

*MATEUSZ PIWOWARSKI, KAROL SZCZYCIŃSKI*

Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

*KESRA NERMEND*

Uniwersytet Szczeciński

## METODA PERSONALIZACJI TREŚCI W SERWISACH INTERNETOWYCH

### Streszczenie

Głównym celem artykułu jest zaprezentowanie autorskiej metody personalizacji zawartości serwisu internetowego. Pierwsza część publikacji zawiera krótki przegląd istniejących metod personalizacji treści w witrynach. W dalszej części znajduje się opis opracowanej metody personalizacji (połączenie metody MinHash w podejściu generowania rekomendacji opartym na podobieństwie użytkowników typu przedmiot–przedmiot oraz metody AHP), a następnie przedstawiane są wyniki badań symulacyjnych dla wybranego zestawu danych.

### Wprowadzenie

Obecnie w internecie funkcjonuje wiele witryn o tej samej tematyce. Konkurują one między sobą w celu pozyskania jak największej liczby użytkowników. Serwis internetowy może osiągnąć przewagę nad pozostałymi, np. opierając się na dobrze prowadzonym marketingu, ale również dzięki zadowoleniu

użytkowników z zawartości, jakie oferują strony. Pierwszy czynnik to domena różnego rodzaju agencji marketingowych, natomiast drugi, czyli satysfakcję **użytkownika z otrzymanych treści**, można zwiększyć poprzez prezentowanie treści jak najbardziej zbliżonych do oczekiwań odbiorców. Aby uzyskać ten efekt, gromadzi się informacje o wszystkich użytkownikach danej witryny. Zgromadzone dane poddawane są analizie, tak aby dostosować do każdego z nich takie treści, które prawdopodobnie trafią w jego gust i oczekiwania. Do realizacji tego zadania wykorzystywane są metody personalizacji treści, które służą właśnie do jak najlepszego dopasowania treści strony do oczekiwań użytkownika.

Celem artykułu jest zaprezentowanie metody dopasowania zawartości witryny na potrzeby systemu zarządzania treścią. Opracowana metoda umożliwi generowanie spersonalizowanej zawartości, która może w większym stopniu zainteresować internautę, a tym samym dłużej zatrzymać go w serwisie internetowym.

## 1. Personalizacja treści w serwisach internetowych

Metody personalizacji treści służą dostosowaniu zawartości strony do oczekiwań użytkownika, który właśnie ją przegląda. Ich celem jest zaprezentowanie takich informacji, które w jak największym stopniu mają szansę go zainteresować, a tym samym zwiększyć jego zadowolenie z witryny. Efektem tego jest dłuższe zatrzymanie internauty w danej witrynie, wzrost jego zadowolenia z witryny i tym samym osiągnięcie przewagi konkurencyjnej nad innymi podobnymi serwisami. Jeśli witryna ma charakter komercyjny, może się z tym wiązać wzrost zysków z prowadzenia danej formy działalności. Naturalne zatem wydaje się stosowanie takich rozwiązań w serwisach np. typu e-commerce, w celu generowania listy propozycji (rekomendacji) produktów do zakupu.

Aby wygenerować rekomendacje, systemy rekomendujące muszą zbierać wiele różnych danych. W konkretnych rozwiązaniach źródła danych mogą się różnić, lecz generalnie można wyróżnić trzy podstawowe składowe<sup>1</sup>:

- przedmioty (ang. items), czyli obiekty, które są przedmiotem rekomendacji;

---

<sup>1</sup> F. Ricci, L. Rokach, B. Shapira, P.B. Kantor, *Recommender Systems Handbook*, Springer, New York 2010.

- użytkownicy (ang. users);
- transakcje (ang. transactions), czyli interakcje zapisane pomiędzy użytkownikiem a systemem rekomendującym.

Mówiąc o generowaniu rekomendacji, należy zaznaczyć, że istnieją dwa główne podejścia do rozwiązania tego problemu:

- podejście oparte na podobieństwie treści (ang. Content-based filtering);
- podejście oparte na podobieństwie użytkowników (ang. Collaborative filtering).

Rozwiązania oparte na podobieństwie treści charakteryzują się tym, że dobrze działają i są efektywne w sytuacjach, kiedy internauta dopiero się zarejestrował i posiadamy o nim mało informacji. Najprostsze algorytmy tego typu generują rekomendacje, wyszukując podobną treść, na przykład: treść tego samego autora, treść z tej samej kategorii czy też treść z podobnym tytułem.

Podejście oparte na podobieństwie użytkowników, w odróżnieniu od podejścia opartego na podobieństwie treści z profilem użytkownika, oparte jest na znajdowaniu użytkowników, którzy dzielą takie same upodobania. Rekomendacje opierane są na założeniu, że „podobni” **użytkownicy lubią podobne rzeczy**. Istotne jest zatem to, co spodobało się użytkownikom, a dlaczego tak jest – pozostaje niewiadomą. W tym podejściu wyróżnia się dodatkowy podział na podejście oparte na sąsiedztwie zorientowane na użytkownika, podejście oparte na sąsiedztwie zorientowane na przedmiot oraz podejście oparte na budowaniu modelu<sup>2</sup>.

Do głównych metod wykorzystywanych przy generowaniu spersonalizowanej treści w systemach rekomendacji zaliczamy<sup>3</sup>:

- naiwny klasyfikator Bayesa;
- metodę k-najbliższych sąsiadów;
- metodę drzew klasyfikacyjnych i regresyjnych;
- metodę MinHash.

W metodzie MinHash (o pozostałych metodach tutaj nie wspominamy) do obliczenia miary podobieństwa między dwoma wektorami A i B wykorzystuje się miarę Jaccarda, zdefiniowaną jako stosunek rozmiarów części wspólnej i sumy.

---

<sup>2</sup> D. Asanov, *Algorithms and Methods in Recommender Systems*, [http://www.snet.tuberlin.de/fileadmin/fg220/courses/WS1011/snet-project/recommendersystems\\_asanov.pdf](http://www.snet.tuberlin.de/fileadmin/fg220/courses/WS1011/snet-project/recommendersystems_asanov.pdf), dostęp 25.12.2011.

<sup>3</sup> J. Koronacki, J. Ćwik, *Statystyczne systemy uczące się*, WNT, Warszawa 2005; D. Hand, H. Mannila, P. Smyth, *Eksploracja danych*, WNT, Warszawa 2005.

W tradycyjnym podejściu do obliczania podobieństwa między dwoma wektorami porównywane są całe wektory. W przypadku podejścia opartego na podobieństwie użytkowników typu użytkownik–użytkownik, każdy przedmiot opisywany jest przez wektor długości ilości wszystkich użytkowników zapisanych w bazie danych. Założyć więc można, że takie wektory posiadają nie 10 komórek, ale przynajmniej kilka tysięcy. Porównywanie dużej ilości długich wektorów jest zadaniem złożonym i bardzo obciążającym zasoby komputera (procesor i pamięć RAM). Metoda MinHash jest sposobem optymalizacji tego problemu. Porównywanie odbywa się przez przekształcenie wektorów do ich krótszych sygnatur, dla których podobieństwo według wybranej miary będzie takie samo lub możliwie bardzo zbliżone do podobieństwa obliczonego dla wektorów w pełnej postaci. Nie można jednak tego zrobić przez losowe wybranie indeksów komórek, które utworzą sygnaturę wektorów. Dzieje się tak, ponieważ w większości przypadków takie wektory są rzadkie. Powodowałoby to częste sygnatury z samymi zerami, gdyż „jedynek” wskazujące użytkowników znajdowałyby się często w innych komórkach. Rozwiązanie tego problemu jest następujące<sup>4</sup>:

- 1) ustal długość sygnatury na 50 komórek;
- 2) utwórz 50 permutacji wszystkich indeksów komórek wektora początkowego;
- 3) weź pierwszą permutację i sprawdzaj wartość komórki (w oryginalnym wektorze) o takim indeksie, jaki wskazuje wartość komórki w wektorze z permutacją (zaczynając od pierwszej), kiedy napotkasz „jedynek” w oryginalnym wektorze, przerwij poszukiwania i jako pierwszą wartość sygnatury wpisz numer indeksu w oryginalnym wektorze, pod którym została znaleziona „jedynek”;
- 4) powtarzaj punkt 3 dla każdej permutacji (w tym przykładzie jeszcze 49 razy);
- 5) otrzymany wektor stanowi sygnaturę (ang. Hash) oryginalnego wektora; warto zauważyć, że gdyby oryginalny wektor też miał długość 50 komórek, to prawdopodobieństwo, że sygnatury dwóch wektorów są równe, jest takie samo, jak podobieństwo dwóch oryginalnych wektorów – oba są równe.

Podstawą skuteczności tej metody jest zależność między wartościami MinHash a podobieństwem Jaccarda. Polega ona na tym, że jeśli permutacje

---

<sup>4</sup> J. Ullman, *Data Mining of Very Large Data*, Symposium on the Effectiveness of Logic in Computer Science in Honour of Moshe Vardi, 2002.

są wybrane losowo, to prawdopodobieństwo, że dadzą te same wartości Min-Hash dla dwóch wektorów, jest takie samo jak podobieństwo Jaccarda dla tych wektorów. Pozwala to na szacowanie prawdopodobieństwa Jaccarda na podstawie tylko części odpowiadających sobie wartości.

Po przeprowadzeniu analizy istniejących podejść i metod generowania spersonalizowanej zawartości w witrynach internetowych zaproponowano autorską metodę generowania treści w systemie zarządzania treścią. Czerpie ona korzyści z metody MinHash i podejścia generowania rekomendacji opartego na podobieństwie użytkowników typu przedmiot–przedmiot oraz wielokryterialnej metody wspomagania podejmowania decyzji AHP (ang. Analytic HierarchyProcess)<sup>5</sup>.

## 2. Metoda dopasowania zawartości w systemie zarządzania treścią

Proponowana metoda polega na zastosowaniu połączenia metody MinHash w podejściu generowania rekomendacji opartego na podobieństwie użytkowników typu przedmiot–przedmiot oraz metody AHP. Metoda MinHash służy w tym rozwiązaniu do znajdowania podobnych przedmiotów na podstawie gustu użytkowników, natomiast metoda AHP pozwala dobrać parametry dla metody MinHash, jednocześnie ukrywając przed użytkownikiem trudność doboru tych parametrów. Odbywa się to poprzez prezentację użytkownikowi interfejsu, w którym proszony jest jedynie o określenie, które trzy kryteria w dziewięciostopniowej skali są dla niego ważniejsze. Jeśli użytkownik nie potrafi zdecydować, metoda przyjmuje założenie, że kryteria są jednakowo istotne i dobiera optymalny wariant ustawień.

Zalety tego rozwiązania są następujące:

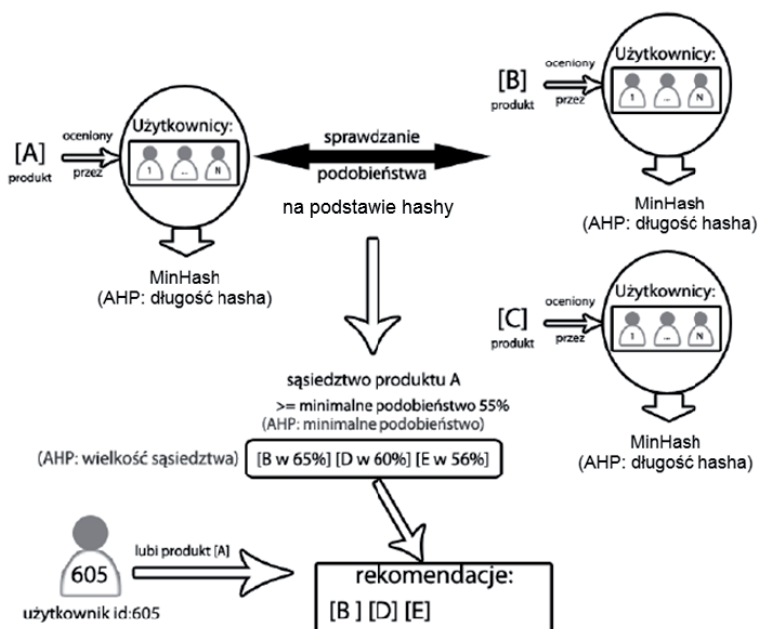
- skuteczne generowanie trafnych rekomendacji w czasie rzeczywistym – wysoka wydajność w oczach użytkownika;
- najbardziej kosztowne obliczenia, czyli znajdowanie podobnych przedmiotów, są wykonywane w trybie offline;
- wielokrotne wykorzystywanie obliczeń wykonanych w trybie offline;

---

<sup>5</sup> T.L. Saaty, *How to make a decision: The Analytic Hierarchy Process*, „European Journal of Operational Research”, vol. 48, 1990.

- możliwość generowania spersonalizowanej treści również dla nowych użytkowników;
- rozwiązanie oparte na gustach **użytkowników**, przy jednoczesnym braku ograniczenia w postaci ich niezmienności, gusta mogą się powoli zmieniać w czasie, tak samo jak rekomendowane treści;
- przyspieszenie kosztownych obliczeń znajdowania podobnych przedmiotów dzięki metodzie MinHash, przy niewielkiej stracie dokładności szacowania podobieństwa;
- rozwiązanie jest proste do zrozumienia oraz proste w wykorzystaniu;
- szybkie i uproszczone sterowanie parametrami rozwiązania dzięki zastosowaniu metody AHP, która pozwala na połączenie wiedzy eksperckiej wraz z preferencjami użytkownika poznawanymi przez porównywanie kryteriów parami.

Na rys. 1 zilustrowano schemat działania proponowanej metody personalizacji treści.



Rys. 1. Schemat działania proponowanej metody personalizacji treści

Źródło: opracowanie własne.

Algorytm metody jest następujący:

- administrator systemu zarządzania treścią określa preferencje poprzez zdefiniowanie ważności trzech kryteriów: szybkości obliczeń, większej ilości rekomendacji, pewności rekomendacji;
- następnie na tej podstawie dobierany jest najlepszy wariant ustawień parametrów (warianty są uprzednio przygotowane przez eksperta): długość sygnatury, wielkość sąsiedztwa oraz minimalne podobieństwo (tabela 1 prezentuje wpływ wyszczególnionych parametrów na uzyskane wyniki);

Tabela 1

Wpływ poszczególnych parametrów na proces dopasowania treści

Parametr	Wpływ po zwiększeniu	Wpływ po zmniejszeniu
Długość sygnatury	Zwiększona dokładność szacowania podobieństwa; zwiększona złożoność i czas obliczeń	Zmniejszona dokładność szacowania podobieństwa; zmniejszona złożoność i skrócony czas obliczeń
Wielkość sąsiedztwa	Zwiększenie maksymalnej ilości możliwych rekomendacji; zwiększona złożoność i czas obliczeń	Zmniejszenie maksymalnej ilości możliwych rekomendacji; zmniejszona złożoność i skrócony czas obliczeń
Minimalne podobieństwo	Zwiększenie pewności prezentowanych rekomendacji; zmniejszenie ilości prezentowanych rekomendacji	Zmniejszenie pewności prezentowanych rekomendacji; zwiększenie ilości prezentowanych rekomendacji

Źródło: opracowanie własne.

- administrator systemu uruchamia proces znajdowania podobnych przedmiotów: tworzone są sygnatury wektorów produktów (wektor produktu zawiera informacje, którzy użytkownicy polubili ten produkt), dla każdej sygnatury obliczane jest podobieństwo z każdą pozostałą, najbardziej podobne produkty, które spełniają wymóg minimalnego zaufania, zostają zapamiętane wraz z wartością podobieństwa w metadanych produktu;
- kiedy **użytkownik polubi dany produkt**, jest to zapamiętywane;
- spersonalizowana zawartość zawiera polecane, podobne produkty, których użytkownik nie polubił do tej pory, a które są podobne do

produktów, jakie użytkownik już polubił, w kolejności od najbardziej podobnych do najmniej.

Prezentowana metoda została wdrożona w systemie zarządzania treścią Squiz Matrix<sup>6</sup> w wersji 4.6.2. Główny kod programu został napisany w języku JavaScript, a do wdrożenia zostały użyte następujące typy assetów systemu Squiz Matrix: Standard Page, JS File, Asset Listing, Trigger, User, News Item. Asset typu: Standard Page – został użyty do przechowywania kodu HTML panelu administracyjnego projektu.

- Assety typu: JS File – zostały stworzone cztery assety tego typu, pierwszy użyty został do przechowywania głównego kodu JavaScript, działającego w panelu administratora, drugi asset został użyty do przechowywania kodu JavaScript działającego po stronie użytkownika, kolejne dwa do przechowywania JavaScriptowej biblioteki jQuery<sup>7</sup> oraz dodatku do tej biblioteki: jQuery UI.
- Assety typu: Asset Listing – stworzone zostało pięć assetów tego typu, pierwszy służy do listowania wszystkich identyfikatorów użytkowników, drugi do listowania wszystkich identyfikatorów przedmiotów, kolejne dwa do listowania identyfikatorów użytkowników, którzy polubili każdy przedmiot oraz piąty do tworzenia spersonalizowanej listy rekomendowanych przedmiotów.
- Assety typu: Trigger – stworzone zostały dwa assety tego typu, pierwszy odpowiedzialny jest za zapisywanie w systemie użytkowników pod przedmiotami, kiedy ci je polubili, natomiast drugi odpowiedzialny jest za zapisywanie sąsiedztwa każdego z produktów.
- Assety typu: User – zostały stworzone w celu przechowywania informacji na temat użytkowników i informacji, jakie przedmioty polubili.
- Assety typu: News Item – zostały stworzone w celu przechowywania informacji o przedmiotach.

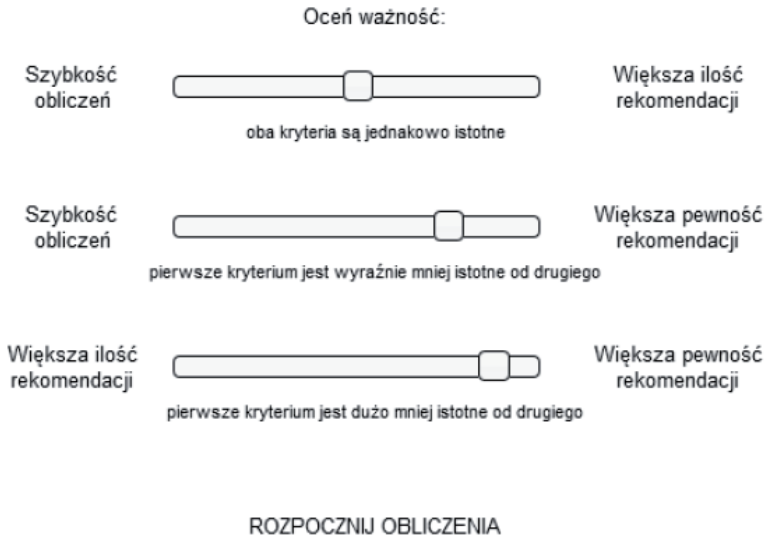
Do zrealizowania panelu administratora została użyta biblioteka jQuery UI<sup>8</sup>. Dzięki temu możliwe było stworzenie bardzo intuicyjnego i łatwego w obsłudze interfejsu użytkownika (rys. 2).

<sup>6</sup> Squiz Matrix: User Manual Library, <http://manuals.matrix.squizsuite.net>, dostęp 3.03.2012.

<sup>7</sup> jQuery: The Write Less, Do More, JavaScript Library, <http://jquery.com>, dostęp 1.03.2012.

<sup>8</sup> jQuery user interface library, <http://jqueryui.com>, dostęp 1.03.2012.





### Zapisywanie sąsiedztwa przedmiotu nr 768 (66/100)

Rys. 2. Interfejs panelu administratora wdrożonego projektu  
Źródło: opracowanie własne.

Interfejs użytkownika pozwala na określanie preferencji poprzez przesuwanie suwaka w stronę jednego lub drugiego kryterium, **w każdej parze** kryteriów. Dodatkowo pod każdym suwakiem znajduje się tekstowa informacja o stopniu dominacji któregoś z kryteriów lub o równości istotności kryteriów, kiedy suwak nie zostanie przesunięty w żadną ze stron. Dodatkowo wyświetlana jest informacja o postępie zapisywania wyników dokonanych obliczeń.

### 3. Praktyczna weryfikacja metody

Do przeprowadzenia symulacji został użyty darmowy i publicznie dostępny zbiór danych z systemu o nazwie Jester<sup>9</sup>. System ten jest systemem

<sup>9</sup> Anonymous Ratings from the Jester Online Joke Recommender System, <http://eigentaste.berkeley.edu/dataset>, dostęp 4.04.2012.

rekomendacji dowcipów realizującym algorytm o nazwie Eigentaste: A Constant Time Collaborative Filtering Algorithm, stworzonym na kalifornijskim uniwersytecie Berkeley. Na stronie projektu dostępne są dwa zbiory. Pierwszy zawiera ponad cztery miliony ocen stu dowcipów od 73 421 użytkowników z okresu od kwietnia 1999 do maja 2003 r. Natomiast drugi zawiera ponad półtora miliona ocen 150 dowcipów od 63 974 użytkowników z okresu od listopada 2006 do maja 2009 r. Do przeprowadzenia symulacji wykorzystany został podzbiór pierwszego zbioru.

Do systemu Squiz Matrix zostało zaimportowanych sto dowcipów z pierwszego zbioru oraz trzy tysiące użytkowników wraz z informacjami, które dowcipy z całej puli stu dowcipów polubili. W oryginalnym zbiorze danych oceny użytkowników były ciągłe w przedziale od  $-10$  do  $10$ . W przypadku braku oceny danego dowcipu przez użytkownika w zbiorze występowała wartość  $9$ . Oceny przed wgraniem do systemu Matrix, celem dostosowania zbioru danych do autorskiej metody, zostały poddane dyskretyzacji. Wszystkie wartości mniejsze niż  $5$  oraz wartość  $99$  zostały zastąpione przez wartość  $0$ , a wszystkie wartości większe lub równe  $5$  i mniejsze lub równe  $10$  zostały zastąpione przez wartość  $1$ .

Drugim elementem wykorzystanym do przeprowadzenia symulacji były trzy zestawy parametrów wejściowych do autorskiej metody. Warianty parametrów zostały stworzone na podstawie obserwacji wyników działania algorytmu i są elementem wiedzy eksperckiej we wdrażanej metodzie. Trzy zestawy ustawień były następujące:

1. hashLength=40 (%), minSimilarity=40 (%), neighbourhoodSize=6
2. hashLength=35 (%), minSimilarity=35 (%), neighbourhoodSize=5
3. hashLength=30 (%), minSimilarity=30 (%), neighbourhoodSize=3

Zmienna hashLength przechowuje wartość długości sygnatury, zmienna minSimilarity wartość minimalnego podobieństwa, natomiast zmienna neighbourhoodSize maksymalną wielkość sąsiedztwa.

Tabele 2, 3, 4 zawierają oceny porównania wariantów pod względem każdego z kryteriów według przyjętej skali ocen, przeprowadzone na podstawie metody AHP.

Tabela 2

Porównanie zestawów wartości pod względem szybkości obliczeń

	Zestaw A	Zestaw B	Zestaw C	Wektor wartości
Zestaw A	1,00	0,33	0,20	0,10
Zestaw B	3,00	1,00	0,20	0,21
Zestaw C	5,00	5,00	1,00	0,69

Źródło: opracowanie własne.

Tabela 3

Porównanie zestawów wartości pod względem ilości rekomendacji

	Zestaw A	Zestaw B	Zestaw C	Wektor wartości
Zestaw A	1,00	0,33	0,20	0,12
Zestaw B	3,00	1,00	3,00	0,55
Zestaw C	5,00	0,33	1,00	0,33

Źródło: opracowanie własne.

Tabela 4

Porównanie zestawów wartości pod względem pewności rekomendacji

	Zestaw A	Zestaw B	Zestaw C	Wektor wartości
Zestaw A	1,00	5,00	7,00	0,72
Zestaw B	0,20	1,00	3,00	0,19
Zestaw C	0,14	0,33	1,00	0,08

Źródło: opracowanie własne.

W procesie symulacji zostały określone następujące preferencje:

- „Szybkość obliczeń” jest jednakowo istotna jak „ilość rekomendacji”;
- „Szybkość obliczeń” jest nieznacznie mniej istotna niż „pewność rekomendacji”;
- „Większa ilość rekomendacji” jest nieznacznie istotniejsza niż „większa pewność rekomendacji”.

Wybrane preferencje spowodowały dobranie przez metodę AHP drugiego wariantu parametrów wejściowych. Tabela 5 prezentuje testy jednostkowe po

wdrożeniu opracowanej metody na danych rzeczywistych. Zawiera wygenerowane rekomendacje dla trzech losowo wybranych użytkowników po polubieniu przez nich pierwszego dowcipu. Testy były powtarzane trzykrotnie dla każdego użytkownika. Wybrani użytkownicy pochodzą z części oryginalnego zbioru danych, która nie została wgrana do systemu. Jeden wiersz w tabeli 5 reprezentuje jednego użytkownika i jego niezależne pojedyncze polubienie wybranego dowcipu.

Tabela 5

## Testy jednostkowe na wybranych użytkownikach

Numery wszystkich dowcipów, które polubiłby użytkownik, gdyby został wgrany do systemu wraz ze zbiorem uczącym	Nr polubionego dowcipu	Nmery rekomendowanych dowcipów	Liczba rekomendacji, które spodobałyby się użytkownikowi [A]	Liczba rekomendacji, które być może będą się podobać (nie zostały ocenione) [B]	Liczba nietrafnych rekomendacji [C]	Komentarz
5, 7, 11, 21, 27, 29, 32, 35, 39, 42, 45, 65, 68, 98	5	21,48	1	0	1	50% trafności z możliwością na więcej
5, 7, 11, 21, 27, 29, 32, 35, 39, 42, 45, 65, 68, 98	65	53, 28, 69, 31, 35	1	4	0	20% trafności, numer 53, mimo iż został uznany za nietrafiony, prawdopodobnie mógłby być uznany przez użytkownika za fajny, gdyż przed dyskretyzacją danych posiadał ocenę, która klasyfikowała go bardzo blisko do znalezienia się w zbiorze polubionych
5, 7, 11, 21, 27, 29, 32, 35, 39, 42, 45, 65, 68, 98	32	27, 31, 52, 54, 69	1	3	1	20% trafności, z możliwością na więcej

7, 14, 21, 22, 23, 27, 28, 29, 32, 36, 38, 40, 41, 42, 43, 47, 50, 54, 55, 57, 61, 64, 65, 66, 67, 69, 88, 89, 90, 91, 94, 96, 97	7	21, 61, 69	3	0	0	100% trafnych rekomendacji
7, 14, 21, 22, 23, 27, 28, 29, 32, 36, 38, 40, 41, 42, 43, 47, 50, 54, 55, 57, 61, 64, 65, 66, 67, 69, 88, 89, 90, 91, 94, 96, 97	23	30, 55, 40, 67	3	0	1	75% trafności, z możliwością na więcej
7, 14, 21, 22, 23, 27, 28, 29, 32, 36, 38, 40, 41, 42, 43, 47, 50, 54, 55, 57, 61, 64, 65, 66, 67, 69, 88, 89, 90, 91, 94, 96, 97	96	89	1	0	0	100% trafności
6, 7, 9, 14, 17, 20, 26, 29, 31, 32, 35, 39, 42, 45, 49, 50, 52, 53, 54, 56, 65, 66, 68, 69, 87	17		0	0	0	0% trafności, wybrany dowcip ma puste sąsiedztwo
6, 7, 9, 14, 17, 20, 26, 29, 31, 32, 35, 39, 42, 45, 49, 50, 52, 53, 54, 56, 65, 66, 68, 69, 87	26	38	0	1	1	0% trafności, jeden dowcip nie został oceniony, natomiast drugi pewnie byłby uznany w subiektywnej ocenie na: „może być”
6, 7, 9, 14, 17, 20, 26, 29, 31, 32, 35, 39, 42, 45, 49, 50, 52, 53, 54, 56, 65, 66, 68, 69, 87	49	53, 36, 47, 65, 69	3	2	0	60% trafności

Źródło: opracowanie własne.

Po analizie powyższej tabeli, mimo iż testy nie są wyczerpujące i pokrywają zaledwie ułamek możliwych wariantów, należy stwierdzić, że metoda dobrze radzi sobie z rekomendowaniem treści. Użytkownik dostaje natychmiastowo rekomendacje, które w mniejszym lub większym stopniu trafiają w jego gust.

## Podsumowanie

W artykule zostały pobieżnie przedstawione obecnie używane podejścia i metody dostosowania treści oraz zaprezentowana została metoda personalizacji bazująca na innych istniejących rozwiązaniach. Opracowana metoda została wdrożona w systemie zarządzania treścią o nazwie Squiz Matrix. Do przeprowadzenia testów został użyty darmowy i publicznie dostępny zbiór danych zawierający informacje o stu dowcipach i ich ocenach. Wykorzystane zostały oceny od trzech tysięcy użytkowników. Po przeprowadzeniu testów jednostkowych należy stwierdzić, że opracowana metoda daje obiecujące wyniki. Dużym autem opisywanej metody jest możliwość szybkiego prezentowania rekomendacji. Wystarczy, że użytkownik polubi jeden przedmiot, a od razu można generować dla niego spersonalizowaną treść. Natomiast dla administratora serwisu internetowego opracowane rozwiązanie jest bardzo proste w obsłudze. Wystarczy, że określi on swoje preferencje odnośnie do trzech kryteriów, a reszta wykonuje się automatycznie. Wydaje się, że warto rozwijać tego typu metody, gdyż już przynoszą one duże korzyści witrynom, w których zostały wdrożone. Wraz z doskonalszymi metodami te korzyści mogą być jeszcze większe.

## Literatura

1. Anonymous Ratings from the Jester Online Joke Recommender System, <http://eigentaste.berkeley.edu/dataset>, dostęp 4.04.2012.
2. Asanov D., *Algorithms and Methods in Recommender Systems*, <http://www.snet.tuberlin>.
3. [de/fileadmin/fg220/courses/WS1011/snet-project/recommendersystems\\_asanov.pdf](http://fileadmin/fg220/courses/WS1011/snet-project/recommendersystems_asanov.pdf), dostęp 25.12.2011.
4. Hand D., Mannila H., Smyth P., *Eksploracja danych*, WNT, Warszawa 2005.
5. jQuery: The Write Less, Do More, JavaScript Library, <http://jquery.com>, dostęp 1.03.2012.
6. jQuery user interface library, <http://jqueryui.com>, dostęp 1.03.2012.
7. Koronacki J., Ćwik J., *Statystyczne systemy uczące się*, WNT, Warszawa 2005.
8. Ricci F., Rokach L., Shapira B., Kantor P.B., *Recommender Systems Handbook*, Springer, New York 2010.

9. Saaty T.L., *How to make a decision: The Analytic Hierarchy Process*, „European Journal of Operational Research”, vol. 48, 1990.
10. Squiz Matrix: User Manual Library, <http://manuals.matrix.squizsuite.net>, dostęp 3.03.2012.
11. Ullman J., *Data Mining of Very Large Data*, Symposium on the Effectiveness of Logic in Computer Science in Honour of Moshe Vardi, 2002.

## **A METHOD FOR PERSONALIZATION OF WEB CONTENT**

### **Summary**

The main purpose of this article is to present a content personalization method in the website. The document is divided into three chapters. The first chapter describes the analysis of existing content personalization methods in websites. The next part contains the description of created author's method (a combination of methods MinHash and AHP) with description of implementation. The last chapter it presents test results for used dataset.

*Translated by Karol Szczyciński*

