Chair for Algorithms
and Data Structures
Prof. Dr. Hannah Bast
Elmar Haußmann

Information Retrieval
WS 2015/2016

http://ad-wiki.informatik.uni-freiburg.de/teaching

UNI FREIBURG

Exam

Tuesday, February 23, 2016, 14:00 - 16:30, Hörsaal 026+036 in Building 101

General instructions:

There are six tasks, of which you can select *five tasks of your choice*. Each task is worth 20 points. If you do all six tasks, we will only count the best five, that is, you can reach a maximum number of 100 points.

You need 50 points to pass the exam. You have 150 minutes of time overall. If you do five tasks, that is 30 minutes per task on average.

You are allowed to use any amount of paper, books, etc. You are not allowed to use any computing devices or mobile phones, in particular nothing with which you can communicate with others or connect to the Internet.

You may write down your solutions in either English or German.

Please write your solutions on this hand-out! You can also use the back side of the pages. Please write your name and Matrikelnummer on the cover sheet (top right corner). If you need additional pages, please write your name and Matrikelnummer on each of them, too.

Important:

For the programming tasks, you can use Python, Java, or C++. None of your functions must be longer than TWELVE lines.

For all other tasks: do not simply write down the final result; it should also be clear how you derived it.

Good luck!

Task 1 (Ranking and evaluation, 20 points)

Consider the following 3 x 5 term-document matrix A:

	D_1	D_2	D_3	D_4	D_5
bla	1	1	2	2	3
bli	1	0	1	2	2
blu	0	1	1	0	1

1.1 (5 points) Let q be the query consisting of the words "bla blu" and assume that only documents D_1 and D_5 are relevant. Order the five documents according to their dot-product similarity with q. Then write down a bit array, where the i-th element says whether the document ranked i-th is relevant. Compute the average precision (AP).

1.2 (5 points) Write a function *compute_ap* that takes a bit array like in Task 1.1 as argument and computes the average precision. You can assume that not all bits are zero.

1.3 (5 points) Prove that the AP is 100% if and only if all the relevant documents are ranked before all the non-relevant documents.

1.4 (5 points) Write A as a product of a 3 x 2 matrix with a 2 x 5 matrix. What exactly does this prove about the rank of A.

Task 2 (Encodings, 20 points)

For this exercise, consider the following simple (unary) encoding scheme:

$c(1) = 1, c(2) = 01, c(3) = 001, c(4) = 0001$, ...

2.1 (5 points) Decode the bit sequence 110000100111001, that is, write down the sequence of integers encoded by this bit sequence. Prove (with a formal argument) that the encoding scheme is prefix-free.

2.2 (5 points) Write a function *decode* that takes a bit array as argument, and returns the encoded sequence (as an array of integers). If the bit sequence contains an invalid code, return the empty array.

2.3 (5 points) Assume the following random process for generating one of the above codes: pick a random bit, where 0 and 1 each occur with probability 1/2; repeat until a 1 is encountered. Let X be the integer encoded by such a code. What is the probability $\Pr(X = x)$ that integer x is encoded? Do not just write down the probability; it should also be clear how you derived it.

2.4 (5 points) Prove that the unary encoding from above is entropy-optimal for the probability distribution from Task 2.3.

Task 3 (Web applications and UTF-8, 20 points)

Consider the following piece of JavaScript:

```
$(document).ready(function() {
  $("#query").keyup(function() {
    var query = $("#query").val();
    $.get("http://www.gugel.de:8888/?q="+ query, function(result) {
      $("#result").html(result);
    })
  })
})
```

3.1 (5 points) Write a piece of HTML (minimal is OK) that makes sense for this piece of JavaScript. You can omit the `<head>...</head>` part.

3.2 (5 points) Assume that a user slowly types `hey` letter by letter into the field with id `query` from the HTML. Specify the first fifteen characters of each of the GET requests sent and to which machine they are sent. Don't bother with the result (neither sending it nor computing it) for this task.

3.3 (5 points) What is the exact number of valid 2-byte sequences in UTF-8? Don't just write down the result but explain how you derived it.

3.4 (5 points) The ISO-8859-1 code of the German umlaut ü is 252. What is the UTF-8 code (as two integers, in decimal)? What is the URL-escaped code? Do not just write down the result; it should also be clear how you derived it.

Task 4 (Perceptrons, 20 points)

Consider a training set of eight integer objects $\{1, 2, 3, 4, 5, 6, 7, 8\}$, where $\{1, 2, 3, 4\}$ are labeled negative and $\{5, 6, 7, 8\}$ are labeled positive.

4.1 (5 points) Execute the first round of the Perceptron algorithm, starting with $w = 0$ and $b = 0$. Go through the objects in their natural (ascending) order.

Reminder: If $w \cdot x - b = 0$, then w and b always need to be updated (and the update should be in the direction such that the prediction is right afterwards).

4.2 (5 points) Implement a function *perceptron* that computes the final values of w and b for the training set above after fifty rounds. In each round, you can go through the training elements in their natural (ascending) order.

4.3 (10 points) Let w^*, b^* be the final values of w, b on the above training set after fifty rounds (like in the function from Task 4.2). Let $\bar{w}^*, \bar{b}^*$ be the final values using the same function but when the training labels are swapped (that is, $\{1, 2, 3, 4\}$ are labeled positive and $\{5, 6, 7, 8\}$ are labeled negative). Prove that $\bar{w}^* = -w^*$ and $\bar{b}^* = -b^*$.

Hint: Introduce w_i, b_i for the values of w, b after the i-th iteration of the function, and $\bar{w}_i, \bar{b}_i$ for the corresponding values when the labels in the training set have been swapped.

Task 5 (Linear algebra, 20 points)

Consider the following collection of four documents, with "words" from the vocabulary $\{a, b, c\}$:

D_1: a , D_2: $a\ a\ b\ c\ c$, D_3: $b\ b$, D_4: c

5.1 (5 points) Compute the *idf* scores for a, b, c (*hint:* they are all integer). Then write down the 3×4 term-document matrix A for the collection above, using standard *tf.idf* scores.

5.2 (5 points) Consider two other documents C_1: $a\ b\ c$ and C_2: $b\ b$ and translate them to vectors with *tf.idf* scores (using the *idf* scores from Task 5.1).

Let S be the 2×4 matrix, where the entry at i, j is the dot-product similarity between C_i and D_j. Show how S can be computed using a matrix multiplication (that involves the term-document matrix A from Task 5.1). Compute S in this way and from the result determine which of the four documents is more similar to C_1 and which more similar to C_2.

5.3 (5 points) Provide a matrix M such that the product of $A \cdot M$ results in a 3×2 matrix, where the first column is the average of the documents closer to C_1 and the second column is the average of the documents closer to C_2.

5.4 (5 points) Write a function *matrix_mult* that multiplies two matrices, each given as a two-dimensional array, using basic arithmetic operations (that is, without using a linear-algebra library). Return the result as another two-dimensional array.

You can assume that the dimensions of the two matrices are compatible. You can assume the existence of functions *num_rows* and *num_cols* that provide the number of rows and columns, respectively, of a given matrix. You can assume the existence of a function *zeroes* that returns an all-zero matrix with given dimensions.

Task 6 (Miscellaneous, 20 points)

6.1 (5 points) We proved that if $\mathrm{ED}(x, y) \leq \delta$, then their padded versions $x' = \$\$x\$\$$ and $y' = \$\$y\$\$$ have at least a certain number of 3-grams in common. What is the purpose of the padding?

6.2 (5 points) Let S be the sum of the number of the rolls of two independent fair dice. What is the probability of the event E that S is an even number? What is the conditional probability $\Pr(S = 12|E)$?

6.3 (5 points) Let X be a random variable which can take two values x_1 and x_2. Prove for which probability distribution of X the entropy $H(X)$ is maximized? *Hint:* $\log_2 p = 1/\ln 2 \cdot \ln p$ and $(p \cdot \ln p)' = 1 + \ln p$.

6.4 (5 points) Assume we have a knowledge base which includes relations for *nationality* (relating persons to countries), *contains* (between geographical regions, where one contains the other), *film* (relating persons to films they acted in), *director* (relating persons to films they directed). Express the query *actors or actresses from films with a european director* in SPARQL.