Examination in IN3120/IN4120

IN3120/IN4120
Final exam
2020-12-18, 15:00-19:00

This is an open-book exam, all aids (textbook, online resources, notes, and so on) are allowed. Please supply references to the resources you have used, if any, and please try to formulate precise and complete answers in your own words rather than copy text from an external resource.

You can submit your answers in English or Norwegian. Please use the language you are the most comfortable with.

It is strictly forbidden to collaborate or communicate with others about the exam during the exam. You may be randomly selected for a conversation to check ownership of your answer [link]. This conversation does not affect the grading or grade, but may result in IFI opting to pursue a case for cheating. You can read more about what is considered cheating on UiO's website [link]. Moreover, information on the website about exams at MN in the autumn 2020 applies [link]. For technical support during the exam, see [link].

The different questions have different weights, as indicated. If you need to ask clarification questions regarding the questions on the exam, you can attend a brief Zoom meeting ("digital trøsterunde") that begins exactly one hour after the start of the exam [link].

1 LINK ANALYSIS [20%]

(a) [9%] Consider four web pages that link to each other as follows: {1: [2, 3, 4], 2: [3, 4], 3: [1], 4: [1, 3]} and assume a teleportation probability of 0.5 in the random surfer model. Write down the transition probability matrix for this web graph.

(b) [5%] Given the transition probability matrix, describe two different ways to compute the PageRank value for each web page.

(c) [6%] Assume a teleportation probability of exactly 0.0. Explain precisely why this might cause a problem when computing PageRank values on an arbitrary web graph.

Fill in your answer here

Format | B I U x_2 x^2 I_x | Ω | Σ | ABC

Words: 0

Maximum marks: 20

2 EDIT DISTANCE [20%]

(a) [7%] Assume Levenshtein-Damerau distance with unit edit costs for inserts, deletes, substitutions and transpositions. The following edit table contains an error. Can you quickly spot it, and clearly explain why this must be an error?

		g	o	b	b	e	l	h	o	b	e	l
	0	1	2	3	4	5	6	7	8	9	10	11
g	1	0	1	2	3	4	5	6	7	8	9	10
o	2	1	0	1	2	3	4	5	6	7	8	9
b	3	2	1	0	1	2	3	4	5	6	7	8
b	4	3	2	1	0	1	2	3	4	5	6	7
l	5	4	3	2	1	1	1	2	3	4	5	6
e	6	5	4	3	2	1	1	2	3	4	4	5
h	7	6	5	4	3	2	2	1	2	3	4	5
o	8	7	6	5	4	3	3	2	1	2	3	4
b	9	8	7	6	5	5	4	3	2	1	2	3
b	10	9	8	7	6	5	5	4	3	2	2	3
l	11	10	9	8	7	6	5	5	4	3	3	2
e	12	11	10	9	8	7	6	6	5	4	3	3

(b) [13%] Assume that you have a dictionary with one million string entries. Given a query string, you need to efficiently locate (within milliseconds) which strings among the one million strings in the dictionary that have an edit distance from the query string that is less than or equal to k. You can assume that k is small, say, less than 4. Describe your approach and how it works.

Fill in your answer here

Format

Words: 0

Maximum marks: 20

3 QUIZ-O-RAMA [20%]

(a) [2%] If you wanted to search for *d*ld* in a permuterm wildcard index, what key or keys would you do the lookup on?

(b) [2%] Give an example of a sentence that falsely matches the wildcard query *mon*h* if the search were to simply use a conjunction of bigrams.

(c) [2%] Consider the postings list [4, 10, 11, 12, 15, 62, 63, 265, 268, 270, 400] with a corresponding list of gaps [4, 6, 1, 1, 3, 47, 1, 202, 3, 2, 130]. Assume that the length of the postings list is stored separately, so the system knows when a postings list is complete. Using variable byte encoding: (i) What is the largest gap you can encode in 1 byte? (ii) What is the largest gap you can encode in 2 bytes? (iii) How many bytes will the above postings list require under this encoding? Count only the space for encoding the sequence of numbers.

(d) [2%] Can the TF-IDF weight of a term in a document exceed 1? Justify your answer.

(e) [2%] Consider the case of a query term that is not in the set of our indexed terms. A standard construction of the query vector would then have a dimension not found in the vector space created from the collection. How would one adapt the vector space representation to handle this case? Explain why.

(f) [2%] John has an inverted index with impact-ordered posting lists, and is looking to implement document-at-a-time scoring. Explain which problem John will quickly face if he goes down this route.

(g) [2%] True or false: At the break-even point, the value of the F_1-score must be higher than the precision value. Justify your answer.

(h) [2%] You want to implement a search feature where you indicate a reference page and then the system retrieves other pages that are similar to the selected reference page. Using the Rocchio relevance feedback algorithm for implementing this feature, what are your weight settings for alpha/beta/gamma?

(i) [2%] True or false: For a two-class classification problem and a linear SVM, there must always be an even number of support vectors for at least one of the classes. Justify your answer.

(j) [2%] True or false: If your query vector is normalized to unit length and all your document vectors are normalized to unit length, then you get the same rank ordering for documents using cosine similarity as you get using Euclidean distance. Justify your answer.

Fill in your answer here

Format

Words: 0

Maximum marks: 20

4 LARGE-SCALE SYSTEMS [30%]

(a) [10%] You are coding part of a distributed search engine, and have partitioned your content into 50 disjoint partitions. All the incoming queries your system sees are really user identifiers. There could be a lot of unique queries, but there is a finite set of possible queries.

Your first idea is to dispatch an incoming query to all 50 partitions in parallel, and then merge the results coming back from each partition. However, only some of the partitions might return a non-empty result set for a query, and it is costly and a wasted effort to dispatch a query to a partition that will produce an empty result set since this incurs a network call and several disk accesses on the node that hosts the partition. Ideally you only want to dispatch the query to those partitions you think might be able to produce non-empty result sets. You are therefore looking to introduce a quick pre-check before dispatching the query to a partition, to see if you can avoid dispatching the query to that partition.

Describe a suitable data structure to use for the pre-check and explain how the data structure works, and describe how you will be using it to solve the problem at hand.

(b) [10%] You are designing a distributed news search engine where in total more than a hundred million news stories will be searchable, where new news stories need to be added to the index more or less all the time as they become published, and where it's imperative that new news stories become searchable almost immediately after they are published.

Describe an index construction strategy that addresses your requirements, and outline some of the complications that your proposed strategy introduces.

(c) [10%] You are working on the design of a new search system and your coworker has left you a note describing three algorithms *algo1*/*algo2*/*algo3* that he cannot get to work well, described in Python-like pseudocode.

For each algorithm, (i) describe where it fits into a typical search system and what it is supposed to do, (ii) explain why it does not work as expected and how it can be changed so that it will work better, and (iii) explain what would have happened if it was deployed in the search system as specified below without being corrected.

```
algo1(cs, v, pp, cp, d):
  ts = foo(v, d)
  s = {}
  for c in cs:
    s[c] = log(pp[c])
    for t in ts:
      s[c] = s[c] * (log(cp[t][c]) + 1)
  return max(s)


def algo2(cs, es, q):
  ms = []
  for i,c in enumerate(cs):
    ds = [d for d in cs[c]]
    ms.append((sum(normalize(d) for d in ds) / len(ds)))
 xs = [abs(m - normalize(q)) for m in ms]
 return xs.index(max(xs))


def algo3(s, m, n):
    for x in s:
      for d,f in postingslist[x]:
        r[d] = weight1(x, s) * weight2(x, d) + r[d]
for d in m:
    r[d] = r[d] / len(d)
return r[0:n]
```

Fill in your answer here

Format

Words: 0

Maximum marks: 30

5 EVALUATION [10%]

(a) [6%] Explain how the NDCG and MAP evaluation metrics work, and discuss some of their relative merits.

(b) [4%] Gold standard relevance judgments (i.e., knowing which documents that are relevant for a given query) can be hard to come by and are sometimes produced manually by human judges. Briefly discuss some of the problems with this, and how you can address or quantify these issues.

Fill in your answer here

Format

Words: 0

Maximum marks: 10