System rekomendacyjny wypożyczalni filmów z rozmytością

Piotr Iśtok
Piotr Jacak

Marzec 2025

Spis treści

1 Opis zagadnienia

W dobie cyfrowej transformacji, gdzie platformy streamingowe oferują miliony filmów, systemy rekomendacyjne stały się kluczowym narzędziem personalizacji. Ich zadaniem jest nie tylko ułatwienie użytkownikom odnalezienia treści dopasowanych do gustu, ale także zwiększenie zaangażowania i satysfakcji odbiorców. Jest to kluczowy problem ogromnych korporacji takich jak np. Netfilx, aby utrzymać użytkowników zainteresowanych filmami w dobie, gdy najmłodsze pokolenia mają ogromne problemy z długością koncentracji uwagi (tzw. "attention span").

Tradycyjne metody rekomendacji, takie jak filtrowanie współpracujące (Collaborative Filtering) czy analiza treści (Content-based Filtering), kiedyś opierały się na ścisłych danych: ocenach, tagach lub metadanych. Jednak rzeczywistość dostarcza wyzwań, które wykraczają poza ramy precyzyjnych algorytmów. Głównym problemem jest niejednoznaczność i rozmytość (ang. fuzziness) danych. Preferencje użytkowników rzadko da się opisać zero-jedynkowymi kategoriami. Bardzo mało osób jest w stanie w jednoznaczny sposób określić swoje preferencje do całego gatunku filmów. Oceny mają charakter stopniowalny. Filmy mogą być np. "tylko częściowo smutne" lub "nieco straszne", jak i również gatunki filmów nie są zawsze jednostkowe np. "sci-fi z elementami komedii". W tym miejscu wkracza logika rozmyta (ang. fuzzy logic) – gałąź logiki, która operuje na pojęciach przybliżonych, takich jak częściowa przynależność do zbioru. Dzięki niej system może interpretować stwierdzenia, które są o wiele bardziej stopniowe takie jak "użytkownik preferuje głównie thrillery, ale czasem sięga po dramaty". Rozmyte systemy rekomendacyjne radzą sobie z niepewnością i gradacją cech, przypisując obiektom (np. filmom) wartości przynależności w przedziale [0, 1], a nie binarnych etykiet. Przykładowo, film "Django" w reżyserii Quentina Tarantino mógłby w 60% należeć do gatunku "western", w 50% do "dramatu", a w 30% do "thrillera", co pozwala na bardziej elastyczne dopasowanie do złożonych preferencji. Integracja rozmytości z AI otwiera drogę do modeli lepiej odzwierciedlających ludzkie myślenie, gdzie granice między kategoriami są płynniejsze od tych zero-jedynkowych, a decyzje opierają się na kompromisach.

2 Opis metody rozwiązania problemu

2.1 Dane i ich przygotowanie

Bardzo ważną częścią tworzenia takiego projektu są również dane, na których będziemy podczas pracy operować. Zdecydowaliśmy się wybrać zbiór danych oferowany przez "GroupLens" z Uniwersytetu Minnesoty. Dataset zawiera 32 miliony ocen 200 tysięcy użytkowników na 85 tysiącach filmów. W zbiorze tym jest również ponad dwa miliony tagów, które pozwolą nam na elastyczność przy wyborze metod tworzenia modelu. Filmy mają sobie przypisane gatunki oraz są ocenione w skali od 1 do 5 przez użytkowników. Do tego każdy film zawiera przyporządkowane sobie ID w bazie filmowej IMDb, więc gdy byłyby potrzebne jeszcze jakieś dodatkowe dane byłaby możlwiość ich znalezienia.

2.2 Collaborative Filtering

Collaborative Filtering (po polsku filtrowanie współpracujące) jest fundamentem współczesnych systemów rekomendacyjnych. Jest on oparty na idei "mądrości tłumu", która mówi, że najlepszą ocenę daje zbiorowa opinia zróżnicowanej i niezależnej grupy osób lub użytkowników, a nie opinia pojedynczego eksperta. Na jej podstawie działają współcześnie takie serwisy jak Quora, Reddit,

czy Stack Exchange. Istotą filtrowania współpracującego jest więc wykorzystanie zbiorowych zachowań użytkowników serwisu do przewidywania indywidualnych preferencji. W przeciwieństwie do metody Content-based Filtering opisanej w sekcji 2.3, która skupia się na cechach obiektów rekomendowanych (w naszym przypadku filmów), Collaborative Filtering ignoruje je i działa tylko i wyłącznie na danych o interakcjach użytkowników z przedmiotami, takimi jak:

- Oceny (np. gwiazdki na IMDb filmów, łapki w górę bądź w dół)
- Częstość zakupów lub wypożyczeń (np. jak często dany film jest wypożyczany)
- Wyświetlenia filmów (np. na takich platformach jak YouTube)
- Czas spędzony podczas oglądania filmu

Na podstawie właśnie tych metadanych algorytm decyduje zgodnie z zasadą, że użytkownicy o podobnej przeszłości będą mieli podobne preferencje w przyszłości, jakie obiekty proponować. Przykładowo, jeżeli użytkownik A ocenił pozytywnie filmy, które oglądał użytkownik B, to filmy, które obejrzał użytkownik B, a nie obejrzał ich użytkownik A, mogą być mu polecone.

Jednak jakie są zalety takiego podejścia do tworzenia systemu rekomendacyjnego?

- Uniwersalność - algorytm ten działa dla rekomendacji dowolnego obiektu, dla którego da się znaleźć jakąś miarę jego oceny.
- Odkrywanie nieoczywistych powiązań - system może połączyć filmy, które albo nie mają wcale wspólnych cech, albo my ich nie jesteśmy w stanie znaleźć, ale i tak użytkownicy je chętnie łączyli (np. filmy z bardzo odległych gatunków filmowych, o innej obsadzie aktorskiej)

Niestety taki sposób tworzenia systemu rekomendacyjcnego ma też swoje wady.

- Cold start (powolny początek) - jest to problem tego systemu rekomendacyjnego, który objawia się tym, że nie da się precyzyjnie rekomendować obiektów, które nie mają żadnych ocen lub nie da się rekomendować obiektów użytkownikom, którzy jeszcze nie weszli w żadne interakcje. Zjawisko można to zaobserwować np. w momencie pierwszego logowania się na platformie streamingowej, która jeszcze uczy się naszych preferencji.
- Sparsity of a user-item matrix (rzadkość danych macierzy użytkownik obiekt) - w realnych implementacjach tego systemu najczęściej użytkownik nie jest w stanie ocenić nawet 10% produktów możliwych do rekomendacji. W 2020 roku serwis Netflix miał w USA dostępnych aż 6 tysięcy filmów i seriali, które oczywiście często się zmieniają, dlatego przeciętny użytkownik nie jest w stanie rzetelnie obejrzeć i ocenić 600 filmów lub seriali, a jak wiemy oferta się bardzo rozrosła przez ostatnie kilka lat.
- Skalowalność - obliczenia dla milionów użytkowników i obiektów są bardzo wymagające sprzętowo i wraz ze zwiększającą się ofertą wymagają ciągłej optymalizacji.

Collaborative Filtering dzieli się na podejścia pamięciowe (memory-based) - jednymi z nich są user-user i item-item - oraz modelowe (model-based) np. drzewa decyzyjne, klasyfikator bayesowski czy sieci neuronowe. Poniżej skupimy się na dwóch głownych metodach pamięciowych.

2.2.1 User-User Collaborative Filtering

Głównym mottem tego systemu jest *"Znajdź ludzi podobnych do ciebie i podsuń ci to, co oni lubią."*. Podejście User-User Collaborative Filtering charakteryzuje się kompleksową analizą podobieństwa między użytkownikami w systemie rekomendacyjnym. Mechanizm działa na zasadzie identyfikacji użytkowników wykazujących zbliżone preferencje i zainteresowania, a następnie wykorzystania tych podobieństw do generowania rekomendacji. Model ten opiera się na założeniu, że użytkownicy o podobnych historycznych wyborach oraz ocenach produktów prawdopodobnie będą mieli zbieżne preferencje również w przyszłości. Algorytm kompleksowo bada wzorce zachowań użytkowników, tworząc wielowymiarową przestrzeń podobieństwa, gdzie każdy użytkownik reprezentowany jest przez wektor swoich dotychczasowych wyborów i ocen. Głównymi zaletami tego podejścia są:

- Intuicyjność
- Łatwość interpretacji
- Wysoki poziom personalizacji rekomendacji
- Dobre wyniki w małych systemach

Jednakże metoda charakteryzuje się także istotnymi ograniczeniami, takimi jak:

- Koszty obliczeniowe
- Aktualizacje w czasie rzeczywistym
- Podatność na manipulacje

Szczególnie należy wyróżnić problem złożoności obliczeniowej oraz wymagania częstej aktualizacji. Dla N użytkowników w systemie, należałoby obliczyć $O(N^2)$ podobieństw, co przy dużej liczbie użytkowników wydaje się nie możliwe do realizacji.

Uważamy, że nasz system, gdyby był wprowadzany do realnego zastosowania operowałby na dużej grupie użytkowników, dlatego nie będziemy korzystać z tego rodzaju Collaborative Filtering.

2.2.2 Item-Item Collaborative Filtering

Głównym mottem tego systemu jest *"Ludzie, którzy ocenili przedmiot X bardzo dobrze, tak jak ty, ocenili również bardzo dobrze przedmiot Y, którego ty jeszcze nie oceniłeś, więc zostanie on tobie polecony"*. Item-Item Collaborative Filtering koncentruje się na analizie podobieństwa między samymi produktami, w przeciwieństwie do podejścia zorientowanego na użytkowników. Kluczowym mechanizmem jest identyfikacja produktów wykazujących zbliżone wzorce ocen i preferencji użytkowników. Metoda ta bada relacje pomiędzy poszczególnymi elementami w systemie, tworząc swego rodzaju sieć powiązań produktowych. Algorytm oblicza stopień podobieństwa między produktami na podstawie ich dotychczasowych ocen, konstruując macierz podobieństwa, która służy do generowania rekomendacji. Głównymi zaletami są:

- Odkrywa nieoczywiste powiązania
- Nie wymaga metadanych
- Mniej kosztowne obliczeniowo od metody user-user

- Przy większej ilości użytkowników od obiektów ocenianych, średnia ocena obiektu często się nie zmienia, dzięki czemu model nie musi być często przebudowywany

Niestety ma również następujące wady:

- Problem powolnego początku, gdy przedmioty nie mają ocen lub użytkownik niczego nie ocenił
- Konieczność przechowywania macierzy podobieństwa między produktami i ocenami użytkowników
- Produkty z małą liczbą ocen mają niską wiarygodność podobieństwa
- Problem z produktami niszowymi lub nowymi

Do generowania rekomendacji wykorzystywany jest następujący wzór

$$\hat{R}_{ui} = \frac{\sum_{j \in N(i)} \text{sim}(i,j) \cdot R_{uj}}{\sum_{j \in N(i)} |\text{sim}(i,j)|} \tag{1}$$

gdzie:

- $\hat{R}_{ui}$ - przewidywana ocena użytkownika u dla przedmiotu i
- $\text{sim}(i,j)$ - podobieństwo między przedmiotami i i j
- R_{uj} - ocena użytkownika u przedmiotu j
- $N(i)$ - zbiór najbardziej podobnych przedmiotów do i

Oczywiście funkcja podobieństwa nie jest jednoznacznie zdefiniowana i może być wykorzystywana jedna z następujących metod:

- Kosinusowa miara podobieństwa
- Korelacja Pearsona
- Indeks Jaccarda

Dzięki wszystkim powyżej wymieniony cechom ta metoda będzie jedną z tych wykorzystanych przy implementacji projektu. Po mimo wad pozostałe metody zostaną tak dobrane, aby je minimalizować.

2.3 Content-based filtering

Content-based filtering, lub tłumacząc na polski, filtrowanie na podstawie treści, analizuje produkty, aby polecać inne pozycje, na podstawie wcześniejszych działań lub opinii użytkownika.

Systemy rekomendacyjne wykorzystują algorytmy uczenia maszynowego oraz techniki analizy danych, aby polecać nowe pozycje i odpowiadać na zapytania. Silnik rekomendacji porównuje profil użytkownika z profilem produktu, aby przewidzieć odpowiednią interakcję użytkownik-produkt i polecić odpowiednią pozycję.

- Profil produktu - reprezentacja produktu w systemie. Składa się ze zbioru cech produktu, które mogą być wewnętrznie ustrukturyzowane lub może być to po prostu opis produktu. W rozważanym przypadku filmy mogą być przechowywane jako zestaw cech: gatunek, data premiery, reżyser, itd.

	Dramat	Komedia	Kryminał
Forrest Gump (1994)	0.9	0.6	0.0
Pulp Fiction (1994)	0.5	0.6	0.7
Ojciec chrzestny (1972)	0.9	0.1	1.0
Kac Vegas (2009)	0.1	1.0	0.4

Tabela 1: Przyporządkowanie kategorii dla filmów

- Profil użytkownika - reprezentacja preferencji i zachowań użytkownika w systemie. Może to być zbiór cech takich jak: polubienia użytkownika, oceny filmów, zapytania, ale także lista oglądanych wcześniej filmów oraz czas ich oglądania.

System rekomendacji CBF reprezentuje użytkowników oraz odpowiednie pozycje w przestrzeni wektorowej. Produkty są konwertowane do wektorów korzystając z metadanych i kluczowych charakterystyk.

2.3.1 Przykład zastosowania CBF

Można rozważyć przykład filmów oraz ich gatunków. To czy film należy do danej kategorii można opisać liczbą w przedziale $[0, 1]$, dzięki czemu skorzystamy z zalet wykorzystania logiki rozmytej. Wartość 1 oznacza, że film z pewnością należy do danego gatunku, wartość 0, że z pewnością nie należy. Przykład w tabeli 1. Analogicznie dla każdego użytkownika należy utworzyć podobny wektor. Na pozycji i-tej będzie wartość z przedziału $[0, 1]$ określająca, jak bardzo użytkownik lubi i-ty gatunek filmu.

Można wyobrazić sobie, że każdy gatunek to jeden wymiar w przestrzeni wektorowej. Wówczas, wektor każdego filmu reprezentuje jego pozycję w tej konkretnej przestrzeni wektorowej. Jeśli dwie pozycje są blisko siebie w określonej przestrzeni wektorowej, to system określi je jako podobne. Ważne jest, aby określić wystarczająco dużo wymiarów. W przeciwnym wypadku, jeśli dwie różne pozycje będą miały te same wartości dla odpowiadających kategorii, system określi je jako identyczne. Również, system poleca użytkownikowi te filmy, których wektory znajdują się najbliżej wektora użytkownika.

2.3.2 Badanie podobieństwa wektorów

Co oznacza, że dwie pozycje są blisko siebie? Aby wyznaczyć odległość między dwoma wektorami, można wykorzystać różne metryki. Poniżej opisane jest kilka z nich (x_i to wartość pierwszego wektora na i-tej pozycji, a y_i to wartość drugiego wektora na i-tej pozycji):

- Metryka euklidesowa: jest najbardziej intuicyjna, mierzy długość hipotetycznej linii łączącej dwa punkty w przestrzeni wektorowej.

$$d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2} \tag{2}$$

- Podobieństwo cosinusowe (cosine similarity): miara kątą pomiędzy dwoma wektorami, osiąga wartości z zakresu $[-1, 1]$. Im wyższa wartość, tym bardziej prawdopodobne, że dwa elementy są podobne. Rekomendowane dla wielowymiarowych przestrzeni wektorowych.

$$cosine(\vec{x}, \vec{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{3}$$

- Iloczyn skalarny: iloczyn cosinusa kąta między dwoma wektorami oraz ich odpowiednich modułów, względem określonego punktu odniesienia (przeważnie początek układu współrzędnych $(0, 0, ..., 0)$. Iloczyn skalarny najlepiej sprawdza się przy porównywaniu elementów o znacznie różnych wartościach wielkości, na przykład w ocenie popularności.

$$dot(\vec{x}, \vec{y}) = \sum_{i=1}^{n} x_i y_i \tag{4}$$

Najczęściej w tej metodzie wykorzystuje się podobieństwo cosinusowe.

2.3.3 Zalety i wady CBF

Zalety:

- Cold start - w przeciwieństwie do systemu CF, system CBF bezproblemowo radzi sobie z nowymi pozycjami. Jest tak, ponieważ system poleca pozycje na podstawie metadanych, a nie na podstawie poprzednich interakcji użytkowników.
- Większa przejrzystość - użytkownik może zobaczyć, dlaczego system polecił akurat ten film - może być ulubiony gatunek, reżyster itd.

Wady:

- Ponownie cold start - system dobrze radzi sobie z nowymi pozycjami, ale słabo z nowymi użytkownikami. Dla nowych użytkowników nie ma zdefiniowanych wektorów gatunków filmów czy poprzednio obejrzanych filmów.
- Często niewystarczająca liczba cech - jeśli użytkownik lubi filmy np. z konkretnym operatorem kamery, a system nie bierze pod uwagę, kto pełnił taką funkcję przy danym filmie, to nie będzie polecał odpowiednich filmów.
- Zbyt oczywiste polecenia - czasami tej metodzie brakuje sposobów, aby polecić użytkownikowi coś nowego i na pierwszy rzut oka nieprzewidywalnego. Takie sposoby oferuje druga metoda, collaborative filtering.

2.4 System hybrydowy

Hybrydowy system rekomendacyjny łączy w sobie zalety dwóch wyżej omówionych systemów rekomendacji:

- Collaborative Filtering
- Content-Based Filtering

Dzięki połączeniu tych metod system eliminuje następujące ograniczenia:

- CBF ma ograniczoną różnorodność - poleca filmy tylko podobne do tych już ocenionych.
- CF ma problem zimnego startu (cold start) - kiedy użytkownik jest nowy i nie ma historii ocen.

2.4.1 Przykład dla rekomendacji filmów

1. Wykorzystanie collaborative filtering:
    - Jeśli użytkownik A i użytkownik B oglądali i polubili te same filmy, to można przypuszczać, że jeśli użytkownik wysoko oceni film X, to użytkowniki B również się spodoba
    - Przykład: użytkownik lubi filmy science-fiction i oglądał *Interstellar* i *Incepcję*, to system może polecić mu film *Blade Runner 2049*.
2. Wykorzystanie content-based filtering
    - Jeśli użytkownik obejrzał filmy Martina Scorsese *Wilk z Wall Street* i *Chłopaki z ferajny* to system może zaproponować film *Czas krwawego księżyca*, również w reżyserii Scorsese.

2.4.2 Działanie hybrydyzacji

Istnieją różne sposoby połączenia obu metod:

- Łączenie wyników - generowanie rekomendacji osobno przez oba systemy, a następnie łączenie ich (przykładowo za pomocą średniej ważonej).
- Użycie jednego modelu jako filtru do drugiego - przykładowo collaborative filtering proponuje listę pozycji, a content-based filtering wybiera te najbardziej dopasowane do użytkownika.
- Połączenie w modelu uczenia maszynowego - dane treningowe zawierają zarówno historię użytkownika (CF) jak i cechy pozycji (CBF) co pozwala na bardziej precyzyjne rekomendacje.

Podsumowując, hybrydowe podejście jest najlepsze, ponieważ obie metody uzupełniają się. Rozwiązany jest problem zimnego startu, a także system może oferować większą różnorodność pozycji, które z pewnością będą dopasowane do preferencji użytkownika.

Literatura

[1] Charu Aggarwal, Recommender Systems: The Textbook, Springer, 2016. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press, 2016.

[2] Elsa Negre, Information and Recommender Systems, Vol. 4, Wiley-ISTE, 2015.

[3] Linden G, Smith B, York J (2003) Amazon.com recommendations: item-to-item collaborative filtering. Industry report, IEEE, pp 76–80

[4] Amita Jain, Charu Gupta (2018) Fuzzy Logic in Recomender Systems, Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications (pp.255-273)