УДК: 004.89

Н.С. Лєсна, С.М. Гайдамака

Харківський національний університет радіоелектроніки, Харків

МЕТОДИ ПОШУКУ ТА ФІЛЬТРАЦІЇ ІНФОРМАЦІЇ З ВИКОРИСТАННЯМ МЕТОДУ КОЛАБОРАТИВНОЇ ФІЛЬТРАЦІЇ

Основна мета цієї роботи присвячена вивченню та аналізу існуючих методів інформаційного пошуку. Докладний огляд методу спільної фільтрації, яка дозволяє обробляти інформацію і використовувати цю інформацію щоб створювати прогнози. Значною перевагою спільної фільтрації є те, що замість того, щоб опитувати кількох друзів про декілька об'єктах, система колаборативної фільтрації може враховувати думки тисяч людей відносно тисячі об'єктів, виробляючи все автоматично і анонімно.

Ключові сова: *колаборативна фільтрація, ранжований список, косинусна міра, системи прогнозування.*

Вступ

У сучасному світі часто доводиться стикатися з проблемою рекомендації товарів або послуг користувачам якої інформаційної системи. У старі часи для формування рекомендацій обходилися зведенням найбільш популярних продуктів: це можна спостерігати і зараз, відкривши той же Google Play.

За останні десять років прикладні програми пройшли шлях від маленьких та порівняно простих додатків до великих та складних систем. Але з часом такі рекомендації стали витіснятися таргетованими (цільовими) пропозиціями: користувачам рекомендуються не просто популярні продукти, а ті продукти, які напевно сподобаються саме їм.

З технологічним способом отримати рекомендацію про товар, фільм або розважальній сайт ми знайомі. Достатньо спитати у друзів. Ми знаємо, що у деяких з друзів смак краще ніж у інших; ми мали можливість впевнитись у цьому тому, що не раз виявлялось, що їм подобається теж саме що і вам. Але по мірі збільшення кількості пропозицій стає все менш практично опитувати невеликі групи людей, оскільки вони можуть просто не знати про всі існуючі варіанти. Саме тоді на допомогу приходить на допомогу те що називають колаборативною фільтрацією.

Основний матеріал

Не так давно компанія Netflix проводила конкурс з призовим фондом в 1 мільйон доларів, завданням якого стояло поліпшення алгоритму рекомендації фільмів. Як же працюють подібні алгоритми?

Зазвичай алгоритм колаборативної фільтрації працює наступним чином: алгоритм обробляє інформацію від великої кількості людей і знаходить в ній меншу групу із такими ж смаками як і у вас. Алгоритм дивиться, які ще речі їм подобаються, об'єднує уподобання і створює ранжований список уподобань.

Розглянемо алгоритм колаборативної фільтрації по схожості користувачів, яка визначається з використанням косинусних заходів.

Припустимо, у нас є матриця оцінок, виставлених користувачами продуктам, для простоти продуктам задані номери 1 – 9 (табл. 1): Задати її можна за допомогою csv-файлу, в якому першим стовпцем буде ім'я користувача, другим - ідентифікатор продукту, третім - виставлена користувачем оцінка. Таким чином, нам потрібен csv-файл з наступним вмістом:

- alex, 1,5.0;
- alex, 2,3.0;
- alex, 5,4.0;
- ivan, 1,4.0;
- ivan, 6,1.0;
- ivan, 8,2.0;
- ivan, 9,3.0;
- bob, 2,5.0;
- bob, 3,5.0;
- david, 3,4.0;
- david, 4,3.0;
- david, 6,2.0;
- david, 7,1.0.

Для початку розробимо функцію, яка прочитає наведений вище csv-файл. Для зберігання рекомендацій будемо використовувати стандартну для python структуру даних dict: кожному користувачеві ставиться у відповідність довідник його оцінок виду «продукт»: «оцінка».

Таблиця 1

Таблиця оцінок продуктів

	1	2	3	4	5	6	7	8	9
alex	5.0000	3.0000			4.0000				
ivan	4.0000					1.0000		2.0000	3.0000
bob		5.0000	5.0000						
david			4.0000	3.0000		2.0000	1.0000		

Інтуїтивно зрозуміло, що для рекомендації користувачеві № 1 – який продукт вибирати потрібно з продуктів, які подобаються якимось користувачам 2-3-4-etc., які найбільш схожі за своїми оцінками на користувача № 1. Як же отримати чисельне вираження цієї «схожості» користувачів? Припустимо, у нас є M продуктів. Оцінки, виставлені окремо взятим користувачем, являють собою вектор в M-мірному просторі продуктів, а порівнювати вектора ми вміємо. Серед можливих заходів можна виділити такі:

- Косинусні міра.
- Коефіцієнт кореляції Пірсона.
- Евклідова відстань.
- Коефіцієнт Танімото.
- Манхеттенська відстань і т.д.

Досить сказати, що в рекомендаційних системах найбільш часто використовуються косинусна міра і коефіцієнт кореляції Танімото. Розглянемо більш детально косинусні міру, яку ми і збираємося реалізувати. Косинусні міра для двох векторів - це косинус кута між ними.

Зі шкільного курсу математики ми пам'ятаємо, що косинус кута між двома векторами - це їх скалярний добуток, поділений на довжину кожного з двох векторів:

$$\cos(\vec{x},\vec{y}) = (\vec{x}\cdot\vec{y}) / \|\vec{x}\|_2 \times \|\vec{y}\|_2 \qquad (1)$$

Реалізуємо обчислення цього заходу, не забуваючи про те, що у нас безліч оцінок користувача представлено у вигляді dict «продукт»: «оцінка»:

```
def distCosine (vecA, vecB):
    def dotProduct (vecA, vecB):
        d = 0.0
        for dim in vecA:
            if dim in vecB:
                d + = vecA [dim] * vecB [dim]
        return d
    return dotProduct (vecA, vecB) / math.sqrt
(dotProduct (vecA, vecA)) / math.sqrt (dotProduct
(vecB, vecB))
```

При реалізації був використаний факт, що скалярний добуток вектора самого на себе дає квадрат довжини вектора - це не найкраще рішення з точки зору продуктивності, але в нашому прикладі швидкість роботи не принципова.

Отже, у нас є матриця переваг користувачів і ми вміємо визначати, наскільки два користувачі схожі один на одного. Тепер залишилося реалізувати алгоритм колаборативної фільтрації.

Вибрати L користувачів, смаки яких найбільше схожі на смаки розглянутого. Для цього для кожного з користувачів потрібно обчислити обрану міру (в нашому випадку косинусну) щодо розглянутого користувача, і вибрати L найбільших. Для Івана з таблиці (табл. 2), наведеної вище, ми отримаємо наступні значення:

Таблиця 2

Результати обчислення

	alex	bob	david	**sum**
ivan	0.5164	0.0000	0.0667	0.5831

Для кожного з користувачів помножити його оцінки на обчислену величину заходи, таким чином оцінки більш «схожих» користувачів будуть сильніше впливати на підсумкову позицію продукту, що можна побачити в таблиці на ілюстрації нижче.

Для кожного з продуктів порахувати суму каліброваних оцінок L найбільш близьких користувачів, отриману суму розділити на суму мір L обраних користувачів. Сума представлена на ілюстрації в рядку «sum» (табл. 3), підсумкове значення в рядку «result».

Курсивом відзначені стовпці продуктів, які вже були оцінені розглянутим користувачем і повторно пропонувати їх йому не має сенсу.

У вигляді формули цей алгоритм може бути представлений як:

$$r_{u,i} = k \sum_{u' \in U} \mathrm{sim}(u,u') r_{u',i}\,, \qquad (2)$$

де функція sim - обрана нами міра схожості двох користувачів, U - безліч користувачів, r - виставлена оцінка, k - нормувальні коефіцієнт:

$$k = 1 / \sum_{u' \in U} |\mathrm{sim}(u,u')| \,. \qquad (3)$$

Було реалізовано механізм рекомендування таким чином, що для створення набору даних необхідні оцінки. Для декількох тисяч людей або предметів це можливо, і буде працювати, але на такому великому сайті як, наприклад, Amazon, мільйони користувачів і товарів, тому порівняння кожного користувача с усіма іншими, а після цього порівняння товарів, яким кожен користувач виставив оцінки, займе недопустимо багато часу. Окрім цього, на сайті, який продає мільйони різних товарів, перекриття смаків може бути дуже малим, тому нелегко вирішити, які користувачі схожі.

Таблиця 3

Загальна таблиця для всіх користувачів

	1	2	3	4	5	6	7	8	9
alex	*2.5820*	1.5492	0.0000	0.0000	2.0656	*0.0000*	0.0000	*0.0000*	*0.0000*
bob	*0.0000*	0.0000	0.0000	0.0000	0.0000	*0.0000*	0.0000	*0.0000*	*0.0000*
david	*0.0000*	0.0000	0.2668	0.2001	0.0000	*0.1334*	0.0667	*0.0000*	*0.0000*
sum	*2.5820*	1.5492	0.2668	0.2001	2.0656	*0.1334*	0.0667	*0.0000*	*0.0000*
result	*4.4281*	2.6568	0.4576	0.3432	3.5424	*0.2288*	0.1144	*0.0000*	*0.0000*

Техніка, яку ми застосовували до цього часу, має назву «колаборативна фільтрація за схожістю користувачів». Альтернатива такому підходу відома під назвою «колаборативна фільтрація за схожістю образів». Коли набір даних дуже великий, колаборативна фільтрація за схожістю образів може давати кращі результати, при тому, що багато обчислень можна виконати заздалегідь, тому користувач отримає рекомендації набагато швидше.

Процедура фільтрації за схожістю образів здебільшого заснована на вже розглянутому матеріалі. Головна ідея полягає в тому, щоб для кожного образу заздалегідь обчислити більшість схожих на нього. Тоді для створення рекомендацій користувачеві достатньо буде знайти ті зразки, яким він виставив найвищі оцінки, і створити зважений список зразків, що були б максимально схожими на ці.

Потрібно звернути увагу на одну суттєву відмінність: хоча на першому кроці необхідно дослідити всі дані, результати порівняння зразків змінюються не так часто, яка результати порівняння користувачів.

Це означає, що не потрібно постійно перераховувати для кожного зразка список схожих на нього. Це можна робити, коли навантаження на сайт невелика, або взагалі на іншому комп'ютері або за допомогою сервісів Cloud. [1]

Висновки

Ми розглянули на прикладі і реалізували один з найпростіших варіантів колаборативної фільтрації з використанням косинусні міри подібності. Важливо розуміти, що існують інші підходи до колаборативной фільтрації, інші формули для обчислення оцінок продуктів, інші заходи схожості.

Коллборативная фільтрація використовує схожість думок різних користувачів для видачі рекомендацій щодо об'єктів. Воно ґрунтується на тому факті, що людські уподобання не розподіляються випадковим чином: в думках групи людей простежуються загальні тенденції.

Значною перевагою спільної фільтрації є те, що замість того, щоб опитувати декількох друзів про декількох об'єктах, система колаборативної фільтрації може враховувати думки тисяч людей у відношенні тисячі об'єктів, виробляючи все автоматично і анонімно.

Список літератури

1. Segaran Toby. Programming Collective Intelligence. / Toby Segaran. – Publisher: O'Reilly Media, August 2007. – 362 p.

2. Jannach D. Recommender Systems / D. Jannach, M. Zanker, A. Felfernig, G. Friedrich An.

3. Introduction. New York: Cambridge University Press 32 Avenue of the Americas, 2011. – 352 p.

4. Melville P. Recommender systems / P. Melville, V. Sindhwani. – Encyclopedia of Machine Learning. 2010.

5. Su X. T.M. Khoshgoftaar Survey of Collaborative Filtering Techniques / X. Su, T.M. . Advances in Artificial Intelligence. 2009.

6. Ricci F. Kantor Recommender Systems Handbook / F. Ricci, L. Rokach, B. Shapira, P.B. Springer, 2011. – 842 p.

7. Koren Y. Collaborative Filtering with Temporal Dynamics / Y. Koren. – KDD'09. 2009.

Надійшла до редколегії 25.04.2013

Рецензент: д-р техн. нвук, доцент К.С. Смеляков, Харківський університет Повітряних Сил ім. І. Кожедуба, Харків.

МЕТОДЫ ПОИСКА И ФИЛЬТРАЦИИ С ИСПОЛЬЗОВАНИЕМ МЕТОДА КОЛЛАБОРАТИВНОЙ ФИЛЬТРАЦИИ

Н.С. Лесная, С.Н. Гайдамака

Основная цель этой работы посвящена изучению и анализу существующих методов информационного поиска. Подробный обзор метода коллаборативной фильтрации, которая позволяет обрабатывать информацию и использовать эту информацию для создания рекомендаций. Значительным преимуществом совместной фильтрации является то, что вместо того, чтобы опрашивать нескольких друзей о нескольких объектах, система коллаборативной фильтрации может учитывать мнения тысяч людей в отношении тысячи объектов, производя всё автоматически и анонимно.

Ключевые слова: *коллаборативная фильтрация, ранжированный список, косинусная мера, системы прогнозирования.*

METHODS OF SEARCHING AND FILTERING WITH USING THE METHODS OF COLLABORATIVE FILTERING

N.S. Lesna, S.M.Gaidamaka

The main objective of this work is devoted to the study and analysis existing methods of information retrieval. Detail reviewing the method of collaborative filtering that allows you to process information and use that information to make recommendations. A significant advantage of collaborative filtering is that, instead of a few friends poll of several objects, collaborative filtering system can take into account the views of thousands of people against thousands of objects, making everything automatically and anonymously.

Keywords: *collaborative filtering, ranked list, cosine measure, the forecasting system.*