دانشکار

{ یک یادگیری کارساز }

بوتکمپ

Artificial Intelligence

About me

Seyed Faridoddin Kiaei

MSc of AI - University of Tehran

+8 years of experience as AI/ML/Data Engineer

MLOps

Session 1

Welcome to MLOps Fundamentals

- **Course Overview**: 8 sessions, 12 hours total
- **Today's Focus**: Understanding MLOps foundations
- **Quick Poll**: Who has struggled with model deployment?

Session 1 Learning Objectives

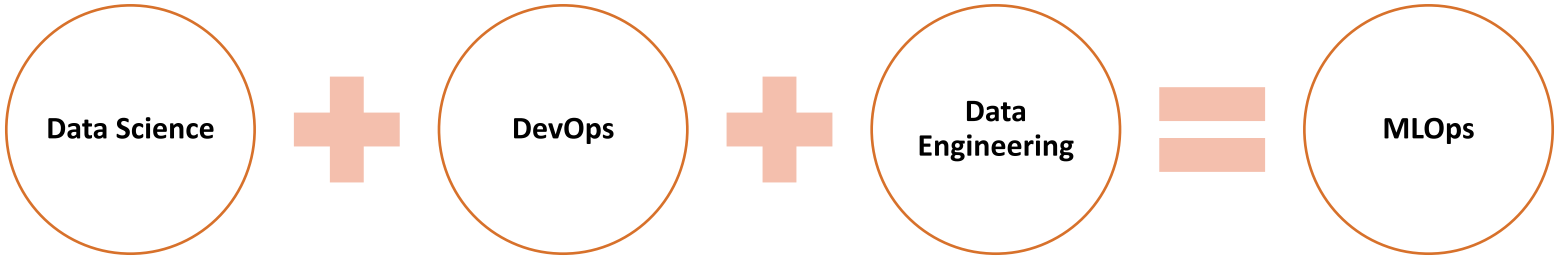
By the end of this session, you will understand:

- What MLOps is and why it's critical
- Key differences: Traditional software vs ML development
- Complete MLOps lifecycle overview
- Key roles and team dynamics
- Real-world examples and common pitfalls

What is MLOps?

Machine Learning Operations:

- **Definition**: Set of practices combining ML, DevOps, and Data Engineering to deploy and maintain ML systems in production reliably and efficiently
- **Purpose**: Bridge the gap between data science experimentation and production systems



Meet VisionaryAI

Our Example Company - AI Services Provider

- **Computer Vision**: Mobile phone factory defect detection
  - Real-time image analysis on production lines
  - Thousands of photos per hour processing
- **NLP**: AI-powered customer support
  - Automatic ticket categorization
  - Response suggestions for human agents
- **Recommender Systems**: E-commerce personalization
  - Product recommendations (cases, chargers, accessories)
  - Millions of users, millisecond response times

The Problem - Without MLOps

VisionaryAI's Initial Struggles:

- Models worked in notebooks (95% accuracy)
- Failed in production environments
- No systematic monitoring or maintenance

"Jupyter Notebook: ✅ Working Great"

"Production: ❌ Everything's Broken"

Real Failure Stories

- **Defect Detection Disaster**
  - Model degraded after 2 weeks in factory
  - Lighting changes, new phone models, dirty cameras
  - No alerts or adaptation mechanisms
- **Support Ticket Catastrophe**
  - Critical security issues misclassified as "general inquiries“
  - Training data bias (summer vs winter issue patterns)
  - Customer satisfaction plummeted
- **Recommendation Nightmare**
  - Winter coats recommended in Florida
  - No real-time location/seasonal adjustments
  - Inventory misalignment

Traditional Software vs ML Development

Traditional Software	Machine Learning
Deterministic behavior	Non-deterministic results
Code-only dependencies	Data + Code dependencies
Predictable failures	Silent performance decay
Stable over time	Degrades without maintenance
Known inputs/outputs	Evolving data distributions

Why ML Development is Unique

Key Challenges:

- **Non-Deterministic:** Same data → different models (random initialization)
- **Data Dependency:** Model behavior tied to training data quality/distribution
- **Silent Failures:** Models degrade without obvious errors
- **Performance Decay:** Real world changes → model becomes less accurate
- **Continuous Evolution:** Requires ongoing monitoring and retraining

MLOps Lifecycle Overview

Data Management

- Collection
- Versioning
- Validation
- Lineage tracking

Model Development

- Experiment tracking
- Reproducible environments
- Systematic validation

Model Deployment

- Packaging
- Automation
- Scaling
- Multiple deployment strategies

Monitoring & Maintenance

- Performance tracking
- Drift detection
- Automated retraining

Data Management Phase

VisionaryAI Examples:

- Defect Detection: Camera feeds, lighting conditions, product variants
- Support Tickets: Customer inquiries, agent responses, resolution outcomes
- Recommendations: User interactions, purchases, inventory, seasonal trends

MLOps Approach:

- Version control for datasets
- Data quality validation
- Lineage tracking (which data → which model)

Model Development Phase

Beyond Jupyter Notebooks:

Systematic Experimentation

- Track parameters, metrics, model versions
- Reproducible environments

Organized Code Structure

- Version control for ML code
- Configuration-driven development

Comprehensive Validation

- Accuracy + bias + fairness + robustness testing

Model Deployment Phase

From Laptop to Production:

- **Consistent Packaging:** Models + dependencies + configurations
- **Automated Deployment:** Repeatable, testable processes
- **Scalable Infrastructure:** Handle varying demand automatically
- **Multiple Strategies:** Batch, real-time, edge deployment options

Monitoring & Maintenance Phase

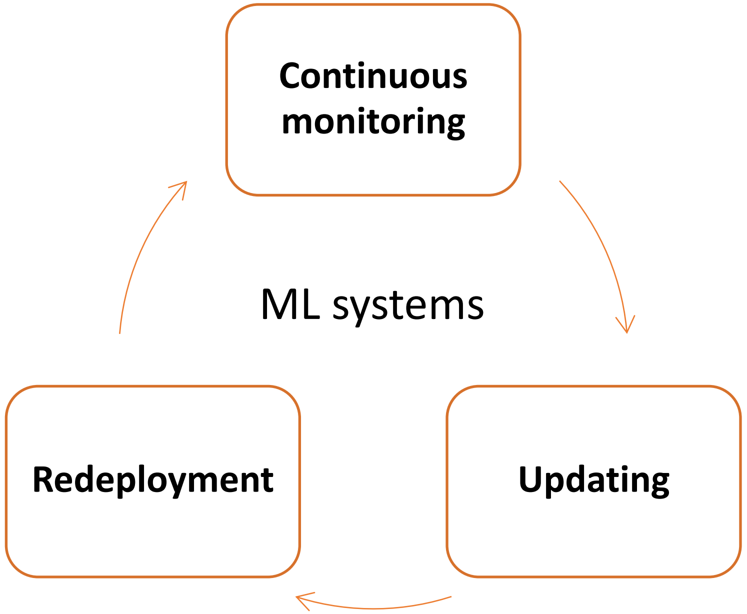
Continuous Vigilance:

- **Performance Monitoring:** System metrics + model-specific metrics
- **Drift Detection:** Data drift + model performance drift alerts
- **Automated Retraining:** Trigger retraining when thresholds crossed
- **Continuous Improvement:** Learn from production feedback

The Continuous Loop

Key Insight: MLOps is never "done"

- New data arrives continuously
- Model performance tracked continuously
- Retraining happens automatically when needed
- Business requirements evolve



Traditional software: Deploy once → Runs unchanged

VisionaryAI's MLOps Success

After Implementing MLOps:

- **Defect Detection:** Auto-adapts to lighting/product changes
- **Support System:** Learns from new ticket patterns continuously
- **Recommendations:** Daily updates with user behavior + inventory
- **Result:** Issues detected in hours (not weeks), customer satisfaction restored

MLOps Team - Key Roles

- **Data Scientists:** Model builders, algorithm experts
- **ML Engineers:** Production-ready ML systems
- **DevOps Engineers:** Infrastructure, CI/CD, monitoring
- **Data Engineers:** Data pipelines, quality, availability
- **Product Managers:** Business requirements, success metrics
- **Domain Experts:** Business context, result interpretation

Your MLOps Journey

As an MLOps Practitioner:

- **Multiple Hats:** Especially in smaller organizations
- **Bridge Builder:** Connect data science with engineering
- **Tool Implementer:** CI/CD, monitoring, deployment systems
- **Collaboration Facilitator:** Help teams work together effectively
- **Key:** Understand how all roles contribute to reliable ML systems

Industry Success Examples

Netflix: 200M+ users, multiple daily model deployments

Uber: Real-time demand forecasting across hundreds of cities

Spotify: Billions of interactions, continuous personalization

Common Success Factors:

- Systematic experimentation
- Automated deployment
- Comprehensive monitoring
- Cross-functional collaboration

Common MLOps Pitfalls

- **Model-Centric Thinking:** Accuracy over reliability
- **Notebook-to-Production Gap:** Treating notebooks as production systems
- **Ignoring Data Drift:** Assuming training data is forever valid
- **Manual Everything:** No automation for deployment/monitoring
- **Siloed Teams:** Poor collaboration between roles

How to Avoid Pitfalls

- **Think Systems-First:** 90% accurate + reliable > 95% accurate + unreliable
- **Clear Boundaries:** Notebooks for exploration, proper code for production
- **Monitor Everything:** Data drift + model drift from day one
- **Automate Early:** If you do it twice, automate it
- **Foster Collaboration:** Shared tools, processes, metrics

Key Takeaways

- **MLOps is Essential:** Not optional in modern ML development
- **ML ≠ Traditional Software:** Unique challenges require specialized approaches
- **Team Sport:** Requires collaboration across multiple disciplines
- **Systems Thinking:** Focus on reliability, not just model accuracy
- **Start Simple:** Basic practices prevent major headaches

Next Session Preview

- **Project Organization:** Structure for success
- **Version Control:** Code + Data + Models
- **Collaboration:** Workflows that actually work
- **Reproducibility:** Environments and dependencies
- **Come Prepared:** Think about your current ML project organization!

Questions & Discussion

Before We Move to Code Demo:

- **Reflect:** What MLOps challenges have you experienced
- **Discuss:** Which VisionaryAI scenario resonates most
- **Ask:** Any questions about today's concepts
- **Next:** Hands-on code demonstration - Messy vs Organized ML Code


دانشگار
THANK YOU
FOR YOUR ATTENTION!