DADS7305: MLOPs

Northeastern University

Instructor: Ramin Mohammadi

February 10, 2026

These materials have been prepared and sourced for the course **MLOPs** at Northeastern University. Every effort has been made to provide proper citations and credit for all referenced works.

If you believe any material has been inadequately cited or requires correction, please contact me at:

`r.mohammadi@northeastern.edu`

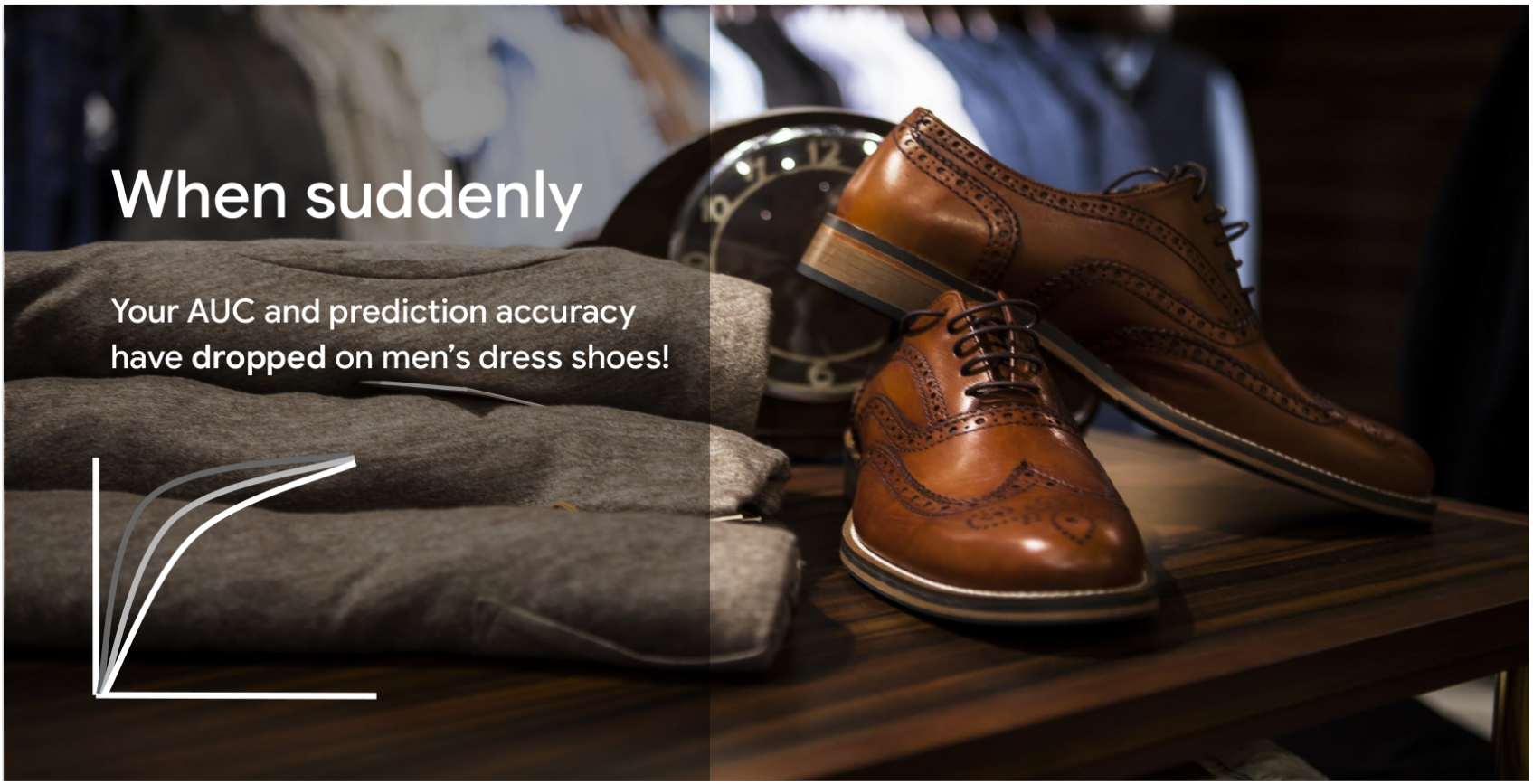
Thank you for your understanding and collaboration.

Data Labeling

Case Study: Degraded Model Performance

You're an Online Retailer Selling Shoes...

- Your model predicts **click-through rates (CTR)**
- Helps you decide how much inventory to order


When suddenly
Your AUC and prediction accuracy
have **dropped** on men's dress shoes!


Why?


How do we know that we have a problem?

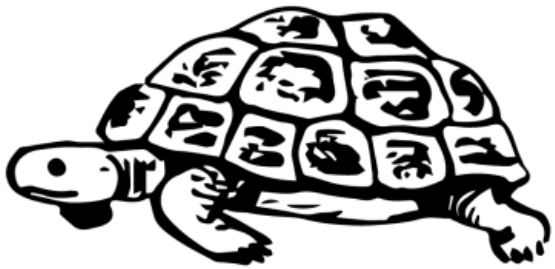
Data Problem

Case Study: Taking Action

- **How to detect problems early on?**
  - Monitor input data distribution
  - Track model performance over time
  - Set up alerting for drift or anomalies
- **What are the possible causes?**
  - Data drift or concept drift
  - Upstream data source changes
  - Labeling inconsistencies or feedback loops
- **What can be done to solve these?**
  - Retrain or fine-tune the model periodically
  - Add new features or filters in preprocessing
  - Improve monitoring and logging

What causes problems?

- Kinds of problems:
  - **Slow** – example: drift
  - **Fast** – example: bad sensor, bad software update



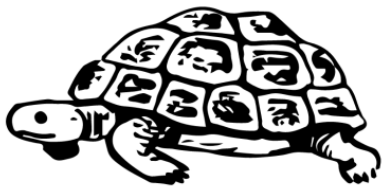
Gradual problems

Data changes

- Trend and seasonality
- Distribution of features changes
- Relative importance of features changes

World changes

- Styles change
- Scope and processes change
- Competitors change
- Business expands to other geos



Sudden problems

Data collection problem	Systems problem
• Bad sensor/camera • Bad log data • Moved or disabled sensors/cameras	• Bad software update • Loss of network connectivity • System down • Bad credentials

Why "Understand" the model?

- Mispredictions do not have uniform cost to your business
- The data you have is rarely the data you wish you had
- Model objective is nearly always a proxy for your business objectives
- Some percentage of your customers may have a bad experience
- The real world does not stand still!

Why "Understand" the model?

- LLMs may hallucinate - generating incorrect or misleading responses
- Prompt sensitivity means small changes can cause unpredictable shifts
- Few-shot examples and context length affect model behavior
- Model outputs can reinforce biases and stereotypes
- Real-world user prompts are diverse and often adversarial

Data Labeling

Data and Concept Change in Production ML

Outline

- Detecting problems with deployed models
  - Data and concept change
- Changing ground truth
  - Easy problems
  - Harder problems
  - Really hard problems

Detecting problems with deployed models

Key points

- Data and scope changes
- Monitor models and validate data to find problems early
- Changing ground truth: label new training data

Easy problems

- Ground truth changes slowly (months, years)
- Model retraining driven by:
  - Model improvements, better data
  - Changes in software and/or systems
- Labeling
  - Curated datasets
  - Crowd-based

Harder problems

- Ground truth changes faster (weeks)
- Model retraining driven by:
  - **Declining model performance**
  - Model improvements, better data
  - Changes in software and/or system
- Labeling
  - Direct feedback
  - Crowd-based

Really hard problems

- Ground truth changes very fast (days, hours, min)
- Model retraining driven by:
  - **Declining model performance**
  - Model improvements, better data
  - Changes in software and/or system
- Labeling
  - Direct feedback
  - Weak supervision

Key Points

Key points

- Model performance decays over time
  - Data and Concept Drift
- Model retraining helps to improve performance
  - Data labeling for changing ground truth and scarce labels

Data Labeling

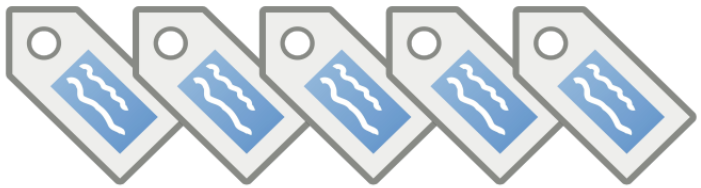
Process Feedback and Human Labeling

Data Labeling

Key points

- Variety of Methods:
  - Process Feedback (Direct Labeling)
  - Human Labeling
  - Semi-Supervised Labeling
  - Active Learning
  - Weak Supervision

Data labeling



Process Feedback — Example: Actual vs predicted click-through

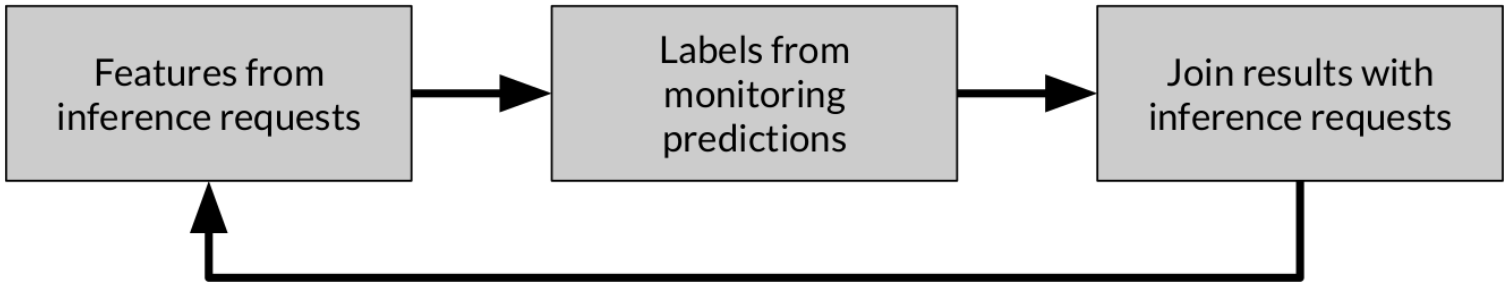
Human Labeling — Example: Cardiologists labeling MRI images

Why is Labeling Important in Production ML?

Key points

- Using business/organization available data
- Frequent model retraining
- Labeling is an ongoing and critical process
- Creating training datasets requires labels

Direct labeling: continuous creation of training dataset



Similar to reinforcement learning rewards

Process Feedback - Advantages

Key points

- Training dataset is created continuously
- Labels evolve quickly with the system
- Captures strong and timely label signals

Process Feedback - Disadvantages

Key points

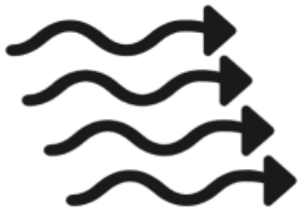
- Hindered by inherent nature of the problem
- Failure to capture ground truth
- Largely bespoke design

Process feedback - Open-Source log analysis tools

Logstash

Free and open source data processing pipeline

- Ingests data from a multitude of sources
- Transforms it
- Sends it to your favorite "stash."

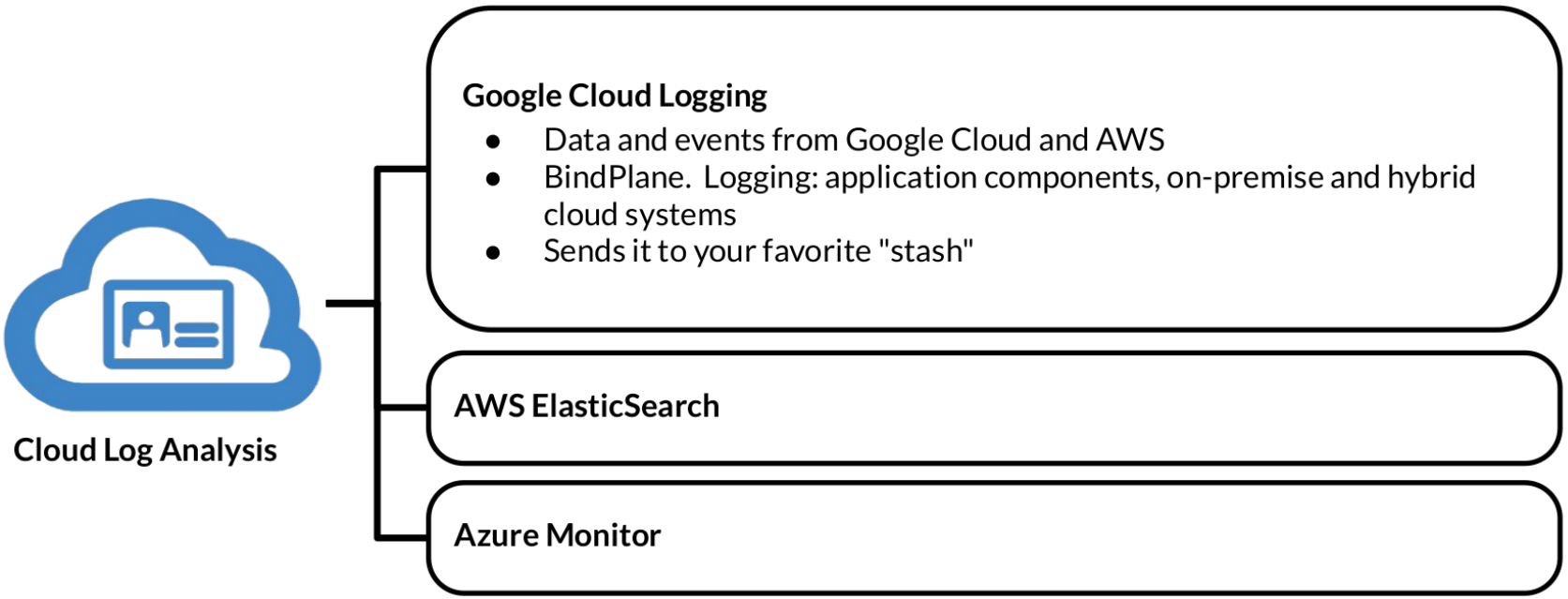


Fluentd

Open source data collector

Unify the data collection and consumption

Process feedback - Cloud log analytics



Human labeling

People (“raters”) to examine data and assign labels manually

Raw data

Unlabeled and ambiguous data is sent to raters for annotation

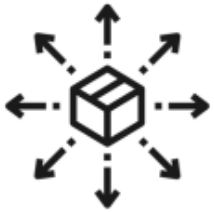
A training data set is ready for use

Human labeling - Methodology

Unlabeled data is collected

Human “raters” are recruited

Instructions to guide raters are created



Data is divided and assigned to raters

Labels are collected and conflicts resolved

Human Labeling - Advantages

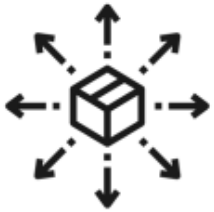
- More labels available
- Enables pure supervised learning

Human labeling - Disadvantages

Unlabeled data is collected

Human “raters” are recruited

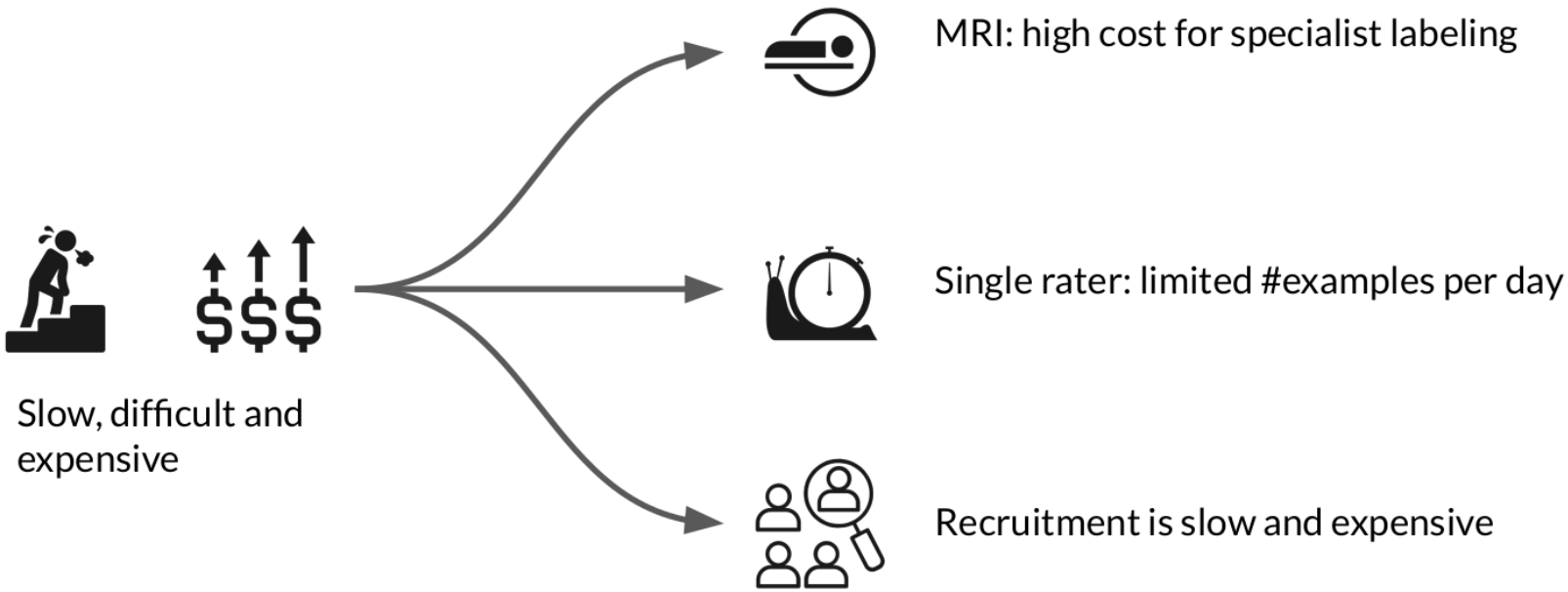
Instructions to guide raters are created



Data is divided and assigned to raters

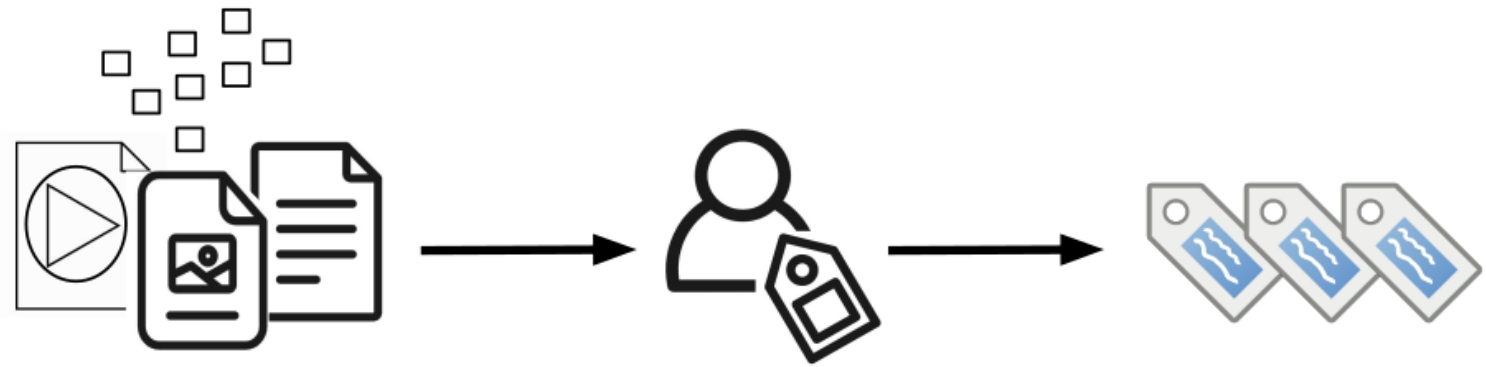
Labels are collected and conflicts resolved

Why is human labeling a problem?



Key points

- Various methods of data labeling
  - Process feedback
  - Human labeling
- Advantages and disadvantages of both



Validating Data

Detecting Data Issues

Outline

- Data issues
  - Drift and skew
    - Data and concept Drift
    - Schema Skew
    - Distribution Skew
    - Prompt Drift
    - Context Window Skew
    - Tool/Action Skew (Agentic AI)
    - Retrieval/Knowledge Base Skew (RAG)
- Detecting data issues

Drift and Skew

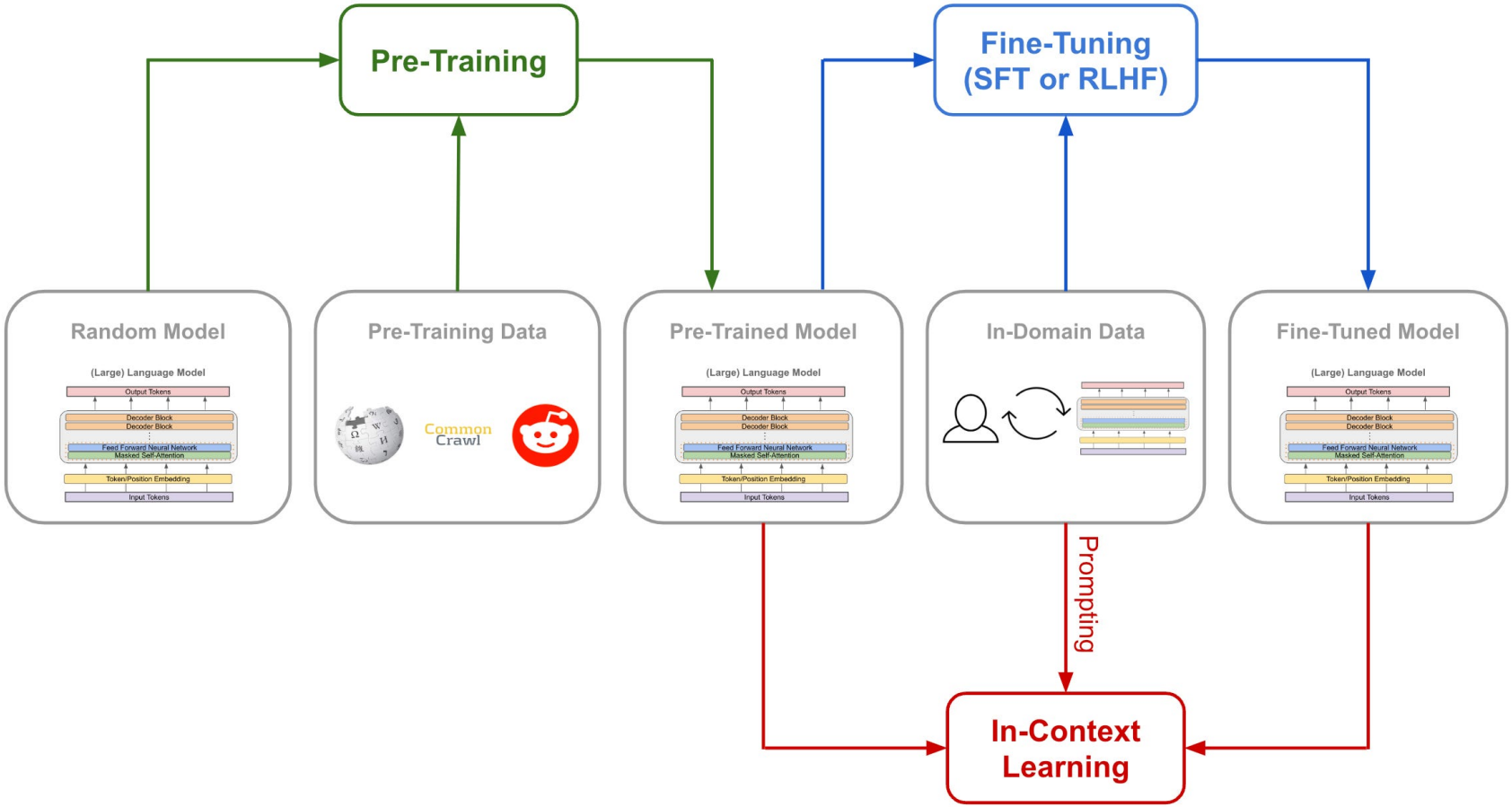
Key points

- **Drift**
  - Changes in data over time
  - Example: data collected once a day evolves gradually
- **Skew**
  - Difference between two static versions or sources
  - Example: training set vs serving set mismatch

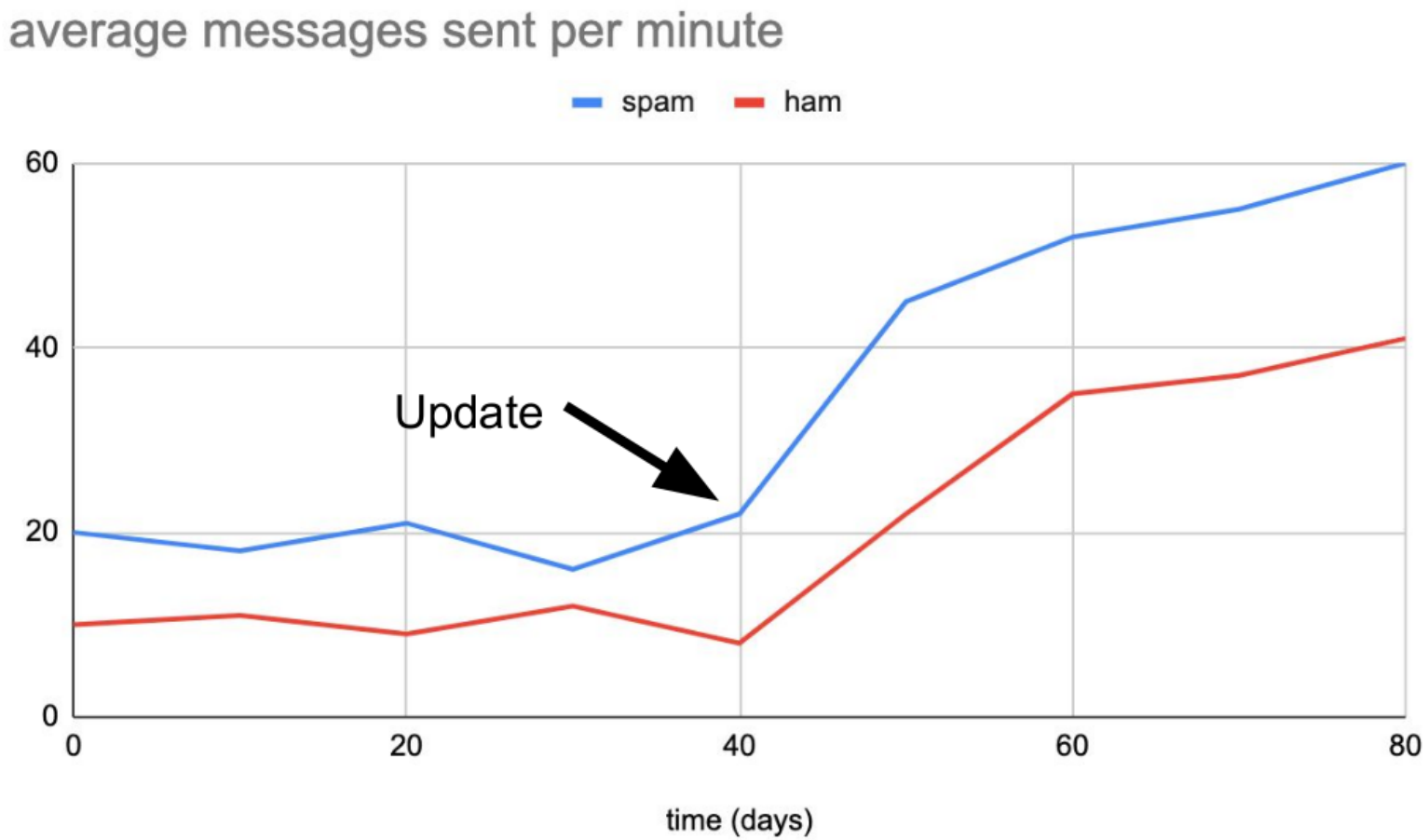
LLM/Agentic-Specific Drift and Skew

- **Prompt Drift**
  - Prompt Drift is the phenomenon where a prompt yields different responses over time due to model changes, model migration or changes in prompt-injection data at inference.
- **Context Window Skew**
  - describes the tendency for models to prioritize information at the beginning and end of a context window, leading to poor performance on tasks requiring understanding of information in the middle of long texts.
- **Tool/Action Skew (Agentic AI)**
  - disproportionate or imbalanced preference for using certain external tools or taking specific actions over others.
- **Retrieval/Knowledge Base Skew (RAG)**
  - a problem where the data within a Retrieval-Augmented Generation (RAG) system is unevenly represented, causing the system to overemphasize some information while neglecting other equally relevant data.

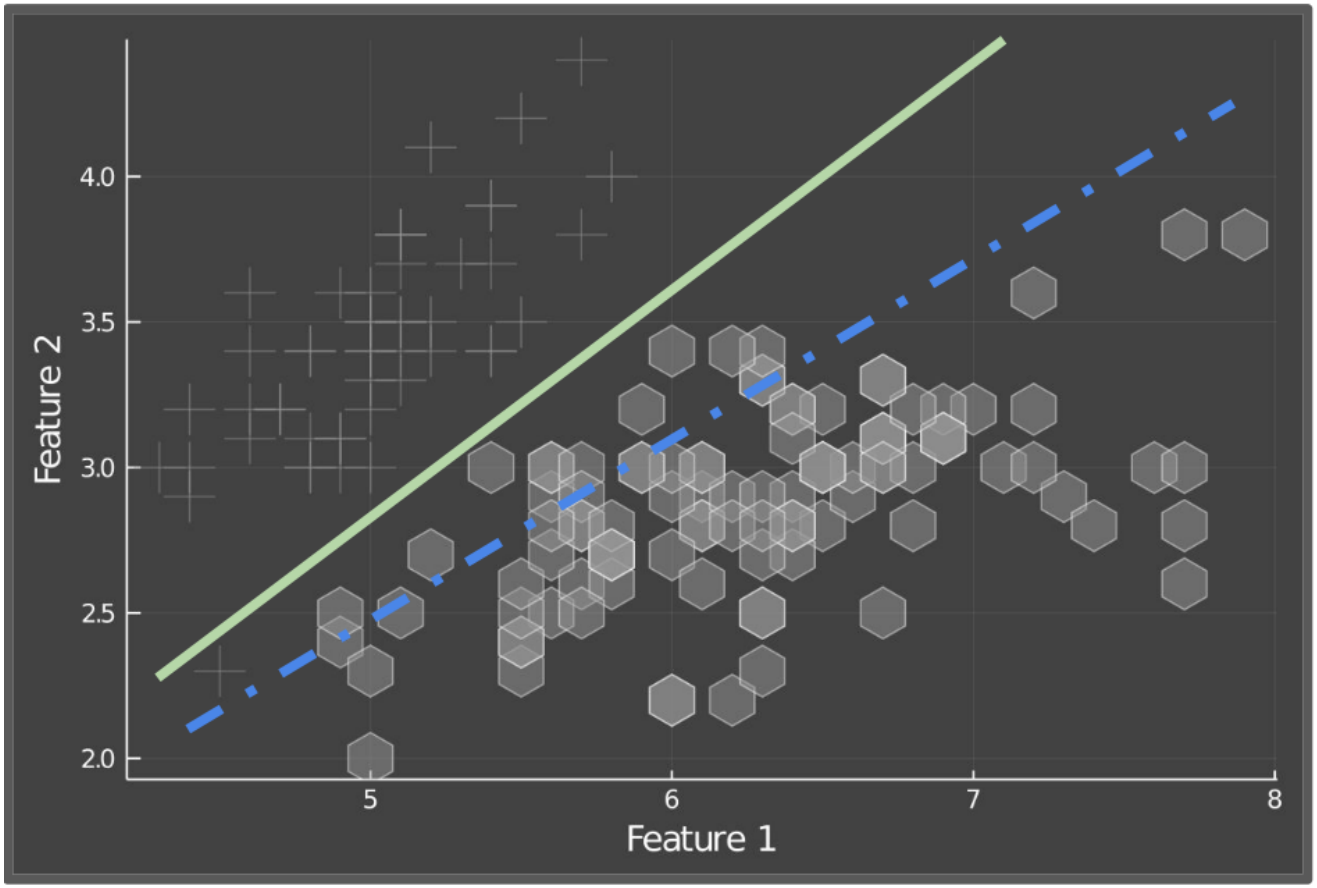
Typical LLM pipeline



Model Decay : Data drift



Performance decay : Concept drift



Detecting data issues

Key points

- Detecting schema skew
  - Training and serving data do not conform to the same schema
- Detecting distribution skew
  - Dataset shift such as covariate or concept shift
- Requires continuous evaluation

Detecting distribution skew

	Training	Serving
Joint	$P_{\text{train}}(y, x)$	$P_{\text{serve}}(y, x)$
Conditional	$P_{\text{train}}(y\|x)$	$P_{\text{serve}}(y\|x)$
Marginal	$P_{\text{train}}(x)$	$P_{\text{serve}}(x)$

Dataset shift $P_{\text{train}}(y, x) \neq P_{\text{serve}}(y, x)$

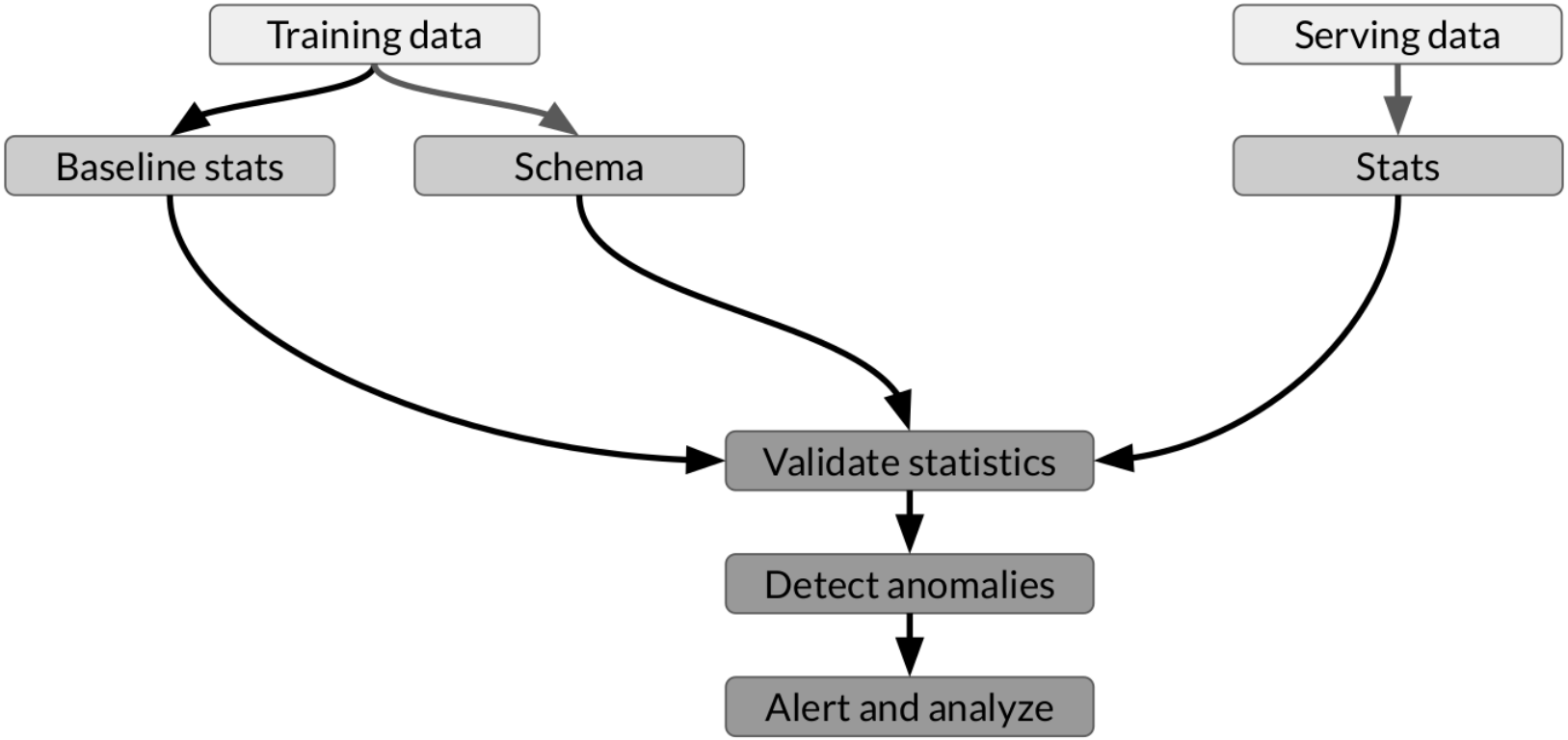
Covariate shift $P_{\text{train}}(y|x) = P_{\text{serve}}(y|x)$

$P_{\text{train}}(x) \neq P_{\text{serve}}(x)$

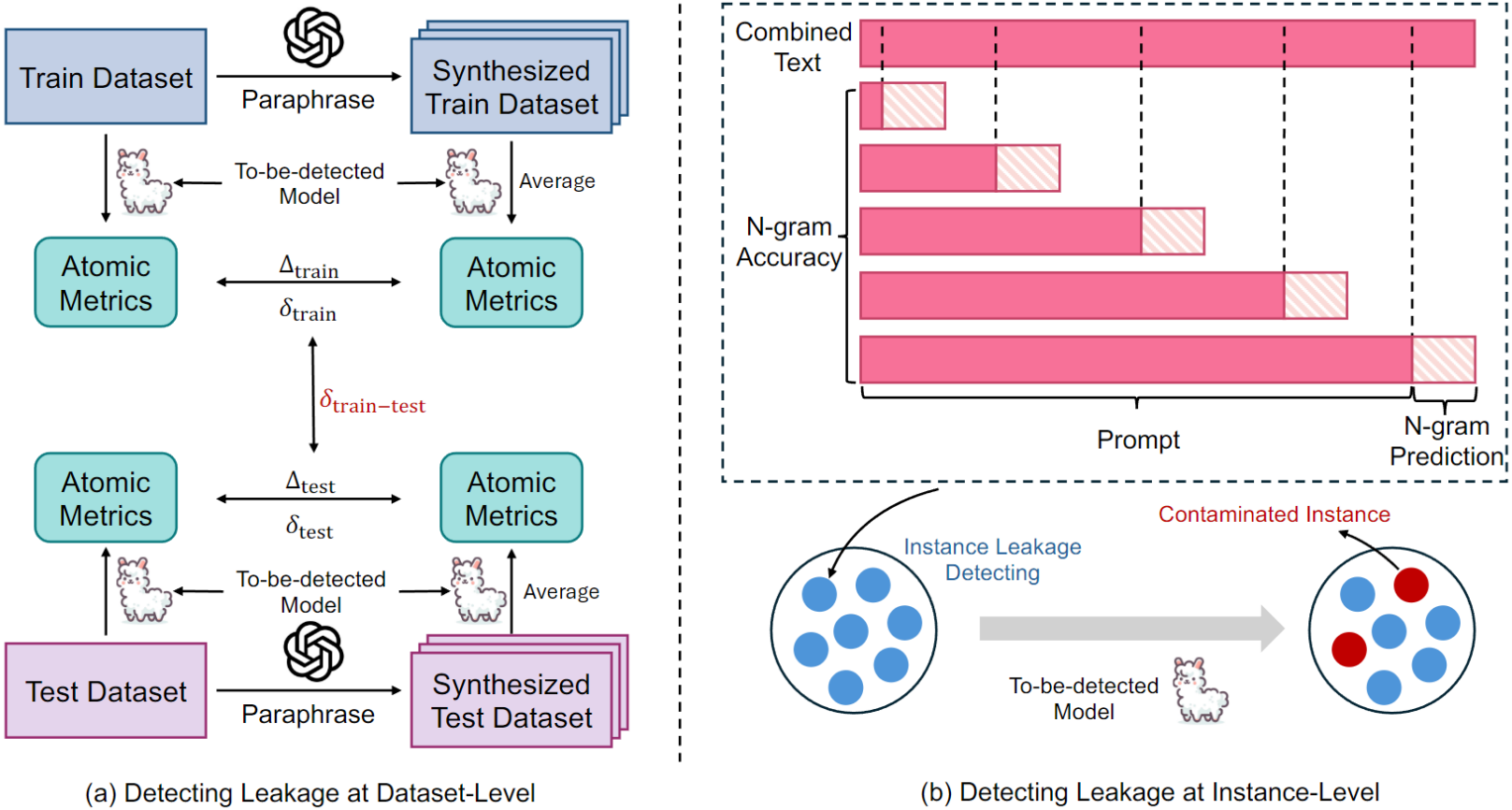
Concept shift $P_{\text{train}}(y|x) \neq P_{\text{serve}}(y|x)$

$P_{\text{train}}(x) = P_{\text{serve}}(x)$

Skew detection workflow



Benchmark Leakage in Large Language Models



(a) Detecting Leakage at Dataset-Level

(b) Detecting Leakage at Instance-Level

Figure: Benchmarking Benchmark Leakage in Large Language Models

Validating Data

TensorFlow Data Validation

TensorFlow Data Validation (TFDV)

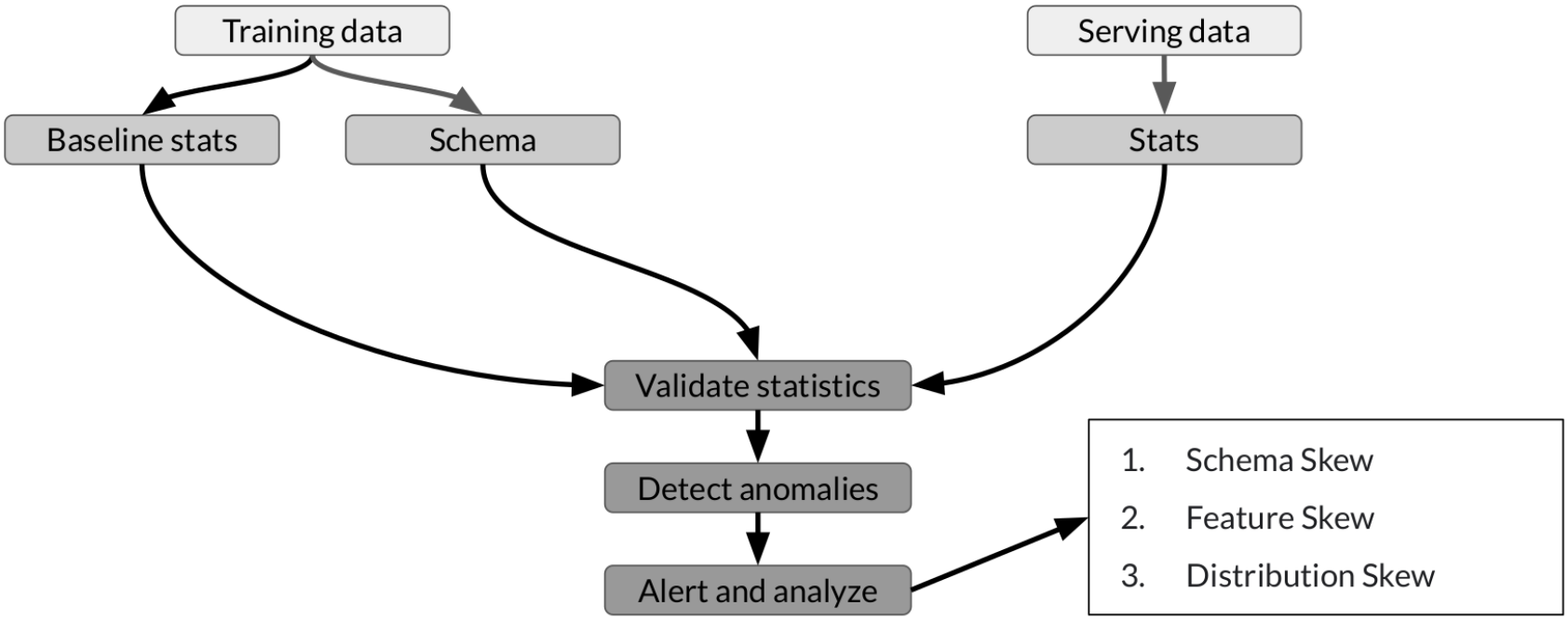
- Understand, validate, and monitor ML data at scale
- Used to analyze and validate petabytes of data at Google every day
- Proven track record in helping TFX users maintain the health of their ML pipelines

TFDV Capabilities

Key points

- Generates data statistics and browser visualizations
- Infers the data schema
- Performs validity checks against schema
- Detects training/serving skew

Skew detection - TFDV

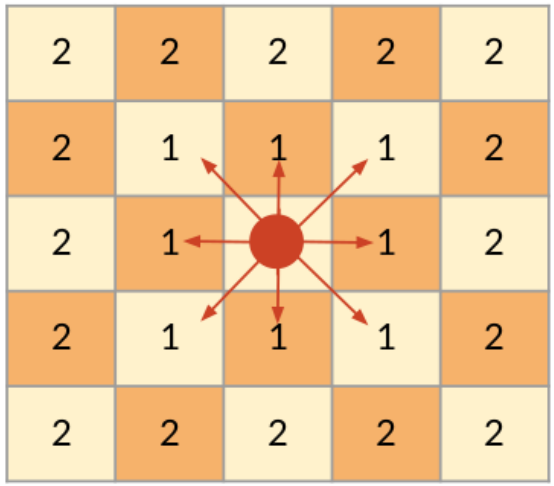


Skew - TFDV

- Supported for categorical features
- Expressed in terms of L-infinity distance (Chebyshev Distance):

$$D_{\text{Chebyshev}}(x, y) = \max_i(|x_i - y_i|)$$

- Set a threshold to receive warnings



Schema Skew

- Serving and training data don't conform to the same schema
  - For example, `int` $\neq$ `float`

Feature Skew

- Training feature values differ from serving feature values
  - Feature values are modified between training and serving time
  - Transformation applied only in one of the two instances

Distribution Skew

- Distribution of serving and training datasets is significantly different
  - Faulty sampling method during training
  - Different data sources for training and serving data
  - Trend, seasonality, or changes in data over time

Key points

- TFDV: Descriptive statistics at scale with embedded Facets visualizations
- It provides insight into:
  - What are the underlying statistics of your data
  - How do your training, evaluation, and serving dataset statistics compare
  - How can you detect and fix data anomalies

Wrap up

- Differences between ML modeling and a production ML system
- Responsible data collection for building a fair production ML system
- Process feedback and human labeling
- Detecting data issues
- Practice data validation with TFDV in this week's exercise notebook
- Test your skills with the programming assignment

Validating Data

Evidently-AI Data Validation

Evidently AI for LLM Data Evaluation

Key points

- Open-source tool for monitoring, evaluating, and validating ML and LLM data
- Provides visual dashboards and reports for dataset health, drift, and quality
- Supports tabular, text, embedding, and LLM output data

Evidently AI Capabilities for LLMs

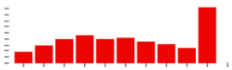
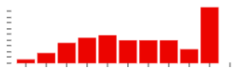
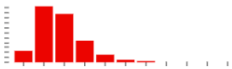
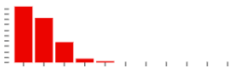
Key points

- Monitors and compares:
  - Prompt and response drift
  - Token length and truncation patterns
  - Embedding distributions
  - Text similarity (e.g., cosine, Euclidean)
- Built-in reports for:
  - Data quality
  - Drift detection
  - Text data analysis

Text Descriptors Drift for column 'Review_Text'

Drift is detected for 100.0% of columns (3 out of 3).

Search

	Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
>	Text Length	num			Detected	Wasserstein distance (normed)	0.137066
>	Non Letter Character %	num			Detected	Wasserstein distance (normed)	0.136386
>	OOV %	num			Detected	Wasserstein distance (normed)	0.131852

Why Use Evidently for LLM Pipelines?

Key points

- Enables continuous validation of LLM input/output at scale
- Detects shifts in:
  - Prompt structure and user queries
  - Generated text diversity, length, and format
  - Embedding-based semantic drift
- Useful for RAG pipelines, fine-tuning data curation, and production monitoring

Labs for This Week

Objective

Briefly describe the learning goal for this week's lab(s).

Lab Activities:

- Lab 1: [TFDV] - [TFDV Tutorial]
- Lab 2: [Evidently-AI] - [Evidently-AI Tutorial]
- Lab 3: [PT Streaming] - [PT Streaming Data Pipeline]

Submission Deadline: [Before the next class]

- Assignment 5: [TFDV] - [Create a data validation of your choice]
- Assignment 5: [Evidently-AI] - [Create a data validation of your choice]

Reading Materials

This Week's Theme

Topic focus: [People + AI Guidebook - Data Collection + Evaluation.pdf]
You should use the worksheet related to this pdf to your project and submit it when its requested.

Required Readings:

- [Benchmarking Benchmark Leakage in Large Language Models]
- [Detection of data drift and outliers affecting machine learning model performance over time]

Be prepared to discuss highlights and open questions in class.

 DeepLearning.AI

 The People + AI Guidebook