

DAIS Qualifying Examination  
*Fall 2007*

Department of Computer Science  
University of Illinois at Urbana-Champaign

Sept. 17, 2007  
Time Limit: 180 minutes

Please read the following instructions carefully before starting the test:

- The examination has 13 questions, including one basic concept question and 12 topic questions. You are required to answer the basic concept question and any **5** of the 12 topic questions of your choice. If you answer more than five topic questions, the committee will randomly select five to grade.
- The basic concept question has **12** subquestions. You are required to answer **9** of these 12 subquestions. If you answer more than 0 subquestions, the committee will randomly select 9 to grade.
- The 12 topic questions are distributed by areas as follows:
  - **Database Systems:** 3 questions
  - **Data Mining:** 3 questions
  - **Information Retrieval:** 3 questions
  - **Bioinformatics:** 3 questions

Each of these 12 questions generally has three parts. Make sure that you answer all the three parts of any topic question that you have chosen to answer.

- Use a separate booklet for each question that you answer. That is, you will use a total of six booklets (1 for the basic concept question and the rest for the 5 chosen topic questions).
- To ensure the fairness of grading, all the exams will be kept anonymous. Your answer sheets should thus only bear the assigned ID but *no names*.
- Please make your answers clear and succinct; you will lose credit for verbose, convoluted, or confusing answers. *Succinctness does count!*
- The questions have been designed to avoid any ambiguity. However, in case you are not sure how to interpret a question, please assume one interpretation, write down your assumption, and solve the problem accordingly.
- On average, you will have 30 minutes for each of the six questions that you have chosen to answer. So plan your time accordingly, and keep in mind that some questions/subquestions may demand more writing than others.

## Required Question (Basic Concepts): Problem 0

You are required to answer 9 out of the following 12 subquestions. Each subquestion is worth 1 point. Please focus on the major point and keep your answer concise; in general, an answer is expected to be no more than two sentences.

- (1) (*two phase commit*) Give a concrete example involving reads and writes of objects  $A, B, \dots$ , at database sites  $S_1, S_2, \dots$ , that shows why many people do not like two phase commit. Use as few objects, data sites, users, and events as you can to show what the problem is.
- (2) (*normalization*) Define *one* of the following normal forms: 3NF, 4NF, BCNF.
- (3) (*multidimensional access methods*) Explain why *one particular kind* of multidimensional index usually works poorly in practice when the number of indexed attributes gets large (e.g., more than 15 dimensions). You can use R-trees, kd-trees, or another well-known kind of index to explain what goes wrong.

**Note: Please start with a new answer sheet.**

- (4) (data cube and OLAP) A data cube may contain different measures which can be categorized as *distributive*, *algebraic*, and *holistic*. Which category each of the following belongs to: (1) standard-deviation; (2) top- $k$  for  $k \leq 10$ ; and (3) Q1 (*i.e.*, 25 percentile)?
- (5) (selection of data mining algorithms) Name one data mining algorithm that best fits each of the following applications: (1) finding strange moving cars in a highway network, (2) predicting which category that a new webpage should belong to if you know the categories of some existing web pages, and (3) determining whether the value of a particular stock is too high, if you know the history of many stocks.
- (6) (data mining applications) List one good application example for each of the following algorithms: (1) density-based clustering, (2) frequent-pattern based classification, and (3) Hidden-Markov Model (HMM).

**Note: Please start with a new answer sheet.**

- (7) (*Vector Space Retrieval Model*) In the vector space retrieval model, what three weighting factors are often considered when designing a term weighting function?
- (8) (*Language Model*) What is a background language model? How is it used in the language modeling approach to information retrieval (*i.e.*, the query likelihood retrieval approach)?
- (9) (*PageRank*) In order to compute PageRank, what matrix needs to be constructed? What pages will have high values according to PageRank?

**Note: Please start with a new answer sheet.**

- (10) (*alignment*) What does the term “affine gap penalty” refer to in the context of sequence alignment ?
- (11) (*Hidden Markov models*) Name two applications of Hidden Markov models in sequence analysis in bioinformatics.
- (12) (*Phylogeny*) For the purpose of evolutionary comparison, what does the term “outgroup” mean, and why is it important ?

## Database Systems: Problem 1. Query optimization

Consider the following relational schema for students and courses at UIUC: Student (sname, snumber), Course (cname, cnumber, cdept), Enrollment(snumber, cnumber, semester, grade). Suppose that for the past 50 years, every year UIUC has had 35,000 students, a fixed set of 100 departments, and a fixed set of 10,000 courses, of which 5,000 are offered in any given semester.

### Part 1

1. Draw or describe simple but believable **table-level histograms** and use them to estimate how many students got A's in CS courses last semester. (For example, don't assume that the CS department did not offer any classes, or no one ever gets A grades.) (1 point)
2. Suppose that your histograms are completely correct. Give one believable reason why your estimate is still probably quite wrong. (1 point)

### Part 2

Once again, consider the query  $Q =$  "how many students got A's in CS courses last semester." Create a reasonably small, believable tuple graph synopsis for the schema above (1 point), embed  $Q$  in it (1 point), and show how the synopsis leads to a more accurate estimate than your histogram technique (1 point). Show just enough of the tuple graph synopsis to justify your answer. (3 points)

### Part 3

Suppose that you work for the UIUC registrar's office, and are in charge of DBMS tuning for the DB described above. You have access to a log containing all the (millions of) queries run against the database last semester through the BANNER system. Suppose that you are running out of space to store new enrollment tuples, and you must reduce the size of your current sets of tuple graph synopses. Describe a reasonable methodology for (a) deciding which synopses to make smaller (2 points) and then (b) deciding which parts of them to compress (3 points). (5 points)

## Database Systems: Problem 2. Search and Crawling

### Part 1

In the “To Search or To Crawl” paper, why is there a contrast between *search* and *crawl*? What are the trade-offs? (2 points)

### Part 2

The paper is concerned of data gathering strategies for *task-centric* tasks. Give examples of two such tasks that may require different data gathering strategies. Identify these tasks, and explain what strategies might be appropriate for them. (2 points).

### Part 3

This paper proposes an interesting framework for “optimizing” data gathering strategies.

- (a) The paper relates the problem to the query optimization problem in DBMS query processing. Compare the two problems, and identify one significant similarity and one significant difference. (2 points)
- (b) The paper assumes *scan* and *query* as the two basic primitives for retrieving documents. Do you think it is necessary to extend beyond these two? Explain why or why not. (2 points)
- (c) In case you would like to extend the framework beyond the two primitives, how would you propose to make the framework extensible? (2 points)

## Database Systems: Problem 3. Ranking in RDBMS

### Part 1

What is the purpose of *ranking* for querying? Give one application scenario for querying a DBMS where ranking might be crucial. To contrast, give another scenario where ranking is irrelevant. (2 points)

### Part 2

Does SQL support ranking or not? Explain your answers either way. If *yes*, what is the construct that supports ranking? If *no*, is there any construct close to it? (2 points).

### Part 3

(a) What does *context-sensitive ranking* in the paper of Agrawal et. al. mean? Contrast it with context-free ranking with examples. (2 point)

(b) Why does context-sensitive ranking complicate query processing? (2 points)

(c) The paper defines “democratic voting” to determine overall the ranking given a set of preferences. We ask you to *disagree* with this particular design, and propose an alternative. Argue why your design addresses the issues. (2 points)

# Data Mining and Data Warehousing: Problem 1

## Part 1

Different data sets may require different cube computation algorithms. Outline one algorithm that is most suited for computing each of the following data cubes: (3 points)

- (a) a data cube with 5 dimensions, dense data, and  $10^7$  rows
- (b) an iceberg cube with 10 dimensions, sparse data, and  $10^6$  rows, and
- (b) a structure that may support OLAP operation for 100 dimensions, sparse data, and  $10^5$  rows.

## Part 2

Multidimensional data modeling is essential at designing data warehouses and data cubes. Although people have saved e-mails in tree-structured mail directories, it is more desirable to construct an e-mail data warehouse so that e-mails can be searched in multi-dimensions, such as based on sender (i.e., `from_list`), recipients (`to_list`), sending/receiving time, topic, keywords in title, length, attachment, keywords in the body, and so on.

- (a) Design such an e-mail data warehouse by drawing star schema. (1 point)
- (b) Outline how such a data warehouse can be implemented efficiently. (2 points)

## Part 3

One may like to construct data cube for data stream for certain applications.

- (a) Give a few typical applications of such a stream data cube, (1 point)
- (b) Based on one such application, outline how such a stream data cube can be implemented efficiently. (1 point)
- (c) if the stream data is high dimensional, can you work out an efficient implementation method? If yes, outline the method. If not, what is the major difficulty? (2 point)

## Data Mining and Data Warehousing: Problem 2

### Part 1

- (a) Discuss why “*null-(transaction) invariance*” is an important property at measuring pattern interestingness in large transaction databases. (1 point)
- (b) What should be the best measure for justifying whether a set of items are strongly correlated in a large set of transactions? Why? (1 point)

### Part 2

- (a) What is the worse-case computational complexity of mining frequent patterns? Why do many good frequent pattern mining algorithm claim that it can usually find the complete set frequent patterns efficiently under a reasonable *min\_support* threshold? (2 points)
- (b) Why that a typical frequent pattern mining algorithm encounters difficulty at finding rather large patterns (such as of size 100)? Can you propose a method that may find such pattern efficiently? (2 points)

### Part 3

- (a) Sequential pattern mining has been used for studying customer shopping behavior. Illustrate three major kinds of sequent pattern mining methods. (2 points)
- (b) To derive desired results, a user may like to enforce constraints on sequent patterns. Discuss how to categorize constraints, and how to push categories deeply into the mining process. (2 points)

## Data Mining and Data Warehousing: Problem 3

### Part 1

- (a) What are the major difficulties of classifications in stream environment vs. nonstream environments? (*1 point*)
- (b) It is often desirable to predict rare events, such as computer network intrusions in a data stream environment. Discuss how to perform high quality classification for such rare events in data streams. (*1 point*)

### Part 2

- (a) Decision-tree is a popular classification method. However, when the training data is too huge to fit in main memory, a typical decision-tree induction may have to do a lot of I/O operations. Present a highly scalable method for decision tree induction. (*2 points*)
- (b) Similarly, present a highly scalable method for support vector machine (SVM) induction. (*2 point*)

### Part 3

- (a) The method proposed by Hong et al. (2007) for pattern-based classification is to first mine the full set of closed frequent patterns and then extract a set of discriminative frequent patterns for effective classification. Explain why this method may lead to high classification accuracy. (*2 points*)
- (b) Can you extend the method for classification of sequential patterns? Outline your proposed method. (*2 points*)

# Information Retrieval: Problem 1. Web Search

## Part 1

- a List at least 3 kinds of information about a web page that we can potentially exploit to improve retrieval accuracy over a standard vector space retrieval model for ranking web pages. (*1 point*)
- b Compare Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG), and point out their similarities and differences. (*2 points*)

The following questions are based on the paper “Optimizing Web Search Using Social Annotations” [Bao et al. WWW 07].

## Part 2

- a The authors proposed two algorithms to exploit social annotations (i.e., SocialSimRank and SocialPageRank). For each algorithm, briefly describe a scenario (e.g., an example query and some example documents) where the algorithm would help improve search accuracy. (*2 points*)
- b The SocialPageRank algorithm is essentially to compute the principal eigenvector for a symmetric matrix  $MM^T$ . What are the three association matrices used to define  $M$ ? As more and more users provide annotations, what kind of pages will get higher SocialPageRank values? (*2 points*)

## Part 3

Annotation spamming is identified as a potential barrier for using the proposed techniques to improve Web search. Briefly describe how a spammer can boost a page’s ranking through malicious annotations. Can you propose some ideas for automatically detecting such malicious annotations? (*3 points*)

## Information Retrieval: Problem 2. Retrieval Model

### Part 1

The Markov Random Field (MRF) retrieval model would rank a document  $D$  w.r.t. query  $Q$  using the following formula:

$$P_{G,\Lambda}(Q, D) = \frac{1}{Z_\Lambda} \prod_{c \in C(G)} \phi(c; \Lambda)$$

- a What does  $C(G)$  denote? (1 point)
- b Sketch how you may use an MRF model to simulate an ordinary TF-IDF weighting retrieval formula (e.g., Pivoted normalization) through defining the potential function  $\phi(c; \Lambda)$  appropriately. (2 points)

### Part 2

In the paper “Latent Concept Expansion Using Markov Random Fields” [Metzler & Croft SIGIR 07], some latent concepts are selected to expand the graph  $G$  to achieve query expansion. A latent concept  $E$  is scored using the following formula:

$$P_{H,\Lambda}(E|Q) \propto \sum_{D \in R_Q} \exp[F_{QD}(Q, D) + F_D(D) + F_{QD}(E, D) + F_Q(E)]$$

- a What is  $R_Q$ ? (1 point)
- b Explain how the term  $F_{QD}(Q, D)$  can intuitively help us select good latent concepts. (2 points)
- c In the evaluation, what kind of measure did the authors use to show that this latent concept model is more robust than a baseline method? (1 point)

### Part 3

In the paper [Metzler & Croft SIGIR 07], the authors tied the parameters of the retrieval model. Can you explain why they wanted to do that? In general, as we introduce more features to this kind of models, we will have more parameters. How do you think we should estimate/set these parameters?

## **Information Retrieval: Problem 3. Sentiment Retrieval**

The questions refer to the following two papers:

[Eguchi & Lavrenko 06] Sentiment Retrieval using Generative Models, K. Eguchi and V. Lavrenko, Proceedings of EMNLP 2006, pp. 345-354.

[Popescu & Etzioni 05] Extracting Product Features and Opinions from Reviews, A. Popescu and O. Etzioni, Proceedings of HLT/EMNLP 2005, pp. 339-346.

### **Part 1**

- a In [Eguchi & Lavrenko 06], two variations of the sentiment retrieval task are considered in evaluation. Give an example query for each variation. (*1 point*)
- b While in most traditional retrieval literature, documents are units for retrieval, in [Eguchi & Lavrenko 06], the so-called “statements” are retrieval units. Briefly explain why the authors want to make such a distinction. (*1 point*)

### **Part 2**

- a In [Eguchi & Lavrenko 06], the ranking of a statement is affected by both topic matching and sentiment matching. Briefly explain how these two pieces of evidence are combined to produce a final ranking of statements. Suppose we completely ignore the sentiment matching part in their retrieval function and only consider ranking based on topic matching, what would the retrieval function look like? (*2 points*)
- b In the relaxation labeling algorithm used in [Popescu & Etzioni 05], a support function is defined. Briefly explain what probability the support function gives and how the support for label  $L$  of word  $w$  at iteration  $m$  (i.e.,  $q(w, L)_{(m)}$ ) is computed. (*2 points*)

**Part 3** Suppose you are to design a retrieval system to support a query where each keyword is tagged with a sentiment preference. For example a query may be like “urbana [+] champaign [-] apartment”, which attempts to find information about apartments in Urbana and Champaign and particularly want the matching of “Urbana” to be in a positive opinion context while that of “Champaign” in a negative opinion context. How would you design a retrieval model/algorithm to score a document for this kind of queries? (*4 points*)

## Bioinformatics: Problem 1. CRM finding - Gupta and Liu, 2005.

### Part 1

What is a cis-regulatory module, and why is it important to find them ? (*2 points*)

### Part 2

Briefly describe the probabilistic framework adopted by Gupta and Liu for finding cis-regulatory modules. (*5 points*)

### Part 3

How will you extend the basic approach of Gupta and Liu to an integrated framework that does not require any given motifs ? (*3 points*)

## Bioinformatics: Problem 2. Evolutionary comparisons, Siepel et al., 2005.

### Part 1

Why is it useful to find evolutionary conserved sequences genome-wide ? (*2 points*)

### Part 2

Briefly describe the phylogenetic Hidden Markov model proposed by Siepel et al. for cross-species comparison. (*5 points*)

### Part 3

Identify a possible future extension of the phylo-HMM approach, and explain how you would approach to do such an extension. (*3 points*)

## Bioinformatics: Problem 3. Motif finding, Narlikar et al., 2006.

### Part 1

What is the motif finding problem ? What model of motifs is used in the Narlikar et al. paper ?  
(2 points)

### Part 2

What is meant by transcription factor structural class ? How can this information be used as an informative prior ? What method does this prior information feed into, as per the Narlikar et al. paper ? (5 points)

### Part 3

Can you propose any other type of information that may be used as a prior within the probabilistic framework of Narlikar et al. ? How would you go about implementing such a prior ? (3 points)