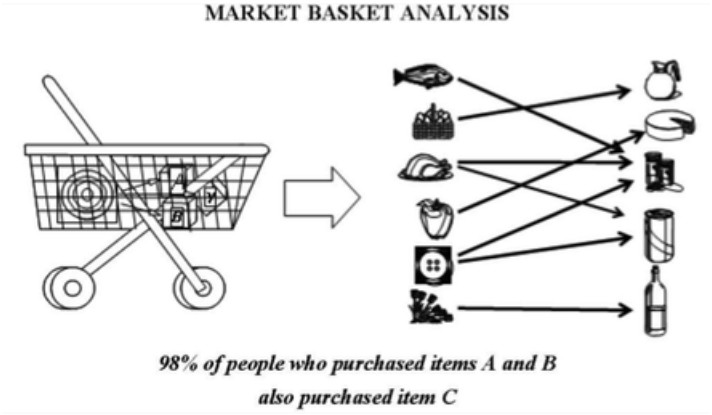
Ćwiczenie 4 (2pkt)

Algorytm Apriori i Reguły Asocjacyjne



Zadanie do wykonania

1) Tworzymy na pulpicie katalog w formacie Imię_nazwisko, w którym umieszczamy wszystkie pliki związane z ćwiczeniem.

2) Czytamy teorię wyliczania zbiorów zdarzeń częstych oraz reguł asocjacyjnych, w razie problemów ze zrozumieniem, analizujemy przykłady na kartce.

3) Generujemy zbiór paragonów za pomocą programu paragon_generator.exe.

4) Do otrzymanego zbioru implementujemy w wybranym języku algorytm Apriori w celu znalezienia zdarzeń częstych, przyjmując próg częstości $\Phi = 2$ **(1pkt)**.

5) Ze zbiorów zdarzeń częstych tworzymy wszystkie możliwe reguły Asocjacyjne **(1pkt)** spełniające warunki,

$$a)\ wsparcie\ reguły * ufność\ reguły \geqslant \frac{1}{10}$$

$$b)\ wsparcie\ reguły * ufność\ reguły \geqslant \frac{2}{10}$$

$$c)\ wsparcie\ reguły * ufność\ reguły \geqslant \frac{3}{10}$$

$$d)\ wsparcie\ reguły * ufność\ reguły \geqslant \frac{4}{10}$$

6) Do wykonania zadania można wykorzystać programy demonstracyjne dostępne w katalogach starter-Cpp lub starter-Csharp,

Algorytm Apriori - Tworzenie częstych zbiorów zdarzeń - teoria

• Rozważając pewien zbiór transakcji D (np zbiór paragonów).

• Ustalamy pewien próg częstości Φ, który mówi nam, że zbiór jest częsty, gdy występuje co najmniej Φ razy w D.

• Znajdujemy zbiór F_1, zawierający jednoelementowe zbiory częste.

• Z kombinacji bez powtórzeń elementów zbioru F_1 tworzymy zbiór C_2, zawierający dwuelementowych kandydatów na zbiory częste.

• Elementy C_2, które mieszczą się w progu Φ tworzą zbiór częstych par F_2.

• Teraz aby znaleźć F_k, Algorytm Apriori tworzy zbiór C_k, k elementowych kandydatów, poprzez łączenie elementów zbioru F_{k-1}, które mają $k-2$ pierwszych wspólnych pozycji.

• W kolejnym kroku przecinamy C_k własnością Apriori. Dla każdego elementu $c \in C_k$ są tworzone i sprawdzane podzbiory o rozmiarze $k-1$, jeżeli dowolny z tych podzbiorów nie jest częsty, c nie może być zbiorem częstym i usuwamy go ze zbioru C_k.

• Spośród pozostałych w C_k kandydatów, sprawdzamy ich częstość i usuwamy te zbiory, które nie przekroczyły progu częstości.

• Kończymy szukanie częstych zbiorów zdarzeń w zbiorze D, gdy F_k zawiera 1 lub nie zawiera żadnego elementu.

Reguły Asocjacyjne - teoria

• Reguły Asocjacyjne wyliczamy ze zbiorów zdarzeń częstych $F_k, F_{k-1}, ..., F_2$ zbioru transakcji D, wybieramy z każdego zbioru częstego $x \in F_k$ podzbiory wielkości $k-1$ i tworzymy reguły postaci,

$$ss \Rightarrow s - ss$$

gdzie, ss jest poprzednikiem (wybranym $k-1$ elementowym zbiorem)
$s - ss$ jest jednoelementowym następnikiem

(Istnieje możliwość budowania reguł, których następnik zawiera więcej niż jeden element, ale najczęściej stosujemy następnik jednoelementowy)

• Reguły możemy wartościować parametrami,

$$Wsparcie\ reguły = \frac{liczba\ obiektów\ ze\ zbioru\ D,\ do\ których\ pasuje\ reguła\ w\ sensie\ poprzednik \Rightarrow następnik}{liczba\ obiektów\ zbioru\ D}$$

$$Ufność\ reguły = \frac{liczba\ obiektów\ ze\ zbioru\ D,\ do\ których\ pasuje\ reguła\ w\ sensie\ poprzednik \Rightarrow następnik}{liczba\ obiektów\ ze\ zbioru\ D,\ do\ których\ pasuje\ reguła\ w\ sensie\ poprzednika}$$

• Jeżeli ustalimy próg jakości reguł $Wsparcie\ reguły * Ufność\ reguły$, akceptujemy tylko reguły, które go przekraczają.

Przykład budowania zbioru zdarzeń częstych (Algorytm Apriori)

Dla zbioru transakcji D postaci,
{kapusta,ogórki,pomidory,kabaczki}
{ogórki,pomidory,kabaczki}
{cytryny,pomidory,woda}
{cytryny,woda,jajka}
{ogórki,grzybki,żołądkowa}
{żołądkowa,ogórki,pomidory}

ustalamy próg częstości $\Phi = 2$

Budujemy zbiory zdarzeń częstych

F_1={{ogórki},{pomidory},{kabaczki},{cytryny},{woda}{żołądkowa}}
Sortujemy F_1 alfabetycznie,
F_1={{cytryny},{kabaczki},{ogórki},{pomidory},{woda}{żołądkowa}}

Teraz kombinacje bez powtórzeń elementów zbioru F_1, tworzą zbiór kandydatów,
C_2={{cytryny,kabaczki},{cytryny,ogórki},{cytryny,pomidory},{cytryny,woda}
{cytryny,żołądkowa},{kabaczki,ogórki},{kabaczki,pomidory},{kabaczki,woda},
{kabaczki,żołądkowa},{ogórki,pomidory},{ogórki,woda},{ogórki,żołądkowa},
{pomidory,woda},{pomidory,żołądkowa},{woda,żołądkowa}}

Do F_2 trafiają ci kandydaci, którzy mają częstość przynajmniej 2,
F_2={{cytryny,woda},{kabaczki,ogórki},{kabaczki,pomidory},
{ogórki,pomidory},{ogórki,żołądkowa}}

Teraz łączymy elementy zbioru F_2, o $k-2$ pierwszych identycznych pozycjach, czyli w tym przypadku na pierwszej pozycji, tworząc zbiór kandydatów,
C_3={{kabaczki,ogórki,pomidory},{ogórki,pomidory,żoładkowa}}

Zbiory zawarte w C_3, przecinamy własnością Apriori, czyli odrzucamy kandydatów, którzy zawierają podzbiór długości 2, nie będący elementem zbioru F_2

Sprawdzamy
Dla {kabaczki,ogórki,pomidory}, podzbiory długości 2 są postaci,
{kabaczki,ogórki} jest w F_2
{kabaczki,pomidory} jest w F_2
{ogórki,pomidory} jest w F_2

Sprawdzamy zbiór

{ogórki,pomidory,żoładkowa}
podzbiory długości 2 są postaci,
{ogórki,pomidory} jest w F_2
{ogórki,żołądkowa} nie jest w F_2, czyli nie jest zbiorem częstym w sensie $\Phi = 2$
{pomidory,żołądkowa} jest w F_2
stąd zbiór {ogórki,pomidory,żoładkowa}, nie jest częsty usuwamy go ze zbioru C_3

C_3={{kabaczki,ogórki,pomidory}}

Teraz sprawdzamy, czy pozostali w C_3 kandydaci mają w D częstość 2,

Zbiór {kabaczki,ogórki,pomidory} ma w D częstość 2, stąd,

F_3={{kabaczki,ogórki,pomidory}}, F_3 zawiera jeden element, warunek stopu został spełniony, kończymy algorytm Apriori.

Liczymy reguły Asocjacyjne

Naszym progiem jakości będzie $wsp * ufn \geqslant \frac{1}{3}$
gdzie, $ufn = ufność\ reguły$, $wsp = wsparcie\ reguły$

ze zbioru F_3 mamy
kabaczki $\wedge$ ogórki $\Rightarrow$ pomidory $wsp = \frac{1}{3}$, $ufn = \frac{2}{2} = 1.0$, $wsp * ufn = \frac{1}{3}$
kabaczki $\wedge$ pomidory $\Rightarrow$ ogórki $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$
ogórki $\wedge$ pomidory $\Rightarrow$ kabaczki $wsp = \frac{1}{3}$, $ufn = \frac{2}{3}$, $wsp * ufn = \frac{2}{9}$

ze zbioru F_2 mamy
cytryny $\Rightarrow$ woda $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$
woda $\Rightarrow$ cytryny $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$
kabaczki $\Rightarrow$ ogórki $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$
ogórki $\Rightarrow$ kabaczki $wsp = \frac{1}{3}$, $ufn = \frac{1}{2}$, $wsp * ufn = \frac{1}{6}$
kabaczki $\Rightarrow$ pomidory $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$
pomidory $\Rightarrow$ kabaczki $wsp = \frac{1}{3}$, $ufn = \frac{1}{2}$, $wsp * ufn = \frac{1}{6}$
ogórki $\Rightarrow$ pomidory $wsp = \frac{1}{2}$, $ufn = \frac{3}{4}$, $wsp * ufn = \frac{3}{8}$
pomidory $\Rightarrow$ ogórki $wsp = \frac{1}{2}$, $ufn = \frac{3}{4}$, $wsp * ufn = \frac{3}{8}$
ogórki $\Rightarrow$ żołądkowa $wsp = \frac{1}{3}$, $ufn = \frac{1}{2}$, $wsp * ufn = \frac{1}{6}$
żołądkowa $\Rightarrow$ ogórki $wsp = \frac{1}{3}$, $ufn = 1.0$, $wsp * ufn = \frac{1}{3}$

Po uwzględnieniu progu jakości $wsp * ufn \geqslant \frac{1}{3}$, reguły o częstości przynajmniej $\Phi = 2$ mają postać,
kabaczki $\wedge$ ogórki $\Rightarrow$ pomidory $wsp * ufn = \frac{1}{3}$
kabaczki $\wedge$ pomidory $\Rightarrow$ ogórki $wsp * ufn = \frac{1}{3}$
cytryny $\Rightarrow$ woda $wsp * ufn = \frac{1}{3}$
woda $\Rightarrow$ cytryny $wsp * ufn = \frac{1}{3}$
kabaczki $\Rightarrow$ ogórki $wsp * ufn = \frac{1}{3}$
kabaczki $\Rightarrow$ pomidory $wsp * ufn = \frac{1}{3}$
ogórki $\Rightarrow$ pomidory $wsp * ufn = \frac{3}{8}$
pomidory $\Rightarrow$ ogórki $wsp * ufn = \frac{3}{8}$, żołądkowa $\Rightarrow$ ogórki $wsp * ufn = \frac{1}{3}$