Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

FIIT-0000-00000

Martin Černák

Dynamické odporúčanie

Bakalárska práca

Študijný program: Informatika
Študijný odbor: 9.2.1 Informatika
Miesto vypracovania: Ústav informatiky a softvérového inžinierstva
Vedúci práce: prof. Ing. Pavol Návrat, PhD.

Máj 2015

Anotácia

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ
Študijný program: Informatika
Autor: Martin Černák
Bakalárska práca: Dynamické odporúčanie
Vedúci práce: prof. Ing. Pavol Návrat, PhD.
Máj 2015

Dynamické odporúčanie v kontexte hudobných dokumentov je vďaka svojmu úzkemu zamerania a vďaka menšej komunite značne nepreskúmané. Existuje niekoľko riešení, ktoré ale nevyužívajú plný potenciál dynamického odporúčania. Jednou z možností ako tieto systémy vylepšiť je začať uvažovať starnutie ako používateľových tak globálnych preferencií. V hudobnóm odvetví môžeme častejšie ako v ostatných vidieť príchod mimoriadne populárnych nových interprétov, piesni a štýlov, ktoré rýchlo vymiznú z povedomia verejnosti, prípadne zostane okolo nich úzka skupina fanúšikov. Kontrastom k nim sú piesne, autori a hudobné štýly, ktoré pretrvávajú dlhodobo v povedomí ľudí a vypadajú, že starnutie na nich nemá vplyv.

Annotation

Slovak University of Technology Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES
Degree Course: Informatics
Author: Martin Černák
Bachelor thesis: Dynamic recommendation
Supervisor: prof. Ing. Pavol Návrat, PhD.
May 2015

Dynamic recommendation in the context of musical documents thanks to its narrow focus and with smaller community largely unexplored. There are several solutions but that don't using full potential of dynamic recommendation. One way to improve these systems is to start thinking of aging user and global preferences. In the music sector this can be more frequent than in other sectors, extremely popular new artists, songs and styles that quickly disapear from public awareness or remain around them a small group of fans. A contrast to them are songs, authors and musical styles that persist long time in the minds of people and looks like aging does not affect them.

POĎAKOVANIE

Chcel by som v prvom rade poďakovať pánu profesorovi Pavlovi Návratovi za jeho odbornú pomoc a motiváciu a za všetky konzultácie ktoré sme spoločne absolvovali. Zároveň by som rád poďakoval učástníkom jeho výskumného seminára za konštruktívnu kritiku, ktorá taktiež prispela k zdokonaleniu tejto práce.

ČESTNÉ PREHLÁSENIE

Čestne prehlasujem, že záverečnú prácu som vypracoval samostatne s použitím uvedenej literatúry a na základe svojich vedomostí a znalostí.

.....

Martin Černák

Obsah

Zoznam obrázkov

Zoznam ukážok

1 Úvod

Ako z jej názvu vyplýva, informatika je predmet zameraný na prácu s informáciami. To čo kedysi bolo najväčším problémom, teda dostať nejaké informácie k používateľom, už dávno nie je problém. Vďaka internetu sa dajú informácie dostať prakticky všade. No teraz čelíme väčšiemu problému. Naša spoločnosť dokáže za účelom zábavy, rozvoja alebo produktivity vyprodukovať neuveriteľné množstvo informácií. Precíznosť archivácie údajov je asi najväčšia v histórií, problém nastáva ak chceme nejaké údaje vyhľadať. Klasický prístup spravovania informácií už nie sú dostačujúce a jednoduché vyhľadávanie už nieje dostatočne efektívne na to aby sme boli schopní nájsť požadované informácie.

Dokonca aj vyhľadávanie ako také prestáva byť dostatočne efektívne, namiesto neho sa dostáva do popredia odporúčanie, ktoré doslova používateľovi ponúkne informácie, ktoré by ho mohli zaujímať, bez toho aby musel vynaložiť akúkoľvek námahu na hľadanie. Aby mohol systém robiť takúto predikciu potrebuje poznať používateľa a to mu umožňuje profilovanie používateľov. Profil používateľa je komplexná vec. Záujmy používateľa môžu byť ovplyvnené jeho demografickými parametrami (vek, vzdelanie, miesto pobytu), záujmami a všeobecnými novinkami ako vydanie nového albumu obľúbenej kapely alebo uvedenie nového zariadenia na trh. Do úvahy musíme brať aj udalosti v živote používateľa, napríklad narodenie potomka tiež v určitom smere ovplyvní používateľove záujmy. Z toho vyplýva že profil musí byť dynamický, a preto je potrebné nejakým spôsobom aj odoberať záujmy, o ktoré používateľ už viac nezaujíma.

Cieľom tohoto projektu je vytvoriť aplikáciu ktorá bude schopná dynamicky odporúčať. Na riešenie hore spomenutých problémov existuje množstvo prístupov. Každý z týchto prístupov má mierne lepšie výsledky v iných situáciách, čiže dosť závisí od domény, pre ktorú bude systém odporúčať. V tomto projekte sa budeme zaoberať doménou hudobných dokumentov (akordy, texty, taby, preklady). Táto oblasť ešte nie je prebádaná, čo nám prináša nové možnosti ako aj nové problémy.

1.1 Použité pojmy a skratky

sedenie - Sekvencia zobrazení dokumentov ktorá je časovo.

akcia - Jedna elementárna interakcia používateľa so systémom, kliknutie na odkaz, zadanie vyhľadávacieho reťazca.

dokument - Je jedno hudobné dielo reprezentovane tabmy, textom, notamy, prekladom alebo iným spôsobom nápomocným k prevedeniu hudobného diela.

vlastnosť dokumentu - Špecifický črt dokumentu ktorý môže ovplivniť používateľové preferencie.

značka dokumentu - Otnačená vlastnosť dokumentu ktorá je rozoznávana vyhľadávacím systémom.

preferencia - Je vlastnosť dokumentu ktorú nejakým spôsobom používateľ preferuje.

užitočnosť - Vlastnosť je hodnota určujúca či je daný dokument preferovaný používateľom.

2 Existujúce riešenia

Počas analýzi som našiel niekoľko riešení problému dynamického odporúčania. Každé z týchto riešení poskytuje rôzne výhody a nevýhody pri rôznych oblastiach nasadenia. To ktorý model je najvhodnejší pre nás je ovplivnené cieľovím artiklom a cieľovou skupinou. Pre potreby tejto práce bolo potrebné najskôr vybrať doménu, v ktorej chcem riešenie implementovať. Následne zanalizovát' prístupy v danej oblasti.

2.1 Odporúčanie hudobných dokumentov

Pre potreby tejto práce som zvolil oblasť hudby, avšak nie v úplne klasickom ponímani, zameral som sa na dokumenty umožňujúce hudobníkom naučiť sa hrať určité hudobné dielo. Za základne vlastnosti hudby sa považuje rytmus, sila a farba tónu. Na zaznamenanie týchto vlastnosti vzniklo viacero zápisov podľa potrieb určitých skupín hudobníkov. Odporúčanie v tejto oblasti je pomerne nové a preto sa budem skôr snažiť najsť riešenia z iných oblastí a preskúmať ich aplikovateľnosť v tejto oblasti.

Na presné odporúčanie akého koľvek artiklu, musíme najskôr najsť nejaké jeho vlastnosti na základe ktorý môžeme odporúčať. Keďže táto doména je úzko spojená z hudbov, budem sa snažiť vychádzať z nej.

Najznámejším spôsobom kategorizovania hudby je rozdelenie na žánre a podžánre. Problém pri žánroch a podžánroch je, že neexistuje jednotá definícia ani spôsob ich kategorizácie. Určovanie zánrov ma nasledujúce typy pravidiel

- **formálne a technické pravidlá aplikované na obsah(sila, výška a farba tónu)**,
- **semiotické pravidla** (abstraktný vopred dohodnuty koncept, napríklad politická situácia),
- **pravidla správania sa** (črty správania sa fanúčikov alebo interpretov daného žánru),

- **ekonomické a jurisdikčné pravidlá** (zákonné a právne aspekty ktoré daný žáner podporujú, napríklad český protestsong[1]).

Tieto pravidlá definoval Franco Fabbri[4].

Pravidlá sú pomerne abstraktné a i napriek viacerím pokusom o vytvorenie kompletne ontológie žánrov či už z akademický alebo komerčných kruhov [2], neexistuej jenotná ontológia hudobných žánrov.

Pre potrebý odporúčania by bolo najvhodnejšie automatické určovanie žánru, ako napríklad navrhli autory článku [9], kde analizovali výšku a snažíli sa odhaliť akcent nôt, čo im umožnilo odhaliť rymtus piesne. Následne sa zamerali na určenie jednotlivých časti hudobného diela ako predohra, hlavná časť, refrén a sloha.

Ďaľším prístupom je nechať označovať vlastnosti dokumentu používateľom, tento prístup používa napríklad služba last.fm[3], ktorá sa následne snaží používa najpopulárnejšie značky ako žánre. Tento prístup je bližšie opísany v článku Paual Lamera[7] a je známy pod názvom socialne značenie (angl. social tagging).

Podobnosť odporúčania hudobných dokumentov a hudby

I keď sú tieto dve domény veľmi podobné, existujú rozdiely. Jeden z rozdielov sú ich vlastnosti dokumentov. Na presnú identifikáciu pesníčky nám stačí názov piesne, autor piesne, interpret a prevedenie. Pri hudobných dokumentoch sa môžu okrem týchto vlastnosti líšiť aj type dokuemntov (taby, akordy, text, preklad), prípadne niektoré dokumenty môžu obsahovať iba časť daného hudobného diela (predohra, medzi-hra, refrén, sólo atď.).

Podobnosť z odporúčaním textu

Ďaľšia doména ktorú je možné využiť je odporúčanie textových dokumentov. To je najčastejšie založené na anlíze výskytu slóv v texte. Avšak jediné typy dokumentov ktoré by sa dali takto analizovať sú preklady a texty.

[1] www.wikipedia.sk

[2] https://www.apple.com/itunes/affiliates/resources/documentation/genre-mapping.html

[3] http://www.last.fm/home

2.2 Rôzne prístupy k odporúčaniu

Hlavný účelom odporúčacích systémov je odhadnúť užitočnosť dokumentu pre používateľa[13], pričom v mnohých prípadoch je potrebné užitočnosť dokumentu odhadovať. Užitočnosť ako taká môže závisieť od veľkého množstva parametrov, skopina týhto parametrov sa všeobence nazýva kontextové premenné. Na základe toho ako systém nakladá z danými údajmy, delím odporúčacie systémy na niekoľko kategórií. Hlavnou charakteristikov systému je **funkcia užitočnosti**.

Filtrovanie na základe obsahu

Pri tomto prístupe odporúčame používateľovy dokumenty podobné tým čo sa mu páčili v minulosti. Funkciou užitočnosti dokumentu je teda podobnosť dokumentu z už zobrazenými dokumentmi používateľa. Podobnosť dokumentov sa zisťuje pomocov porovnávania značiek dokumentov.

Kolaboratívne filtrovanie

Toto je najpopulárnejší spôsob implementácie odporúčania. Najjednoduchšia a originálna implementácia odporúča aktívnemu používateľovy dokumenty ktoré sa páčili ľuďom z podobným vkusom. Podobnosť používateľov je založená na histórí hodnotenia dokumentov používateľov. Takže úlohou funkcie užitočnosti je najsť najpodobnejších používateľov a vrátiť dokument ktorý mal najkladnejšie hodnotenia od týchto používateľov.

Najväčším problémom kolaboratívneho filtrovania, tzv. problém studeného štartu. Spočíva v tom že ak pribudne do zbierky nový dokument, nemá ešte žiadne hodnotenia, takže sa nebude nikomu odporúčať.

Demograficke odporúčanie

Tieto odporúčacé systémi odporúčajú používateľovy dokumenty na základe jeho demografického profilu (vek, národnosť, jazyk atď.). Za istú fomru tohoto odporúčania môžeme považovať multijazičnosť dnešných stránok. Zaujímavým príkladom je aj domovska stránka google, ktorá sa vo významné dni zobrazuje v

rôznych krajinách rôzne[1].

Znalostne odporúčanie

Zanlostné odporúčacie systémy odporúčajú na základe znalostí o tom ako nejaká vlastnosť dokumentu ovplivňuje užitočnosť dokumentu pre používateľa. V princípe ide o systém ktorý dáva používateľovy otázky a na základe zistených faktov mu odporučí dokument. Takéto odporúčanie sa najčastejšie využíva pri zákazníckej podpore, veľmi dobrým príkladom na tento prístup je zákaznícka podpora Microsoftu[2].

Komunitné odporúčanie

Toto odporúčanie ja veľmi podobné s kolaboratívným filtrovaním, avšak na rozdiel od neho upradnostňuje implicitne dané priateľstvá medzi používateľmi. Tento druh odporúčania zažíva rozkvet nejmä v poslednej dobe spolu z rozkvetom používania socialnych sieti. V dokumente[13] sa dokonca uvádza že v špecialnych prípadoch sú efektívnejšie ako kolaboratívne filtrovanie. Tento druh odporúčania sa často nazýva aj sociálne odporúčanie. Funkcia užitočnosti v tomto prípade najskôr zistí vzťahy medzi používateľmy a preferencie priateľov používateľa a na základe ich preferencií určí užitočnosť dokumentov pre používateľa.

Hybridné odporúčanie

Toto odporúčanie kombinujé vlastnosti predchádzajúcich prístupov na vyriešenie ich vázjomných problémov. Napríklad časté riešenie je kombinovanie colaboratívneho filtrovania z filtrovaním založeným na obsahu, kedy v podstate filtrovanie na základe obsahu rieši problém studeného štartu pre kolaboratívne filtrovanie. Tak isto sa v poslednej dobe zvikne kombinovať kolaboratívne filtrovanie z komunitným odporúčaním vďaka ich dobrým výsledkom.

[1] http://www.google.com/doodles/
[2] https://support.microsoft.com/sk-sk

Spätná vezba

Aby odporúčacie systémy mali ako odporúčať, potrebujú zistiť preferované vlastnosti dokumentov zo korpusu odporúčaných dokumentov. Z tohot dôvodu je potrebné nejakým spôsobom zistiť ktoré vlastnosti používateľ preferuje. Na toto zisťovanie slúži spätná väzba. Spätnú väzbu v zásade delíme na dve skupiny,

- **implicitná spätná väzba** (je získavanie spätnej väzby používateľa z jeho akcií ktoré nesúvysia priamo z hodnotením, napríklad stiahnutie dokumentu, prípadne jeho vytlačenie, hlavnou výhodou je že nevyžaduje vedomí zásah používateľa, avšak zvyšuje technycké nároky na systém)
- **explicitná spätná väzba** (je vedome ponúknutie spätnej väzby od používateľa, napríklad ak používateľ označí že sa mu dokument páči, podľa [2] zvykne byť efektívnejšia, avšak vyžaduje vedomy zásah používateľa).

Explicitnú spätnu väzbu môžeme ďalej deliť na základe toho, akú hodnotu nám vracia napríklad na **binárnu**, „páči sa my“ a „nepáči sa mi“, alebo **z hodnotu** (napríklad pridelovanie 1 až 5 hviezdičiek). S týmto priamo súvysí aj konštrukcia používateľských profilov.

2.3 Používateľské profily

Ďalšou dôležitou súčasťou odporúčacích systémov je spôsob akým konštruujú používateľské profili. Rôzne druhy profilov umožňujú použítie rôznych algoritmov avšak môžu mať rôzny dopad na pamäťovú či výkonovú stránku systému[2].

Binárny vektor

V tomto prípade sú preferencie reprezentované vektorom v dvojrozmernom priestore kde jeden rozmer sú značky dokumentov a druhý sú používatelia. Vektor je tvorený binárnou hednotou kde 1 na pozícií p_{ij}, pri predpoklade že j je identifikátor j.-teho používateľa a i je identifikátor i-tej značky, znamená že i-ta značka je preferencia používateľa j. V tabuľke 1 môžeme vidieť príklad v ktorom máme používateľa p_0 ktorý nepreferuje žiadné značky, následne Používateľa p_1 ktorý preferuje značky z_1, z_2 a Používateľa p_2 ktorý preferuje značku z_0.

z	p_0	p_1	p_2
z_0	0	0	1
z_1	0	1	0
z_2	0	1	0

Tabuľka 1: *Ukážka modelu profilu ako binárneho vektora*

Vahovaný vektor

Tento profil je veľmy podobný z predchádzajúcim profilom, avšak namiesto bynarných hodnôt sú hodnotamy súradníc užitočnosti daných preferencií 2.

z	p_0	p_1	p_2
z_0	0	0	2
z_1	0	2	0
z_2	0	5	1

Tabuľka 2: *Ukážka modelu profilu ako váhovaného vektora.*

Trojrozmenrý vektor

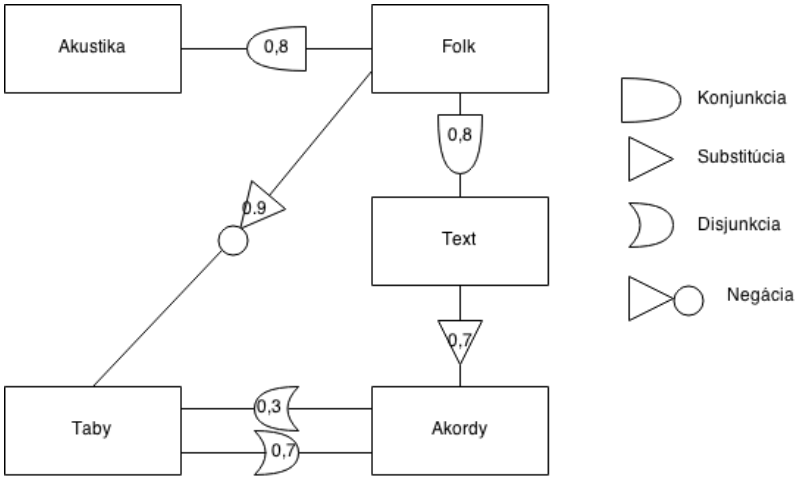
V prípade že vieme aj rozdeliť značky do domén alebo kontextov, môžeme do vektorového profilu pridať ešte jednú súradnicu ktorá reprezentuje doménu dnej značky. Napríklad ak odporúčame text a značky sú reprezentované slovamy ktoré sa našli v texte, tak značke ktorá pochádza z nadpisu môžeme automaticky priradiť väčšiu hodnotu.

Profil sémantickej siete

Profil sémantickej siete (angl. Semantic network profile) je semantická sieť ktorá je vybudovaná pre konkrétneho používateľa a vyjadruje vzťahy medzi značkamy ktoré používateľ preferuje.

Sémantická sieť[17] je orientovaný graf v ktorom sú vrcholmi značky, zatiaľ čo hrany sú ich vzťahy. napríklad v [2] sú použité vzťahy typov.

- konjunkcia,



Obr. 1: *Ukážka sémantickej siete*

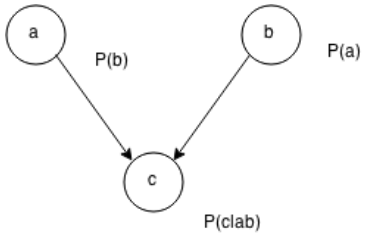
- disjunkcia,
- substitúcia,
- negácia.

Takéto semantické siete sa používajú najmä pri dopĺňaní slov do vyhľadávacích fráz, kde v prípade krátkej vyhľadávacej frázy môžeme zredkuvať jej viacznačnosť doplnením slóv ktoré majú v sémantickej sieti najsilnejšiu pozitívnu väzbu. Na obrázku 1 môžeme vidieť príklad takejto siete.

Teda ak by mi prišiel od používateľa výraz *folk*, tak môžem výraz rozšíriť slovamy *akustika* a *text*. Keď budem pokračovať, slovo text sa dá nahradiť slovom akordy (keďže v podstate akordy sú text doplnení o skratky akordov). Následne však taby už nemôžem doplniť aj keď majú disjunktný vzťah z akordamy, pretože majú silnú negativnú väzbu z folkom. Teda výsledný výraz by bol *folk a akustika akordy*.

Bayesova sieť

Ďalšou možnosťou ako ukladať používateľské dáta je bayesová sieť. Bayesová sieť vychádza z bajesovej teorémy, o ktorej využití v kontexte odporúčania sa budeme zaoberať neskôr. Bayesová sieť slúži na výpočet pravdepodobnosti hypotézy



Obr. 2: *Ukážka bayesovej siete*

pri zmene jej evidencie. Jej vrcholmy sú latentné premenné (závisia od ostatných premenných) a jej hranamy ich vzťahy. Takže v pípade že sa mi zmení hodnota jednej premennej v grafe, po hranách viem upraviť hodnoty všetkých premenných ktoré od nej závysia.

Napríklad na obrázku 2 môžeme vydieť bayesovú sieť zloženú z troch premenných a, b a c. Tieto sú vo vzájomnom vzťahu. V prípade zmený hodnoty a alebo b sa hodnota c automaticky prepočíta.

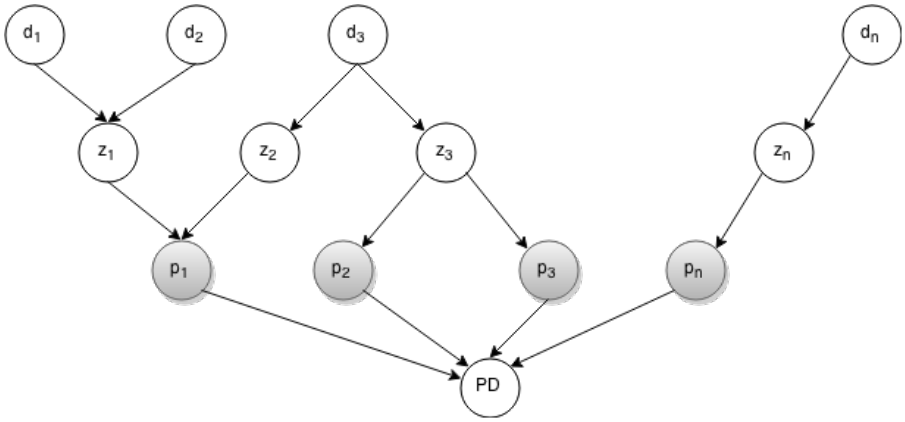
Ako príklad ako využiť takúto sieť môžem uviesť zjednodušený príklad z prezentácie [5], kde najhornejšie vrcholy (tie ktoré nezávyseli od iných premenných) boli dokumenty $D = d_1, d_2...d_n$ a pod nimy boli ich značky $Z = z_{1,z\ 2}...d_n$, následne ak sme chcely odporúčať (príklad je aplikovaný na vyhľadávací dotaz, avšak dotaz môže byť nahradení používateľským profilom), na najspodnejšie body sme pripojili preferencie používateľa $P = p_1, p_2...p_n$. Náčrt siete môžeme vydieť na obrázku 3.

Pričom značky a dokumenty si môžem držať predpočítané v pamäti, a tak isto aj používateľské preferencie. V prípade odporúčania ich iba poprepájam.

Bayesov theorem

Bayesová teoréma sa zaoberá vyplivom nových poznatkov na existujúce domienky o určitej hypotéze. Vďaka nej vieme kombinovať nové dáta z existujúcimy poznatkami. Matematicky je táto teoréma vyjadrená v kontexte vyhľadávania pomocov rovnice 1 a rovnice 2.

$$P(P|d) = \frac{P(d|P) * P(P)}{P(d)} \tag{1}$$



Obr. 3: *Ukážka odporúčacej bayesovej siete*

$$P(NP|d) = \frac{P(d|NP) * P(NP)}{P(d)} \tag{2}$$

Rovnice obsahujú,

- pravdepodobnosť že dokument je užitočný P,
- prevdepodobonsť že dokuemnt nie je užitočný NP,
- pravdepodobnosť že vrátený dokument d je užitočný $P(P|d)$,
- $P(d|P)$ prevdepodobnosť že ak je vrátený užitočný dokument, je to dokument d,
- pravdepodobnosť vrátenia užitočného dokumentu $P(P)$,
- pravdepodobnosť výberu dokumentu $P(d)$,
- pravdepodobnosť neužitočnosti dokumentu $P(NP|d)$,
- pravdepodobnosť $P(d|NP)$ že ak je ak je vrátený neužitočný dokument, je to dokument d,
- $P(NP)$ je pravdepodobnosť že je vrátený neužitočný dokument,
- pravdepodobnosť $P(d)$ vrátenia dokumentu d.

To či je dokumentu užitočný sa následne určuje tým, či je $P(P|d)$ väčšie ako $P(NP|d)$.

2.4 Váhovanie značiek

Ak mám dokument alebo profil reprezentovaný značkamy, je potrebné zistiť ako veľmi sú tieto značky prefereované. Nie včetky značky majú rovnakú váhu, takže potrebujeme dosiahnuť aby užitočnosť dokumentu závisela od unikátnosti značiek v ňom. Základy váhovania sú popísané v prezentácií Hinrich Sch¨utze [15].

Frequencia pojmov

Frequencia pojmov (angl. Term Frequency ďalej TF), je jeden z najjednoduchších a najstarších prístupov k váhovaniu značiek, pri tomto prístupe sa jednoducho počíta počet výskytov značiek v dokumente. Existuje niekoľko druhov tohoto váhovania.

Binárne váhovanie znamená že napríklad, nepočítam počet výskytov slova v texte ale beriem iba či sa v texte nachádza alebo nie. Matematicky vyjadrené rovnicou **??**.

$$w_{TF_BIN}(t_i) = \begin{cases} 1 & \text{ak } t_i \in d_j \\ 0 & \text{ak } t_i \notin d_j \end{cases} \tag{3}$$

Kde d_j je dokument a t_i je slovo.

Čistá frequencia sa dá tiež použiť, v tomto prípade sa váha slova určuje podľa počtu jeho výskytov v texte. Matematicky vyjadrené rovnicou 4

$$w_{TF_RAW}(t_i) = t_{i_{d_j}} \tag{4}$$

Kde $t_{i_{d_j}}$ je počet výskytov slova t_i v dokumente d_j.

Logaritmická váha sa taktiež používa najmä kôly tomu, že relevancia dokumentu nerastie proporcionalne z počtom výskytov slova v dokumente. Toto môžem matematicky vyjadriť napríklad rovnicou **??**.

$$w_{TF_LOG}(t_i) = \begin{cases} 1 + \log 10 t_{i_{d_j}} \text{ ak } t_{i_{d_j}} > 0 \\ 0 \qquad\qquad\qquad\quad \text{inak} \end{cases} \tag{5}$$

Existuje ešte viac spôsobov ako sa dá vyhodnotiť frequencia pojmov, medzi ktoré patrí napríklad dvojita normalizácia 0.5 (angl. double normalization 0.5) alebo k-dvojita normalizácia (angl. double normalization K) [15].

Frequencie pojmov, inverzna frequencia dokumentov

Frequencia pojmov, inverzná frequencia pojmov (angl. Term Frequency, Inverse Document Frequency ďalej TF*IDF) je prístup, pri ktorom zahrňujeme do užitočnosti aj počet dokumentov, ktoré majú danú značku. Základom je zníženie užitočnosti často sa vyskytujúcim značkám. Toto znižovanie je reprezentované rovnicou 6 ďalej IDF.

$$idf_i = log10\frac{N}{dt_i} \tag{6}$$

Kde dt_i je počet dokumentov v ktorých sa pojem t_i nacháza.

Výsledna rovnica TF*IDF je rovnica 7 ktorá je vlastne súčin TF a IDF.

$$w_{TF*IDF} = w_{TF} * log10\frac{N}{dt_i} \tag{7}$$

Presonalizované BM25 váhovanie

Tento model je jeden zo štatystických modelov. Tu uvedená verzia je jeho modifikácia podľa S. Cronen a spol. [3] ktorá je matematický reprezentovaná rovnicou 8.

$$w_{BM25}(t_i) = \log \frac{(r_{t_i} + 0.5)(N - n_{t_i} + 0.5)}{(n_{t_i} + 0.5)(R - r_{t_i} + 0.5)} \tag{8}$$

,

Kde N je počet všetkých dokumentov, n_{t_i} je počet dokumentov obsahujúcich pojem t_i, R je počet dokumentov ktoré používateľ už navštívil a r_{t_i} je počet dokumentov ktoré už používateľ navštíciľ obsahujúcich pojem t_i

2.5 Evolúcia používateľských preferencií

Preferencie používateľa sa z časom menia, môžu vznikať nové a zanikať staré, prípadne sa vracať predošlé. Na základe toho môžeme používateľské preferencie rozdeliť na

- krátkodobé preferencie,
- dlhodobé preferencie,
- sezónne preferencie.

Odhalenie krátkodobých záujmov je pomerne trivialne, stačí agregovať používateľové záujmi za časové obdobie ktoré považujeme za „krátku dobu“ a vrátiť značky ktoré používateľ preferoval najčastejšie.

Dlhodobé záujmy

Problém dlhodobých záujmov je o dosť komplikovanejší, väčšina riešení ktoré som preskúmal používala na tento problém kombináciu rôznych váhovacích algoritmov a štatystických metód. Napríklad v článku [**?**], kde použíli už spomínané váhovacie algoritmy.

Používateľský profil je reprezentovaný trojrozmerný váhovaným vektorovým profilom a zoznamom zobrazených dokumentov (navštívené url). Následne sa najskôr vytvorí zoznam adekvátnych dokumentov zo značiek. Následne na porovnávanie značiek dokumentov a profilov sa používajú tri rôzne algoritmy.

Jednoduché porovnávanie kde sa vlastne spočítajú váhy značiek ktoré majú spoločné používateľ a dokument podľa vzorca 10.

$$u_j(d_i) = \sum_{t=1}^{N} {}_{z_t} f(z_t) * u(z_t) \tag{9}$$

Vysledkom je funkcia užitočnosti pre dokument i $u_j(d_i)$, N_{z_i} je počet unikátnych značiek v dokumente d_i, $f(z_t)$ je počet výskytvo značky z_t v dokumente a $u(z_t)$ je vypočítaná užitočnosť značky z_t.

Porovnávanie unikátnych značiek, teda zanedbávame váhu určenú váhovacími algoritmamy a iba spočítame unikátne značky podľa vzorca **??**.

$$u_u(d_i) = \sum_{t=1}^{N} {}_{z_i} u(z_t) \tag{10}$$

Jazykový model (angl. Language model) ktorý generuje unigramovy jazykovy model (angl. unigram language model) kde užitočnosť značiek je použitá ako ako frekvencia značiek v rovnicy 11.

$$u_{lm}(d_i) = \sum_{t=1}^{N} {}_{z_i} \log u(z_t) + 1 \sum_{j=1}^{N} {}_{z_i} \tag{11}$$

Ďalej sa ešte výsledne dokumenty poskytnuté jednym z týchto algoritmov ešte filtruju algoritmom PClick, ktorý pracuje z históriou navštívených dokumentov. Tento algoritmus vracia iba dokumenty ktoré používateľ v histó často navštevoval. Matematický je to vyjadrené rovnicou 12. Tento algoritmus berie do úvahy aj vyhľadávacií dotaz.

$$u_{pclick}(d_i) = \frac{|Zobrazenia(d_i, u_j, q_n)|}{|Zobrazenia(q_n, u_j)| + \beta} \tag{12}$$

Kde $Zobrazenia(d_i, u_j, q_n)$ je počet zobrazení dokumentu d_i a $Zobrazenia(q_n, u_j)$ je celkový počet zobrazení dokumentu d_i pre vyhľadávací dotaz q_i.

Ďalším možné riešenie som našiel v článku Ferida Achemoukh a spol. [**?**]. Toto riešenie využíva hľavne štatystické metódy. Používateľa modeluje ako bayesovú sieť ktorú modeluje pre niekoľko sedení ktoré sa skladajú z niekoľkých používateľových interakcií zo systémom.

3 Návrh riešenia

V tejto kapitole sa budeme zaoberať zvolenými technológiamy a abstraktným návrhom aplikácie.

3.1 Platforma

Aplikáciu som sa rozhodol vypracovať ako webovú aplikáciu, najmä z dôvodu jednoduhšej dostupnsoti aplikácie pre používateľov. Nezaťažuje používateľov inštaláciou a tak isto eliminuje potrebu synchronizácie zariadení. Čo by mohol byť problém vzhľadom na očakávany objem dát spravovaný aplikáciou.

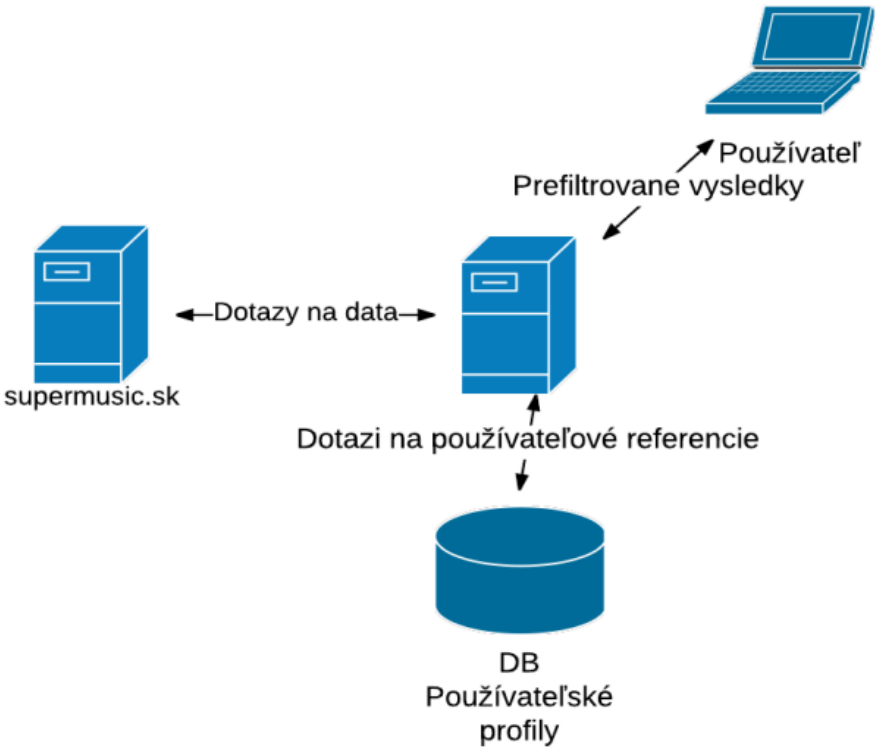
Ďalšou výhodou je prenositeľnosť, nieje potrebne robiť špecialne úpravy pre rôzne operačné systémy a prípadne ich rôzne verzie. Najmä pri použití knižníc ktoré zabezpečujú túto premostiteľnosť ako javascriptová knižnica jqery a na nej postavený css framework bootstrap. Tieto technológie zároveň odstraňujú nutnosť špecialnych úprav aplikácie pre zariadenia z rôzným spôsobmi interakcie (dotykové zariadenia a klasické stolové počítače) a rôzne veľkosti zobrazovacích zariadení.

Na strane servera som sa rozhodol použiť jazyk PHP spolu z jeho aplikačným rámcom (angl. framework) yii2. Tento aplikačný rámec umožňuje premostiteľnosť z viacerímy databázovými riešeniamy ako napríklad MySQL, PostogreSQL alebo SQLite.

Implementácia je postavená okolo dvoch základných funkcionalít, ktoré sa budú navzájom dopĺňať, pôjde vyhľadávanie v hudobných dokumentoch ktorého hlavnou úlohou bude naplniť používateľské profily preferenciami a zostavovanie spevníkov na základe týchto preferencií.

3.2 Vyhľadávanie hudobných dokumentov

Vyhľadávanie bude pracovať nad už existujúcou databázou hudobných dokumentov. V podstate bude využívať už funkčné vyhľadávanie na tejto stránke, akurát na základe používateľových preferencií rozšíri jeho vyhľadávacie reťazce o ďalšie slová ktoré presnejšie špecifikujú používateľov zámer.



Obr. 4: *Náčrt funkčnosti aplikácie.*

3.3 Zostavenie spevníka

Aplikácia bude podporovať funkcionalitu automatického generovania spevníka, kedy si používateľ zvoli používateľov s ktorými si chce ísť zahrať a aplikácia automatický vygeneruje spevník zložený s nejpreferovanejších hudobných diel daných používateľov.

3.4 Krawler ?

Tento komponenet prehľadáva databázu ktorá je cieľom môjho odporúčača, využíva k tomu abecedne zobrazenie záznamov databázy. Databáza sa nedá zobraziť od do, takže granularitu zobrazenie stránok som musel určiť pokusom, najskôr som si zobrazoval všetky troj písmenkove názvy, čo bolo 30*26*26 zobrazení (20280), čo ale trvalo príliš dlho, tak som v tretej sade prehľadával iba každé štvrté písmenko, čo zredukovalo počet stranok na 3380.

3.5 Indexing

Jestvuje veľa spôsobov ako sa dá označovať a vyhľadať obsah, ja som sa počas prieskumu zameral na try:

Priama tagovacia tabuľka

Vytvoril som tabuľku tagov, kde bol každý tag fyzický priamo vložený spolu z id dokumentu ku ktoremu sa viaže, tento prístup ale nebol dostatočne rýchli na vygenerovanie, ani na vyhľadávanie. Pri vyhľadávaní nad 118989 značkamy označujúcimi 47002 dokumentov zabral 44.6984 sekúnd. Nepomohlo ani zindexovanie podľa mena.

Model vektorovho priestoru (angl. vector space model

Pri použití tohto modelu zabral dotaz 0.006 sec.

Tento model sa v MySQL nazýva model prírodzeného jazyka (angl. Natural Language Model), ktorý porovnáva vlastnosti dokumentov na základe abstrakcie priestoru, v ktorom sú jednou dimenziou vlastnosti jedného dokuemntu a druhou

vlastností druhého dokumentu, prípade vyhľadávacieho reťazca alebo používateľ-ský profil. Následne sa vracajú dokumenty ktoré majú najpodobnejší smer vektora k požadovanej fráze.

V MySQL je tento prístup implementovaný pomocov nasledujúcej rovnice[1]:

$$w_d = \frac{\log(dtf_d) + 1}{\sum_{i=1}^{t} \log(dtf_i) + 1} . \frac{U}{1 + 0.0115 * U} . \log \frac{N}{nf}$$

- dtf_d je sila (množstvo koľko krát sa nachádza pojem v text v prípade analízy textu) vlastnosti vyhodnocovaného dokumentu
- dtf_i sila i-tej vlastnosti
- U počet unikátnych vlastnosti dokumentu
- N počet všetkých dokumentov
- nf je počet dokumentov ktoré obsahuje danú vlastnosť

Rovnica sa dá rozdeliť na tri časti.

Základná časť

Je to primárna rovnica určujúca váhu pojmu.

Normalizačný faktor

Spôsobý, že ak je dokument kratší ako preiemerná dĺžka dokuemntu, jeho relevancia stúpa. [16]

Inverzná frequencia

Zabezpečuje že menej časté pojmy majú vyššiu váhu.

[1] http://dev.mysql.com/doc/internals/en/full-text-search.html

3.6 Filtrovanie bezvýznamných značiek (angl. stopwords)

Niektoré slová sú pri vyhľadávaní a indexovaní zbytočné. Síce sa dá použiť tf*idf ktorý redukuje váhu slov na základe ich unikátnosti, ale tieto slová aj tak musí systém spracovať, ja som sa rozhodol použiť kombináciu českých, anglických a slovenských slóv z projektu TODO: Ako ? google code stop-words

3.7 Váhovanie Dokumentu

$$w(d_j) = \sum_{i=1}^{N} w(t_i)$$

- N Počet značiek v dokumente,
- t_i I-ty pojem v dokumente,
- d_j J-ty dokument.

Literatúra

[1] Amr Ahmed, Yucheng Low, Mohamed Aly, Vanja Josifovski, and Alexander J. Smola. Scalable distributed inference of dynamic user interests for behavioral targeting. august 2011.

[2] Peter Brusilovsky. User profiles for personalized information access. School of Information Sciences University of Pittsburgh, USA, 2009.

[3] S. Cronen-Townsend and W. B. Croft. Quantifying query ambiguity. 2002.

[4] Franco Fabbri. A theory of musical genres: Two applications. 1980. [navštívené 27 apríla 2015].

[5] information$_r etrieval. Probabilisticinformationretrievalparti : Survey. [navštívené 10 januara 2015]$.

[6] Karin Kosina. Music genre recognition. 2002.

[7] Paul Lamere. Social tagging and music information retrieval. 2009. [navštívené 24 apríla 2015].

[8] DIK L. LEE. Document ranking and the vector-space model. 1997.

[9] Namunu C. Maddage, Li Haithou, and Mohan S. Kankanhalli. Music structure analysis statistics for popular songs. November 2009.

[10] Prem Melville, Raymond J. Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. júl 2002.

[11] Gašpar Peter. Odporúčanie s využitím osobných vyjadrení. 2014.

[12] Yves Raimond, Samer Abdallah, and Mark Sandler. The music ontology. 2007.

[13] Francesc Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. *RECOMMENDER SYSTEMS HANDBOOK*, chapter Recommendation Techniques, pages 11 – 14. Springer New York Dordrecht Heidelberg London. [navštívené 1 máj 2015].

[14] Arnold Schoenberg. *Fundamentalls of musical composition*. Faber and Faber Limited, 1967.

[15] Hinrich Sch¨utze. Introduction to information retrieval. [navštívené 6 januara 2015].

[16] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. 1996.

[17] John F. Sowa. Semantic networks. [navštívené 4 februára 2015].

[18] Kiryl Tsishchanka. Exponential growth and decay. 2010.

[19] Tong Zhu. Nonlinear p´olya urn models and self-organizing processes. 2009.