

DAIS Qualifying Examination
Fall 2004

Department of Computer Science
University of Illinois at Urbana-Champaign

September 24, 2004
Time Limit: 180 minutes

- You are required to answer *six* questions. If you answer more than six questions, the committee will randomly select six to grade.
- Out of the six questions answered, at least *two* must be from the area of *Database Systems*. The remaining questions will be your choice. The distribution of the questions by areas is as follows.
 - **Database Systems:** 5 questions
 - **Data Mining:** 3 questions
 - **Information Retrieval:** 3 questions
- For each question, there will three subquestions. The first is to test the basic knowledge related to a topic, the second is to test current research related to the topic, and the third is to test research capability.
- To ensure the fairness of the grading, all the exams will be kept anonymous. Your answer sheets should 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. *Simplicity does count!*

Database Systems: Problem 1

Part 1

In the development of access methods, for tree-based indexing techniques, we have seen B-tree (1972), R-tree (1984), and then GiST (1995), among others. This series of trees represents a sequence of generalization.

Describe what generalizations exist in this series of trees. That is, how does R-tree generalize B-tree? How does GiST generalize R-tree?

Part 2

These trees are said to be *balanced*.

- (1) Why is this property desirable?
- (2) Are there non-balanced index trees that have been proposed? If your answer is *yes*, give an example of such a non-balanced tree. If your answer is *no*, give another example of a balanced tree.

Part 3

GiST is supposed to be a generic index tree, which can be “customized” for a specific type of data with respect to a specific type of queries. This problem asks you to propose some type of data and associated queries, and customize GiST for such data and queries:

- (a) Propose a type of data and associated queries that cannot be appropriately indexed by B-tree or R-tree. Explain why your choice cannot be indexed by these trees.
- (b) Describe your design of GiST: What is your *key* predicate? How does the *consistent* method work?

Database Systems: Problem 2

Part 1

1. Describe ACID properties (1-2 sentences per property is sufficient).
2. Briefly describe in several sentences the idea of checkpointing an undo/redo log.
3. What is a serializable schedule (in 3-4 sentences at most)?

Part 2

1. Briefly describe two-phase locking. Name at least one person who did significant work on this.
2. What are the main ideas of a typical transaction recovery method?

Part 3

1. Briefly describe how transaction management and concurrency control for relational databases change when applied to distributed databases. What new issues arise? What solution directions have been considered?
2. Briefly discuss how transaction management and concurrency control change when applied to data integration systems.

Database Systems: Problem 3

Part 1 Suppose the mediated schema of a data integration system consists of the following two relations:

Movie(title,dir,year,genre)
Schedule(theatre,title,time).

Suppose you has a source

V(theatre,genre),

that is, source V contains a table, each row of which is a tuple that lists a theatre and the genre of movies playing in that theatre.

1. Show how you would do schema mediation using LAV? GAV? (That is, how you would relate the mediated schema relations to V?)
2. Can you answer this query in each case: “Give me the theatres playing comedy movies”?

Part 2

1. Briefly explain the need for recursive rewriting of a query in data integration context. When does such a need arise? Give an example. What do you think can be done in such cases?
2. Briefly explain what schema matching is, and sketch at least one solution direction (any solution direction that has been studied is fine; you don't need more than a paragraph here). How is schema matching relate to query translation in the context of building data integration systems on the Deep Web? Are they the same problem?

Part 3

1. Name at least one more major problem in data integration that people are working on, or that you think people should work on. Briefly argue why the problem is important. Suggest some solution directions (at high level only, you don't need to go into detail).
2. How is peer-to-peer data management different from data integration? Name at least one research group that works in this area. Take a major problem in data integration, examine how the problem changes in the context of peer-to-peer, and briefly suggest some solution directions.

Database Systems: Problem 4

Part 1

What is a DataGuide, what is it used for, and how is it used?

Part 2 What is the equivalent of a DataGuide in the XML world? How does it differ from a DataGuide?

Part 3 Suppose that we have a large set of XML documents, and we decide to store them in Lore and query them with the Lorel language. What, if any, will be the shortcomings of this approach?

Database Systems: Problem 5

Part 1

How can a data stream management system be used to address problems in network security?

Part 2

Describe five ways in which a data stream management system's requirements are different from the requirements for a traditional DBMS.

Part 3 It would be good to have the equivalent of a "relational algebra" for queries over streams. What are the challenges in providing such a "stream algebra"? How might we overcome those challenges (give the general idea, not full details)?

Data Mining and Data Warehousing: Problem 1

Part 1

Assume a base cuboid of 10 dimensions contains only three base cells: (1) $(a_1, d_2, d_3, d_4, \dots, d_9, d_{10})$, (2) $(d_1, b_2, d_3, d_4, \dots, d_9, d_{10})$, and (3) $(d_1, d_2, c_3, d_4, \dots, d_9, d_{10})$, where $a_1 \neq d_1$, $b_2 \neq d_2$, and $c_3 \neq d_3$. The measure of the cube is *count*.

- (a) How many **nonempty** cuboids will a complete data cube contain?
- (b) How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?
- (c) how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is " $count \geq 2$ "?

Part 2

There are several typical cube computation methods, such as multiway array computation (Zhao, et al. SIGMOD'1997), BUC (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), and StarCubing (Xin et al., VLDB'2003).

Briefly describe these three methods (i.e., use one or two lines to outline the key points), and compare their feasibility and performance in the following conditions:

- (a) computing dense full cube of low dimensionality (e.g., less than 8 dimensions),
- (b) computing iceberg cube around 10 dimensions with highly skewed data distribution, and
- (c) computing sparse iceberg cube of high dimensionality (e.g., over 100 dimensions).

Part 3

A cell c is a *closed cell* if there exists no cell d such that d is a specialization of cell c (i.e., d is obtained by replacing a $*$ in c by a non- $*$ value) and d has the same measure value as c . A **closed cube** is a data cube consisting of only closed cells.

- (a) How many closed cells in the full cube of Part 1?
- (b) Proposed an algorithm that computes closed iceberg cubes efficiently.

Data Mining and Data Warehousing: Problem 2

Part 1

Decision tree induction is a popular classification method. Taking one typical decision tree induction algorithm C4.5 (Quinlan 1993), briefly outline the method of decision tree classification.

Part 2

There are many classification methods developed in research. However, some of them may not be scalable to very large data sets. RainForest (Gehrke, Ramakrishnan, and Ganti, VLDB'98) and Sprint (Shafer, Agrawal, Mehta, VLDB'96) are two well-known algorithms that perform scalable decision tree induction.

- (a) By comparison with C4.5, outline how RainForest builds classification models in very large databases.
- (b) Compare and outline the major differences of two scalable decision tree induction algorithms, RainForest and Sprint.

Part 3

The decision-tree induction with RainForest may work well for large database of low dimensionality. However, sometimes we need to build classification models for high-dimensional datasets. Propose a classification method that may work well in a data set of not too small size (e.g., one million tuples) but high dimensionality (e.g., 100 dimensions).

Data Mining and Data Warehousing: Problem 3

Part 1

There are many cluster analysis methods proposed in statistics, pattern recognition, and data mining research. Name 10 cluster analysis methods you know and group them into a few classes based on their analysis methodology.

Part 2

BIRCH (Zhang, Ramakrishnan, and Livny, SIGMOD'96) and CLARANS (Ng and Han, VLDB'94) are two interesting clustering algorithms that perform effective clustering in large data sets.

- (a) Outline how BIRCH performs clustering in large data sets.
- (b) Compare and outline the major differences of the two scalable clustering algorithms: BIRCH and CLARANS.

Part 3

Recently, there are many new applications that require us to perform data mining in huge volume of fast evolving data streams. For example, one may like to use cluster analysis method to detect outliers of computer network intrusion by comparing the behavior of current data stream with that of the previous ones. Propose an efficient clustering algorithm that detects such outliers in fast evolving data streams.

Information Retrieval: Problem 1. Text Categorization

Part 1

- a For a language with N words, what is the total number of (free) parameters of a language model based on multi-Bernoulli event model and one based on multinomial event model, respectively? Please keep in mind any probability constraints.
- b Given a multi-Bernoulli language model, how would the probability of a document change if we expand the document by adding one extra word occurrence to the end? Will it increase or decrease? What if we have a multinomial language model, instead of a multi-Bernoulli model, would you have the same conclusions?

Part 2

- a Explain what is Laplace smoothing and why smoothing is needed when estimating probabilities of language models for text categorization.
- b Suppose we use a Naive Bayes classifier to classify an email message as either a spam or legitimate email, and our vocabulary has only three words, w_1, w_2, w_3 . Use a multinomial distribution event model as shown in the following table to classify a three-word message $m = "w_2w_3w_2"$. Show your calculations.

$p(\text{SPAM}) = 0.2$	$p(\text{LEGITIMATE}) = 0.8$
$p(w_1 \text{SPAM}) = 0.1$	$p(w_1 \text{LEGITIMATE}) = 0.3$
$p(w_2 \text{SPAM}) = 0.1$	$p(w_2 \text{LEGITIMATE}) = 0.2$
$p(w_3 \text{SPAM}) = 0.8$	$p(w_3 \text{LEGITIMATE}) = 0.5$

Part 3 The Naive Bayes classifier makes a conditional independence assumption regarding word occurrences. Suggest an extension of Naive Bayes that can capture some dependence between word occurrences.

Information Retrieval: Problem 2. Language Models for Retrieval

Part 1

- a A common way of using language models for retrieval is to rank documents based on $p(Q|D)$, where Q is a text query, D is a text document, and $p(Q|D)$ is computed based on a unigram language model. Explain why such a ranking strategy would, in general, favor a document with more occurrences of query terms as we intuitively would like.
- b Ranking documents based on $p(Q|D)$ can be justified using the following formula

$$p(D|Q, U) \propto p(Q|D, U)p(D|U)$$

where U is a user variable. Explain what is the role of $p(D|U)$ and how we may use it to incorporate some desirable retrieval heuristics.

Part 2 The task of cross-lingual retrieval is to retrieve documents in language A (e.g., Chinese) with a query in a different language B (e.g., English). The translation models for retrieval proposed in [Berger & Lafferty SIGIR 99] can be adapted to support cross-lingual retrieval in the following way: Let $Q_{English} = q_1 \dots q_m$ be an English query and $D_{Chinese} = d_1 d_2 \dots d_n$ a Chinese document. We can rank documents by the following conditional probability:

$$p(Q_{English}|D_{Chinese}) = \prod_{i=1}^m p(q_i|D_{Chinese})$$

Assume that we have a translation model $t(X|Y)$, which gives us the probability of a Chinese word Y being translated into an English word X . Use the translation model $t(X|Y)$ and a Chinese document language model $p(Y|D_{Chinese})$ to refine the formula so that the conditional probability $p(Q_{English}|D_{Chinese})$ can be computed essentially based on these two models. What kind of training data would we need to estimate $t(X|Y)$?

Part 3

Suppose we estimate the conditional probability $p(D|Q, U)$ directly using the following formula

$$p(D|Q, U) = \prod_{i=1}^n p(d_i|Q)$$

where $D = d_1 \dots d_n$ is a document and $Q = q_1 \dots q_m$ is a query in the same language. And we estimate $p(w|Q)$ using the relative frequency count of word w in Q smoothed with some background language model so that it gives a non-zero probability to every word in our vocabulary. This formula will not perform well as a retrieval formula. Why?

Information Retrieval: Problem 3. PageRank

Part 1

The PageRank algorithm can be described as follows:

$$\vec{R} = (1 - \alpha)M\vec{R} + \alpha\vec{p}$$

where \vec{R} is a vector of importance values for all pages (i.e., rank values), and M is a link matrix defined based on the web graph. Explain why we need α and \vec{p} and explain how to interpret them when we view PageRank as computing the stationary probabilities of a random surfing model. In the standard PageRank, assume the total number of pages is N , what's the value of \vec{p} ?

Part 2

In the topic-sensitive PageRank algorithm proposed in [Haveliwala WWW 2002], which component of the PageRank equation (see part 1) is exploited to make PageRank topic-specific and how? Use a few sentences to describe how the PageRank scores are used to rank the documents for a query.

Part 3 In [Haveliwali www 2002], 16 basis topics are defined based on the open directory categories. Can you think of any better way of defining the basis topics? What are the advantages and disadvantages of using many more (e.g., 1,000) basic topics, if any?