



دانشکده مهندسی کامپیوتر

سیستم توصیه گر موسیقی بر اساس محتوا

پایان نامه یا رساله برای دریافت درجه کارشناسی
در رشته مهندسی کامپیوتر گرایش نرم افزار

محیا خزائی

استاد راهنما:

سید صالح اعتمادی

شهریورماه ۱۳۹۸

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

تأییدیه ی صحت و اصالت نتایج

باسمه تعالی

اینجانب محیا خزائی به شماره دانشجویی ۹۳۴۳۲۰۸۱ دانشجوی رشته مهندسی کامپیوتر مقطع تحصیلی کارشناسی تأیید می‌نمایم که کلیه ی نتایج این پایان‌نامه/رساله حاصل کار اینجانب و بدون هرگونه دخل و تصرف است و موارد نسخه‌برداری شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. در صورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مؤلفان و مصنفان و قانون ترجمه و تکثیر کتب و نشریات و آثار صوتی، ضوابط و مقررات آموزشی، پژوهشی و انضباطی ...) با اینجانب رفتار خواهد شد و حق هرگونه اعتراض در خصوص احقاق حقوق مکتسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم. در ضمن، مسؤولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی‌صلاح (اعم از اداری و قضایی) به عهده‌ی اینجانب خواهد بود و دانشگاه هیچ‌گونه مسؤولیتی در این خصوص نخواهد داشت.

نام و نام خانوادگی: محیا خزائی

تاریخ: ۱۰ شهریور ۱۳۹۸

مجوز بهره‌برداری از پایان‌نامه

بهره‌برداری از این پایان‌نامه در چهارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما به شرح زیر تعیین می‌شود، بلامانع است:
بهره‌برداری از این پایان‌نامه/ رساله برای همگان بلامانع است.

نام استاد : سید صالح اعتمادی

تاریخ: ۱۰ شهریور ۱۳۹۸

چکیده

سیستم‌های توصیه‌گر امروزه نقش بسیار مهمی در شبکه‌های اجتماعی و برنامه‌های کاربردی دارند. وظیفه‌ی اصلی این سیستم‌ها محدود کردن فضای حالت برای انسان‌ها به منظور بالا بردن کیفیت و بهره‌وری از موقعیت‌های به وجود آمده است. این جست و جو وظیفه‌ی سیستم‌های خودکار و هوش مصنوعی می‌باشد. برخی از این توصیه‌ها بر اساس متادیتاها (اطلاعاتی که به صورت اجمالی یک محتوا را توصیف می‌کند مانند نام و هشتگ‌ها و ...) و برخی دیگر بر اساس خود محتوا انجام می‌شود. بررسی محتوای یک اثر مانند موسیقی موضوع آسانی نیست چراکه درک یک ماشین و یک انسان از این محتوا یکسان نمی‌باشد. در این مقاله سیستم توصیه‌گر موسیقی فقط بر اساس محتوا مورد تحقیق قرار گرفته‌است. اطلاعات اولیه‌ی سیستم تعداد دفعاتی می‌باشد که هر کاربر به هر آهنگ گوش داده‌است. به این منظور از *implicit collaborative filtering* استفاده شده است و این تحلیل با دو مورد از روش‌های *SVD* (جداسازی مقادیر یگانه) مورد بررسی قرار گرفته است. محتوای هر آهنگ نیز ابتدا به *mfcc*های مرتبه اول تا سوم تبدیل شده سپس با خوشه‌بندی محدوده‌ی آن کاهش پیدا کرده و در آخر برای راحتی محاسبات *quantize* می‌شود. در نهایت محتوای هر آهنگ به عنوان داده، و اطلاعات حاصل از *matrix factorization* برای هر آهنگ به عنوان برچسب به یک ماشین داده می‌شود تا فرایند یادگیری ماشین انجام شود. از این به بعد، برای هر آهنگ جدیدی که به سیستم اضافه می‌شود، می‌توان برچسب‌های مربوط به آن را استخراج کرد و با ضرب کردن دوباره‌ی ماتریس‌ها دریابیم این آهنگ با ذائقه‌ی کدام کاربران مطابقت دارد.

واژه‌های کلیدی: سیستم توصیه‌گر، موسیقی، محتوا، *collaborative filtering*

فهرست مطالب

۱	فصل ۱: مقدمه
۲	۱-۱- مقدمه
۳	فصل ۲: مروری بر منابع
۴	۱-۲- مقدمه
۴	۲-۲- مروری بر ادبیات موضوع
۵	۳-۲- نتیجه‌گیری
۷	فصل ۳: روش تحقیق
۸	۱-۳- مقدمه
۸	۲-۳- محتوا
۸	۳-۲-۱- علت انتخاب روش
۹	۳-۲-۲- تشریح کامل روش تحقیق
۱۲	فصل ۴: نتایج و تفسیر آنها
۱۳	۱-۴- مقدمه
۱۳	۲-۴- محتوا
۱۴	فصل ۵: جمع‌بندی و پیشنهادها
۱۵	۱-۵- مقدمه
۱۵	۲-۵- محتوا
۱۵	۳-۵- جمع‌بندی
۱۶	مراجع

فصل ۱:

مقدمه

۱-۱- مقدمه

در اپلیکیشن‌های معروف پخش و معرفی موسیقی از سیستم‌های متنوعی برای توصیه‌گر استفاده می‌شود. برخی از آن‌ها بر اساس متادیتا^۱ این توصیه را انجام می‌دهند از مواردی مثل نام خواننده، نام آلبوم، سبک آهنگ و در مواقعی متن ترانه‌ها استفاده می‌کنند. اگر این داده‌ها را برای هر شخص به صورت انفرادی استفاده کنیم، دایره‌ی عمل و پیشنهادات بسیار محدود می‌شود. اما تصور کنید از این که یک آهنگ مورد علاقه‌ی چند نفر باشد بتوانیم ویژگی‌های سلیقه‌ی آنان را شناسایی کنیم و تناظر این ویژگی‌های سلیقه‌ای به ویژگی‌های آهنگ را پیدا کنیم روند پیشنهاد دادن موسیقی بسیار معقول‌تر می‌شود.

^۱ metadata

فصل ۲:

مروری بر منابع

۲-۱- مقدمه

در بسیاری از مقالات و کارهای علمی انجام‌شده به بررسی این مسئله که چگونه درک انسان از موسیقی را به عهده‌ی ماشین بگذاریم پرداخته‌شده‌است. این روش‌ها عبارتند از لیست‌های پخش موسیقی متناسب با سلیقه‌ی افراد و ادامه دادن پخش موسیقی با توجه به رفتار کاربر برای ادامه‌ی گوش دادن یا رد کردن آن‌ها، توصیه کردن یک موسیقی خاص به یک فرد خاص با توجه به آنالیز ویژگی‌های آن آهنگ و ویژگی‌های سلیقه‌ی هر فرد و روش‌های دیگری که می‌توان توسط آن‌ها موسیقی یا موسیقی‌هایی را به یک فرد پیشنهاد داد.

۲-۲- مروری بر ادبیات موضوع

در مقاله‌ای^[۱] برچسب‌های اجتماعی، کلمات کلیدی مرتبط با برخی از منابع در وب به حساب آمده‌است. در مورد موسیقی، برچسب‌های اجتماعی تبدیل به یک جزء مهم از سیستم‌های پیشنهاد دهنده Web2.0 شده‌اند، که به کاربران امکان می‌دهد لیست‌های پخش (playlist) را براساس شرایط وابسته به استفاده، مانند "استراحت" یا "پیاده روی" که برای آهنگ‌های خاصی استفاده می‌شود، ایجاد کنند. در این مقاله، روش پیش بینی این برچسب‌های اجتماعی را مستقیماً از فایل‌های MP3 پیشنهاد می‌کند. با استفاده از مجموعه‌ای از طبقه‌بندی‌های تقویت‌شده (boosted classifiers)، ویژگی‌های صوتی را بر روی برچسب‌های اجتماعی جمع‌آوری شده از وب نمایش می‌دهیم. برچسب‌های اتوماتیک (auto tags) اطلاعاتی را درباره موسیقی ارائه می‌دهند که اگر این سیستم وجود نداشت غیرقابل برچسب گذاری بودند یا برچسب‌های ضعیفی داشتند و امکان افزودن موسیقی به یک توصیه کننده اجتماعی وجود نداشت. این روش از "cold-start problem" معمول در چنین سیستمی جلوگیری می‌کند. مشکل cold-start problem زمانی به وجود می‌آید که در یک سیستم‌هایی یک شی (یا آهنگ) تازه وارد است و سیستم توصیه گر درباره‌ی آن هیچ اطلاعاتی ندارد که بداند به چه کسانی پیشنهاد بدهد. همچنین می‌تواند برای هموار سازی (Smoothing) از فضای برچسب‌ها استفاده کرد تا تخمینی در باره‌ی شباهت در فضای آهنگ‌ها به دست آورد. در این مقاله به بررسی شبکه‌ی عصبی و مدل SVM برای بررسی عملکرد پرداخته‌شده است.

در مقاله‌ای دیگر^[2] با تکیه بر این که سبک‌های موسیقی مرزهای مشخصی ندارند و منحصر به فرد نیستند سیستمی طراحی شده است تا بتواند برچسب‌گذاری را به کمک طبقه‌بندی ویژگی‌های موسیقی برحسب چندین برچسب و ریزدانه‌ها با استفاده از سه روش مختلف داده‌های صوت، متن و تصویر موسیقی انجام دهد. این کار را به کمک یادگیری عمیق و شبکه‌های `mlp` و `cnn` انجام می‌دهد و نشان می‌دهد وجود اطلاعات بر پایه‌ی تمام این اطلاعات موجب عملکرد بهتر سیستم نسبت به حالتی که فقط به یک یا دو تای آن‌ها توجه شود می‌شود.

در مقاله‌ای دیگر^[3] به موضوع قابل تامل و جدیدی پرداخته شده که هدف نهایی آن ادامه دادن لیست‌های پخش (playlist) به کمک نمونه‌های دستی و ویژگی‌های موسیقی است. تولید خودکار لیست پخش موسیقی یک فرم خاص از توصیه موسیقی است. به طور کلی کاربر یک مجموعه از آهنگ‌ها را دریافت می‌کند که آن یک بازه‌ی منسجم گوش دادن به موسیقی را تعیین می‌کند. فرضیه‌ی این مقاله این است برای توصیه کردن لیست جدید باید از روی یک لیست که به صورت دستی آماده شده است یادگیری انجام شود. به این ترتیب محدودیت مسائل تک کاره (ad hoc) به مسئله تحمیل نمی‌شود. از روش‌های Collaborative filtering می‌توان برای رسیدن به این داده‌ها استفاده کرد. هرچند به دلیل بسیار کم بودن این لیست‌های پخش و گرایش (bias) بودن آن‌ها به سمت آهنگ‌های بسیار معروف و نبودن بسیاری از آهنگ‌ها در این لیست‌ها باعث یک مشکل بزرگ در داده شده و به تنهایی توصیه نمی‌شود. برای غلبه بر این مسئله، در این گزارش یک مدل جایگزین را بر اساس یک طبقه‌بندی کننده‌ی آهنگ به پخش پیشنهاد شده است که ساختار زیرین از لیست‌های پخش واقعی را یاد می‌گیرد در حالی که ویژگی‌های ترانه مشتق شده از صوتی، برچسب‌های اجتماعی و گزارش‌های مربوط به گوش دادن مستقل را دارا می‌باشد. آزمایش‌ها بر روی دو مجموعه داده از لیست‌های پخش شده توسط دست نشان‌دهنده عملکرد رقابتی در مقایسه با فیلترینگ مشارکتی است هنگامی که آموزش کافی داده‌ها در دسترس است و عملکرد قوی‌تری را در هنگام توصیه به نواختن و نواختن ترانه‌ها انجام می‌دهد.

۲-۳- نتیجه‌گیری

با توجه به کارهای علمی انجام شده روی این موضوع به چند مورد خاص پی می‌بریم. اولاً این که جمع‌آوری داده‌ها و برچسب‌های اصلی برای این موضوع کاری دشوار است و باید برای جلوگیری از جهت‌گیری خاص

سیستم به دقت انتخاب شود. ثانياً تمام مقالاتی که از محتوای آهنگ برای یادگیری ماشین استفاده شده بود سیگنال‌های موسیقی به mfcc تبدیل شده بود. همچنین نتایج نشان داده که یادگیری عمیق و شبکه‌های عصبی جواب‌های بهتری نسبت به classifierهای دیگر دارند.

در نتیجه‌گیری آخر فصل، با توجه به بررسی انجام شده بر روی مراجع تحقیق، افق‌های تازه‌ای که پیش‌روست یا خلأهای تحقیقاتی بیان می‌شود.

فصل ٣:

روش تحقیق

۳-۱- مقدمه

در این پژوهش ما می‌خواهیم از طریق بازخورد غیر مستقیمی که افراد نسبت به آهنگ‌ها داشتند، سلیقه‌ی آن‌ها را متوجه شویم و آهنگ‌های مناسب آن‌ها را پیشنهاد دهیم. این رویه به این صورت است که تعداد دفعات گوش دادن هر کاربر به هر آهنگ را در نظر می‌گیریم و فرض می‌کنیم هر چقدر کاربر به یک آهنگ بیشتر گوش داده باشد به آن آهنگ علاقه‌ی بیشتری دارد.

در این مسیر فاکتورهای مربوط به محتوای آهنگ و فاکتورهای مربوط به سلیقه یا سبک را به یک شبکه‌ی عصبی یا رگرسیون می‌دهیم. در این فرایند، فاکتورهای سلیقه همان برچسب‌های مرسوم در فرایند یادگیری ماشین می‌باشند. بعد از فرایند یادگیری، هر زمان یک آهنگ جدید به سیستم بدهیم، فاکتورهای سلیقه یا برچسب‌های آن را استخراج می‌کند و به کاربرانی که آن فاکتورهای سلیقه را دارند پیشنهاد می‌کند.

۳-۲- محتوا

در ادامه به علت انتخاب روش و همچنین شرح کامل روش تحقیق شامل جمع‌آوری داده‌ها پردازش ماتریس و خود آهنگ‌ها پرداخته شده‌است.

۳-۲-۱- علت انتخاب روش

در این مقاله^[4] به صورت کامل درباره‌ی مزیت‌های استفاده از شبکه عصبی عمیق در راستای این فرایند بحث شده‌است و در مقایسه با نمونه‌های مشابه خود که از اطلاعات بیشتری هم استفاده کرده‌بودند، به نتایج دقیق‌تری دست یافته‌است.

۳-۲-۲- تشریح کامل روش تحقیق

□ جمع‌آوری داده‌ها

جمع‌آوری داده‌ها یکی از چالش برانگیزترین مسائل حوزه‌ی هوش مصنوعی می‌باشد. در این مسئله ما نیاز به محتوای آهنگ‌ها و اطلاعات مربوط به کاربرانی را داشتیم که به آن‌ها گوش یا امتیاز داده بودند. به علت به روز بودن این موضوع و تولید ارزش افزوده برای سایت‌های بزرگ پخش موسیقی مانند spotify، youtube و غیره بعد از مدتی api‌های مورد استفاده در مقالات قدیمی از دسترس خارج شدند. در نهایت ما از قسمتی از داده‌های مورد استفاده قرار گرفته در یک مقاله^[۴] برای پیدا کردن داده‌ی مربوط به میزان گوش دادن هر کاربر به هر آهنگ پیدا کردیم به این ترتیب که در یک فایل به ترتیب شناسه‌ی کاربر، شناسه‌ی آهنگ و تعداد دفعات گوش دادن ذخیره شده‌است که تعداد رکوردهای این فایل به ۴۸ میلیون می‌رسد. در این داده [۵] اطلاعات کاربران محرمانه می‌باشد اما یک فایل دیگر وجود دارد که مشخص می‌کند نام خواننده و نام آهنگ منطبق بر چه شناسه آهنگی می‌باشد. حجم فایل‌های اول بسیار زیاد است (حدود ۳ گیگ) که ما آن را از حالت ذکر شده به حالت دیکشنری در آوردیم و در فایل‌های pickle ذخیره‌سازی کردیم تا برای استفاده به راحتی به شکل یک ماتریس در رم قرار بگیرد.

اما در آن مقاله توسط یک سایت توانسته بودند به ۲۳ ثانیه از محتوای هر آهنگ دسترسی پیدا کنند که آن سایت دیگر این خدمات را ارائه نمی‌داد. به همین منظور یک قطعه کد نوشته شد که در آن از ماژول youtube-dl استفاده شده است، به صورت خودکار از فایلی که در آن نام آهنگ و نام خواننده و شناسه‌ی آهنگ موجود بود خط به خط نام آهنگ و سپس نام خواننده را می‌خواند و در صفحه‌ی سایت youtube جست و جو می‌کند سپس اولین نتیجه‌ی موجود را باز کرده و ویدیوی مورد نظر را به صورت صوتی ذخیره می‌کند. سپس آن آهنگ را که به فرمت mp3 بود به کمک برنامه‌ی ffmpeg به فرمت wav در می‌آورد. سپس این فایل را با نام شناسه‌ی آهنگ ذخیره می‌کند تا در ادامه دسترسی به آن برای برنامه آسان‌تر باشد.

□ پردازش محتوای آهنگ

برای قابل فهم شدن محتوای آهنگ دانلود شده باید آن را از حالت موج (wav) خارج کنیم چرا که این حالت خام بسیار فضای حالت موضوع را بزرگ کرده و شبکه‌ی عصبی برای درک آن باید زمان بسیار زیادی صرف کند و از طرفی میزان داده‌های ما برای حل مسئله با این فضا کافی نمی‌باشد. به همین دلیل از mel-frequency cepstral coefficients استفاده می‌کنیم. این فاکتور (به اختصار mfcc) فرکانسهای ملودی را از

اختلاف فرکانس‌ها به دست می‌آورد و بر حسب نوع نمونه‌گیری ضبط موسیقی و طول آن می‌تواند تعداد متفاوتی داشته‌باشد. علاوه بر این اختلاف در تحقیقات ثابت شده‌است نه تنها به صورت مرتبه اول بلکه اختلاف مرتبه دوم و اختلاف مرتبه سوم فاکتورهای mfcc بسیار در توصیف آهنگ به صورتی که انسان درک می‌کند کاربرد دارد^[۴]. در مسیر انجام پروژه فایل‌های حاوی اطلاعات mfcc با ۳ مرتبه اختلاف در فایل‌هایی با نام شناسه‌ی - آهنگ و پسوند mfcc ذخیره شده‌اند.

از آن جایی که طول آهنگ‌های مختلف متفاوت است و ممکن است rate ثابت آن‌ها با یکدیگر متفاوت باشد، باید داده‌های ورودی خود را یکدست کرده تا درک اختلاف آن‌ها برای ماشین ساده‌تر شود. به این منظور ما در بخش قبل اختلاف‌های فرکانسی را در چند مرتبه مورد بررسی قرار داده بودیم. حال به شمارش تعداد هر کدام از مقادیر اختلاف می‌پردازیم. سپس تعداد هر کدام از اختلاف‌ها را نگهداری می‌کنیم که به این حالت از نگهداری داده bag of words گفته می‌شود. اما این حالت هم بسیار گسترده است و باز برای همه‌ی آهنگ‌ها داده‌های مناسب و یکپارچه‌ای تولید نمی‌شود. حالا از الگوریتم k-mean به منظور خوشه‌بندی گستره (range) این اختلاف‌ها می‌پردازیم و ۴۰۰۰ mean انتخاب شده‌است و در هر آهنگ هر فاکتور mfcc را به نزدیک‌ترین mean نسبت می‌دهیم. برای یادگیری جای درست این mean‌ها به منظور متناسب بودن برای هر آهنگی، بهتر است فرایند یادگیری آن روی تمامی آهنگ‌ها انجام شود، اما به علت بالا بودن حجم همه‌ی آهنگ‌ها به یادگیری mean‌ها روی چند آهنگ به صورت اتفاقی اکتفا می‌کنیم.

در پایان این مرحله از کار هر فایل آهنگ با شناسه‌ی خود و پسوند bow موجود است که به همین صورت برای هر شبکه‌ی عصبی یا رگرسیونی قابل استفاده است.

□ پردازش ماتریس اطلاعات کاربران و آهنگ‌ها

همان‌طور که گفته شد، در این قسمت از کار به علت کار با فایل‌های بسیار بزرگ لازم است از ساختمان داده‌های مناسب و الگوریتم‌هایی با پیچیدگی مناسب استفاده شود چرا که در غیر این صورت عملاً پردازش این داده‌ها غیرممکن می‌شود. در باره‌ی ساختار ماتریس‌ها بهتر است از حالت sparse و به علت استفاده از hashing در پیاده‌سازی set در زبان پایتون بهتر است از این ساختمان داده استفاده شود.

فرایند جداسازی ماتریس به فرایندی می‌گویند که یک ماتریس را به دو ماتریس تبدیل کنیم که حاصل ضرب آن‌ها ماتریس اولیه شود. رسیدن به این موضوع از روش‌های مختلفی از جمله gradient decent امکان پذیر است. به این صورت که در دو ماتریس ثانویه مقادیر تصادفی می‌گذاریم و حاصل ضرب آن را اندازه می‌گیریم، در فضای چند بعدی اختلاف این ماتریس حاصل ضرب با ماتریس اولیه را می‌سنجیم و سپس در

خلاف جهت این اختلاف مقادیر را تغییر می‌دهیم تا در نهایت حاصل ضرب منطبق شود بر ماتریس اولیه. حال فرض کنید ماتریس Q که هر سطر متعلق به یک کاربر و هر ستون متعلق به یک آهنگ می‌باشد. اگر بتوانیم دو ماتریس H و W را بیابیم که حاصل ضرب آن‌ها برابر با Q باشد، سطرهای ماتریس اول متعلق به یک کاربر و ستون‌های آن متعلق به فاکتورهای مختلف سلیقه/علاقه می‌باشد. سطرهای ماتریس دوم متعلق به فاکتورهای سلیقه/علاقه می‌باشد و ستون‌های آن متعلق به یک آهنگ می‌باشد. به این ترتیب، اگر ما بتوانیم فاکتورهای مربوط به سلیقه/علاقه برای یک آهنگ را پیدا کنیم، با ضرب کردن ماتریس دوم در ماتریس اول می‌توانیم بفهمیم این آهنگ با فاکتورهای سلیقه/علاقه کدام کاربر تناسب بیشتری داشته و این آهنگ را به آن کاربر پیشنهاد کنیم. به این فاکتورها فاکتورهای نهان می‌گوییم که برای هر آهنگ با توجه به ماتریس داده‌ها قابل تولید است و آن‌ها به صورت شناسه‌ی آهنگ و پسوند LF (حالت اختصاری $Latent$ Factor) ذخیره می‌کنیم.

به منظور بررسی عملکرد کل سیستم ابتدا یک حالت آزمایشگاهی از کل مسئله تولید شد. ۵ آهنگ مختلف شامل ۲ آهنگ راک و ۳ آهنگ آرام کلاسیک دانلود شد و یکی از آهنگ‌های کلاسیک را از فرایند یادگیری خارج کردیم. دو الی سه کاربر فرضی در نظر گرفته شد که علاقه‌ی یکی از آن‌ها به آهنگ‌های راک است. یعنی برای مثال آهنگ‌های راک را یک یا دو بار گوش کرده و آهنگ‌های کلاسیک را گوش نکرده است. کاربری با سلیقه‌ی کاملاً متفاوت در نظر گرفته شد که به آهنگ‌های کلاسیک گوش داده و به آهنگ‌های راک گوش نداده است. در این فرایند به منظور تولید فایل‌های LF به جای شبکه عصبی از رگرسیون خطی استفاده کردیم. بعد از جایگذاری آهنگ جدید در ماتریس مربوط به آهنگ/سلیقه و ضرب کردن در ماتریس کاربر/سلیقه نتیجه‌ی به دست آمده ماتریس مربوط به کاربر/آهنگ نتایج مطلوبی به ما می‌داد. آهنگ کلاسیکی که به منظور ارزیابی سیستم کنار گذاشته شده بود، برای فردی که به آهنگ‌های کلاسیک گوش داده بود عداد بالاتری نسبت به فردی که به آهنگ‌های راک گوش داده بود کسب می‌کرد.

فصل ۴:

نتایج و تفسیر آنها

۴-۱ مقدمه

نتایج تولیدشده حاصل از این پژوهش به علت تفاوت راه‌حل‌های به کار رفته در مقاله‌های قبلی، متفاوت است که در ادامه نتایج آنها خواهد آمد.

۴-۲-۴ محتوا

همان‌طور که در بخش ۳-۲-۱ اشاره شد نتایج حاصل بر روی ۴ موسیقی با دو سبک کاملاً متفاوت به کمک یکی از ساده‌ترین روش‌های رگرسیون یعنی رگرسیون خطی نتیجه‌ای بسیار عالی گرفت. در یک بررسی دیگر ما قسمت خاصی از داده را به شرح زیر جدا کردیم. آهنگ‌ها را بر اساس بیشترین میزان کاربر که به آن‌ها گوش داده‌اند جدا کردیم. سپس سعی کردیم ۱۰ آهنگ انتخاب کنیم که کاربرانی که به ۵ آهنگ اول گوش داده‌اند کمترین اشتراک را با کاربرانی که به ۵ آهنگ دوم گوش داده‌اند داشته‌باشند. و انتظار داشتیم با حذف کردن یکی از آهنگ‌های گروه دوم و استفاده‌ی آن به عنوان تست سیستم نتایج مناسبی بگیریم. اما فراموش کردیم که این جداسازی داده باعث می‌شود اطلاعات داده‌ی واقعی دستکاری شده و سیستم نتواند در آن به جواب خوبی برسد.

فصل ۵:

جمع‌بندی و پیشنهادها

۵-۱- مقدمه

مباحث اصلی که در این پژوهش به حصول رسید عبارت است از این که سیستم جداسازی ماتریس باید ارتباط خوبی با نوع داده داشته‌باشد. سپس این که نوع تحلیل ما از محتوای موسیقی با توجه به تمام محاسبات انجام‌شده روی آن همچنان بسیار خام می‌باشد و سیستم‌های عمیق می‌توانند آن را تحلیل کنند نه سیستم‌های ساده.

۵-۲- محتوا

به دلیل زیاد بودن صفر در ماتریس اصلی این که از سیستم non-negative matrix factorization استفاده کنیم نتیجه‌ی خوبی کسب نمی‌کند چرا که بعد از factorization ماتریس و اضافه کردن ستون جدید و ضرب مجدد آن‌ها طیف اعداد تولیدشده بسیار متفاوت از ماتریس اول خواهند بود، به این معناست که برای داده‌های بزرگ و صفر دار از سیستم weighted matrix factorization استفاده کنیم. همچنین در مقاله مرجع از یک سیستم یادگیری عمیق استفاده‌شده بود که به هیچ وجه قابل مقایسه با یک سیستم رگرسیون خطی نمی‌باشد و این عمل به منظور میزان تحلیل پذیری این داده‌ها توسط سیستم‌های ساده‌تر می‌باشد.

۵-۳- جمع‌بندی

با توجه به جواب قابل قبول سیستم رگرسیون خطی در یک مثال بسیار ساده می‌توان نتیجه گرفت با انجام پیش پردازش‌های قوی‌تر و بررسی‌شده‌تر می‌توان از سیستم‌های سبک‌تر نیز برای گرفتن نتیجه‌ی مطلوب کمک گرفت و با این روش سرعت این سیستم‌ها را بالا برد.

مراجع

- [1] Eck, Douglas, et al. "Automatic generation of social tags for music recommendation." *Advances in neural information processing systems*. 2008.
- [2] Oramas, Sergio, et al. "Multi-label music genre classification from audio, text, and images using deep features." *arXiv preprint arXiv:1707.04916* (2017)
- [3] Vall, Andreu, et al. "Music playlist continuation by learning from hand-curated examples and song features: Alleviating the cold-start problem for rare and out-of-set songs." *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems*. ACM, 2017.
- [4] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR)*, 2011.

Abstract:

Today recommender systems have a critical role in social networks and applications. The main duty of these systems is to limit the search space for people in order to increase the quality of applications. This search is duty of autonomous systems and artificial intelligence machines. Some of these recommenders based on metadata and some of them based on content. It's not a simple work to study content of music because a machine has not the same vision to music as the vision of human. But these systems can make suitable recommendations like human recommend. In this thesis we just talk about content-based recommendation. The main data-set consist of the count of user-item pair. To analyze this matrix, we used implicit collaborative filtering and investigated two methods of SVD. For music content analysis we used mfccs and then used clustering to decrease dimension of data and finally used quantization to make it calculatable. The content of each music become the data and latent factors that extracted from matrix factorization become the label to train the machine. After that, for every new music we can compose the latent factors and reproduced matrixes can tell us witch user may love this song.

Keywords: content-based, collaborative filtering, mfcc



IU | ST

**Iran University of Science and Technology
Computer Department**

Content-Based Music Recommendation

**A Thesis Submitted in Partial Fulfillment of the Requirement for the
Degree of Bachelor of Science in Software**

**By:
Mahya Khazaei**

**Supervisor:
Sauleh Etemadi**

September 2019