KONCEPCJA SYSTEMU ANALIZY RELACJI PRODUKTÓW

Samuel Jendryke

XENOS Schutweg 5, 5145 NP Waalwijk

Luty, 2026

Wprowadzenie

Celem projektu jest stworzenie narzędzia analitycznego, które na podstawie historii zamówień automatycznie identyfikuje grupy produktów często kupowanych razem. Zamiast opierać się na strukturze kategorii lub intuicyjnych założeniach, system będzie wykorzystywał rzeczywiste dane sprzedażowe do wykrywania zależności pomiędzy artykułami.

Projekt jest pierwszym etapem większej koncepcji związanej z przyszłą optymalizacją procesów magazynowych i logistycznych. Na tym etapie nie zajmujemy się jeszcze rozmieszczaniem produktów w magazynie ani symulacją tras kompletacji. Skupiamy się na zbudowaniu solidnej bazy danych o tym, które produkty faktycznie występują razem w zamówieniach oraz jaką mają rzeczywistą wagę operacyjną. Oprócz samych klastrów system pozwoli również określić ich względną ważność na podstawie historii zamówień, co ułatwi interpretację wyników i późniejsze wykorzystanie ich w kolejnych etapach projektu.

System ma działać możliwie bezobsługowo — użytkownik uruchamia analizę i otrzymuje wynik w postaci przypisania produktów do klastrów, ewentualnie podając kilka podstawowych parametrów. Prostota użytkowania nie oznacza jednak uproszczeń technologicznych. Projekt zakłada wykorzystanie zaawansowanych metod analizy grafów i klastrowania, tak aby uzyskane wyniki były jak najbardziej wiarygodne i przydatne w dalszych zastosowaniach.

Takie rozwiązania nie są wymysłem ani eksperymentem. Analiza produktów kupowanych razem jest od lat stosowana w dużych sklepach internetowych i systemach analitycznych. Powód jest prosty — kategorie produktów rzadko pokazują, jak klienci naprawdę robią zakupy. To, że coś jest w tej samej kategorii, nie znaczy, że jest kupowane razem, i odwrotnie.

Po co to robimy? Przy tysiącach SKU bardzo trudno jest zauważyć powtarzalne schematy wyłącznie na podstawie obserwacji. W praktyce wiele decyzji w magazynie opiera się dziś głównie na intuicji i przypuszczeniach, ponieważ brakuje narzędzia, które pokazuje rzeczywiste powiązania między produktami. Automatyczne tworzenie klastrów oraz określenie ich znaczenia w realnych zamówieniach nie rozwiązuje wszystkich problemów, ale daje bardziej obiektywny punkt odniesienia. Dzięki temu kolejne decyzje — czy to logistyczne, czy analityczne — mogą opierać się na danych, a nie tylko na doświadczeniu.

1. Zakres i cele projektu

Zakres projektu obejmuje stworzenie narzędzia, które pozwoli analizować relacje pomiędzy produktami na podstawie historii zamówień i automatycznie tworzyć grupy artykułów często kupowanych razem. Dodatkowym elementem analizy będzie określenie względnej ważności powstałych klastrów na podstawie rzeczywistych danych zamówień, co pozwoli lepiej zrozumieć ich znaczenie operacyjne. Celem jest uzyskanie czytelnych i powtarzalnych wyników, które w przyszłości będą mogły zostać wykorzystane w kolejnych projektach, np. związanych z optymalizacją magazynu. System ma być prosty w użyciu, ale jednocześnie oparty na rzetelnej analizie danych i przygotowany na dalszy rozwój. Projekt od początku zakłada podejście, w którym priorytetem jest jakość, wiarygodność i stabilność wyników — wybór metod i rozwiązań nie będzie uzależniony od łatwości implementacji technicznej, lecz od ich realnej wartości analitycznej.

W ramach projektu powinny zostać uwzględnione:

- wykorzystanie rzeczywistych danych zamówień jako głównego źródła informacji o powiązaniach między produktami,
- opracowanie sposobu automatycznego pobierania danych wejściowych poprzez integrację z jednym z istniejących systemów,
- stworzenie jednego, spójnego i powtarzalnego sposobu grupowania artykułów zamiast polegania na intuicji lub ręcznych decyzjach,
- określenie względnej ważności klastrów na podstawie historii zamówień,
- wykorzystywanie możliwie najlepszych dostępnych metod i algorytmów w celu uzyskania wiarygodnych i stabilnych wyników analizy,
- maksymalne uproszczenie obsługi narzędzia — uruchomienie analizy powinno wymagać jedynie podstawowych działań użytkownika,
- przygotowanie systemu tak, aby w przyszłości mógł pracować na różnych zestawach danych oraz stanowił podstawę kolejnych projektów,
- niezależność wyników od istniejących kategorii produktów,
- możliwość porównywania wyników dla różnych okresów czasu,
- działanie narzędzia w sposób bezpieczny, bez wpływu na bieżące systemy operacyjne,
- prezentowanie wyników w sposób zrozumiały i łatwy do interpretacji,
- ograniczenie dodatkowych zależności technologicznych do minimum.

2. Uproszczony schemat działania systemu

Dane

- pobranie danych zamówień z wybranego okresu w postaci wierszy: order_id, order_date, sku_id, qty (opcjonalnie),
- przygotowanie listy produktów przypisanych do każdego zamówienia,
- wstępne oczyszczenie danych. Pominięcie zamówień zawierających tylko 1 SKU, opcjonalne ograniczenie bardzo dużych zamówień w celu redukcji szumu.

Obrazowo: lista zamówień, gdzie każde zamówienie to zbiór produktów, np.

Zamówienie 1001 → [SKU_A, SKU_B, SKU_C].

Określanie wystąpień razem

- identyfikacja produktów, które pojawiły się w tych samych zamówieniach,
- zliczenie wszystkich par SKU występujących wspólnie.

Obrazowo: tabela pokazująca ile razy dwa produkty były razem, np.

SKU_A + SKU_B → 120 razy,
SKU_A + SKU_C → 15 razy.

Obliczanie wagi między SKU

- wyliczenie siły powiązań między produktami przy pomocy wskaźnika **Lift**,
- odfiltrowanie słabych lub przypadkowych relacji,
- przygotowanie grafu powiązań produktów.

Obrazowo: sieć połączeń między produktami, gdzie linie są grubsze dla produktów częściej kupowanych razem.

Tworzenie klastrów

- pogrupowanie produktów na podstawie grafu relacji z użyciem algorytmu **Louvain**,
- przypisanie każdego produktu do konkretnego cluster_id.

Obrazowo: lista grup produktów, np.

Klaster 1 → [SKU_A, SKU_B, SKU_D],
Klaster 2 → [SKU_X, SKU_Y].

Określanie ważności klastrów

- analiza zamówień pod kątem tego, jak często pojawiają się produkty należące do poszczególnych klastrów,
- nadanie klastrom względnej ważności na podstawie ich udziału w zamówieniach.

Obrazowo: podsumowanie pokazujące znaczenie klastrów, np.

Klaster 1 → 68% zamówień,
Klaster 2 → 21% zamówień,
Klaster 3 → 11% zamówień.

3. Dane wejściowe

Aby możliwe było tworzenie klastrów produktów, niezbędne jest pozyskanie podstawowych informacji o zamówieniach. Minimalny zakres danych powinien być jak najprostszy, tak aby analiza była stabilna i łatwa do utrzymania.

Kluczowym założeniem projektu jest maksymalne uproszczenie procesu po stronie użytkownika. Wprowadzanie danych ręcznie nie wchodzi w grę — przy tysiącach SKU i dużej liczbie zamówień byłoby to niepraktyczne i podatne na błędy. Dlatego dane powinny być dostarczane automatycznie poprzez integrację aplikacji z systemem, który już je posiada.

Wymagane dane

- order_id — identyfikator zamówienia,
- order_date — data zamówienia (do określenia zakresu czasowego analizy),
- sku_id — identyfikator produktu.
- qty (opcjonalnie) — ilość produktu w zamówieniu.
  Na obecnym etapie nie jest ona potrzebna do tworzenia klastrów, jednak warto ją pobierać, aby zachować kompletność danych i umożliwić przyszłe analizy.

Wstępne przygotowanie danych

- przed rozpoczęciem analizy zamówienia zawierające tylko jeden produkt (1-SKU) powinny zostać odfiltrowane, ponieważ nie wnoszą informacji o relacjach między produktami i mogą zaburzać obliczenia statystyczne,
- każde zamówienie traktowane jest jako zbiór unikalnych SKU (bez duplikatów w obrębie jednego zamówienia).

Możliwe źródła danych

- **Opcja A — integracja z WMS Locus (rekomendowana)**
  System Locus posiada archiwalną bazę danych LTT, z którą możliwe jest bezpośrednie połączenie i pobranie danych zamówień z określonego zakresu czasu. To podejście pozwala uzyskać dokładnie te informacje, które są potrzebne do analizy, bez dodatkowych pośredników i bez konieczności rozbudowy istniejącej infrastruktury. Ze względu na prostotę wdrożenia i natychmiastową dostępność danych jest to obecnie najlepsze rozwiązanie.
- **Opcja B — integracja z systemem ORP**
  Alternatywą byłoby połączenie aplikacji z systemem ORP. W tym przypadku konieczne byłoby opracowanie integracji we współpracy z firmą Wolfpack. Może to wiązać się z dodatkowymi kosztami, zależnością od zewnętrznych prac oraz potencjalnym wydłużeniem czasu realizacji projektu.
- **Opcja C — integracja z innymi systemami (np. Yellowfin)**
  Teoretycznie możliwe jest również wykorzystanie danych z innych platform analitycznych, takich jak Yellowfin. Wymagałoby to jednak wcześniejszego sprawdzenia dostępnych metod integracji oraz oceny, czy dane są dostępne w odpowiedniej formie.

Rekomendowane podejście

Ze względu na to, że integracja z bazą LTT systemu Locus zapewnia bezpośredni dostęp do potrzebnych danych i nie wymaga dodatkowych warstw pośrednich, jest to najbardziej racjonalne i najszybsze rozwiązanie na obecnym etapie projektu.

4. Określanie wystąpień produktów razem

Po pobraniu danych zamówień kolejnym etapem jest ustalenie, które produkty faktycznie pojawiają się razem w tych samych zamówieniach. Na tym etapie system nie analizuje jeszcze siły powiązań ani nie tworzy klastrów — jego zadaniem jest jedynie zidentyfikowanie wszystkich współwystąpień produktów.

Każde zamówienie traktowane jest jako lista SKU. Następnie dla każdego zamówienia tworzony jest zestaw par produktów, które wystąpiły razem. W praktyce oznacza to, że jeśli w jednym zamówieniu znajdują się trzy produkty, to zapisujemy wszystkie ich kombinacje jako potencjalne relacje.

Przykład:

Zamówienie 1001 → [SKU_A, SKU_B, SKU_C]

Na podstawie takiego zamówienia powstają pary:

SKU_A + SKU_B
SKU_A + SKU_C
SKU_B + SKU_C

Każde kolejne zamówienie zwiększa licznik wystąpień tych par. W efekcie otrzymujemy prostą tabelę pokazującą, ile razy dane produkty pojawiły się razem.

Obrazowo:

Para produktów	Liczba wspólnych wystąpień
SKU_A + SKU_B	120
SKU_A + SKU_C	15
SKU_B + SKU_D	87

Na tym etapie nie oceniamy jeszcze, czy relacja jest silna czy przypadkowa — zapisujemy wyłącznie fakty wynikające z danych zamówień. Powstała lista współwystąpień stanowi podstawę do kolejnego kroku, w którym obliczana będzie waga relacji między produktami.

5. Obliczanie siły powiązań między produktami

Po ustaleniu, które produkty występują razem w zamówieniach, kolejnym krokiem jest określenie, jak silna jest ta relacja. Samo zliczenie wspólnych wystąpień nie wystarcza — popularne produkty będą pojawiać się razem z wieloma innymi tylko dlatego, że są często kupowane. Dlatego do oceny jakości relacji wykorzystywany jest wskaźnik Lift, a następnie stosowana jest filtracja relacji w celu zachowania tylko wiarygodnych powiązań.

Czym jest Lift

Lift to wskaźnik pokazujący, czy dwa produkty są kupowane razem częściej, niż wynikałoby to z przypadku. Pozwala odróżnić sytuacje, w których produkty naprawdę mają ze sobą związek zakupowy, od takich, gdzie tylko często pojawiają się w zamówieniach, bo są popularne. W praktyce Lift odpowiada na pytanie:

Czy obecność produktu A w zamówieniu zwiększa prawdopodobieństwo pojawienia się produktu B?

Wzór:

$Lift(A,B) = P(A \cap B) / (P(A) * P(B))$

Forma praktyczna:

$Lift(A,B) = (together\_count * total\_orders) / (count\_A * count\_B)$

Filtracja relacji

Po obliczeniu wartości Lift stosowana jest filtracja, której celem jest usunięcie przypadkowych lub zbyt słabych relacji przed budową grafu produktów.

Na tym etapie uwzględniane są w szczególności:

- minimalna liczba wspólnych wystąpień pary produktów (min_edge_support),
- minimalny próg wartości Lift (lift_threshold).

Filtracja dotyczy relacji między produktami, a nie samych SKU — dzięki temu możliwe jest zachowanie klastrów sezonowych lub promocyjnych, przy jednoczesnym ograniczeniu wpływu przypadkowych współwystąpień wynikających z małej liczby zamówień.

Przykładowe relacje po obliczeniu Lift:

Para SKU	Together Count	Lift
SKU_A + SKU_B	120	2.4
SKU_A + SKU_C	3	5.8
SKU_D + SKU_E	18	1.3
SKU_X + SKU_Y	1	9.5

Załóżmy progi:

- min_edge_support = 10
- lift_threshold = 1.2

Po filtracji zostają tylko relacje spełniające oba warunki:

Para SKU	Together Count	Lift
SKU_A + SKU_B	120	2.4
SKU_D + SKU_E	18	1.3

Relacje takie jak SKU_A + SKU_C czy SKU_X + SKU_Y zostają odrzucone — mimo wysokiego Lift — ponieważ liczba wspólnych wystąpień jest zbyt mała, aby uznać je za stabilne statystycznie.

Interpretacja wartości Lift

Lift ≈ 1 — brak realnej relacji,
Lift > 1 — produkty kupowane razem częściej niż losowo,
Lift < 1 — produkty pojawiają się razem rzadziej niż można by oczekiwać.

Przykład:

Liczba zamówień: 10 000

SKU_A występuje w 2000 zamówień
SKU_B występuje w 1000 zamówień
Razem SKU_A i SKU_B wystąpiły w 600 zamówieniach

Lift = (600 * 10000) / (2000 * 1000)
Lift = 3

Wartość 3 oznacza, że produkty pojawiają się razem trzy razy częściej, niż wynikałoby to z przypadku.

Sposób obliczania oraz filtracji relacji na podstawie Lift ma bezpośredni wpływ na jakość klastrów w kolejnym etapie analizy. Dlatego konieczne jest przetestowanie różnych wartości parametrów, takich jak próg Lift czy minimalna liczba wspólnych wystąpień, i wybór najbardziej optymalnej konfiguracji. W niektórych przypadkach sam Lift może okazać się niewystarczający — wtedy analizę można uzupełnić o dodatkowe miary, np. Confidence lub indeks Jaccarda, aby zwiększyć stabilność i wiarygodność wyników.

Po zastosowaniu filtracji powstaje zestaw relacji o wystarczającej sile i stabilności statystycznej, który stanowi podstawę do budowy grafu produktów wykorzystywanego w kolejnym etapie — tworzeniu klastrów.

6. Tworzenie klastrów produktów

Po obliczeniu siły powiązań między produktami kolejnym etapem jest ich pogrupowanie w większe zbiory, czyli klastry. Celem nie jest stworzenie nowych kategorii ani sztucznych podziałów, lecz odnalezienie naturalnych grup produktów wynikających bezpośrednio z danych zamówień.

Do tego zadania wykorzystywany jest algorytm Louvain. Algorytm powstał do analizy dużych sieci powiązań i jest powszechnie stosowany w data science do wykrywania naturalnych skupisk danych.

W naszym przypadku sieć powiązań tworzą produkty, a relacje między nimi wynikają z wcześniejszego etapu obliczeń (Lift). Każdy produkt jest elementem grafu, a siła połączeń odzwierciedla to, jak często produkty występują razem w zamówieniach.

Na czym polega działanie algorytmu

Algorytm analizuje całą strukturę relacji jednocześnie i wyszukuje takie grupy produktów, w których powiązania wewnętrzne są silniejsze niż powiązania z resztą zbioru. Proces przebiega iteracyjnie:

- początkowo każdy produkt stanowi osobną grupę,
- algorytm sprawdza, czy przeniesienie produktu do innej grupy poprawia jakość podziału,
- produkty z silnymi relacjami zaczynają tworzyć klastry,
- struktura jest upraszczana i analizowana ponownie,
- proces kończy się, gdy dalsze zmiany nie poprawiają wyniku.

Liczba klastrów nie jest definiowana ręcznie — wynika bezpośrednio z danych.

Zastosowanie w logistyce i systemach magazynowych

Podejście oparte na analizie relacji między SKU nie jest nowością i stosowane jest w nowoczesnych systemach logistycznych, szczególnie w obszarze slotting optimization i analizy zachowań zamówień.

Przykłady systemów wykorzystujących podobne koncepcje:

- Manhattan Associates WMS — optymalizacja rozmieszczenia produktów na podstawie relacji zamówień,
- Blue Yonder (JDA) — analiza powiązań SKU i rekomendacje layoutu magazynu,
- SAP EWM — zaawansowane strategie slottingu uwzględniające częstotliwość współwystępowania produktów,
- rozwiązania 3PL i e-commerce wykorzystujące tzw. order affinity lub basket analysis.

W wielu magazynach analizy tego typu są ukryte wewnątrz systemów jako część większych modułów optymalizacyjnych. Projektowane narzędzie realizuje tę samą ideę w sposób bezpośredni — tworząc niezależną warstwę analityczną opartą wyłącznie na danych zamówień.

Przykładowy rezultat

Efektem działania algorytmu jest przypisanie każdego produktu do konkretnego klastra:

Sku Id	Cluster Id
SKU_001	1
SKU_002	1
SKU_003	1
SKU_014	2
SKU_015	2
SKU_021	3

Można to również przedstawić jako zestawienie grup:

Klaster	Sku
Klaster 1	SKU_001, SKU_002, SKU_003
Klaster 2	SKU_014, SKU_015
Klaster 3	SKU_021

Wynikiem tego etapu jest uporządkowana lista sku_id → cluster_id. Otrzymane klastry stanowią uporządkowaną warstwę analityczną opisującą rzeczywiste powiązania między produktami i stają się punktem wyjścia do kolejnych etapów analizy, takich jak określanie znaczenia poszczególnych klastrów czy dalsze wykorzystanie danych w przyszłych projektach.

7. Określanie ważności klastrów

Po utworzeniu klastrów możliwe jest określenie ich rzeczywistej ważności operacyjnej na podstawie historii linii zamówień. Celem tego etapu nie jest zmiana struktury klastrów, lecz określenie, które grupy produktów generują największe obciążenie w procesie kompletacji.

Na czym polega analiza

- każda linia zamówienia zostaje przypisana do klastra na podstawie sku_id,
- zliczana jest łączna liczba linii zamówień należących do każdego klastra,
- obliczany jest udział liczby linii danego klastra względem wszystkich linii zamówień w analizowanym okresie,
- uzyskany udział traktowany jest jako miara względnej ważności klastra — im większy udział, tym częściej klaster uczestniczy w operacjach kompletacji i tym większe generuje obciążenie operacyjne.

Przykład:

Linie zamówień:

Zamówienie 1001 → SKU_A (Klaster 1), SKU_B (Klaster 1), SKU_X (Klaster 3)
Zamówienie 1002 → SKU_C (Klaster 1), SKU_Y (Klaster 2)

Zliczenie linii:

Klaster 1 → 3 linie
Klaster 2 → 1 linia
Klaster 3 → 1 linia
Razem → 5 linii

Podsumowanie ważności klastrów:

Klaster 1 → 60% udziału w liniach zamówień
Klaster 2 → 20%
Klaster 3 → 20%

Interpretacja wyników

- wyższy procent oznacza większe rzeczywiste wykorzystanie klastra w pracy magazynu,
- metryka odzwierciedla częstotliwość operacji kompletacji, a nie tylko obecność klastra w zamówieniach,
- dzięki analizie na poziomie linii zamówień ograniczony zostaje wpływ pojedynczych popularnych SKU na wynik końcowy.

Na obecnym etapie jest to analiza dodatkowa, która nie wpływa na sam proces tworzenia klastrów, ale pomaga lepiej interpretować otrzymane wyniki.

8. Możliwe przyszłe zastosowania w magazynie

Obecny projekt kończy się na utworzeniu klastrów produktów oraz określeniu ich względnej ważności. Na tym etapie system nie ingeruje w layout magazynu ani w bieżące procesy operacyjne. Uzyskane dane stanowią jednak solidną podstawę do dalszych analiz i potencjalnych usprawnień w przyszłości.

Najważniejsze możliwe kierunki wykorzystania wyników:

- **Wsparcie optymalizacji layoutu magazynu i tras kompletacji**
  klastry pokazują, które produkty faktycznie występują razem w zamówieniach, co może wspierać decyzje dotyczące rozmieszczenia SKU, projektowania stref operacyjnych oraz skracania tras pickera — zarówno w kompletacji pojedynczych zamówień, jak i w batch pickingu.
- **Optymalizacja uzupełniania lokacji (replenishment)**
  analiza klastrów może pomóc w identyfikacji produktów, które często zużywają się razem. Dzięki temu możliwe staje się:
  - planowanie uzupełnień całych grup produktów zamiast pojedynczych SKU,
  - ograniczenie sytuacji, w których picker trafia na wiele pustych lokacji w tej samej trasie,
  - lepsze planowanie kolejności replenishmentu w zależności od ważności klastra.
- **Obiektywizacja decyzji operacyjnych**
  zamiast opierać się wyłącznie na intuicji, możliwe będzie porównywanie planowanych zmian z rzeczywistymi danymi o relacjach między produktami.
- **Podstawa do przyszłych symulacji i narzędzi optymalizacyjnych**
  klastry oraz ich ważność mogą zostać wykorzystane w kolejnych projektach, np. do symulacji wariantów layoutu magazynu lub analizy wpływu zmian na proces kompletacji.
- **Wykrywanie potencjalnie nieoptymalnych decyzji layoutowych**
  analiza klastrów pozwala identyfikować sytuacje, w których produkty często występujące razem w zamówieniach znajdują się w fizycznie odległych lokalizacjach magazynu. Może to stanowić sygnał do ponownej oceny rozmieszczenia SKU oraz wspierać bardziej świadome decyzje dotyczące zmian layoutu, w oparciu o rzeczywiste dane operacyjne.

Warto podkreślić, że są to potencjalne zastosowania wynikające z możliwości analitycznych projektu. Wdrożenie konkretnych zmian w magazynie wymagałoby osobnego etapu projektowego oraz dodatkowych analiz operacyjnych.