Inteligentne systemy decyzyjne

Temat: System rekomendacyjny produktów pszczelarskich

Autor: Tomalik Kacper

1. Istota problemu

Obecnie coraz częściej wykorzystuje się systemy rekomendacyjne. Powodem takiego stanu rzeczy jest ich niezwykła skuteczność której efektem jest na przykład zwiększenie przychodów sklepu poprzez trafne propozycje produktów. Głównym założeniem systemu rekomendacyjnego stworzenie potrzeby na produkt w chwili jego wyświetlenia poprzez np "uzupełnienie zamówienia". Najlepszym przykładem może być kupno gwoździ. Użytkownik taki może potrzebować młotka do ich wbicia czy desek które będzie przybijał. Jeżeli system to przewidzi można go uznać za skuteczny.

Zgodnie z tematem projektu użytkownik powinien dostawać kolejne propozycje produktów z oferty sklepu na podstawie zamówień składanych przez innych klientów. Oczywiście rekomendacje są uzależnione od danych na których system pracuje i w jaki sposób je przetwarzają. W tym wypadku system przechowuje dane o 50 klientach (nazywane w owych systemach termami)

2. Budowa termu opisującego klienta.

[nazwa - nr kolumny w tablicy TFM znajdującej sie ponizej]

- a. Godzina zakupu 21
- b. Wiek klienta 22
- c. Płeć klienta 23
- d. Miasto 24
- e. Województwo 25
- f. Lista produktów zamówionych przez klienta 20 produktów lecz nie ma przeciwwskazań by modyfikować ilość produktów ponieważ aplikacja bez większych problemów jest w stanie się przeskalować.

```
∘ PierścieńDoPowałki - 1
○ SitkoPająkDoPowałki - 2
○ Przegonka8-wylotowa - 3
○ EtykietyNaMiódGryczanyE217-100Szt - 4

    PodkarmiaczkaPowałkowa41 - 5

  PokarmCiastoFondant15kgDiamant - 5
○ PodkurzaczTypuDadantNierdzewnyDuży - 6

    Książka, ŁatwePszczelarstwo" (drGerhardLiebig) - 7

    RękawicePszczelarskieZOwczejSkóryDługieXL - 8

  MiotełkaPasiecznaDwurzędowaZKońskiegoWłosia - 9
  Fajkadołapaniamatekpszczelich - 10
  BluzaPszczelarskaZKapeluszemRozpinanaXL - 11
   KapeluszPszczelarskiBIAŁY - 12
  WiaderkoOdstojnikNaMiódZZaworem33lBiały - 13
   DrutDoRamek-StalNierdzewna-025Kg - 14
   UchwytDoRamekNierdzewny - 15
   DwufunkcyjnyNóżPasiecznyZDłutem - 16
   RamkaWielkopolskaHoffmanowska-10Szt - 17
  UniwersalnySkrobakPasieczny - 18
   NakrętkiNaSłoikZMiodemND47-10Szt - 19
  UlikOdkładowyWielkopolski - 20
```

Każdy nowy klient jest porównywany do owych 50 klientów. Sposób zapisu informacji jest realizowany binarnie (0,1) ponieważ interesują nas produkty które wchodziły w skład zamówienia a nie ich ilość.

3. Realizacja projektu oraz metodyka obliczania

Mając termy oraz dane na których będziemy mogli pracować, możemy przystąpić do stworzenia macierzy TFM która zawiera informacje o wszystkich użytkownikach.

Przykładowe wypełnianie macierzy TFM z poziomu kodu aplikacji.

```
temp.Add(new dItem(10, 40, true, "Koszalin", "ZP", "User" + (temp.Count + 1))
{ PierscienDoPowalki = true, SitkoPajakDoPowalki= true, PodkarmiaczkaPowalkowa= true });
```

A tak prezentuje się nasza macierz TFM.

11001000000000		0 10 40	1 Kosza		User1
$0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ $		0 11 50	6 Kosza.		User2
1100110000000	000000	0 13 60	0 Kosza.	lin ZP	User3
0000001001000	000000	0 16 40	1 Kosza.	lin ZP	User4
0000001011000	000000	0 15 50	1 Kosza.	lin ZP	User5
0000001011010	000000	0 17 60	1 Białoga	ard ZP	User6
0000001011001	000000	0 14 30	1 Białoga	ard ZP	User7
00000000000000	0110100	0 15 40	Ø Białoga	ard ZP	User8
00000000000000	0110110	0 12 50	1 Białoga	ard ZP	User9
00000000000000	0110100	0 16 20	1 Białoga	ard ZP	User10
0 0 0 0 0 0 1 0 1 1 0 1 0	1000000	0 13 40	1 Karl:	ino ZP	User11
0 0 0 0 0 0 1 0 1 1 0 1 0	1000010	0 12 50	<pre>0 Karl:</pre>	ino ZP	User12
0000000011010	1000010	0 13 60	1 Karl:	ino ZP	User13
0000001001000	1000010	0 15 30	1 Karl:	ino ZP	User14
0000000001000		0 18 30	<pre>0 Karl:</pre>	ino ZP	User15
0 0 0 0 0 0 0 0 0 0 0 0		1 17 40	1 Gor:		User16
	0101100	1 18 50	1 Gor:	zów Lubuskie	User17
$0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 1 \; 0 \; 0$	0100100	1 19 50	6 Gor:	zów Lubuskie	User18
0000001100100	0100100	1 20 30	6 Gor:	zów Lubuskie	User19
0000001100000	0000100	1 21 30	1 Gor:	zów Lubuskie	User20
00000000000000	0011100	0 16 50	1 Zielona Go	óra Lubuskie	User21
00010000000000	0000011	0 15 40	1 Zielona Go	óra Lubuskie	User22
0000110000000	000000	0 14 40	Ø Zielona Go	óra Lubuskie	User23
0 0 0 0 0 0 1 0 1 1 0 1 0	1000000	0 13 40	Ø Zielona Go	óra Lubuskie	User24
00000000000010		1 15 50	1 Zielona Go		User25
00010000000000		0 16 20	1 Pozi		User26
		0 17 20	9 Pozi	nań Wielkopolskie	User27
0 0 0 0 0 0 1 0 1 1 0 1 0		0 18 30	1 Pozi		User28
00000000000010		1 19 30	1 Pozi		User29
		0 18 50	9 Pozi		User30
0000001011000		0 16 50	0 Gdai		User31
00000000000000		0 15 20	1 Gdar		User32
0000000001000		0 14 20	1 Gdar		User33
0000001100000		1 18 50	1 Gdar		User34
00000000000010		1 19 30	0 Gdai		User35
		0 18 50	1 Warsza		User36
		1 18 50	0 Warsza		User37
		0 17 10	0 Warsz		User38
00000000000000		0 19 20	0 Warsza		User39
		0 16 20	1 Warsz		User40
1 1 0 0 1 1 0 0 0 0 0 0 0		0 16 20	1 Lub		User41
0000000000000		0 17 20	0 Lub		User42
		0 17 20	0 Lub.		User43
		1 18 30	1 Lub		User44
		0 19 30	0 Lub		User45
		0 17 20		lce Świętokrzyskie	User46
0 0 0 0 0 0 1 0 1 1 0 0 1		0 16 20		lce Świętokrzyskie	User47
0 0 0 0 0 0 0 0 1 1 0 1 0		0 16 30		lce Świętokrzyskie	User48
0 0 0 0 0 0 1 0 0 1 0 0 0		0 15 30		lce Świętokrzyskie	User49
0000001100000	0000100	1 18 20	0 Kie	lce Świętokrzyskie	User50

4. Wyznaczanie odległości euklidesowej

Następnym krokiem jest wyznaczenie odległości pomiędzy naszym klientem a klientami których posiadamy już w bazie.

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_n - y_n)^2}$$
 gdzie:

x to nasz klient

y to klient już istniejący w bazie.

Dodatkowo algorytm został ograniczony do liczenia kolumn które zostały wypełnione przez klienta. To oznacza że w procesie liczenia biorą udział tylko kolumny w których występują wartości '1' w wierszu klienta.

Nastepnie szukamy takie d(x,y) które jest najmniejsze ale rózne od 0 ponieważ d=0 oznacza dokladnie takiego samego użytkownika wiec nic on nie zasugeruje. Po odnalezieniu takiej osoby proponujemy naszemu klientowi to co zamówiła "najbliższa mu" osoba, a czego jeszcze klient nie zamówił.

Implementacja wzoru w kodzie:

```
for (int i = 0; i < tfm.Count; i++)
{
    double value = 0.0;
    for (int j = 0; j < convert[i].Count; j++)
    {
        var a = convert[i][j];
        var b = convert[convert.Count - 1][j];
        bool c = a.Equals(b);
        double temp = c ? 0.0 : 2.0;

        if (!b.Equals("0"))
            value += Math.Pow(Math.Abs(temp), 2.0);
    }
    value = Math.Pow(value, 1.0 / 2.0);
    this.Add(value);
    proposition.Add(new Tuple<double, int>(value, i));
}
```

oraz wyszukanie najbliższego klienta:

```
proposition = proposition.OrderBy(x => x.Item1).ToList();
```

gdzie osoba pierwsza na liście jest najbliżej.

Pozostaje nam tylko już wyświetlić rekomendowane produkty. Podczas wyświetlania znajdują się określenia Mało/Średnio/Dużo i są wyznaczane na podstawie **offset**=(d(x,y).Max-d(x,y).Min)/3 w wyniku czego powstają grupy:

- Mało dla najbliższych klientów < d(x,y).Min() + offset
- Średnio dla klientów < d(x,y).Min + offset * 2
- Dużo dla klientów < d(x,y).Min + offset * 3

5. Działanie aplikacji - Studium Przypadków

Jest to pierwsza część aplikacji gdzie podajemy swoje dane a nastepnie wyświetlana jest lista produktów które można zamówić i wyświetlana jest az do wciśniecia klawisza "esc".

Po wyborze produktu (w tym wypadku 0.PierscienDoPowalki) otrzymujemy informacje na temat najbliższego nam zamówienia skłądającego się z "słownego określenia odleglosci" - "wartości liczbowej odleglosci" - "nazwy użytkownika" -"oraz wypunktowanej listy produktów wraz z numerem produktu w menu"

```
Malo - 03,46 - User1
1. SitkoPajakDoPowalki
4. PodkarmiaczkaPowalkowa
```

Na podstawie tego zamowienia zostało wybrane 1.Sitko Pająk Do Powalki w wyniku czego najlepsza rekomendacja wygląda następująco.

```
Malo - 03,46 - User1
4. PodkarmiaczkaPowalkowa
```

Teraz żeby sprawdzić działanie zostanie wybrana opcja(6.EtykietaNaMiódGryczany) która nie jest sugerowana, a oto wynik:

```
Malo - 04,00 - User1
4. PodkarmiaczkaPowalkowa
```

Jak widać dalej najbliższy użytkownik sie nie zmienił ale za to odleglosc z 3,46 zmieniła się na 4,00. A tak prezentuje się w tej chwili term naszego klienta oraz najbliższego uzytkownika - user1.

6. Wnioski

Aplikacja została wykonana bez większych przeszkód, a wszystkie zamierzone funkcjonalności zostały zaimplementowane. Rekomendacje działają poprawnie w zgodzie z jej bazą wiedzy. Należy pamiętać iż system taki wraz z rosnącą ilością zamówień może coraz lepiej rekomendować produkty.

Realizacja owego projektu była bardzo ciekawym doświadczeniem ponieważ miałem okazję poznać coś szeroko rozpowszechnionego oraz wciąż popularnego na rynku. Jak się okazało nie jest to aż takie trudne.