Data Mining and Applied Statistics

1. Introduction to statistics, probability theory, distributions
2. Statistical inference, central limit theory, confidence intervals, hypothesis testing
3. Data Mining Process, Data Preprocessing
4. Association Rule Mining
5. Clustering
6. Classification
7. Regression
8. Ensemble learning, Preprocessing

Cloud Computing

1. Introduction
2. Technology & Types
3. Parallel Computing
4. Distributed System
5. Big Data
6. Map Reduce & Hadoop
7. Map Reduce Design Patterns
8. DM, LA, IR in the cloud

Cloud Computing

1. Cloud Computing INDEX-II

Information Retrieval & Text Mining

1. Boolean Retrieval
2. Term Vocabulary & Posting lists
3. Dictionary & Relevant Retrieval
4. Index construction & Compression
5. Scoring, Term-Weights, Vector space Model
6. Evaluation, Relevance, Query Expansion
7. Text clustering & Classification
8. Recommender Systems
9. Social Network Analysis
10. Opinion Mining
11. Information Extraction

Semantic Web Technologies

1. Introduction
2. RDF-I
3. RDF-II
4. OWL-I
5. OWL-II & upper layers
6. Network, webarchitecture, HTTP
7. Web technologies
8. Web mining, crawling 2-0
2-phase locking protocol
- read, write locks
- strong, strict 2-phase locking
- R-10, wait, rel.
- growing lock set
- 2PL ops is serializable

Deadlocks
- wait-for graph
- about one
- prevention
  - timestamp
  - priority
  - combination

4. Stored Procedures, Triggers, Active Databases
- Stored procedures: code shared among sel/psm (queries + updates)
- Triggers: (event based)
  - Calls procedures
- Active Databases:
  - Supports triggers
    - Events
    - Condition
    - Action
    - Insert / update
    - cascading
    - Statement from level
    - Applications
      - Constraint violations
      - Views
      - Logging
      - Stats, authorization
Parallel databases
- parallel processing tech + phys
  - shared mem
  - shared disk
  - shared nothing
- throughput / response time
- speed-up vs. sequential:
  - (faster proc.)
  - (more proc.)
- intra / inter query parallelism
  - round robin
  - hash range
- data partitioning (sharding)
- parallel impl. of rel. op.
  - split & merge parallel streams of data
  - sort
  - project
  - join
- parallel merge sort
- parallel hash join
- parallel nested loop join

6. Distributed Databases Database Design

- data modeling ➔ refinement (relational)
  ➔ (ER)
  ➔ File, SQL Tables
    ➔ (physical) (conceptual)
  ➔ conceptual schema: relational model + functional dependencies
  ➔ normalization: eliminate anomalies
  ➔ ER ➔ reln. tables

- problematic DB design
  ➔ redundancy
  ➔ anomalies
  ➔ solution: decomposition
    ➔ lossless-join, dependency preservation
  ➔ how: functional dependency, normal join
Applied Statistics

1. Introduction to statistics, probability theory, and distributions

- Data lifecycle: Business understanding → Data Understanding → Deployment → Eval → Modeling → Data Prep

- Statistical → descriptive
  - inferential (sample)
  - population → sample → statistic
describe

- Nominal
  - Categorical
    - Ordinal
  - Interval
    - Discrete
    - Continuous

- Measures of location: mean, median, mode, skew.
- Measures of dispersion: range, variance, quantile

- Decision support: window queries, top-k, online aggregation

- Indexing: bitmap, join, views

- Rollup / drill down
  - Bottom / top

- n-dimensional aggregate
  - Generate powerset of aggregating columns

- Dimension hierarchies
  - Synchronize at-depth levels
  - Rollup / drill-down
  - Pivot
  - Cross-tabulation
- linear regression
  - target variable 'y' linearly dependent on attributes

- isotonic regression
  - non-linear, correct pairs of forms

- SVM for regression

- local regression - kNN + linear regression

- decision tree + regression
  \[ \text{reg tree} - \text{model tree} \]
  \[ \text{(piecewise constant fn)} \quad (\text{piecewise linear}) \]

- use ANN for regression

- Ensemble
  - online learning
    - wisdom of crowds
    - ensembling predictions from different learners
    - accuracy, diversity
    - voting
      - class
        - reg
          - (most voted) (avg)
    - bagging
      - different samples for different learners
      - learn models & combine
      - randomization
        - (randomize the learning alg, initialize input data)
        - eg: random forests
      - weighted voting with confidence values
      - boosting
        - set of classifiers one after another
        - data classifiers focus on errors by rank
        - iterations
      - stacking
        - learn a function to combine predictions
model-based
- item-based
  (similarity between items)
  (cosine sim)
  - ARM
  - prob methods
  - bayesian

- content-based reco
  - utilize item information
  - keyword overlap
  - content rep/user-profile
    - knn
  - hybrid
    - voting, weights

- evaluating
  - accuracy
  - mean squared error
  - precision, recall, F-1

9. Social Network Analysis

- store of data: web crawl, blogs, social sites, community data
- networks:
  - graph < directed/un-directed
  - vertex, arc, edge
  - degree distribution
    - freq count
    - prob dist
    - avg degree
  - graph density
    - L infinity
  - clustering coeff
    - (density of connections in neighborhood)
  - adjacency matrix, bipartite graph
  - paths
    - n/2 diameter, avg distance
community detection
- set of frequently interacting actors
- connected graph, components, bridge, cut-vertex, giant component, isolates
- SCC / USCC
  (div) (undiv)
- web: centralities, Pagerank and subgraph communities
- k-cores: max undiv subgraph in which each vertex has at least k deg
  - clique: max connected complete subgraph
  - divisive hierarchical clustering (remove "weaker")
  - edge betweenness: no of shortest paths that pass through
- islands
- prominence: centrality / prestige (in, out) (target)
- power-law distributions (degree dist)
- small-worldness (clustering coeff, shortest path length)

10. Sentiment Analysis & Opinion Mining
- classify polarity of given text
  - polarity values: +ve, -ve, neutral
  - decline / feature level
  - mine opinion in un-generated content
    - regular comparison
    - opinion, target of opinion, opinion holder
- entity $t$ = $A$ (finite set of aspects)
  - $\{a_1, a_2, \ldots, a_n\}$
  - model of entity
  - model of review
    - contains $\cup_{s \in S} \{s\}$
    - choose a word to desc entity for pos
    - +, -, neutral
Cloud Computing

1. Introduction

- supercomputer, grid $\rightarrow$ cloud
  $\rightarrow$ cloud

  $\downarrow$

  BDP utility XaaS web2.0

- principles
  - pooled resources: available to all
  - virtualization: high utilization of I/T infrastructure
  - elasticity: dynamic scalability
  - automation: no manual
  - metered billing: pay as

- benefits
  - economic
  - agility
  - efficiency
  - security
web browser - client/server

servers - sshd, httpd: listening on port

multithreaded web server - model: ked by servlets

URL: address of doc

html: describe structure of doc

tag attributes
anchor key
tables, images

content vs presentation

CSS - display, html docs

6. Networking

- Local hierarchy
  - layers, protocols, interfaces
  - n/w arch
  - set of layers + protocols
    ▼ services ▼ rules of conv
    ▼ primitives

- OSI Reference model

App - Presentation - Session - Transport
  ▼ Phy - DL - N/w

- Sending data: connectionless / connection oriented
  (ckt) (pkt)

- TCP/IP
  connection oriented, UDP: connectionless

- IPv4, IPv6 addresses
  classes A, B, C, D, E, F; CIDR
6. Typing, Memory management

- In statically typed languages
  - for every operation, types of data on which it is applicable
- Weak/strong typing
  - type-safe
  - prevent value of one type to be treated as another
- Static/dynamic typing
  - Compile-time (no-type in variable) during prog.
- Activation records: allocated when proc is entered & deallocated when proc is exited
  - (info needed by single of proc) exec
Software requirement analysis & specifications
- Customer based
- Requirements gathering
  - Functional
  - Non-functional
- Requirements analysis
  - Identify, list, define, describe
  - Data flow diagram
  - Entity relationship diagram
  - Data dictionary
- Requirements documentation
  - SRS: software req. spec.
    - Functional
    - External interfaces
  - Performance
  - Design constraints

Software design
- Design: given SRS doc., propose a solution

Software engineering
- Software project planning
  - Size estimation
  - LOC, function cost
  - Overall cost estimation
  - Risk estimation

Software Quality Assurance
- Coupling: degree of independence between modules (independence)
  - Low: high (better)
  - Coupling < control
  - External common content

Checkers:
- Functionally related 2 modules are
  - Software project planning
  - Size estimation
  - LOC, function cost
  - Overall cost estimation
  - Risk estimation

Software design
- Design: given SRS doc., propose a solution
- reusability
- scalability
- complexity
- cost
- open-standard protocol stack

- Technical details
  - XML data transfer
    - SOAP: HTTP+XML
  - client + UDDI registry + endpoint
    - WSDL
    - UDDI
  - component model
    - interface
    - implementation

-- Fundamentals of Computing

1. Digital
   - Description logic, arithmetic
   - computers binary: simple, easy to build, more efficient
   - logic: valid inferences & correct reasoning
   - short is T/F
   - connectives: ∧, ∨, →
   - truth tables
   - boolean logic in computers
     - bits 0, 1
     - logic gates: AND, OR, NOT
   - boolean expr → boolean dt
   - more gates: XOR, NAND, NOR
   - 2^n boolean function
   - \{T, ∧, ∨\} is universal set of boolean func
   - NAND, NOR are also universal
Python

- interpreted
- dynamically typed
  - object oriented
  - classes, inheritance, exceptions
  - functions as independent entities, types, variables, indicators

- Types
  - tuples (immutable) (fast) - 1D index
  - lists (mutable) (slow)
  - strings (immutable)
  - extend, + (new object)

- Dictionaries
  - maps (key-value)

- boolean exp
  - if, for, break, continue

- abuse + range() [anti-pattern]

- list comprehensions
  - gen a list by applying function to each member

- functions: no overloading, lambda, first-class objects
- inheritance, import
- strings, % formatting