

PRACA INŻYNIERSKA

# System rekommendacji produktów

wykorzystujący filtrację opartą na treści, logikę  
rozmytą i modele probabilistyczne



---

**Piotr Smoła**

Nr albumu: 125162 • Kierunek: Informatyka

Promotor: dr inż. Piotr Grochowski

# Problem i motywacja

29%

wzrost przychodów Amazon  
po wdrożeniu rekomendacji

## Przeładowanie informacyjne

Tysiące produktów → klient porzuca zakupy

## Czarne skrzynki

Amazon Personalize, Google AI - brak kontroli nad algorytmami

## Brak interpretowalności

Gotowe rozwiązania nie wyjaśniają "dlaczego" polecono produkt



## Rozwiązanie

Modułowy system łączący 3 komplementarne metody rekomendacji - implementacja od podstaw bez gotowych bibliotek ML (TensorFlow, Surprise), zapewniająca pełną kontrolę, interpretowalność i dostosowanie do specyfiki e-commerce.

Praca zespołowa z Dawidem Olko - Filtracja współpracy, analiza sentymencu, reguły asocjacyjne

# Cel i zakres pracy



## Cel główny:

Opracowanie i wdrożenie zaawansowanego mechanizmu rekomendacyjnego w ramach funkcjonalnej platformy e-commerce

### Filtracja oparta na treści

Ważone wektory cech +  
podobieństwo kosinusowe

### Logika rozmyta

System TSK  
6 reguł IF-THEN

### Modele probabilistyczne

Łańcuch Markowa + Naive Bayes

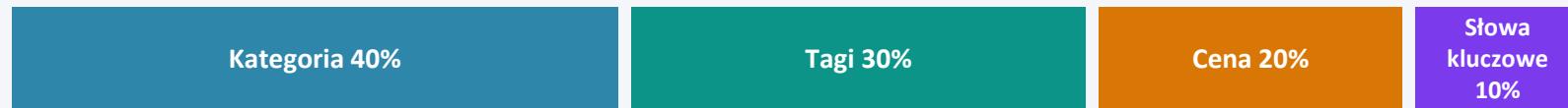
### Dodatkowe cele realizacyjne:

- ✓ Architektura trójwarstwowa: Django + React + PostgreSQL
- ✓ Optymalizacja wydajności (cache, bulk operations, threshold pruning)
- ✓ Ewaluacja na zbiorze: 500 produktów, 20 użytkowników, 200 zamówień

# Filtracja oparta na treści

$$\cos(\theta) = (\mathbf{A} \cdot \mathbf{B}) / (\|\mathbf{A}\| \times \|\mathbf{B}\|) \rightarrow \text{wynik} \in [0, 1]$$

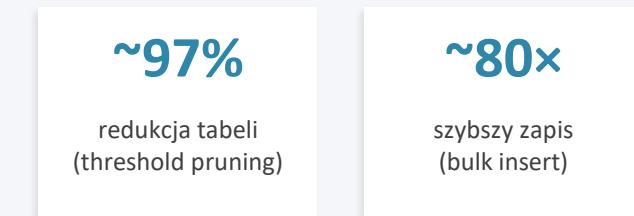
Wagi cech produktu:



Proces:

1. Ekstrakcja ważonych cech → wektor rzadki
2. Obliczenie cosinus similarity dla par produktów
3. Zapis par z similarity > 0.2 (próg 20%)
4. Bulk insert: partie po 1000 rekordów

Wyniki optymalizacji:



Rozwiązuje problem zimnego startu dla nowych produktów - wystarczą cechy (kategoria, tagi, cena)

# Logika rozmyta - Sugeno (TSK)

## 1 Fuzzyfikacja

9 stopni przynależności

## 2 Ewaluacja reguł

T-norma (min)  
T-conorma (max)

## 3 Agregacja

Suma ważona aktywacji reguł

## 4 Defuzzyfikacja

Średnia ważona stałych  
 $\rightarrow$  score  $\in [0,1]$

Zmienne wejściowe ( $3 \times 3$  zbiory rozmyte):

### Cena

cheap · medium · expensive

Progi: 100, 500, 1200, 2000 PLN

### Jakość

low · medium · high

Rating: 2.5 / 3.5 / 4.5

### Popularność

low · medium · high

Zamówienia: 2 / 10 / 30

6 reguł IF-THEN - wybrane przykłady:

Reguła	Warunek IF	Waga
R1: High Quality Bargain	quality HIGH $\wedge$ price CHEAP/MEDIUM	0.9
R2: Popular in Category	category_match HIGH $\wedge$ popularity HIGH	0.7
R4: Category Quality	category_match HIGH $\wedge$ quality MED/HIGH	0.85
R5: Premium Match	user.sensitivity < 0.4 $\wedge$ price EXP $\wedge$ quality HIGH	0.8

+ 2 dodatkowe reguły: Price Sensitive (0.6), Quality-Price Balance (0.75)

## Klasyfikacja: Sugeno zerowego rzędu (TSK-0)

Poprzedniki rozmyte (jak Mamdani), ale następni to stałe liczbowe  $\rightarrow$  średnia ważona zamiast centroidu.

Sugeno-0 jest szczególnym przypadkiem Mamdaniego (Ross, 2010: "Fuzzy Logic with Engineering Applications")

# Modele probabilistyczne

## Łańcuch Markowa (1. rzędu)

*Predykcja sekwencji zakupowych*

$$P(s_j|s_i) = \text{count}(s_i \rightarrow s_j) / \sum_k \text{count}(s_i \rightarrow s_k)$$

- Stany: 48 kategorii produktów
- Trening: pary sąsiadujących stanów z zamówień
- Predykcja: sortowanie wg  $P(\text{przejścia})$ , top-k
- Rozkład stacjonarny:  $\pi = \pi P$
- Wykrywanie cykli (przerywanie >2 powtórzeń)

## Naiwny klasyfikator Bayesa z wygładzaniem Laplace'a

*Predykcja zakupu + ryzyka odejścia*

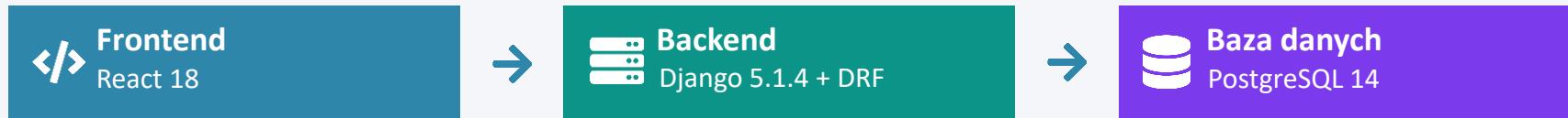
$$P(C|X) = P(C) \times \prod_i P(x_i|C) / P(X)$$

- 5 cech: zamówienia, avg\_order, dni\_od\_ostatniego, kategoria, częstotliwość
- Wygładzanie Laplace'a:  $(\text{count}+1)/(\text{total}+|\mathcal{V}|)$
- Obliczenia w log-przestrzeni (zapobieganie niedomiarowi zmiennoprzecinkowego)
- Predykcja: will\_purchase / will\_not\_purchase
- Ryzyko odejścia: will\_churn / will\_not\_churn

### Integracja:

$\text{score} = 0.6 \times P_{\text{category}}(\text{Markov}) + 0.4 \times P_{\text{purchase}}(\text{NB}) \rightarrow$  Markov lepiej przewiduje kategorię, NB moduluje prawdopodobieństwem zakupu

# Architektura systemu



Docker Compose — 3 kontenery | docker-compose up | Zero konfliktów zależności | Izolacja serwisów

## Mechanizmy optymalizacyjne:

### Cache 3-poziomowy

5min / 30min / 2h

### Bulk operations

~80x szybszy zapis

### Threshold pruning

Redukcja ~97%

### Prefetching

Eliminacja N+1

# Komplementarność metod

Kryterium	Content-Based Filtering	Logika rozmyta	Modele probabilistyczne
Źródło danych	Cechy produktów (kategoria, tagi, cena)	Cechy + profil użytkownika	Sekwencje zakupowe + cechy behawioralne
Problem zimnego startu	Rozwiązyany	Profil domyślny	Wymaga historii
Interpretowalność	Wysoka (wspólne cechy)	100% (reguły IF-THEN)	Średnia (prawdopodobieństwa)
Personalizacja	Niska (cechy produktu)	Wysoka (profil cenowy)	Najwyższa (analiza behawioralna)
Najlepsze zastosowanie	Nowe produkty bez historii	Wrażliwość cenowa + interpretowalność	Predykcja sekwencji + analiza odejścia

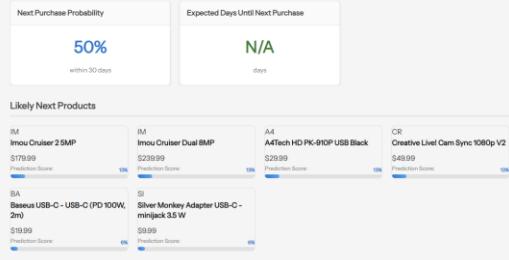
## Kluczowy wniosek

Każda metoda adresuje inny wymiar problemu rekommendacji. CBF dla nowych produktów, Fuzzy Logic dla personalizacji z interpretowalnością, modele probabilistyczne dla głębszej analizy behawioralnej. Optymalne rezultaty osiąga się przez ich synergiczne zastosowanie.

# Prezentacja aplikacji

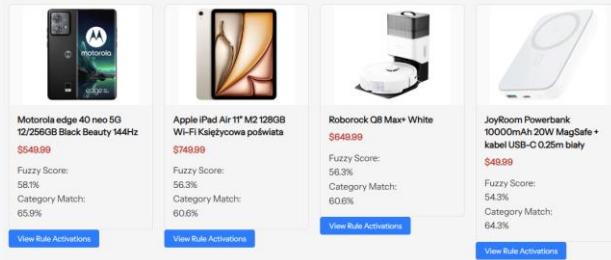
## Next Purchase Predictions

Our Markov chain models analyze your shopping sequences to predict what you're likely to buy next.



## Rekomendacje w panelu klienta - Markow

## Recommended Products



## Rekomendacje w panelu klienta - Logika rozmyta

### Debug Tools - ML Methods Inspector

Inspect internal workings of all 7 machine learning methods used in SmartRecommendations.

Collaborative Filtering    Sentiment Analysis    Association Rules    Content-Based    Fuzzy Logic    Probabilistic

**Content-Based Filtering Debug Information**

Select Product to Analyze: — Select a Product —

Algorithm: Content-Based Filtering (Cosine Similarity)  

$$\text{cost}(B) = (A|B) / (|A| * |B|)$$

Name: Microsoft Surface Laptop Studio i7/32GB/2TB/Geforce

Formula: 
$$\text{cos}(A, B) = \frac{\sum_{i=1}^n a_i b_i}{\sqrt{\sum_{i=1}^n a_i^2} \sqrt{\sum_{i=1}^n b_i^2}}$$

**Database Statistics**

Total Products	500
Saved Similarities	16038
Percentage Saved	64.6%
Threshold	20%

Only similarities > 20% are saved to database



## Filtracja oparta na treści - strona główna

## Panel debugowania - administrator

# Wyniki i ewaluacja

**500**

produktów

**20**

użytkowników

**200**

zamówień

**48**

kategorii  
(stany Markova)

**~4000**

par podobieństw  
(CBF)

## Kluczowe osiągnięcia:

### Implementacja od podstaw

Bez gotowych bibliotek rekomendacyjnych — pełne zrozumienie algorytmów

### 3 komplementarne metody

CBF + Fuzzy Logic + Probabilistyczne — pokrywają wzajemne słabości

### Pełna interpretowalność Fuzzy Logic

Każda rekomendacja wyjaśniona regułami IF-THEN (GDPR)

### Wydajne optymalizacje

Cache 3-poziomowy, bulk ops, threshold pruning — real-time odpowiedzi

### Kompletna platforma e-commerce

Katalog, koszyk, zamówienia, opinie, panel admin + debug algorytmów

# Podsumowanie i rozwój

## Zidentyfikowane ograniczenia

### Cold start:

Markov + NB wymagają historii zakupów (CBF kompensuje)

### Filter bubble CBF:

Rekomenduje tylko podobne - brak odkrywania nowych kategorii

### Skalowalność >10k:

$O(n^2)$  CBF wymaga ANN (LSH/HNSW) dla dużych katalogów

### Brak sezonowości:

Nie uwzględnia kontekstu czasowego

## Kierunki dalszego rozwoju

### Deep learning:

Autoencodery, LSTM/GRU dla lepszych sekwencji

### Meta-learner:

Automatyczny dobór najlepszej metody per kontekst

### Real-time rekommendacje:

Celery + Redis, opóźnienie 30-60s

### Markov wyższego rzędu:

Uwzględnienie 2-3 ostatnich zakupów

## Wniosek końcowy

Implementacja od podstaw jest możliwa i celowa w kontekście edukacyjnym. Zrealizowany system stanowi kompletne rozwiązanie e-commerce z trzema komplementarnymi metodami rekommendacyjnymi, gotowe do wdrożenia w środowisku produkcyjnym.

# Dziękuję za uwagę

---

Piotr Smoła • 125162

System rekomendacji produktów wykorzystujący filtrację opartą na treści,  
logikę rozmytą i modele probabilistyczne

Promotor: dr inż. Piotr Grochowalski • Uniwersytet Rzeszowski, 2026

