem Grund ist geplant, eine neue Distanzfunktion zu definieren, bei der die Parameter und deren Einfluss leichter manipuliert werden können. Zu allererst soll die Genredistanz mittels der Levenshtein-Distanz berechnet werden. Die Division soll des Weiteren aus der Formel weichen und durch eine simple Addition der Faktoren ersetzt werden. Der Jahresfaktor soll durch eine Einteilung der Filme in Dekaden leichter zu berechnen sein und seinen kleinen Einfluss beibehalten. Außerdem sollen alle Faktoren normiert und mit variablen Parametern multipliziert werden, um deren Einfluss auf die Gesamtdistanz feingranularer steuern zu können. Durch diese Neudefinition der Distanz können neben der einfacheren Begreifbarkeit der Formel auch die Kriterien einer Metrik erfüllt werden.

Die besten Werte für die zuvor genannten Parameter sollen dann in weiterer Folge durch Testreihen ermittelt und verifiziert werden. Diese Verifikation soll durch automatisierte Clusteringalgorithmen erfolgen. Des Weiteren ist geplant, den OPTICS-Algorithmus zur Visualisierung der Clusterbildung zu verwenden.

Sobald diese distanzbasierten Vorschläge vernünftige Ergebnisse liefern, sollen schlussendlich auch Benutzerdaten in die Vorschlagsberechnung miteinbezogen werden. Dazu ist angedacht, ein iteratives Verfahren zur Distanzkorrektur zu entwickeln, damit die bereits ermittelten Distanzen nur aktualisiert und nicht bei jedem Durchlauf des RS neu berechnet werden müssen.

Referenzen

Adomavicius, Gediminas, Ramesh Sankaranarayanan, Shahana Sen, und Alexander Tuzhilin. *Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach.* Herausgeber: ACM. New York, 1. Januar 2005.

Das, Duco, und Herman ter Horst. *Recommender Systems for TV.* Herausgeber: AAAI. Eindhoven, 1998.

Golbeck, Jennifer, und James Hendler. *FilmTrust: Movie Recommendations using Trust in Web-based Social Networks.* Maryland, 2006.

Herlocker, Jonathan J., Joseph A. Konstan, und John Riedl. *Explaining Collaborative Filtering Recommendations.* Herausgeber: ACM. Minneapolis, Minnesota, Dezember 2000.

Li, Peng, und Seiji Yamada. *A Movie Recommender System Based on Inductive Learning.* Herausgeber: IEEE. Tokyo, 2004.

Mak, Harry, Irena Koprinska, und Josiah Poon. *INTIMATE: A Web-Based Movie Recommender Using Text Categorization.* Sydney, 2003.

Pazzani, Michael J., und Daniel Billsus. *Content-Based Recommendation Systems.* Herausgeber: Springer Verlag Berlin Heidelberg. 2007.

Perny, Patrice, und Jean-Daniel Zucker. *Preference-base Search and Machine Learning for Collaborative Filtering: the "Film-Conseil" Movie Recommender System.* Herausgeber: CEPAD. Paris, 2001.

Shaw, Blake. *Graph Embedding and Nonlinear Dimensionality Reduction.* New York, 2011.

Symeonidis, Panagiotis, Alexandros Nanopoulos, und Yannis Manolopoulos. *MoviExplain: A Recommender System with Explanations.* Herausgeber: ACM. New York, Oktober 2009.