Machine Learning Engineer Interview Challenge

Background

An e-commerce company wants to improve its product recommendation system to increase customer engagement and sales. The current system uses basic collaborative filtering but struggles with cold start problems, lacks personalization for new users, and doesn't effectively utilize contextual information like browsing behavior and seasonal trends.

The challenge is to design and implement a more sophisticated recommendation system that addresses these limitations while ensuring scalability for millions of users and products.

Objective

Build a production-ready machine learning system that generates personalized product recommendations based on historical user behavior, product metadata, and contextual information. Your solution should emphasize not just model accuracy but also system architecture, deployment strategy, and operational considerations.

Dataset Overview

- **User-Product Interactions**: 2 million user-product interactions (views, cart additions, purchases)
- **User Profiles**: Demographic and account information for 200,000 users
- **Product Catalog**: Metadata for 50,000 products including categories, descriptions, prices
- **Browsing Sessions**: Sequential user actions during website visits
- **Temporal Data**: 12 months of historical data with seasonal patterns

Scope of Work

1. Data Pipeline Development

- Design ETL processes for combining multiple data sources
- Implement feature engineering for user, product, and contextual features
- Create training/validation/test splits that respect temporal ordering

2. Model Development

- Develop and compare at least two different recommendation approaches
- Address the cold-start problem for new users and products
- Incorporate contextual and sequence information into the model

3. System Architecture

- Design a scalable architecture for training and serving recommendations
- Plan for model updates and feature freshness
- Implement appropriate caching and performance optimizations

4. Evaluation Framework

- Define offline evaluation metrics aligned with business objectives
- Develop an A/B testing strategy for production deployment
- Create monitoring for model performance and data drift

5. Deployment Strategy

- Design a CI/CD pipeline for model deployment
- Implement model versioning and rollback capabilities
- Ensure the system can handle production traffic patterns

Technical Requirements

- Well-documented, modular code with appropriate testing
- Containerized application components using Docker
- Clear separation between training and inference pipelines
- Efficient data processing suitable for large-scale deployment
- Consideration of latency requirements for real-time recommendations

Evaluation Criteria

Your solution will be evaluated based on:

- Technical implementation quality and adherence to ML engineering best practices
- System design choices and scalability considerations
- Model performance on recommendation quality metrics
- Operational excellence (monitoring, deployment, maintenance)
- Code quality, documentation, and reproducibility

Discussion Questions

- How would your system handle a 10x increase in users or products?
- What strategies would you implement to continuously improve recommendation quality?
- How would you measure the business impact of your recommendation system?
- What are the trade-offs between model complexity and serving latency?
- How would your system adapt to changing user preferences over time?

Deliverables

- Complete code repository with setup instructions
- System architecture diagram and documentation
- Performance analysis of recommendation models
- Deployment and operational strategy document
- Brief presentation outlining your approach and key decisions

Final Notes

This challenge evaluates your ability to build complete machine learning systems that go beyond model development to address the entire ML lifecycle. Focus on creating a solution that not only performs well in theory but would also work reliably in a production environment.