Univ.-Prof. Dr.-Ing. Matthias Boehm
Technische Universität Berlin
Faculty IV - Electrical Engineering and Computer Science
Berlin Institute for the Foundations of Learning and Data (BIFOLD)
Big Data Engineering (DAMS Lab) Group

October 12, 2024

Exam Architecture of Machine Learning Systems (SoSe 2024)

Important notes: The working time is **90min**, and lecture materials or any kind of mobile devices are not allowed. Please, make sure to put your name and matriculation number on the top right of the first page of the task description, and each additional piece of your own paper. You may give the answers in English or German, written directly into the task description.

Task 1 Parameter Servers (16 points)

(a) Describe the overall system architecture of *data-parallel parameter servers*, explain its components and interaction among these components. (**10 points**)

(b) Describe synchronous (BSP) and asynchronous (ASP) *update strategies* in data-parallel parameter servers and name their advantages and disadvantages. (**6 points**)

	Synchronous Updates	**Asynchronous Updates**
Description		
Advantages		
Disadvantages		

Task 2 Data Preparation (22 points)

(a) Given the input data below, apply *recoding and one-hot encoding* to the categorical columns A and C, and *binning* with three equi-width bins to the numerical column B. (**10 points**)

A	**B**	**C**
Low	0	S
High	3.1	M
Med	7	L
Low	9	XL
Low	15	M
Low	7	M
Med	4	L
High	12	XL
High	13	L

(b) Explain *feature hashing* and what is its advantage over recoding? (**3 points**)

(c) Describe the text encodings bag-of-word and word-embeddings. (**6 points**)

- **Bag-of-Words:**

- **Word Embeddings:**

(d) What is *data augmentation* and name two concrete techniques. (**3 points**)

Task 3 Model Selection (13 points)

(a) Describe the task of *hyper-parameter tuning by example of Grid Search*. Furthermore, assume seven hyper-parameters with 10 discretized values each, how many models do we need to train? (**8 points**)

- **Hyper-parameter Tuning:**

- **Grid Search:**

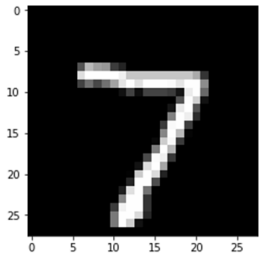
- **Example #Models:**

(b) Explain *Bayesian Optimization* as a more directed search strategy, and how it balances exploitation and exploration? (**5 points**)

Task 4 Model Debugging (8 points)

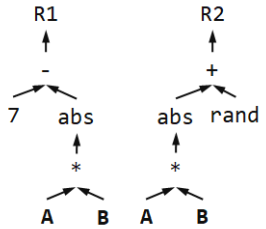
(a) Explain the concept of a *confusion matrix* and describe it in detail. (**4 points**)

(b) Explain the concept of *occlusion-based explanations* by example of classifying below handwritten digit as a seven. (**4 points**)

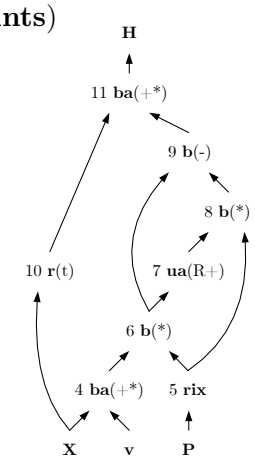


Task 5 Compilation Techniques (19 points)

(a) Describe the purpose of the rewrite *common subexpression elimination (CSE)* and sketch an algorithm to perform CSE on a directed acyclic graph (DAG) of operators. (**5 points**)



(b) Given the directed acyclic graph (DAG) below, perform *shape inference* and determine the dimensions (number of rows and columns) of the intermediates produced by operations (4) through (11). The input matrices have the following dimensions: $\mathbf{X}$ (50,000 × 700), $\mathbf{v}$ (700 × 3), and $\mathbf{P}$ (50,000 × 4). (**4 points**)



ba (+*) .. **binary aggregate** (matrix multiply)
rix .. **right indexing**
r(t) .. **transpose**
b(*) .. **binary elementwise multiplication**
b(-) .. **binary elementwise subtraction**
ua(R+) .. **unary aggregation, row-wise summation**

Original Expression:

```
Q = P[, 1:3] * (X %*% v);
H = t(X) %*%
    (Q - P[, 1:3] * rowSums(Q));
```

(c) Explain the concept of *operator fusion* (or loop fusion) and how it can improve runtime performance. (**3 points**)

(d) Assume an example chain of matrix multiplications ($\mathbf{A}\,\mathbf{B}\,\mathbf{C}\,\mathbf{D}\,\mathbf{E}$), describe the problem of *matrix multiplication chain optimization*, and a dynamic programming algorithm for solving it efficiently. (**7 points**)

Task 6 Data Access Optimizations (13 points)

(a) Assume an n-by-m matrix $\mathbf{X}$ with sparsity $\frac{\text{nnz}(\mathbf{X})}{n \cdot m}$ (fraction of number of non-zeros to cells). In the table below, indicate via a ✓ which matrix block representation is the *most space-efficient* one for each of the five different shape/sparsity scenarios (assuming 4 Byte integer and floating point data types for indexes and values). (**5 points**)

Shape, Sparsity	**Dense**	**Compressed Sparse Rows (CSR)**	**Coordinate (COO)**
1,000 × 1,000, 0.7			
1,000 × 1,000, 0.5			
1,000 × 1,000, 0.1			
20,000 × 50, 0.01			
200 × 5,000, 0.001			

(b) Describe min-max quantization of an `FP64` (64bit floating point) representation into `UINT8` (8bit integer). Why does such an encoding increasing training and/or inference performance? (**8 points**)

Task 7 Model Deployment (9 points)

(a) Consider a deployed model M in a cloud serving environment and assume 1000s of concurrent client requests. Explain three strategies for improving model scoring throughput at the serving site. (**9 points**)