FREIE UNIVERSITÄT BOZEN
LIBERA UNIVERSITÀ DI BOLZANO
FREE UNIVERSITY OF BOZEN · BOLZANO

Fakultät für Informatik | Facoltà di Scienze e tecnologie informatiche | Faculty of Computer Science

Information Search and Retrieval

Written Examination

16.7.2013

FIRST NAME		**LAST NAME**	
STUDENT NUMBER		**SIGNATURE**	

Instructions for students:

Write First Name, Last Name, Student Number and Signature where indicated. If not, the examination can not be marked.

Do not speak to any other student during the examination. If you speak to another student, your examination will be cancelled.

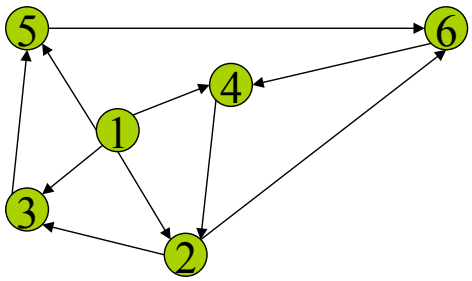
Use a pen, not a pencil.

Write neatly and clearly.

Reply to the following questions. Keep the reply short; do not write more than 10 lines of text for each question.

You cannot consult any material.

1. Consider the graph depicted below: it represents a set of 6 web pages and their hyperlinks.



Write the 6x6 matrix $A_{ij} = \beta M_{ij} + (1-\beta)/N$, where:

- N is the number of nodes;
- M is the row stochastic 6x6 matrix with entries $M_{ij}=1/\deg(i)$ if there is an hyperlink from node i to node j and deg(i) is the number of hyperlinks exiting from node i (outbound links) and $M_{ij}=0$ otherwise;
- $\beta=0.85$.

Describe one method for computing the Google page rank vector of this simple web graph using the matrix A.

2. Adapt the Intersect algorithm for Boolean queries of the type “Brutus OR Cesar” (note that the following original Intersect is for queries of the type “Brutus AND Cesar”).

```
INTERSECT(p1, p2)
 1  answer ← ⟨ ⟩
 2  while p1 ≠ NIL and p2 ≠ NIL
 3  do if docID(p1) = docID(p2)
 4        then ADD(answer, docID(p1))
 5             p1 ← next(p1)
 6             p2 ← next(p2)
 7        else if docID(p1) < docID(p2)
 8                then p1 ← next(p1)
 9                else p2 ← next(p2)
10  return answer
```

3. We have a two-word query. For one term the postings list consists of the following 16 entries:

   [4,6,10,12,14,16,18,20,22,32,47,81,120,122,157,180]

   and for the other term the postings list is:

   [47, 122]

   Work out how many comparisons would be done to intersect the two postings lists if there are skip pointers on the first list between the following pairs of entries in positions: (1, 3), (3, 5), (7, 12).

```
INTERSECTWITHSKIPS(p1, p2)
 1  answer ← ⟨ ⟩
 2  while p1 ≠ NIL and p2 ≠ NIL
 3  do if docID(p1) = docID(p2)
 4        then ADD(answer, docID(p1))
 5             p1 ← next(p1)
 6             p2 ← next(p2)
 7        else if docID(p1) < docID(p2)
 8                then if hasSkip(p1) and (docID(skip(p1)) ≤ docID(p2))
 9                        then while hasSkip(p1) and (docID(skip(p1)) ≤ docID(p2))
10                                do p1 ← skip(p1)
11                        else p1 ← next(p1)
12                else if hasSkip(p2) and (docID(skip(p2)) ≤ docID(p1))
13                        then while hasSkip(p2) and (docID(skip(p2)) ≤ docID(p1))
14                                do p2 ← skip(p2)
15                        else p2 ← next(p2)
16  return answer
```

4. Compute the Jaccard coefficients (based on bigrams representation) between the query ‘carg’ and each of the terms that contain the bigram 'ar', as follows: ar -> (car, bar, target, argo). Based on that computation what is the spell corrected word of ‘carg’?

   Remember: $Jacc(A,B) = |A \cap B| / |A \cup B|$

5. Consider the following three documents:

D1 = “free car park ride car”

D2 = “park ride free ride”

D3 = “free meal ride”

a) Compute the tf-idf representation of these 3 documents. idf_t is $\log_{10}(N/df_t)$, where N is the number of documents and df_t is the document frequency of the term t. Use the log-frequency weighting for tf, i.e., the tf component of tf-idf is obtained by the original term frequency $tf_{t,d}$ using the following formula:

$$w_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d}, & \text{if } \mathrm{tf}_{t,d} > 0 \\ 0, & \text{otherwise} \end{cases}$$

b) Length normalize the documents and find whether D2 or D3 is closest to D1 (cosine similarity).

6. Consider an information need (query) for which there are 5 relevant documents in the collection. Contrast two systems run on this collection. Their top 10 results are judged for relevance as follows (the leftmost item is the top ranked search result):

   System1: RNRNNRNNRR

   System2: NRNNRRRNNR

   What is the MAP of each system (considering this unique information need)?

   Which has a higher MAP?

   Definition of MAP: if the set of relevant documents for an information need q_j is $\{d_1, ..., d_{m\_j}\}$ and R_{jk} is the set of documents retrieved until you get d_k, then:

$$\mathrm{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \mathrm{Precision}(R_{jk})$$

7. Consider the following training set

   D1: "American Boston American Rome" -> Class=Y

   D2: "American Boston Chicago" -> Class=Y

   D3: "American Italy" -> Class=Y

   D4: "Rome Italy American American Chicago" -> Class=N

   Predict the class of the following test document using the **Multinomial** model

   D5: "American Rome Italy Italy"

   Calculate the required $P(c_j)$ and $P(x_k \mid c_j)$ terms as follows:

   For each class c_j in C

   $docs_j \leftarrow$ subset of documents for which the target class is c_j

   $Text_j \leftarrow$ single document containing all $docs_j$

   for each word x_k in *Vocabulary*

   $n_{jk} \leftarrow$ number of occurrences of x_k in $Text_j$

   $n_j \leftarrow$ number of words in $Text_j$

$$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$

$$P(x_k \mid c_j) \leftarrow \frac{n_{jk} + \alpha}{n_j + \alpha |Vocabulary|}$$

   α=1.

8. What is the entropy of a labeled collection where the "hot" elements are ten times the "cold" ones? Is it larger than the entropy of a collection where the "hot" and "cold" elements have equal size? The definition of entropy of the collection S is as following ($p(S_c)$ is the probability that an element is *c* (*hot* or *cold*)):

$$E(S) = \sum_{c \in \{hot, cold\}} - p(S_c) \log_2 (p(S_c))$$

9. Consider a two classes classification problem that is **not** linearly separable.
    a. Make an example of such a problem by plotting some training points in the two dimensional input space [0, 1] x [0, 1].
    b. Make a second example of a classification problem, in the same input space, that is not linearly separable but where there is a linear classifier (e.g. Naïve Bayes) that can obtain the Bayes error rate (i.e., the smallest achievable misclassification error for that problem).

10. Consider a **user-based** Collaborative Filtering recommender system. If an active user u_0 has four neighbor users (u_1, u_2, u_3, u_4) with Pearson Correlations: 0.7, 0.6, 0.2, and 0.7 respectively. Assume further that there are two items i_1 and i_2, in the profiles of these four neighbors, and their ratings are: (1, 2, 4, 5) = $(r_{11}, r_{21}, r_{31}, r_{41})$ and (4, 5, 1, 2) = $(r_{12}, r_{22}, r_{32}, r_{42})$. What is the best item to recommend to u_0 (among these two)? Explain the rationale of the reply by computing the predicted rating for i_1 and i_2, i.e., r^*_{01}, and r^*_{02} for u_0 using the following formula:

$$r^*_{uj} = K \sum_{v \in N_j(u)} w_{uv} r_{vj}$$

w_{uv} is the Pearson Correlation between user u and v, $N_j(u)$ is the neighbor of users similar to u (in our case is composed by $u_1, \ldots, u_4$), and:

$$K = 1 \Big/ \sum_{v \in N_j(u)} |w_{uv}|$$

11. Reply to the following questions:

    a. Define the notion of precision and the recall of a recommender system.

    b. Explain how to convert a user rating for an item in a 1 to 5 scale a relevant/not-relevant judgment.

    c. Let us assume that in a movie catalogue of 1000 items there are 300 movies that are then judged as relevant for user u. What is the precision and recall of the top-10 recommendations if 4 out of 10 are among the 300 that are relevant for user u.

    d. Consider two recommender system algorithms A and B. Can A have a better precision than B but also a higher Mean Absolute Error? Explain the reply with an example, considering one test user, assuming that you are testing A and B on top-5 recommendations, and you know the true ratings and the predicted ratings for the top-5 recommendations produced by the two algorithms for this user.

12. The context-aware recommendation method called “reduction-based” is a pre-filtering approach that generates rating predictions in a particular contextual condition (e.g., time=weekday) by reducing the multi-dimensional context-aware rating prediction problem to a standard bi-dimensional one. This is achieved by considering only the ratings acquired in that specific contextual condition. How the reduction-based approach deals with the potential problem that there could not be enough ratings acquired in a given contextual condition to train a reliable prediction model for rating prediction under that condition?