

МОДЕЛІ РОБОТИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

Оскільки мережа Internet продовжує зростати з експоненціальною швидкістю, разом із цим зростає і складність багатьох веб-сайтів. Користувальницькі інтерфейси можуть допомогти користувачам знаходити інформацію, яка відповідає їхнім інтересам, шляхом персоналізації веб-сайту. Рекомендаційні системи забезпечують персоналізовану інформацію, вивчаючи інтереси користувача аналізуючи його дії.

Більшість рекомендаційних систем використовують один із двох основних методів: колаборативну фільтрацію або фільтрування на основі вмісту.

Колаборативна фільтрація підходить системам, які ґрунтуються на моделі попередньої поведінки користувачів. Модель може бути побудована виключно з поведінки одного користувача або (більш ефективно) від поведінки інших користувачів, які мають подібні риси. При врахуванні поведінки інших користувачів, колаборативна фільтрація використовує знання про групи, щоб сформувати рекомендації на основі спільних рис. По суті, рекомендації базуються на автоматичній співпраці декількох користувачів і фільтрації на тих, хто виявляє подібні риси або поведінку.

Blogs	Marc	Megan	Elise	Jill
Linux	13	3	11	-
OpenSource	10	-	-	3
Cloud Computing	6	1	9	-
Java Technology	-	6	-	9
Agile	-	7	1	8

Articles read per user

Cluster	1	2	1	2
----------------	---	---	---	---

Рис. 1. Приклад колаборативної фільтрації

Інший метод систем рекомендацій - це фільтрація на основі вмісту. У цьому методі профіль користувача створюється для надання інформації про типи елементів, які користувач любить, на основі ключових слів, які використовуються для опису елементів. Метод фільтрації на основі вмісту створює профіль для кожного елемента (на основі набору дискретних атрибутів та функцій), який використовується для характеристики елемента в системі. Потім система створює профіль для користувача на основі зваженого вектора характеристик елемента (з предметів, які користувач раніше оцінював або купив, і від елементів, які користувач переглядає). Вага означає важливість кожної функції для користувача. Існує багато можливих способів обчислення цих ваг: баєсів класифікатор, кластерний аналіз, дерева рішень, штучні нейронні мережі. Незалежно від методики обчислення ціль вагового вектора однакова - оцінити ймовірність того, що користувачу сподобається запропонований елемент.

Цей підхід може використовувати історію переглядів сторінок, наприклад, які блоги читає користувач та характеристики цих блогів. Якщо користувач часто читає статті про Linux або, ймовірно, залишатиме коментарі в блогах про розробку програмного забезпечення, фільтрація на основі вмісту може використовувати цю історію для ідентифікації та рекомендації аналогічного вмісту (статті на Linux або інших блогах про розробку програмного забезпечення). Цей вміст можна визначити вручну або автоматично вилучати на основі інших методів подібності.

Blogs	Elise
Linux	11
OpenSource	-
Cloud Computing	9
Java Technology	-
Agile	1

Articles read per user

Similar content
Linux
OpenSource
Cloud Computing

Ranked blogs

Рис. 2. Приклад фільтрації контенту

Повертаючись до рисунку 1, зосередимось на користувачі Elise. Якщо використовується рейтинг блогу, який вказує на те, що користувачі, які читають про Linux, також можуть читати про «OpenSource» та «Cloud Computing», можна легко рекомендувати на основі її поточних звичок Elise прочитати про «OpenSource». Цей підхід, показаний на рисунку 2, ґрунтується виключно на вмісті елементів, з якими взаємодіє один користувач, а не на поведінці інших користувачів у системі.

Основною проблемою фільтрації на основі вмісту є те, що цей метод залежний від якості опису метаданих елементів та користувачів. Це приносить багато обмежень системі, які колаборативна фільтрація допомагає обійти. Останній метод не залежить від якості та наявності метаданих елементів, а лише від поведінки користувачів. Таким чином можна зробити висновок, що для систем, які мають повний опис метаданих елементів доцільніше використання фільтрації на основі вмісту, а для систем з обмеженою кількістю даних або її відсутності - колаборативну фільтрацію.