**Odporúčací systém pre študentov stredných škôl**

Daniel Ondo

1AIb, 2017 – 2018

**Abstrakt**

Hlavným cieľom tejto práce je oboznámiť sa s existujúcimi metódami odporúčacích systémov a pomocou týchto metód vytvoriť taký odporúčací systém ktorý dokáže študentom odporučiť na základe doteraz získaných výsledkov absolventov taký študijný program na našej škole kde by mali najväčšiu šancu úspešne uspieť. Ďalším cieľom je rozšíriť tento odporúčací systém tak aby sme mohli získať presnejšie odporúčania.

**Kľúčové slová**

Odporúčací systém, kolaboratívne filtrovanie, odporúčanie na základe najbližších užívateľov, odporúčanie na základe najbližšieho obsahu

# Zoznam skratiek a značiek

**RS** – odporúčací systém

**CF** – kolaboratívne filtrovanie

**Akt.U**. – aktuálny užívateľ

**r**- výsledok Pearsanovho koeficientu korelácie

**ra,p** – hodnotenie položky p uživateľom A

**rb,p** – hodnotenie položky p uživateľom B

**(𝑟̅𝑎 )** -priemer všetkých hodnotení uživateľom A

**(𝑟̅b )**-priemer všetkých hodnotení uživateľom B

**𝑝𝑟𝑒𝑑(𝑎, 𝑝)** – predpoveď hodnotení pre položku p uživateľom A

**TPR** - true positive rate

**FPR** – false positive rate

**SVD** - Singular Value Decomposition

# 1. Úvod

Veľa študentov po ukončení gymnázia pokračuje v štúdiu na vysokých školách. Študenti ktorý dosiahli po štyroch rokoch štúdia lepšie výsledky sa hlásia na vysoké školy na západné Slovensko. A práve tu vzniká hlavná motivácia tejto bakalárskej práce. Naša škola by chcela týchto študentov osloviť a ponúknuť im možnosť akú im iná škola neponúkne. Táto možnosť by spočívala v odporúčacom systéme ktorý by dokázal odhadnúť pomocou dosiahnutých výsledkov zo strednej školy úspešnosť na študijnom programe na našej škole pomocou výsledkov študentov ktorý už tento odbor absolvovali. Tento systém by nepomohol len našej škole získať viac študentov ale môže pomôcť aj samotným študentov si vybrať študijné odbory. Keďže veľká časť žiakov ani po absolvovaní strednej školy nevie kde ďalej pokračovať. A práve pre týchto študentov môže byť užitočný tento odporúčací systém a s ktorým by mohli priniesť správne rozhodnutia do budúcnosti.

* 1. **Odporúčací systém**

Hlavnou úlohou RS je generovanie vhodného obsahu na základe existujúcich poznatkov o užívateľovi, ktoré by pre užívateľa mohli byť zaujímavé. Tieto systémy sa začali rozvíjať v minulom desaťročí na podnet veľkých internetových spoločnosti ako napr. Google, Netflix, Amazon, atď.. Keďže v posledných rokoch sa tieto stránky extrémne rozrástli a začali ponúkať množstvo produktov a práve preto bolo potrebné vytvoriť tieto systémy aby zákazníkom ušetrili čas. Vzhľadom nato že RS je relatívne mladá oblasť informatiky existuje už veľa typov RS. Ako napr.:

* Odporúčanie na základe najbližších užívateľov
* Odporúčanie na základe najbližšieho obsahu
* SlopeOneRecommender
* SVDRecommender
* KnnItemBasedRecommender
* TreeClusteringRecommender

Každý z týchto typov majú svoje kladné ale aj záporné stránky. V ďalšej sekcií opíšem niektoré tipy RS.

Taktiež existujúisté modely odporúčania, ktoré sú identifikovateľné v dnešných internetových obchodoch a komunitách. Tieto sa líšia spôsobom zberu informácií, mierou aktivity používateľov a mierou automatizácie.Ako napr.:

* Široké odporúčacie zoznamy
* Komentáre a hodnotenie zákazníkov
* Notifikačné služby
* Odporúčanie založené na asociáciách produktov
* Vysoký stupen personalizácie

**2.Typy RS**

**2.1 Kolaboratívne filtrovanie**

Keďže je to jeden z najstarších typov RS jeho slabiny a možnosti su dobre známe. Základnou myšlienkou CF je generovanie odporúčaní uživateľovi s využitím údajov existujúcej uživateľskej základne.

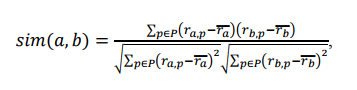
**2.1.1 Odporúčanie na základe najbližších užívateľov** **(User-based nearest neighbor recommendation)**

Funguje na princípe nájdenia užívateľov, ktorý majú podobné záujmy ako aktuálny užívateľ ktorí sa nazývajú aj ako najbližší susedia. Následne hľadáme obsah ktorý zaujal oboch užívateľov.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Položka 1 | Položka 2 | Položka 3 | Položka 4 |
| Akt.U. | 3 | 5 | 2 | **?** |
| Užívateľ 1 | 5 | 2 | 5 | 2 |
| Užívateľ 2 | 3 | 5 | 1 | 5 |
| Užívateľ 3 | 5 | 3 | 4 | 2 |

**Tabuľka 1. Príklad na User-based nearest neighbor recommendation**

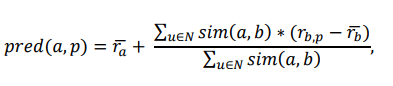
Prvým krokom je výpočet podobnosti aktuálneho užívateľa s ostatnými užívateľmi. Podobnosť medzi dvoma užívateľmi je realizovaná pomocou Pearsonovho korelačného koeficientu:



Výstup Pearsonovho koeficientu korelácie je číslo v intervale od -1 do 1, kde -1 značí, že veličiny sú nepriamo závislé, 1 značí priamu závislosť medzi veličinami a žiadnu závislosť ak je výsledok 0.

Druhým krokom je výber užívateľov , ktorý majú s aktuálnym užívateľom najväčšiu podobnosť.

Posledným krokom je výpočet predpovedi či aktuálny užívateľ preferuje položku alebo nie. Predpoveď vypočítame vzorcom:



Veľkou nevýhodou tohto riešenie je to že pri veľkých serveroch ktoré obsahujú milióny užívateľ a milióny položiek je nereálné aby sa mohli rýchlo vypočítať odporúčania.

**2.1.2 Odporúčanie na základe najbližšieho obsahu(Item-based Collaborative filtering)**

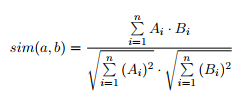
V roku 2001 bola predstavená technika s názvom „Item-based collaborative filtering“, ktorá je prispôsobená pre veľké množstva dát, nakoľko oproti odporúčaniu na základe najbližších susedov umožňuje lepšie využitie predspracovania. Z predspracovaných dát je potom možné generovať odporúčania v reálnom čase.

Základnou myšlienkou algoritmov na základe najbližšieho obsahu je generovanie odporúčaní s využitím podobnosti medzi vecami, obsahom, a nie podobnosti medzi užívateľmi, ako v predchádzajúcom prípade.

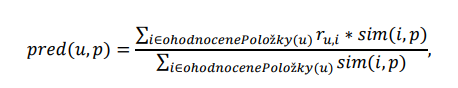
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Položka 1 | Položka 2 | Položka 3 | Položka 4 |
| Akt.U. | 3 | 5 | 2 | **?** |
| Užívateľ 1 | 5 | 2 | 5 | 2 |
| Užívateľ 2 | 3 | 5 | 1 | 5 |
| Užívateľ 3 | 5 | 3 | 4 | 2 |

**Tabuľka 2. Príklad na Item-based Collaborative filtering**

Pre určenie podobnosti medzi vecami je možné využiť viacero algoritmov ako napr. Pearsonov koeficient korelácie, kosínusovú podobnosť alebo upravenú kosínusovú podobnosť. Z experimentálneho porovnania týchto troch algoritmov sa ukazuje, že upravená kosínusová podobnosť je najvhodnejšia metóda pre určovanie podobnosti medzi vecami. Vzorec, pre výpočet kosínusovej podobnosti:



Potom čo sme všetky podobnosti pre všetky položky vypočítali môžeme predpovedať ohodnotenie položky 4 aktualného užívateľa vypočítaním váženej sumy ohodnotenia, ktoré aktualný užívateľ vykonal u položkách ,ktoré boli vybrané ako najviac podobné pomocou vzorcu:



**2.1.3Slope One Recommender**

Slope one recommender je odporúčanie ktoré predikuje potenciálne ohodnotenie položiek aktívnym používateľov na základe ohodnotení ostatných používateľov. Algoritmus Slope One je založene na rozdielu popularity medzi ohodnoteniami položiek používateľov. Jedným z spôsobov ako tento rozdiel určiť medzi dvoma položkami je jednoduché odčítanie ich priemerných ohodnotení. Táto vypočítaná hodnota je ďalej použitá na predpoveď ohodnotenie jednej z týchto položiek ďalším používateľom.

**2.2Nevýhody kolaboratívneho filtrovania**

Týchto problémov môže nastať viac. Niektoré som opísal nižšie a aj s možnými riešeniami.

**2.2.1 Problém studeného štartu**

Tento problém nastáva pri nových používateloch o ktorých nemáme dostatočné informácie , nevieme čo preferujú,čo majú radi a preto nie je možmé vytvoriť odporúčanie. Sú to používatelia ktorý si žiadnu položku nekúpili alebo žiaden produkt neohodnotili. Tento problém je možné vyriešiť pri registrácií nového používateľa kde systém získa informácie o jeho preferenciach. Dalšou možnosťou je využitie hybridných odporúčacích systémov ktoré maju v sebe zakomponované aj jednoduchšie odporúčania(najpopularnejšie produkty,najpredávanejšie produkty,...).

**2.2.2 Problém sivých oviec**

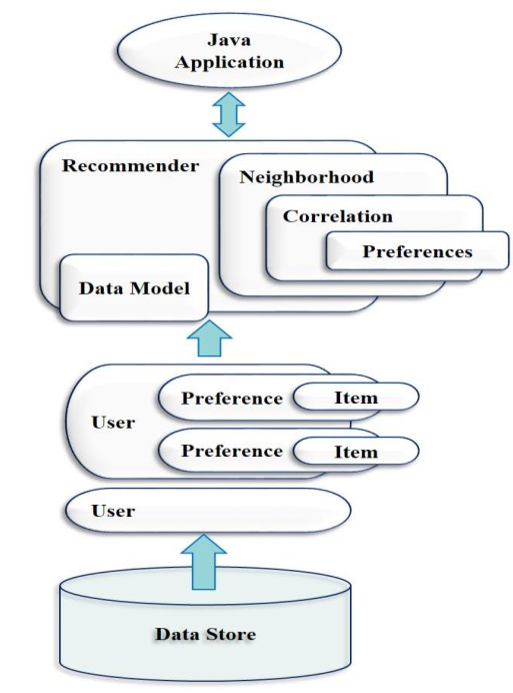
Sivými ovcami rozumieme takých uživatelov,ktorých v poravnaní s ostatnými uživatelmi je tažké zaradiť do určitej skupiny založené na potrebách použivatelov. Títo používatelia v porovnaní s ostatními používatelmi majú nízky korelačný koeficienty ,z toho dôvodu ,že sa ich potreby prekrývajú len z časti. Títo uživatelia predstavujú hrozbu ostatným uživatelom tím že týto používatelia nedostanú presné a správne odporúčania z dvôvodu ovplyvnení vysledkov. Dalším problémom je ten, že môžu v určitých prípadoch ovplyvniť odporúčanie celej komunity. Tento problém je riešený spojením odporúčaní založený na obsahu a kolaboratívneho filtrovania.

**2.2.3 Riedkosť dát**

Pri vytváraní odporúčacieho systému sa predpokladá že budeme mať dostatok informacié a dát k vytvoreniu odporúčaniu. V praxi však to neplatí. V realných aplikáciach kde je prítomné velký súbor položiek, majú matice ohodnotení tendenciu byť velmi riedke,čo spôsobuje to že používatelia ohodnotia len malú časť položiek, ktoré sa nachádzajú v systéme. Tento stav má za následok vytváraní menej presné odporúčanie. Jedným riešením je že sa využijú dodatočné informácie o uživateloch.

**3.Apache Mahout**

Apache Mahout je platforma s otvoreným kódom, poskytujúca nástroje pre strojové učenie, ktorá je schopná pracovať s veľkým množstvom dát a v prípade niektorých algoritmov aj s distribuovaným výpočtom . Na rozdiel od Apache Hadoop nie je Apache Mahout len platforma, ale poskytuje priamo sadu nástrojov, komponent, z ktorých je možné vybudovať, okrem iného, odporúčací systém bez nutnosti vlastnej implementácie odporúčacích a podobnostných algoritmov. Architektúru RS postaveného v Apache Mahout je možné vidieť na obrázku.

****

**Obrázok 1.Apache Mahout architectúra**

Základné komponenty pre tvorbu RS poskytované knižnicou Apache Mahout sú :

**DataModel:** ukladá a pristupuje k dátam o užívateľoch a produktoch, ktoré sú potrebné pre výpočet odporúčania

**UserSimilarity:** poskytuje informáciu o tom, ako sú si dvaja užívatelia podobní na základe rôznych metrík alebo výpočtov

**ItemSimilarity:** poskytuje informáciu o podobnosti produktov pri odporúčaní na základe najbližšieho obsahu

**UserNeighborhood:** poskytuje informácie o užívateľovom najbliž- šom susedstve na základe výstupov z UserSimilarity komponentu

**Recommender:** „spracováva“ dáta z ostatných komponentov a z nich generuje odporúčania

Pre generovanie odporúčaní komponentom Recommender je v Apache Mahout možné využiť viacero techník:

**GenericUserBasedRecommender** – odporúčanie na základe najbližších susedov

**GenericItemBasedRecommender** – odporúčanie na základe najbližšieho obsahu

**SlopeOneRecommender**

**SVDRecommender**

**KnnItemBasedRecommender**

**TreeClusteringRecommender**

Tak ako aj Hadoop, aj Mahout je schopný pracovať v distribuovanom prostredí, vďaka čomu je schopný vykonávať výpočty aj na veľmi rozsiahlych dátových štruktúrach. Z techník kolaboratívneho filtrovania je možný distribuovaný výpočet pre odporúčanie na základe najbližšieho obsahu a maticovej faktorizácie .

**4.Záver**

Teoretická časť tejto práci zahŕňa naštudovanie jednotlivých typov RS. Ako jednotlivé tipy fungujú a v ktorých prípadoch sa ich oplatí využiť. Táto časť práci je úspešne zvládnutá . Praktická časť je rozsiahlejšia. Prvým a najdvôležitejším krokom bola analýza a úprava dát. Kedže dáta boli rozdelé v rôznych excelovských súboroch. Tieto dáta bolo potrebné spárovať po tomto kroku sme ich mohli zredukovať lebo niektoré hodnotenia neobsahovali žiadné informácie o úspešnosti študenta. Nasledujúcim krokom bolo vytvorenie štatistických údajov z týchto dát. Po úprave dát a vytvorení štatistických údajov sme študentov rozdelili do dvoch skupín a to na úspešných ktorý ukončili úspešne štúdium na odbore a na neuspešných. Po tomto kroku sme už mohli jednoducho vyrárať base line pre jednotlivé odbory a to pomocou dvoch hodnôt. Pomocou priemeru známok zo strednej školy úspešných študentov a priemeru známok neuspešných študontov na danom odbore. Z týchto dvoch hodnôt sme získali stredovú hodnotu ktorá nám určuje base line daného odboru. Z tohto údaju vieme vyhodnotiť FPR a TPR.Prvé vyjadruje že mu jeho zvolený program odporúčame a on nebol úspešný a TPR vyjadruje že študent bol úspešný na jeho zvolenom odbore a tento odbor mu odporúčame.Posledným krokom bude získanie odporúčacie skóre pomocou odporúčacích typov ktoré sú naimplementované v knižnici Apache Mahout.

**Poďakovanie**

**RNDr.Erik Bruoth,PhD.**– pomoc a vedenie bakalárskej práce.

**Obrázky:**

[**https://ai2-s2-public.s3.amazonaws.com/figures/2017-08-08/1bc6a7c20997587a7bbf46beff7721c422784e66/2-Figure1-1.png**](https://ai2-s2-public.s3.amazonaws.com/figures/2017-08-08/1bc6a7c20997587a7bbf46beff7721c422784e66/2-Figure1-1.png)

**Zdroje**

<https://is.muni.cz/th/436348/fi_m/jakimov.pdf>

<https://otik.uk.zcu.cz/bitstream/11025/17880/1/dp.pdf>

<https://is.muni.cz/th/359672/fi_m/dp_odporucanie_reklamy.pdf>

<http://www2.fiit.stuba.sk/~kapustik/ZS/Clanky0809/kimak/index.html>

<https://theses.cz/id/f1gzug/Diplomova_prace_Bc_Kortus_Lukas_Final.pdf>