COMP6714 Review

Wei Wang

weiw AT cse.unsw.edu.au

School of Computer Science and Engineering
Universities of New South Wales

November 21, 2019

Course Logisitics

- **THE** formula:

$$mark = \begin{cases} 0.20 \cdot (ass1 + proj1) + 0.60 \cdot exam & \text{, if } exam \geq 40 \\ 39FL & \text{, otherwise.} \end{cases}$$

- Exam date: 2 Dec 2019, 17:45 – 20:00 (Check your own timetable `https://student.unsw.edu.au/exams` for details and possible updates)
- Pre-exam consultations:
  - 28 Nov (Thu): 1-3pm, K17-508
  - 29 Nov (Fri): 1-3pm, K17-508
- Sample exam papers to be released soon.
- Course survey or private messages to me on the forum.

(1) The final exam mark is important and you must achieve at least 40! (2) Supplementary exam is only for those who cannot attend final exam.

About the Final Exam

- **Time**: 10 minutes reading time + 2 hr closed-book exam.
- **Accessories**: UNSW Approved Calculator. Note: watches are prohibited.
- Designed to test your *understanding* and familiarity of the core contents of the course.
- 100 (8 questions)
  - Q1: short answer questions
  - Q2–Q8:
    - choose any 5 to answer.
    - others will require some "calculation" or more steps.

About the Final Exam . . .

- Read the instructions carefully.
- You can answer the questions in *any* order.
- Some of the "Advanced" Methods/algorithms/systems are not required, unless explicitly mentioned here.

Tip: *Write down intermediate steps, so that we can give you partial marks even if the final answer is wrong.*

Disclaimer: *We will go through the main contents of each lecture. However, note that it is by no means exhaustive.*

Boolean Model

- incidence vector
- semantics of the query model (`AND`/`OR`/`NOT`, and other operators, e.g., `/k`, `/S`)
- inverted index, positional inverted index
- query processing methods for basic and advanced boolean queries (including phrase query, queries with `/S` operator, etc.)
- query optimization methods (list merge order, skip pointers)
- **Not required**: next-word index

Preprocessing

- typical preprocessing steps: tokenization, stopword removal, stemming/lemmatization,

Index Construction

- Why we need dedicated algorithms to build the index?
- BSBI: Blocked sort-based indexing
- SPIMI: Single-pass in-memory indexing
- Dynamic indexing: Immediate merge, no merge, logarithmic merge

Vector Space Model

- What is/why ranked retrieval?
- raw and normalized tf, idf
- cosine similarity
- tf-idf variants (using SMART notation): e.g., lnc.ltc
- basic query processing method: document-at-a-time vs term-at-a-time
- exact & approximate query optimization methods (heap-based top-k algorithm, MaxScore algorithm, etc.)

Evaluation

- Existing method to prepare for the benchmark dataset, queries, and ground truth
- For unranked results: Precision, recall, F-measure
- For ranked results: precision-recall graph, 11-point interpolated precision, MAP, etc.
- **Not required**: NDCG, Kappa (κ) measure for inter-judge (dis)agreement

Probabilistic Model and Language Model

- Probability ranking principle (intuitively, how to rank documents and when to stop)
- derivation of the ranking formula of the probabilistic model
- the BM25 method
- Query-likelihood *unigram* language model with *Jelinek-Mercer smoothing*.

Learning to Rank

- Motivation
- Setup, jargons, and basic ideas of Machine Learning
- List-wise L2R
- **Not required**: The details of the SVM L2R model and other advanced variations.

Link Analysis

- The pagerank algorithm
- **Not required**: Personalized PR

Language Models

- Definition, usage, and evaluation (perplexity)
- *n*-gram LM
  - Parameter learning, including various smoothing
- **Not required**: Neural LM

Vector Semantics

- Motivation, taxonomy, and concepts
- Sparse vectors: PPMI weighting and its variants
- High-level understanding of word2vec skip-gram model
- **Not required**: Maths details of Word2vec;