UNIVERSITY
OF SKÖVDE

School of Humanities and Informatics

WRITTEN EXAMINATION

Course Big Data Programmering

Sub-course

Course code IT 739A

Credits for written examination 5,5

Date 2024-03-20

Examination time 08:15-12:00

Examination responsible Richard Senington

Teachers concerned Gunnar Mathiason, Richard Senington

Aid at the exam/appendices No

Other

Instructions

- ☐ Take a new sheet of paper for each teacher.
- ☒ Take a new sheet of paper when starting a new question.
- ☒ Write only on one side of the paper.
- ☒ Write your name and personal ID No. on all pages you hand in.
- ☒ Use page numbering.
- ☒ Don´t use a red pen.
- ☒ Mark answered questions with a cross on the cover sheet.

Grade points A-F

Examination results should be made public within 18 working days

Good luck!

Total number of pages 3

IT739A, Exam, Mar 20th, 2024, at 8:15-12:30

Instructions that must be followed:

- The max number of points listed with each question indicates how thoroughly you are expected to answer the question.
- Number your answer clearly with the same numbering as the question and its sub questions.
- Be clear in our writing. Sometimes fewer but more concise formulations are better than writing a lot of text.

The grading of your answers considers:

- The **correctness** of your answer/explanation
- The **clarity and logical cohesion** of your answer/explanation
- The **conciseness** of your answer/explanation

Grading levels:

- There are 5 questions (with sub questions), which give maximum 10 points each. 1) You need to pass 5 points per each of these 5 questions to pass the exam (giving 25 points). 2) Accumulated points above that threshold linearly determines the exam grading (exam maximum is 50 points).
- Exam grading range is A to F, which also determines the final course grading: A: 45-50 points, B: 40-44 points, C:35-39 points, D: 30-34 points, E: 25-29 points

Examination goals:

This exam assesses the following Course Curriculum goals

- *critically reflect on the needs for programming tools for Big Data,*
- *prove understanding of the development of tools for Big Data programming,*
- *show an in-depth understanding of paradigms that are frequently used for Big Data programming,*

For information: The course Assignment assesses the following Course Curriculum goals

- *demonstrate abilities to make use of chosen programming tools for Big Data programming and data analysis and*
- *execute a small data analysis project using Big Data programming.*

Exam questions:

1. Keras and Deep Learning (10 points):

1. Deep Learning: What is the role of an *optimizer* when propagating updates to weights in a neural network? (1p)

2. Convolutional networks:
    a. Motivate why convolutional networks typically use fewer trainable weights compare to fully connected network.
    Also, under which assumptions about your data can you reach the benefit of using convolutional layers
    (tip: this has to do with re-occurring patterns)? (3p)
    b. Explain and differentiate between the terms *feature map* and *response map* that are used to describe the data transformation in a convolutional operation. (2p)
3. Explain why it is critical to choose a suitable loss function when selecting hyperparameters for a neural network design, and how does that choice influence the learning capabilities of that network? (4p)

2. Tensorflow and application (10 points):

1. Explain the concept of a *tensor data structure*. Also, what is the difference between *rank* and *dimensionality* for data that is formulated as a tensor? (3p)
2. With the Tensorflow *Data APIs* you can build *data pipelines*:
    a. What is a Tensorflow data pipeline and what is the benefit of using one when having large data sets to (re-)process? (2p)
    b. How can you improve GPU utilization when using data pipelines and *pre-fetching*? (2p).
3. Describe two separate advantages of structuring your code logic as a computation graph? How can you inspect and debug such graphs in Tensorflow? (3p)

3. Programming and deployment for Big Data (10 points):

1. When evaluating the performance of a trained model, you need to reserve some hold-out data to use as a *test set*? Why do you need to do that? Also, explain what is meant with *leakage* of information when you try to improve a model's accuracy by tweaking the data or the data preprocessing. (2p)
2. Data can come in many "shapes". One common data "shape" is tabular data, such as with a Python or R "data frame". A tabular data set often include multiple such tables together. There are also other shapes, which are not based on tables. Describe three different types of data structure that can be encountered in Data Science projects? You can choose to explain tabular data as one of these three structures. (3p)
3. Describe a way to *detect overfitting* using plotting of the *accuracy* and *loss* parameters during neural network training. (3p)

4. There are several different purposes of big data analysis. One is to *predict* some output from a many given examples of data set of inputs resulting in some output. Briefly explain two other different purposes of data analysis. (2p)

4. Architecture, databases and the issues of big data (10 points):

1. Variety is one of the 5Vs of Big Data. Discuss how different database architectures can account for this required property of big data. (4p)
2. Describe the 2 primary types of virtualisation and explain how virtualisation can help to manage big data applications. (3p)
3. Discuss how distributed databases can support computation clusters in performing their actions when retrieving data. (3p)

5. Functional programming and Spark (10 points):

1. Data analytics can be performed in a variety of ways. One approach is to use database queries directly to perform analysis of the data stored in a database. Another commonly used approach is to make use of Spark or the earlier Hadoop framework. Compare and contrast each of these technologies including scalability of storage and processing, sophistication of analytics that can be performed and illustrate your answer with example applications that are suited to each. (4p)
2. Describe and discuss the benefit of using pure functional programming for the description of computations that are to be executed on a cluster. (2p)
3. In this question please write a spark/scala program or function to perform the following task.
    - You have several datafiles stored in a document database, and you have a list of the keys/names of each file. Each file is a csv type file with 5 columns; name, age, bank balance, number of children/dependents and employer.
    - Your function must; load the files, filter each for age/40, and compute both the average bank balance and the average number of dependents. Note that you might have to merge several datasets to achieve this.

    Due to being a closed book exam, while accuracy in the code is desirable we will be more concerned with correct use of functional structures and intention than with strict code syntax.(4p)