Final Review

DSCI 551

Wensheng Wu

SQL

- Select…from…where…group by…having…order by…limit…offset…
- MySQL does not support intersect and except
- Subqueries:
  - =, in, >= all/any
  - (not) exists
- Aggregation
  - sum, min/max, count, avg
  - group by
  - having

SQL

- Join
  - Theta-join
  - Natural-join
  - Outer join (left and right, no full outer in MySQL and workaround)
- DML vs DDL
  - Create/alter table …
- Translate SQL into query plan in relational algebra
  - Or implement it using Python
- Recall homework 3

Constraints & views

- PK, FKs
  - When & how to enforce FKs (set null, cascade)
- Virtual vs materialized views
  - View unfolding
  - Reusing materialized views

Data representation & external sorting

- Fixed & variable-length record
- Packing records into a block
- External sorting
  - Two-way vs multi-way
  - Sorting phase & multiple merging phases: input, output, sizes
  - Costs

Indexing

- Clustered vs non-clustered index
- MySQL creates index automatically for …
- Composite index

- Search in B+tree & costs
- Insertion & deletion:
  - strategy to maintain the balance (splitting, rotating, and merging)
  - Effect on tree structure

Query execution

- One pass:
  - Selection, projection
  - Group by, distinct (constraint on memory)
  - Join (one of relations fits in memory)

- NLJ
  - Block-based algorithm
  - Costs: putting smaller relation in outer

Two pass join algorithms

- Sort-merge join
  - Pass 1: sort
  - Pass 2: merge
  - Constraints: B(R) + B(S) <= M * (M-1)

- Partitioned hash join
  - Pass 1: hashing R/S into (M-1) buckets
  - Pass 2: merge Ri and Si
  - Constraints: e.g., min(B(R), B(S)) <= (M-1)*(M-2)

Variations

- Simple-sort based join
  - Completely sort R & S
  - Merge

Index-based algorithm

- Selection
  - Costs: clustered index vs non-clustered index
- R(A,B) Join S(A,C): with index S.A
  - Costs: index clustered vs non-clustered
- Compare it to NLJ
- Zig-zag join: clustered indexes on R.A and S.A
- Recall homework 4

MongoDB

- Find()
  - Pattern matching
  - Query/filter operators: $gt, $lt, …, $and, $or, $not, $in, $all, $elemMatch, $exists, etc.
  - projection
  - sort
  - distinct // note it can't follow find()
  - count
  - skip
  - limit
- update (upsert): $set, multi:true, etc.

MongoDB

- Aggregate()
  - Pipeline: $match, $group ($sum, $avg, …), $lookup, $project, $sort, $limit, etc.
  - $filter will NOT be required for final
- Know how to translate SQL into MongoDB
- Recall lab 3

Hadoop MapReduce

- Job tracker, task tracker

- Map function & reduce function
  - Input, output, logic (be able to write pseudocode, see in class examples)
  - Compare it to Python map() and reduce() // recall examples in lecture

- Shuffling task:
  - recall diagram and discussions

Hadoop MapReduce

- Combiner
  - Commutative and associative
- SQL implementation
  - Join will NOT be covered in final exam
- Recall hw5

Spark dataframe

- Projection
  - E.g., country[['Continent', 'Region']] or country.select('Continent', 'Region')
- Selection
  - E.g., country[country.GNP > 10000] or country.filter('GNP > 10000)
- distinct/dropDuplicates()
- groupBy & agg // min, max, avg/mean, sum, count
  - E.g., how to implement select continent, count(*) from country group by …
- How to implement having? (e.g., filter())
- orderBy(…) // what about descending?
- limit(…)
- Join, natural join, outer join

Spark dataframe

- How to translate SQL into dataframe operation?

- Union, subtract, intersection (not required for final)

Spark RDD

- Creation: textFile, parallelize

- Transformations
  - map, flatMap, mapValues
  - filter (see lecture for examples)
    - E.g., finding all even numbers in a list of integers
  - groupByKey
  - reduceByKey(f) // understand how Spark applies function f in reduction

Spark RDD

- Actions: collect, reduce, max, min, sum, count, mean, aggregate
  - how reduce works on individual partitions in parallel?
  - how to implement sum/min/max using reduce?
  - how to implement count/mean using aggregate? (see example in lecture on mean)
- Differences between transformation and action
- Recall wordcount example shown in class

DynamoDB

- NOT required for final

Final exam (dates/times as shown in syllabus)

- MW 10am section:
  - 12/7, Monday, 8-10am
- MW 3:30pm section:
  - 12/7, Monday, 2-4pm
- Tuesday 3:30pm section:
  - 12/8, Tuesday, 2-4pm
- Online on Blackboard, similar to your midterm
  - closed-book and notes