Zadanie 2
Wybierz jedno z poniższych zadań i wykonaj do 18 czerwca 2007, końcowy efekt proszę przesłać na adres email osoby prowadzącej ćwiczenia i wykładowcy.

PageRank

1. W pliku tekstowym *T* będącym parametrem dla Twojego programu podane są adresy URL, po jednym w wierszu. Twój program powinien odczytać strony odpowiadające tym adresom, wyłuskać z nich wszystkie linki, i utorzyć na ich podstawie graf sieci. Graf ma składać się tylko ze stron, których adresy były umieszczone w pliku *T* (więc linki prowadzące do innych stron ignorujemy).
   Uwagi.
   a) Możesz (ale nie musisz) przyjąć, że strony odpowiadające adresom w pliku *T* zostały zgromadzone na Twoim lokalnym dysku, a nazwy odpowiednich plików/katalogów odpowiadają adresom stron zgodnie z konwencją stosowaną przez wget.
   b) Uwzględniaj adresy względne w linkach, ignoruj zakładki w obrębie strony (tekst zaczynający się od # w adresie).
2. Dla grafu sieci utworzonego według powyższego opisu zastosuj algorytm PageRank do utworzenia rankingu stron. Wykonaj zadanie dla różnych prawdopodobieństw teleportacji (0.1; 0.2; 0.3) i wybierz eksperymentalnie jedno z nich.
3. Utwórz wyjściowy plik tekstowy zawierający następujące informacje o każdej stronie z grafu:
   a) adres URL,
   b) wartość rankingu,
   c) lista słów występujących w kotwicach linkujących na tę stronę.
   Twoja lista powinna być uporządkowana według rankingu.
4. Utwórz indeks dla słownika składającego się ze słów występujących w kotwicach, lista każdego słowa *d* ma zawierać strony *s* wskazywane przez to słowo (tzn. słowo *d* występuje w co najmniej jednej kotwicy wskazującej na *s*).
5. Napisz program, który dla podanego słowa *s* wypisuje (uporządkowane według rankingu) adresy wszystkich stron wskazywanych przez to słowo.

Jako wynik oddajesz:
   A) kody źródłowe;
   B) programy wykonywalne wraz z dokładną instrukcją ich uruchamiania
   C) ranking dla przykładowych, wybranych przez Ciebie danych (wg formatu opisanego w 3.)
   D) raport końcowy:
      - opis użytych algorytmów i struktur danych, problemy implementacyjne i ich rozwiązanie;
      - jakie ograniczenia na rozmiar danych ma Twój program, jak one zależą od dostępnej pamięci operacyjnej/dyskowej;
      - opis danych testowych, oraz wyników rankingu; wybierz kilkanaście zapytań, porównaj wyniki Twojego programu z wynikami jakie uzyskasz w programie będącym Twoim rozwiązaniem zadania programistycznego 1 do tego wykładu (ranking oparty na modelu wektorowym) i skomentuj różnice.

Zadanie powstało na podstawie http://www.cs.cornell.edu/courses/cs430/2006fa/assignments.html

Klasyfikacja dokumentów

Zaimplementuj następujące metody klasyfikacji dokumentów:

1. naiwny klasyfikator Bayesa (r. 13), pamiętaj o „wygładzaniu" zerowych prawdopodobieństw;
2. klasyfikator Rocchio (r. 14), parametry dobierz eksperymentalnie w oparciu o testowane dane;
3. klasyfikator liniowy oparty o SVM (r. 15 – tu możesz wykorzystać gotowe pakiety oprogramowania).

Danych wejściowe umieszczone są w katalogu o następującej strukturze:

- wszystkie dane treningowe (są to pliki tekstowe) są umieszczone w katalogu `training`
- wszystkie dane testowe są umieszczone są w katalogu `test`
- w pliku `klas.txt` znajduje się klasyfikacja wszystkich plików testowych i treningowych, każdy wiersz opisuje jeden plik, składa się z nazwy pliku i cyfry 1 bądź 0 oznaczającej przynależność do testowanej kategorii lub nie (nazwa i cyfra oddzielone spacją).

Twój algorytm powinien „uczyć się" w oparciu o zestaw treningowy i być testowany na zestawie testowym.

Jako wynik oddajesz:

E) kody źródłowe;
F) programy wykonywalne wraz z dokładną instrukcją ich uruchamiania
G) raport końcowy:
   - opis użytych algorytmów i struktur danych, problemy implementacyjne, wykorzystane pakiety oprogramowania;
   - wyniki klasyfikacji dla danych dostarczonych przez wykładowcę, w tym
     1. klasyfikację dokumentów testowych
     2. wyznacz wartość miary F1 i accuracy dla poszczególnych algorytmów
     3. w przypadku klasyfikatora Bayesa: 10 dominujących cech dla obu klas (czyli słów o największych wartościach P(w|c)*P(c)).

Zadanie powstało na podstawie http://lit.csci.unt.edu/~classes/CSCE5200/Assignments/Assignment3.txt

---

Wybrany artykuł naukowy

Wybierz artykuł naukowy omawiający zagadnienie/kwestie nie podane na wykładzie. Wybór uzgodnij z wykładowcą i/lub ćwiczeniowcem. Następnie przygotuj:

1. prezentację multimedialną dotyczącą treści artykułu,
2. wystąpienie w oparciu o prezentację, w 2 wersjach: 60 minut lub 20 minut.

Namiarów na ciekawe artykuły możesz (ale nie musisz) szukać w podrozdziałach „References and further reading" poszczególnych rozdziałów IIR i slajdów do tej książki.

Poniżej znajduje się lista (będzie ulegać zmianom) tematów preferowanych:

- artykuł przeglądowy o semantic web;
- przegląd zagadnień z zakresu web crawling;
- o szybkim liczeniu pagerank (np. http://elearning.unistrapg.it/webclass/mod/resource/view.php?id=87);
- ...