



Politechnika Wrocławska

Wydział Informatyki i Zarządzania

kierunek studiów: Informatyka

specjalność: Projektowanie Systemów Informatycznych

Praca dyplomowa - magisterska

TITLE

TITLE EN

Katatzyna Biernat

słowa kluczowe:

KEYWORDS

krótkie streszczenie:

SHORT ABSTRACT

Promotor:	dr inż. Bernadetta Maleszka
	<i>imię i nazwisko</i>	<i>ocena</i>	<i>podpis</i>

Do celów archiwalnych pracę dyplomową zakwalifikowano do:*

a) kategorii A (akta wieczyste)

b) kategorii BE 50 (po 50 latach podlegające ekspertyzie)

* niepotrzebne skreślić

pieczęć wydziałowa

Wrocław 2016

Niniejszy dokument został złożony w systemie L^AT_EX.

Spis treści

Rozdział 1. Cel pracy	1
Rozdział 2. Wstęp	3
Rozdział 3. Przegląd istniejących rozwiązań	5
3.1. Filtrowanie w oparciu o aktywność użytkownika	5
3.1.1. Explicit/implicit feedback	5
3.1.2. Najczęściej spotykane problemy	6
3.2. Filtrowanie kolaboratywne	6
3.2.1. Najczęściej spotykane problemy	6
3.3. Popularne serwisy wykorzystujące algorytmy rekomendacji	8
3.3.1. Rekomendacja muzyki	8
3.3.2. Rekomendacja filmów	8
3.3.3. Platformy typu e-commerce	9
3.3.4. Inne serwisy	9
Rozdział 4. Model systemu	11
Rozdział 5. Algorytmy	13
5.1. Filtrowanie kolaboratywne	13
5.1.1. Matrix Factorization	13
5.1.2. Biased Matrix Factorization	13
5.1.3. SVD++	13
5.2. Filtrowanie z analizą zawartości	13
5.2.1. Konstrukcja sieci neuronowej	13
5.2.2. Uczenie sieci neuronowej	13
5.3. Algorytmy hybrydowe	13
5.4. Analiza złożoności i poprawności	13
Rozdział 6. Ocena eksperymentalna	15
6.1. Opis metody badawczej	15
6.2. Środowisko symulacyjne	15
6.3. Metodologia	15
6.4. Przeprowadzone eksperymenty	15
Rozdział 7. Wnioski	17
Rozdział 8. CHAPTER 1	19

8.1. SECTION	19
8.2. Section 2	19
8.2.1. Subsection 1	19
Dodatek A. Appendix 1	21
Bibliografia	23

ABSTRACT PL

Streszczenie

ABSTRACT EN

Abstract

Rozdział 1

Cel pracy

Celem pracy jest zaproponowanie i zbudowanie hybrydowego algorytmu rekomendacji. Składowymi docelowego algorytmu są metody kolaboratywnego filtrowania oraz metody filtrowania z analizą treści.

Rozdział 2

Wstęp

Wraz z rozwojem Internetu zmienił się sposób dostępu do informacji. Kiedyś to użytkownik musiał walczyć pozyskanie wiedzy; dzisiaj to informacje walczą u uwagę użytkowników. W świecie zalanym wiadomościami koniecznym wydaje się być zastosowanie filtra, który odsieje interesującą i wartościową zawartość od tej niechcianej. Tak też z pomocą przychodzą zautomatyzowane mechanizmy rekomendacji.

Jednakże sama idea rekomendacji nie jest niczym nowym. Co więcej, zjawisko to możemy zaobserwować w naturze – na przykład wśród mrówek, które podążają wyznaczoną (rekomendowaną) ścieżką feromonową w poszukiwaniu pożywienia.

Ludzie od niepamiętnych czasów posiłkowali się opiniami innych aby ułatwić sobie dokonanie wyboru, od najbliższego grona znajomych do ekspertów i autorytetów.

Wraz z rozwojem nauk informatycznych problem rekomendacji stał się problemem interesującym badaczy. Za pierwszy system rekomendacji uznaje się *Tapestry* stworzony w laboratoriach Xerox Palo Alto Research Center w 1992 roku. Motywacją było odfiltrowanie rosnącej liczby niechcianej poczty elektronicznej [9].

Wkrótce później idea ta została rozszerzona przez takich graczy jak Amazon, Google, Pandora, Netflix, Youtube, Yahoo etc. aż do formy, jaką znamy dzisiaj: systemu, który sugeruje użytkownikom produkty, filmy, muzykę, strony internetowe na podstawie ich aktywności w sieci [16].

Wielkie koncerny internetowe stale poprawiają jakość swoich algorytmów rekomendacji. Najlepszym przykładem jest tutaj Netflix, który w październiku 2006 zorganizował ogólnodostępny konkurs na najlepszy algorytm. Zadaniem uczestników było ulepszenie algorytmu Cinematch. Już po siedmiu dniach od ogłoszenia konkursu trzy zespoły zdołały przebić Cinematch o 1.06% [2][3]. 18 września 2009 Netflix ogłosił, że zespół BellKor's Pragmatic Chaos poprawił Cinematch o 10,06% osiągając wynik $RMSE = 0.8567$. Tym samym wygrał nagrodę w wysokości \$1,000,000 i zakończył konkurs [1][4].

Systemy rekomendacji ulepszone są nieustannie, o czym świadczy chociażby organizowana rokrocznie konferencja *ACM International Conference on Recommender Systems*. Tematyka ta poruszana jest także na konferencjach *European Conference on Information Retrieval*, *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases* i wielu innych. Mimo dużego stopnia

zaawansowania wciąż istnieje pole manewru do ulepszania algorytmów rekomendacji i co za tym idzie zwiększanie zadowolenia użytkowników, które z kolei prowadzi do osiągania korzyści biznesowych.

Rozdział 3

Przegląd istniejących rozwiązań

Tradycyjnie wyróżniamy następujące techniki rekomendacji: [7]

- **filtrowanie w oparciu o aktywność użytkownika** (eng. content-based), technika koncentrująca się na danych historycznych. Użytkownikowi rekomendowane są elementy, które podobne są do tych wybieranych przez niego w przeszłości;
- **filtrowanie kolaboratywne** (eng. collaborative filtering), technika polegająca na odnajdywaniu użytkowników o podobnych gustach i sugerowaniu lubianych przez nich elementów aktualnie aktywnemu użytkownikowi;
- **filtrowanie demograficzne** (eng. demographic), technika koncentrująca się na sugerowaniu aktywnemu użytkownikowi elementów popularnych pośród użytkowników z tej samej okolicy bądź w podobnym przedziale wiekowym;
- **filtrowanie z analizą domeny wiedzy** (eng. knowledge-based), technika dobierająca kolejne elementy na podstawie określonej domeny wiedzy na temat tego, jak dany element spełnia potrzeby i preferencje użytkownika;
- **filtrowanie z analizą społecznościową** (eng. community-based), technika dobierająca rekomendacje dla użytkownika w zależności od preferencji innych użytkowników z jego sieci społecznościowej. W myśl zasady ”powiedz mi kim są twoi przyjaciele a powiem ci kim jesteś”;
- **hybrydowe systemy rekomendacji**, to kombinacja dowolnych powyższych technik.

Każda z tych technik ma swoje wady i zalety w zależności od kontekstu, w którym ma być stosowana.

3.1. Filtrowanie w oparciu o aktywność użytkownika

@TODO

3.1.1. Explicit/implicit feedback

Informacje na temat preferencji użytkownika mogą być zbierane na różne sposoby. Jeżeli użytkownik jawnie pozostawia informacje można mówić o bezpośredniej infor-

macji zwrotnej (explicit feedback). Do takich informacji należą: ocena konkretnych elementów, tzw. łapka w górę lub w dół, komentarz itp.

Jednakże nawet jeżeli użytkownik nie jest skory do zostawiania tego typu śladów, to i tak można wiele na jego temat wywnioskować korzystając z informacji zwrotnych niejawnych (implicit feedback). System bierze wówczas pod uwagę aktywność użytkownika taką jak: historia zakupów, historia przeglądarki a nawet ruchy myszką. W przypadku serwisu z muzyką czy filmem cenną informacją będzie fakt, czy użytkownik wysłuchał lub obejrzał dany materiał do końca czy też wyłączył go po paru sekundach. [14][18]

3.1.2. Najczęściej spotykane problemy

Aby rekomendacja była skuteczna użytkownik powinien ocenić jak najwięcej elementów. Problematici są zatem użytkownicy, którzy dopiero co dołączyli do serwisu oraz tacy, którzy nie są aktywni i rzadko zostawiają po sobie ślad [13].

3.2. Filtrowanie kolaboratywne

Tradycyjnym i zarazem najprostszym podejściem do metody filtrowania kolaboratywnego jest rekomendowanie aktywnemu użytkownikowi elementów, które inni użytkownicy o podobnym guście uznali za atrakcyjne[7][10]. Użytkownicy o podobnym guście to osoby, które oceniły konkretne elementy podobnie jak aktywny użytkownik.

W przypadku filtrowania kolaboratywnego można wyróżnić dwa główne podejścia: oparte o regułę sąsiedztwa (ang. *neighborhood methods*) oraz oparte o modele ukrytych czynników (ang. *latent factor models*)[17].

Rysunek 3.1 pokazuje filtrowanie kolaboratywne oparte o regułę sąsiedztwa, zorientowane na użytkownika. Joe ocenił pozytywnie trzy filmy. System odnajduje innych użytkowników, którzy także ocenili te trzy filmy i dodatkowo kilka innych. Każdy z tych użytkowników pozytywnie ocenił film „Saving Private Ryan”, zatem jest to pierwsza rekomendacja dla Joe.

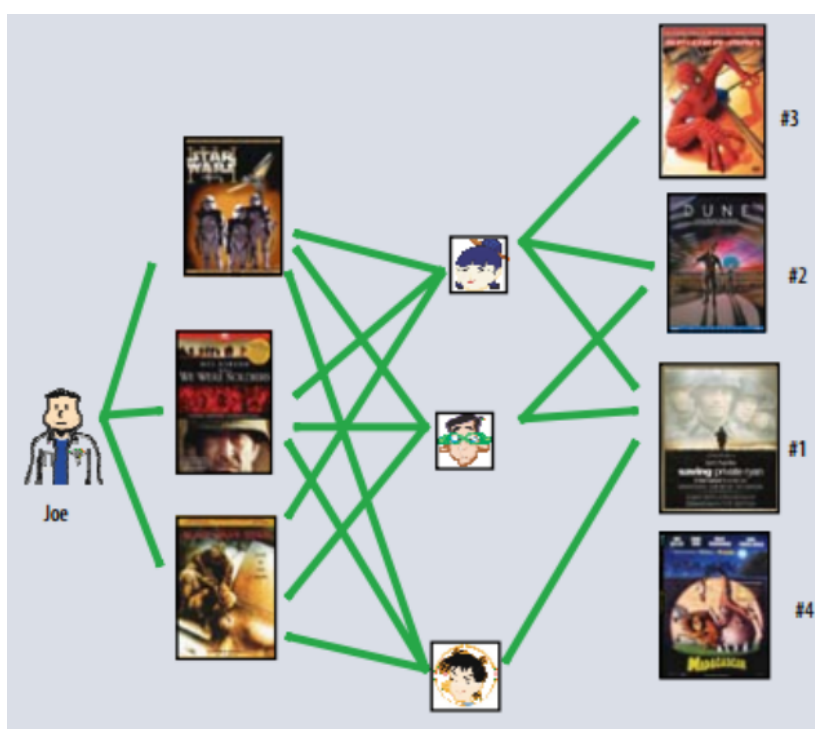
Rysunek 3.2 pokazuje w sposób uproszczony podejście z wykorzystaniem ukrytych czynników. W układzie współrzędnym oznaczeni są użytkownicy wedle swoich preferencji oraz konkretnych cech (np. płeć) a także filmy, które stanowią odpowiedź na dany zestaw preferencji/cech [17].

3.2.1. Najczęściej spotykane problemy

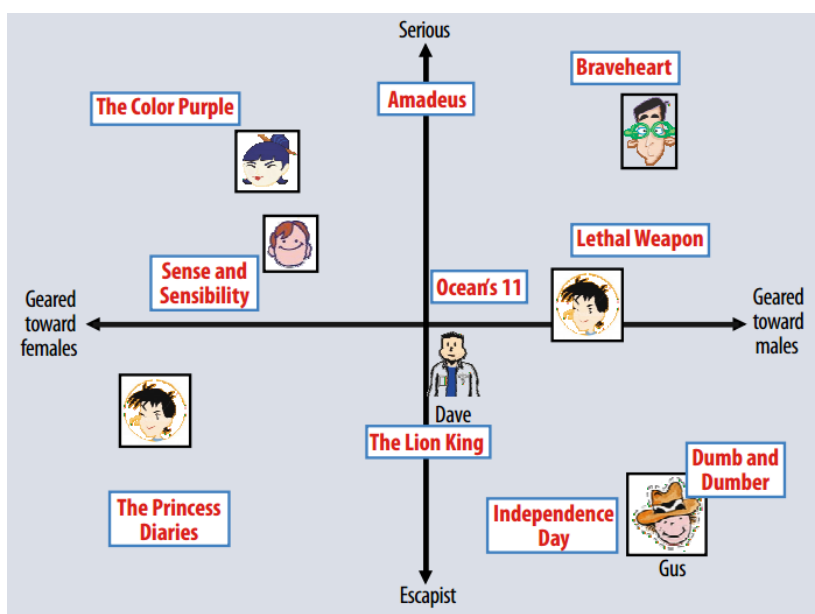
Jednym z problemów klasycznego podejścia do kolaboratywnego filtrowania jest brak uwzględnienia dynamiki zmian w gustach użytkowników. Ten sam użytkownik na przestrzeni kilku lat lub miesięcy może zupełnie inaczej ocenić ten sam film bądź piosenkę. Rozwiązaniem jest dodanie czynnika czasu podczas obliczania wag kolejnych ocen. [11][12][17].

Innym problemem jest tzw. zimny start (eng. cold start). Polega on na tym, że użytkownicy nowi w systemie ocenili zbyt mało elementów, aby można było zbudować dla nich dobre rekomendacje[19][15].

Powszechnym zjawiskiem jest tzw. efekt długiego ogona. Rysunek 3.3 przedstawia jak rozkłada się procentowa ilość ocen danych elementów w zależności od ich popularno-



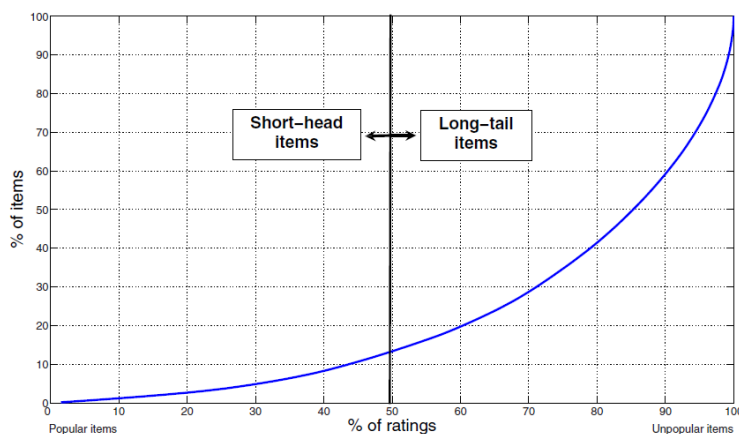
Rys. 3.1: Filtrowanie kolaboratywne metodą sąsiedztwa zorientowanego na użytkownika[17].



Rys. 3.2: Filtrowanie kolaboratywne z wykorzystaniem modeli ukrytych czynników[17].

ści. Jeżeli algorytm rekomendacji nie wspiera mniej popularnych elementów, to istnieje ryzyko, że użytkownicy nie otrzymają możliwości eksplorowania nowych, niszowych materiałów[15][5].

Systemy rekomendacji wykorzystujące filtrowanie kolaboratywne nie są skalowalne.



Rys. 3.3: Problem długiego ogona: 50% ocen dotyczy 10-12% najpopularniejszych elementów w systemie[15].

Złożoność rośnie proporcjonalnie do ilości użytkowników i elementów. Wielkie koncerny internetowe takie jak Twitter wykorzystują klastry i maszyny z bardzo dużą ilością pamięci aby zachować płynność działania serwisu [8].

3.3. Popularne serwisy wykorzystujące algorytmy rekomendacji

@TODO - dokończyć

Algorytmy rekomendacji napotkać można praktycznie w większości dużych serwisów internetowych.

3.3.1. Rekomendacja muzyki

- **YouTube** – serwis powstały w 2005 roku, pozwalający na bezpłatne umieszczanie, odtwarzanie, ocenianie i komentowanie filmów. Od 2006 roku przejęty przez Google [?]. YouTube buduje profil użytkownika w oparciu o jego aktywność w serwisie. Brane pod uwagę są polubienia (łapka w górę), subskrypcje, udostępnianie a także informacje czy użytkownik obejrzał film do końca czy tylko pewien jego procent [?]. Techniki rekomendacji stosowane przez serwis to przede wszystkim asocjacyjna eksploracja danych i licznik wspólnych odwiedzin danego wideo w czasie trwania pojedynczej sesji [?].
- **LastFM** – internetowa radiostacja oferująca rozbudowany mechanizm rekomendacji piosenek "Audioscrobbler".
- **Pandora**

3.3.2. Rekomendacja filmów

- **Netflix**
- **Filmweb**
- **IMDB**

3.3.3. Platformy typu e-commerce

- Allegro
- Amazon

3.3.4. Inne serwisy

- Google search

Rozdział 4

Model systemu

Rozdział 5

Algorytmy

5.1. Filtrowanie kolaboratywne

5.1.1. Matrix Factorization

5.1.2. Biased Matrix Factorization

5.1.3. SVD++

5.2. Filtrowanie z analizą zawartości

5.2.1. Konstrukcja sieci neuronowej

5.2.2. Uczenie sieci neuronowej

5.3. Algorytmy hybrydowe

5.4. Analiza złożoności i poprawności

Rozdział 6

Ocena eksperymentalna

6.1. Opis metody badawczej

6.2. Środowisko symulacyjne

6.3. Metodologia

6.4. Przeprowadzone eksperymenty

Rozdział 7

Wnioski

Rozdział 8

CHAPTER 1

8.1. SECTION

Algorytm 1

Alghoritm 1

$T \leftarrow$ text under analysis
for each word $w \in T$ **do**
 $S_w \leftarrow FIND_SENTIMENT(w)$
 if $S_w = POSITIVE$ **then**
 $Sentiment[POSITIVE]++$
 else if $S_w = NEGATIVE$ **then**
 $Sentiment[NEGATIVE]++$
 else
 $Sentiment[NEUTRAL]++$
 end if
end for
return $\arg \max_x Sentiment[x]$

Rys. 8.1: Schema 1

GRAPHIC

8.2. Section 2

8.2.1. Subsection 1

Subsubsection 1
Definicja 1
Definicja - pierwsza

Dodatek A

Appendix 1

Spis rysunków

8.1 Schema 1	19
------------------------	----

Spis wzorów

Spis algorytmów

1 Alghoritm 1	19
-------------------------	----

Bibliografia

- [1] Netflix Prize: Forum. <http://www.netflixprize.com/community/viewtopic.php?id=1537>. Data dostępu: 2016-06-08.
- [2] Netflix Prize (I tried to resist, but...). <https://www.snellman.net/blog/archive/2006-10-15-netflix-prize.html>. Data dostępu: 2016-06-08.
- [3] Netflix Prize Rankings. http://www.hackingnetflix.com/2006/10/netflix_prize_r.html. Data dostępu: 2016-06-08.
- [4] Netflix Prize Rules. <http://www.netflixprize.com/rules>. Data dostępu: 2016-06-08.
- [5] Celma O. *The Long Tail in Recommender Systems*, pages 87–107. Springer-Verlag Berlin Heidelberg, 2010.
- [6] Christian Desrosiers G. K. chapter A Comprehensive Survey of Neighborhood-based Recommendation Methods.
- [7] Francesco Ricci, Lior Rokach B. S. chapter Introduction to Recommender Systems Handbook.
- [8] Gupta P., Goel A., Lin J., Sharma A., Wang D., Zadeh R. Wtf: The who to follow service at twitter. In *Proceedings of the 22nd international conference on World Wide Web*, pages 505–514. ACM, 2013.
- [9] Huttner J. From Tapestry to SVD: A survey of the algorithms that power recommender system. Master’s thesis, Haverford College Department of Computer Science, 05 2009.
- [10] J.B. Schafer, D. Frankowski J. H. S. chapter Collaborative filtering recommender systems, page 291–324.
- [11] JiuJun Cheng, Yingbo Liu H. Z. X. W., Chen F. A New Recommendation Algorithm Based on User’s Dynamic Information in Complex Social Network. *Mathematical Problems in Engineering*, 2015(ID 281629), 2015.
- [12] Ke Ji, Runyuan Sun W. S. X. L. Next-song recommendation with temporal dynamics. *Knowledge-Based Systems*, 88 (2015) 134–143(0950-7051), 2015.
- [13] Maleszka M., Mianowska B., Nguyen N. T. A method for collaborative recommen-

- dition using knowledge integration tools and hierarchical structure of user profiles. *Knowledge-Based Systems*, 47:1–13, 2013.
- [14] Pasquale Lops M. d. G., Semeraro G. chapter Content-based Recommender Systems: State of the Art and Trends.
 - [15] Rubens N., Kaplan D., Sugiyama M. Active learning in recommender systems. In Kantor P., Ricci F., Rokach L., Shapira B., editors, *Recommender Systems Handbook*, pages 735–767. Springer, 2011.
 - [16] Sharma R., Singh R. Evolution of Recommender Systems from Ancient Times to Modern Era: A Survey. *Indian Journal of Science and Technology*, 9(20), 2016.
 - [17] Yehuda Koren, Robert Bell C. V. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
 - [18] Yehuda Koren R. B. chapter Advances in Collaborative Filtering.
 - [19] Zhang H.-R., Min F., He X., Xu Y.-Y. A hybrid recommender system based on user-recommender interaction. *Mathematical Problems in Engineering*, 2015, 2015.