CSE 234 Fall 2021 Final Exam
Answers

Q1. [20 x 2pts] For each of the following questions, select the correct option.

1. Which of these technical debts in ML workflows is caused primarily by the use of disparate libraries and programming languages within the same workflow?
   (A) Concept drift  (B) Entanglement of features
   (C) Feedback loops  (D) Glue code

   ANSWER: (D)

2. Which of the following data-parallel execution schemes for SGD uses asynchrony?
   (A) Horovod  (B) Model Hopper Parallelism
   (C) MapReduce/Spark  (D) UDAF Model Averaging

   ANSWER: (B)

3. Deequ helps reduce technical debt in data sourcing by automating much data validation in a standardized manner. But the tradeoff is that it still introduces a new technical debt–which one?
   (A) Pipeline jungles  (B) Feedback loops
   (C) Abstraction debt  (D) Scale-related errors

   ANSWER: (C)

4. Consider the following 4 data processing programs shown on the Roofline plot. Which one will benefit the most from optimizations that reduce memory stalls?
5. Which of the following ML methodologies helps reduce the amount of labeled data needed when training DL models?

(A) Transfer learning  (B) Automated ML
(C) Early stopping  (D) None of the rest

ANSWER: (A)

6. Which function in the UDA abstraction operates on a data tuple at a time?

(A) Initialize  (B) Transition  (C) Merge  (D) Finalize

ANSWER: (B)

7. Which of these data regulations gives end users the right to erasure of the data collected from them?

(A) FERPA  (B) HIPAA  (C) GDPR  (D) None of the rest

ANSWER: (C)

8. What is the name of the new programming paradigm ushered in by DL tools?

(A) Differentiable programming  (B) Integrable programming
(C) Learned programming  (D) Dataflow programming

ANSWER: (A)
9. Which consistency model in Parameter Server introduces a new knob to tune the consistency-concurrency tradeoff?
   (A) Sequential    (B) Eventual    (C) Bounded Delay    (D) None of the rest

   ANSWER: (C)

10. Which form of DL is now the most popular form of ML on tabular data?
    (A) Transformers    (B) CNNs    (C) RNNs    (D) None of them

    ANSWER: (D)

11. Consider the following chart showing Pareto tradeoffs of 4 different ML models on two metrics of interest. Suppose you are told that model D is Pareto-optimal. For which of the following metrics on the X axis will that make sense?

   (A) Training cost (dollars)    (B) Training throughput (examples per second)
   (C) Space footprint (bytes)    (D) Inference latency (seconds)

   ANSWER: (B)

12. Which of these is a key difference between a data warehouse and a feature store?
    (A) Modalities of data managed    (B) Supports offline querying
    (C) Supports online querying    (D) Used in data science applications

   ANSWER: (A)
13. Which stage of a typical entity matching workflow helps mitigate the impact of quadratic time complexity?
   (A) Pairwise check   (B) Blocking   (C) Clustering   (D) None of the rest

   **ANSWER:** (B)

14. In which form of DL do dynamic computational graphs arise commonly?
   (A) Transformers   (B) CNNs   (C) RNNs   (D) None of them

   **ANSWER:** (C) or (D). The question had some ambiguity that I missed. Dynamic graphs are not necessarily “common” in RNNs; many RNNs do use padding to convert inputs to fixed lengths. But it is far more common in RNNs than in CNNs or Transformers.

15. At what granularity does Horovod invoke Ring-Allreduce communication steps?
   (A) Once per training session   (B) Once per SGD epoch
   (C) Once per data mini-batch   (D) It is a flexible configuration

   **ANSWER:** (C)

16. Which of the following is a common effect of a key-key join as a feature engineering step to gather more features for ML on tabular data?
   (A) Raises bias   (B) Reduces variance
   (C) Reduces Bayes noise   (D) None of the rest

   **ANSWER:** (C)

17. Which of the following components of MLFlow is analogous to TFX’s ML Metadata to help ML users log their experimentation lineage and metadata?
   (A) Tracking   (B) Projects   (C) Models   (D) Model Registry

   **ANSWER:** (A)
18. In what way does factorized ML primarily help improve ML on tabular data?

(A) Raises accuracy  
(B) Reduces runtime  
(C) Enables multi-node scalability  
(D) Automates ML model tuning

**ANSWER: (B)**

19. In what programming language are most of TensorFlow’s kernels for tensor arithmetic implemented written under the hood?

(A) Java  
(B) Rust  
(C) Python  
(D) C++

**ANSWER: (D)**

20. Which of the following is a common systems-level optimization employed in feature engineering systems?

(A) Materialized views and caching  
(B) Hybrid task+data parallelism  
(C) Learned scheduling algorithms  
(D) Reduced precision with quantization

**ANSWER: (A)**

---

**Q2. [10pts] Scalable systems for deep learning training.**

1. (4pts) Briefly explain why Parameter Server is a poor fit for data-parallel DL training on a multi-node cluster.

2. (6pts) Briefly explain 2 techniques in PyTorch DDP that enable it to scale data-parallel DL training to a multi-node cluster much better than Parameter Server.

**Q3. [15pts] Programmatic data labeling.**

1. (6pts) Briefly explain 1 pro and 1 con of programmatic labeling vs manual labeling.

2. (9pts) Briefly describe a concrete ML application where programmatic labeling is likely to help reduce manual labeling effort and explain why with a concrete example of a feasible programmatic labeling heuristic. Divide your answer into the following prompts.

**Q4. [15pts] Scalable systems for GBDT training.**
1. (6pts) Succinctly explain why MapReduce/Spark is a poor fit for data-parallel GBDT training on a multi-node cluster.

2. (3 x 3pts) Succinctly explain exactly how each of the following techniques in XGBoost help GBDT training in terms of the Pareto tradeoff axes discussed in class, viz., accuracy, runtime, data scalability, usability, manageability, etc.

   (a) Sharding of data into column blocks.
   (b) Weighted quantile sketch.
   (c) CPU cache-aware staging of gradient statistics in DRAM.

Q5. [15pts] MapReduce for Ordinary Least Squares (OLS) Linear Regression.

You are given a large tabular dataset \((Y, X)\) on HDFS in which all variables are numeric. The feature vector \(X\) has only dozens of features. Let \(X\) and \(Y\) represent a matrix view of the features and target, respectively. You want to fit an OLS model \(W\) as follows.

\[
W = (X^T X)^{-1} X^T Y
\]

Assume you are provided a matrix inversion function in a library that works fine for small in-memory matrices. Write succinct pseudocode (or explain precisely) for computing \(W\) using a single MapReduce job that is scalable along the number of examples/rows. Write only your final answer in the sections on this page and next.

(Hint: The Gramian \(X^T X\) will be small; its inverse can be computed on one machine.)

**ANSWER:** Row-wide sharding; makes it scalable to any number of examples/rows.

**Basic idea of Map and Reduce functions:**
Compute 2 separate matrices in a data-parallel manner: \(X^T X\) and \(X^T Y\). Both can be neatly decomposed as sums over the examples. Each Mapper computes the partial sums over its examples for each of those 2 matrices. A single global Reducer adds the respective partial matrices up from all Mappers to obtain the global 2 matrices. It then inverts the Gramian and multiplies that with the second matrix to obtain \(W\).


1. (6pts) Succinctly explain 2 reasons why it is important in practice to have compilers like TVM that speed up DL inference.

2. (3 x 3pts) Succinctly explain the specific impact of each of the following techniques in TVM on memory stalls between the processor and DRAM.

Suppose you need to tune 2 hyperparameters of a DL model (say, learning rate and regularizer). You decide to compare Hyperband and plain old Grid Search using task parallelism on a cluster with 8 worker nodes.

You run Hyperband with the following knobs: $R = 16; \eta = 4$. It yields the following brackets in the same table format from the paper as explained in class:

<table>
<thead>
<tr>
<th>i</th>
<th>s = 2</th>
<th>s = 1</th>
<th>s = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$n_i$</td>
<td>$r_i$</td>
<td>$n_i$</td>
</tr>
<tr>
<td>0</td>
<td>16</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>16</td>
<td></td>
</tr>
</tbody>
</table>

1. (2pts) What is the total number of model configurations trained by Hyperband?
   ANSWER: Same 25 as above!

2. (2pts) Suppose all models have the same runtime per epoch of 10min. What is the runtime of the longest running model in the Hyperband workload?
   ANSWER: 16 * 10min = 160min.

3. (5pts) What is the aggregate resource usage time of this Hyperband workload on the cluster? (Hint: Add up the runtimes of all models across all workers.)
   ANSWER: $(16*1 + 4* (4-1) + 1 * (16-4) + 6*4 + 1 * (16-4) + 3*16) * 10min = 1240min.$

4. (2pts) Next, you run Grid Search with 3 values for each hyperparameter. All models run for 16 epochs. What is the aggregate resource usage time of this workload?
   ANSWER: $(3 * 3 * 16) * 10min = 1440min.$

5. (3pts) You find that Grid Search consumes more resources. So, you decide to run it for fewer epochs. What is the largest number of epochs you can afford to run the
above Grid Search specification without exceeding the aggregate time of the earlier Hyperband workload?

**ANSWER:** Floor of 124 / (3 * 3), i.e., 13 epochs.

6. **(6pts)** Given the above workload calculations and cluster size, briefly describe 1 pro and 1 con of Hyperband over Grid Search for task-parallel model selection.

**ANSWER:** Hyperband does not waste resources on configs with poor learning behavior but grid search does. Hyperband’s degree of parallelism reduces over time, underutilizing the cluster; grid search is a one-shot specification that can be scaled to use the whole cluster well. Other correct answers are also possible.

---

**Q8. [20pts] Essay on managing end-to-end ML workflows.**

1. **(3 x 2pts)** Briefly describe a concrete end-to-end ML application workflow at a hypothetical or real Web company that relies on a mix of tabular and text data and requires realtime prediction on a user-facing website. Divide your answer into the following prompts:

   (a) What is specific ML task and prediction target? What is its business rationale?
   (b) What are the features used by the ML model? What are their data sources?
   (c) Why will non-ML rule-based approaches not suffice here?

2. **(7pts)** Succinctly explain 1 concrete way in which the mix of features and data sources you use causes technical debt in your ML workflow. Then explain how using a feature store can help pay off that specific technical debt.

3. **(7pts)** Succinctly explain 1 concrete way in which the requirement of realtime prediction causes technical debt in your ML workflow. Then explain how using an ML platform such as TFX or MLFlow can help pay off that specific technical debt.

---

**Extra Credit Question. [2 x 5pts] Feature type inference and bias-variance tradeoff.**

Consider the task of feature type inference discussed in class as part of the ML Data Prep Zoo. That is, given a tabular data file with only raw attribute types known (strings, numbers, etc.), we need to infer feature types for ML (categorical, numeric, etc.)

A common failure scenario for rule-based feature type inference is when a categorical feature is stored as integers, e.g., ZipCode, product code, etc. An ML model will still be able to consume it by treating it like other numeric features. However, the mis-inference has interesting implications for the bias-variance tradeoff of the ML model built.
1. How exactly does mis-inferring a categorical integer as a numeric feature alter the bias-variance tradeoff of a logistic regression model? Explain succinctly.

**ANSWER:** Bias goes up and variance goes down. When represented as a categorical, it is typically one-hot encoded (or any other encoding scheme), which leads to more features. But using it as a number leads to a single numeric feature.

2. How exactly does mis-inferring a categorical integer as a numeric feature alter the bias-variance tradeoff of a decision tree model (CART)? Explain succinctly.

**ANSWER:** Variance goes up but bias is unaffected. CART decision nodes can split on integers or strings; so, bias is not affected. It can also recover the categorical nature of that feature using arbitrary partitions on the integer domain. But this can lead to very tall trees that can overfit more.