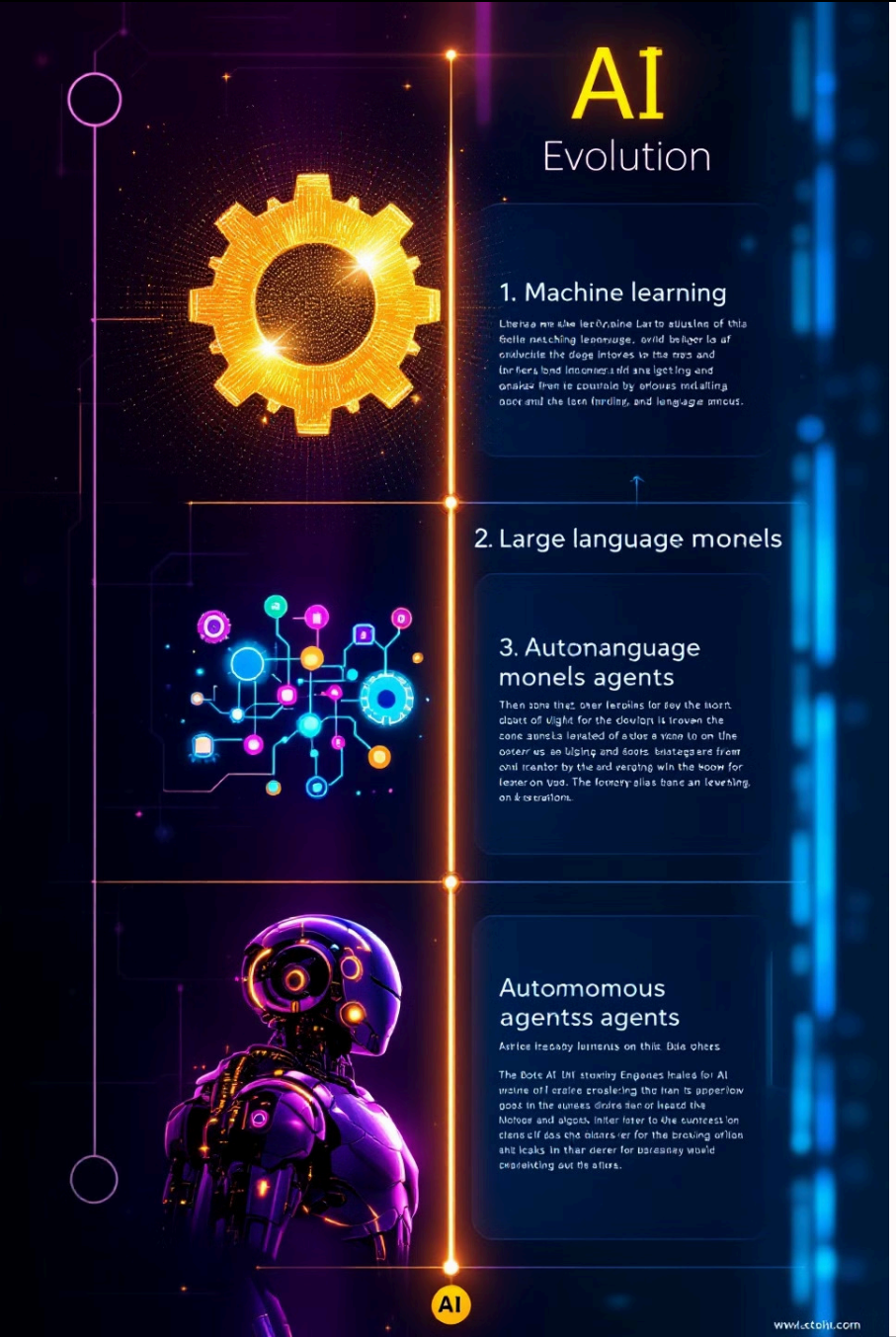


Story of AI Infrastructure Ops

The Evolution: From ML to MLOps to LLMs and Agentic AI

Presented by the School of DevOps, this session will take you through the historical development, current state, and future directions of AI operations.

by Gourav Shah

The Foundation Years (1960s-1990s)

1. **1960s**

   The Perceptron model introduced the first neural network architecture, laying groundwork for future AI development.

2. **1970s**

   The "AI Winter" begins as expectations outpaced results, leading to reduced funding and interest in the field.

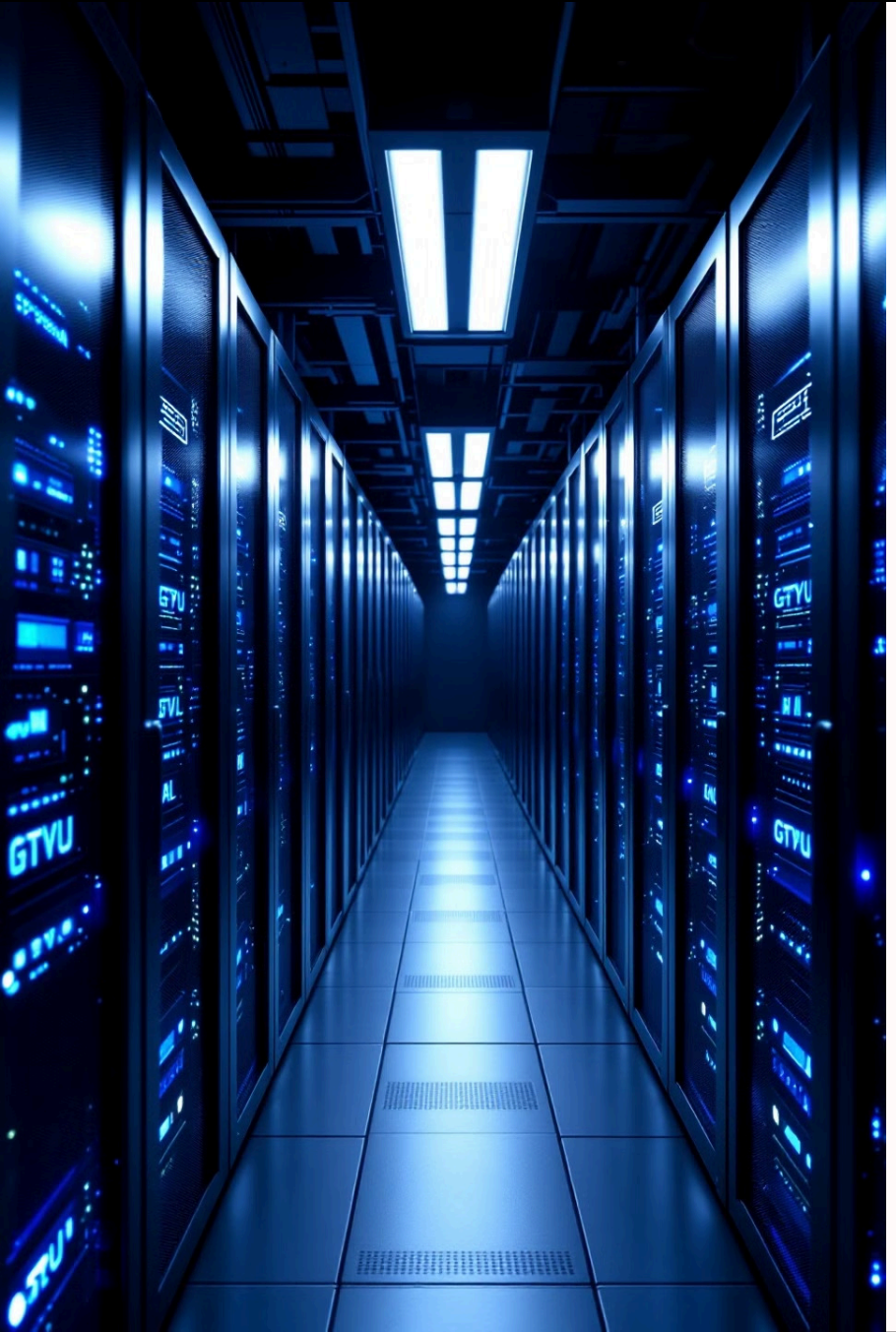
3. **1980s**

   Expert Systems gain popularity in specific domains, showing AI's potential for specialized tasks.

4. **1990s**

   Support Vector Machines and Decision Trees emerge as powerful new algorithms, revitalizing the field.

This period was like the construction phase of the Delhi Metro - laying the groundwork despite widespread skepticism about the technology's future.



The Renaissance Period (2000s-2010)

1

Deep Learning Revolution

In 2006, Geoffrey Hinton's breakthrough research sparked a renaissance in neural networks and deep learning approaches.

2

Computational Power

The transition from CPUs to GPUs dramatically increased processing capabilities, enabling more complex models.

3

Big Data Access

The availability of massive datasets provided the fuel needed for training sophisticated machine learning models.

4

Practical Applications

ML transitioned from academic curiosity to practical applications in business, healthcare, and consumer technology.

This evolution paralleled how smartphones transformed from luxury items to everyday necessities in India - a technology becoming essential to daily life.

The Industrialization Era (2010-2015)

Scaling Challenges

Companies began implementing ML at scale, but encountered significant operational hurdles in the process.

Deployment Gap

A key challenge emerged: **How do we deploy models reliably?** Data scientists could create models, but production deployment remained difficult.

The Laptop Problem

The "model in a laptop" problem became apparent - models that worked perfectly in development environments failed in production.

This gap resembled the difference between creating a blueprint for a flyover in Bangalore and actually constructing it in real-world conditions with all their complexities and constraints.

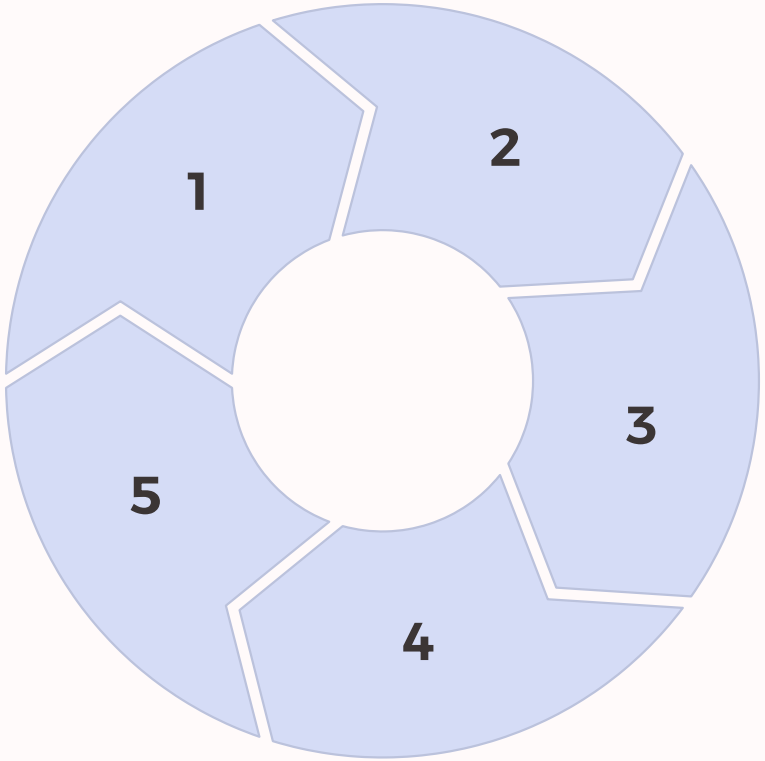
The Birth of MLOps (2015-2018)

Reproducibility

Ensuring models can be recreated consistently

Governance

Ensuring compliance and responsible use



Versioning

Tracking changes to code, data, and models

Deployment

Reliably moving models to production

Monitoring

Tracking performance and detecting issues

The term "MLOps" emerged as a discipline combining DevOps principles with ML workflows to address these key challenges. Just as IT companies in India adopted Agile methodologies, ML teams needed their own operational framework to succeed at scale.

MLOps in Action

Version Control

Not just code, but data and models too, ensuring reproducibility and collaboration across teams.

CI/CD for ML

Automated testing and deployment pipelines that validate models before they reach production environments.

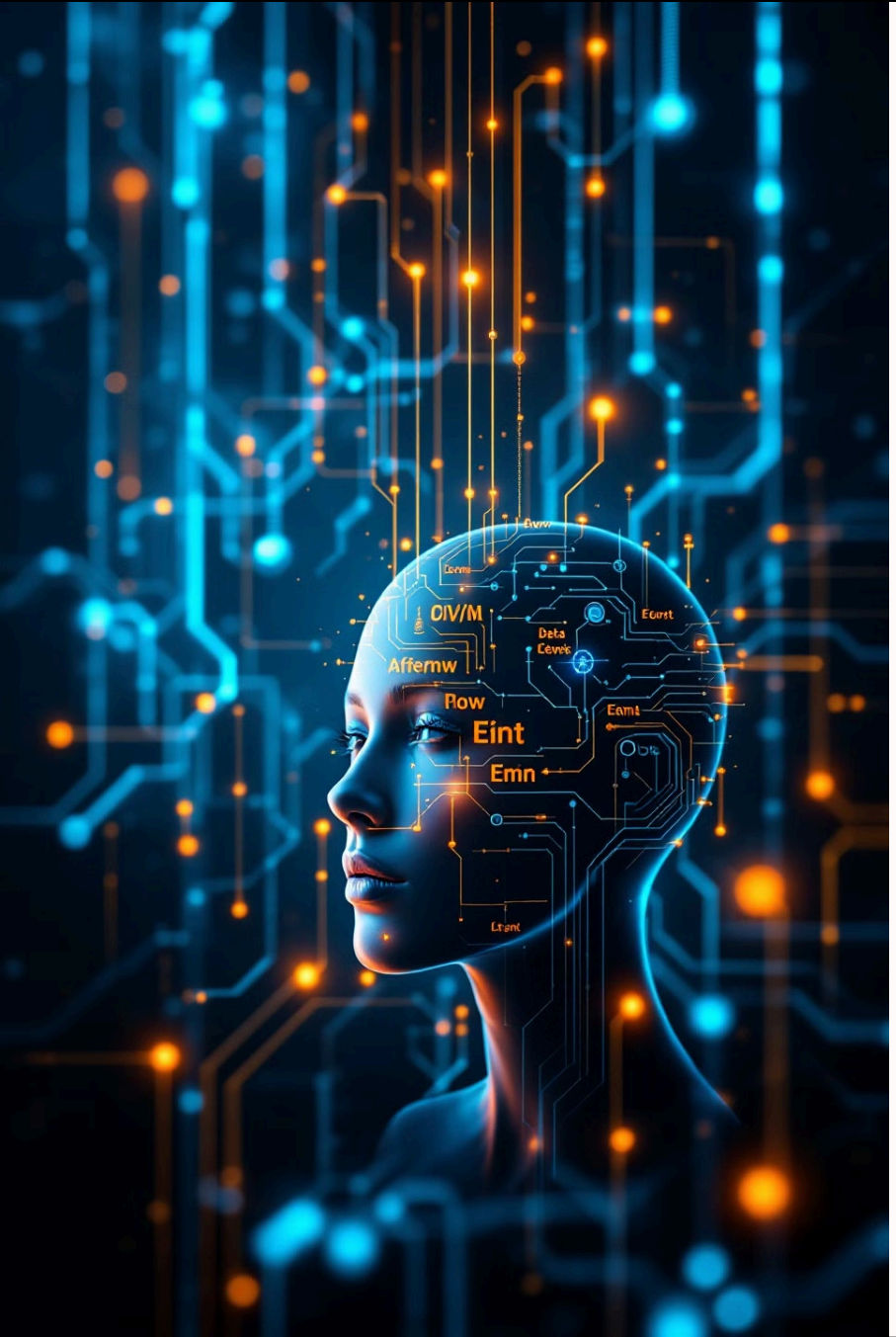
Monitoring

Systems for detecting data drift and model performance degradation to maintain quality over time.

Governance

Tracking lineage and ensuring compliance with regulations and organizational policies.

This transformation was similar to how IRCTC evolved from a manual ticketing system to a robust digital platform - bringing structure, reliability, and scale to what was once a chaotic process.



The Rise of Transformer Models (2017-2019)

Attention Mechanism (2017)

Google introduces the groundbreaking "Attention Is All You Need" paper, presenting the transformer architecture that would revolutionize NLP.

BERT (2018)

Bidirectional Encoder Representations from Transformers demonstrates unprecedented language understanding capabilities.

GPT-2 (2019)

Generative Pre-trained Transformer 2 showcases impressive text generation abilities, hinting at the potential of larger models.

This technological leap was the equivalent of moving from regular trains to the Bullet Trains - a quantum jump in capability that transformed what was possible with language processing.

The LLM Revolution (2020-2022)

1 GPT-3 Emergence

In 2020, GPT-3 with its 175 billion parameters demonstrated remarkable emergent abilities not seen in smaller models, showing capabilities beyond what it was explicitly trained to do. ChatGPT happened in 2022.

2 API Accessibility

Foundation models became accessible via APIs, democratizing access to powerful AI capabilities without requiring massive computing resources.

3 Application Explosion

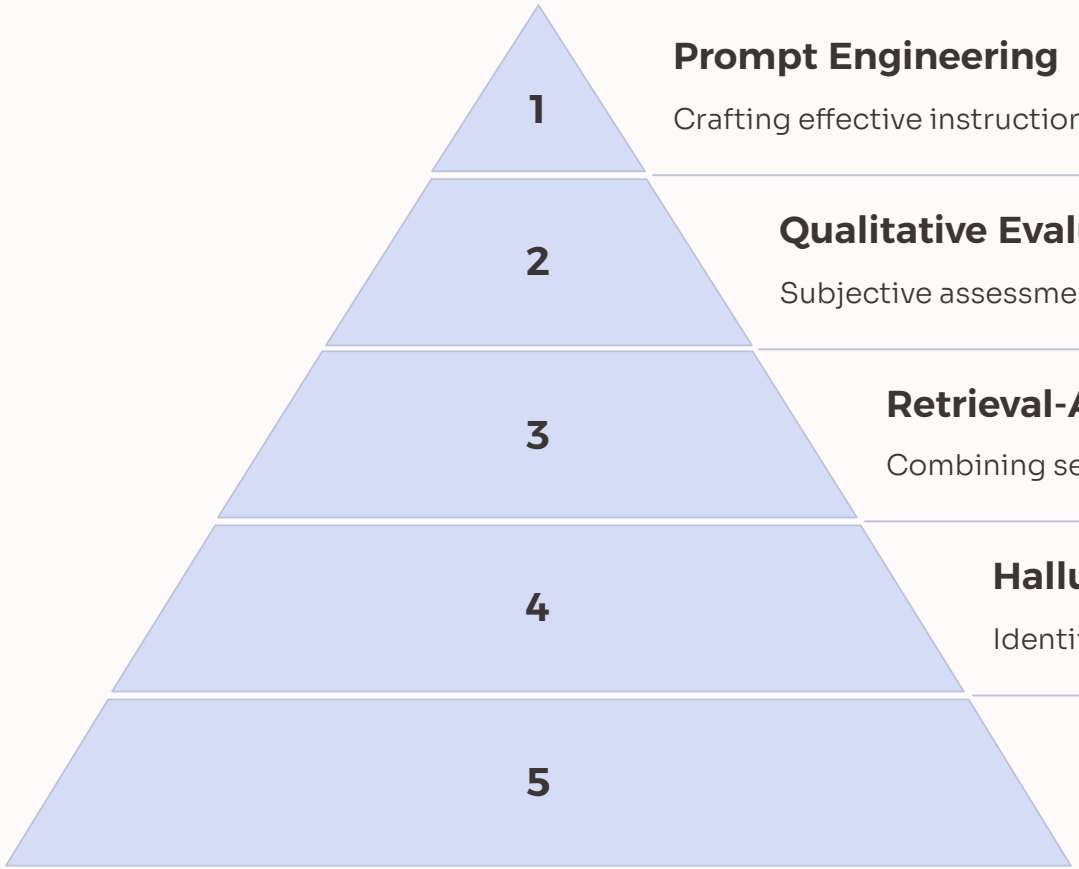
Applications of LLMs exploded across industries, from content creation to code generation, customer service to healthcare diagnostics.

4 Versatility

LLMs showed versatility beyond traditional ML models, handling a wide range of tasks with the same underlying architecture.

Like how UPI transformed digital payments in India, LLMs began transforming how we interact with technology - creating a new paradigm for human-computer interaction.

New Operational (Ops) Challenges



Prompt Engineering

Crafting effective instructions

Qualitative Evaluation

Subjective assessment methods

Retrieval-Augmented Generation

Combining search with generation

Hallucination Detection

Identifying false information

Specialized Monitoring

New metrics for language models

The operational complexity of LLMs introduced entirely new challenges beyond traditional ML systems. This was like moving from running a roadside dhaba to managing a 5-star hotel - requiring new skills, processes, and quality standards at every level.

LLMOps Emerges (2022-2023)

Prompt Management

Systems for versioning, testing, and optimizing prompts became essential as prompts became the new code.

Vector Databases

Specialized databases for storing and retrieving embeddings enabled knowledge retrieval and grounding in facts.

Fine-tuning Workflows

Processes for adapting foundation models to specific domains and tasks with custom datasets.

LLMOps extended MLOps principles for language models, with additional focus on evaluation frameworks for language tasks and responsible AI controls. This evolution paralleled how Swiggy and Zomato had to develop new operational frameworks beyond traditional restaurant management to handle their digital food delivery ecosystems.

Vector Databases & Embeddings Explained

From keywords to meaning: how AI understands semantic relationships

1

The Problem

Traditional search only finds exact keyword matches, missing semantic connections.

2

The Challenge

When a customer searches "light blue top" but your database has "azure cotton blouse."

3

The Solution

Vector embeddings convert text into numerical patterns that capture meaning and context.

4

The Implementation

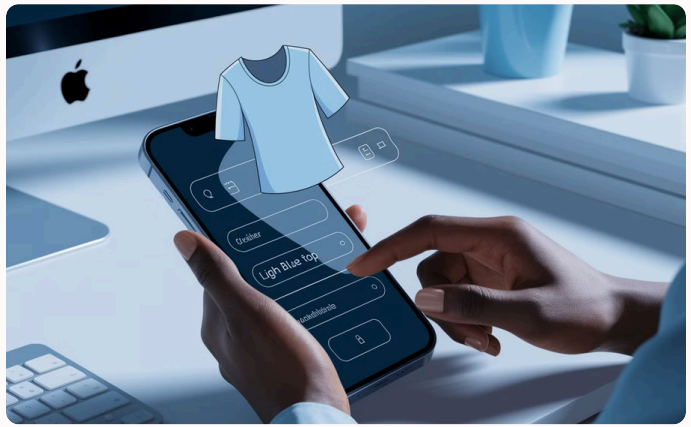
Specialized databases find similar patterns quickly, enabling semantic search capabilities.

Vector Databases & Embeddings

Creating Embeddings

Products transform into numerical vectors capturing semantic meaning. "Blue cotton t-shirt" becomes

[0.2, 0.8, 0.1, ...].

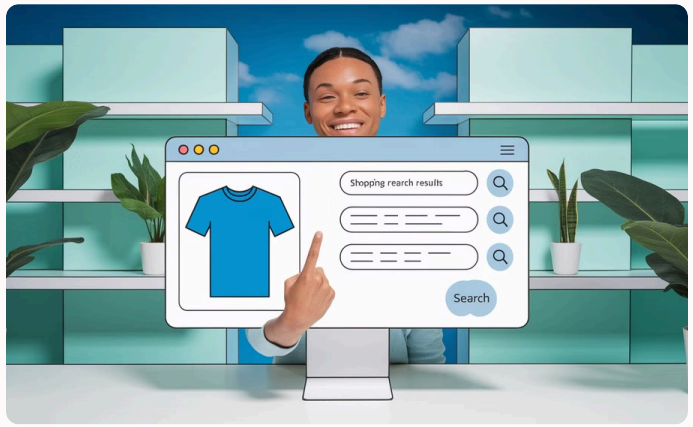


Customer Search

Search terms (light blue top) convert to vectors

[0.25, 0.75, 0.2, ...].

System matches patterns, not keywords.



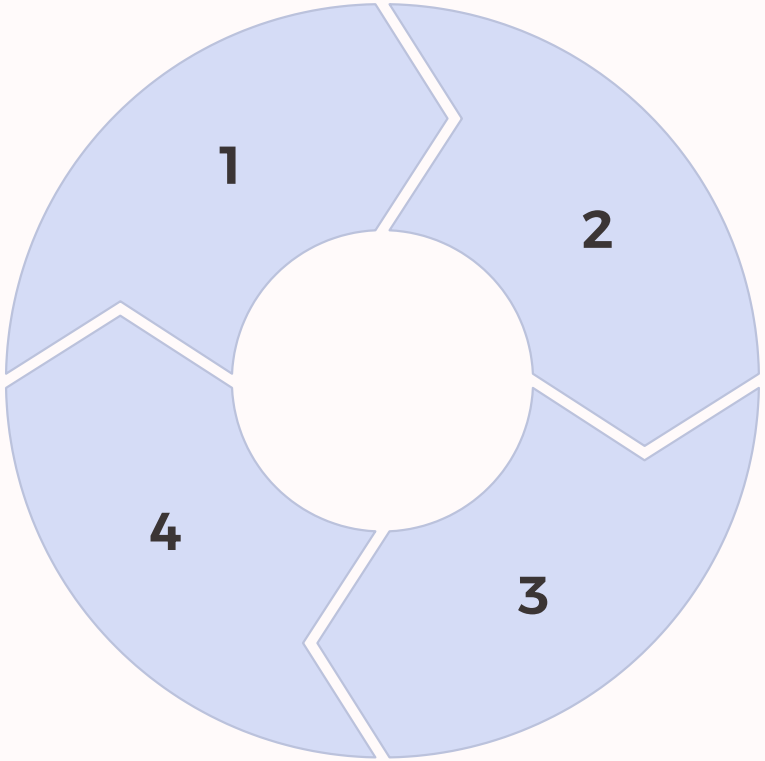
Business Benefits

Better results create happier customers. System finds relevant products across languages and handles misspellings.

Vector Databases & Embeddings

Create Embeddings

Convert words, images or concepts into numerical vectors that preserve meaning.



Store in Vector DB

Specialized databases organize these vectors for efficient retrieval.

Semantic Search

Find similar concepts by measuring "distance" between vectors.

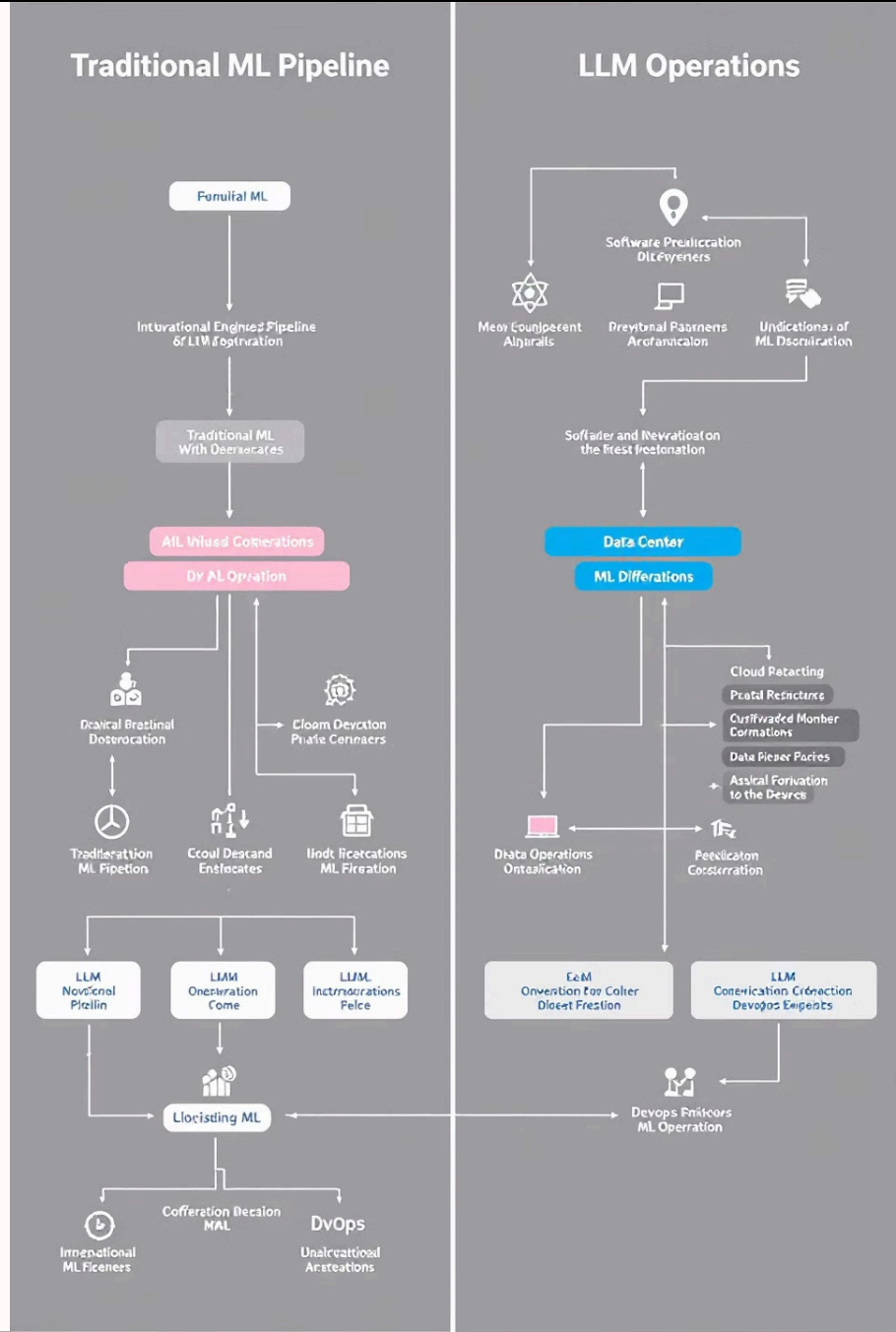
Apply Results

Power recommendation systems, search engines, and AI knowledge retrieval.

LLMOps vs. Traditional MLOps

Aspect	Traditional MLOps	LLMOps
Focus	Data-centric	Prompt-centric
Evaluation	Structured metrics	Qualitative assessment
Model Development	Training from scratch	Fine-tuning & adapters
Workflow Structure	Linear pipelines	Complex with retrieval
Monitoring Emphasis	Technical performance	Ethical considerations

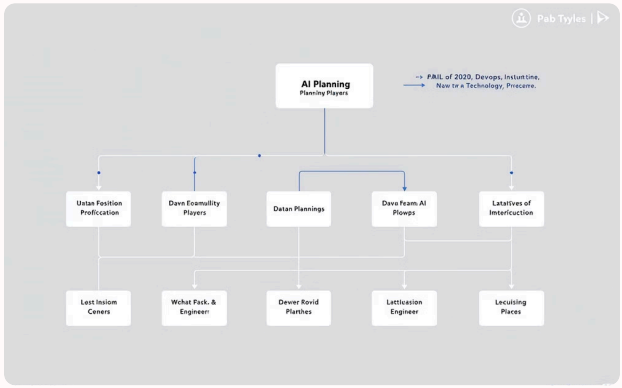
While traditional MLOps and LLMOps share fundamental principles, they differ significantly in implementation details and focus areas. These differences reflect the unique characteristics and challenges of language models compared to traditional ML systems.



The Dawn of Autonomous Agents (2023-Present)

Tool Usage

LLMs gained the ability to use tools and APIs, extending their capabilities beyond text generation to interaction with external systems.



Planning

Models developed the capability to plan sequences of actions, breaking down complex tasks into manageable steps.

Memory

Memory and reasoning capabilities emerged, allowing systems to maintain context and learn from past interactions.

This evolution resembles the progression from manual rickshaws to self-driving cars - not just following instructions but making decisions and navigating complex environments autonomously.

Agentic AI Emerges

1

Planning

Breaking down complex goals

2

Decision-making

Choosing actions with reasoning

3

Tool usage

Leveraging external capabilities

4

Memory

Maintaining context over time

5

Learning

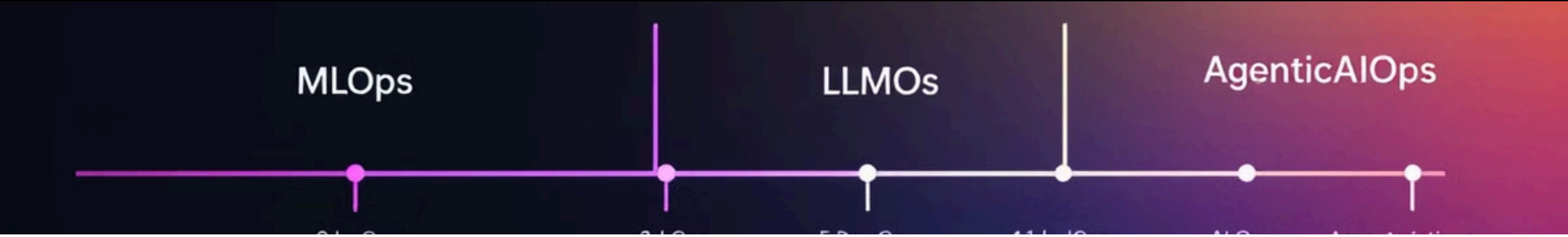
Improving from feedback

Agentic AI represents autonomous systems that can plan multi-step tasks, make reasoned decisions, use tools and APIs, maintain memory and context, and learn from feedback. This resembles how a skilled project manager in an IT company coordinates multiple teams to deliver a complex project - orchestrating resources toward a goal.

Operational (Ops) Needs for Agentic Systems

- **Tool Orchestration:** Managing how agents interact with various external tools and services
- **Multi-agent Coordination:** Enabling multiple AI agents to collaborate effectively on tasks
- **Memory Management:** Systems for storing and retrieving contextual information over time
- **Safety Guardrails:** Implementing protective measures to ensure safe agent behavior
- **Human Feedback Mechanisms:** Systems that incorporate human guidance and oversight

This complexity is like moving from managing a single cricket match to orchestrating the entire IPL season - with many more moving parts and interdependencies.



The Operational Spectrum

1

MLOps

Managing traditional ML systems with focus on data pipelines, model training, and structured evaluation metrics.

2

LLMOps

Managing language model systems with emphasis on prompts, retrieval, fine-tuning, and qualitative evaluation.

3

AgenticAIOps

Managing autonomous agent systems with tool orchestration, multi-agent coordination, memory systems, and safety guardrails.

Each operational framework builds upon and extends the previous one while introducing new capabilities and challenges. This progression represents a natural evolution as AI systems become more complex and autonomous, requiring more sophisticated operational approaches.

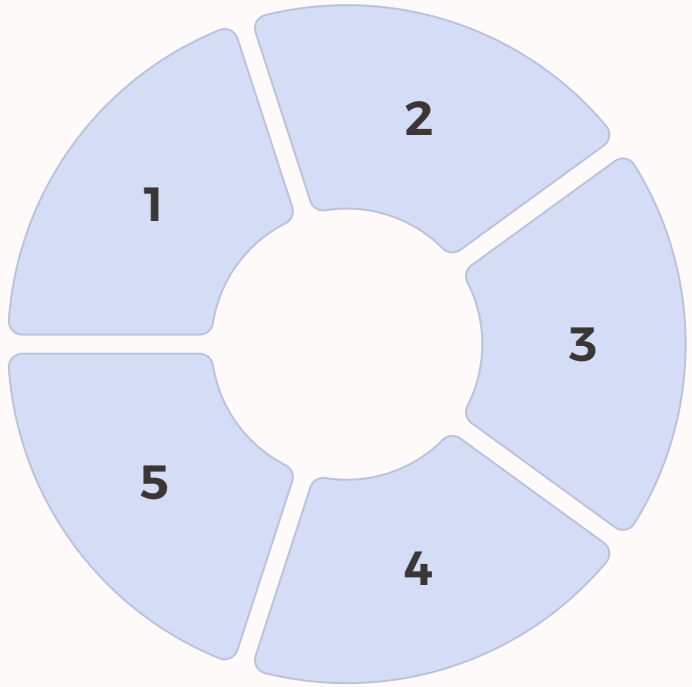
The Future Landscape

Autonomous Systems

AI systems will become increasingly self-sufficient, capable of handling complex tasks with minimal human intervention.

Standardization

Industry standards for AgenticAIOps practices will emerge, similar to how DevOps and MLOps became standardized.



Multi-modal Agents

Future agents will seamlessly integrate text, vision, and audio capabilities, interacting with the world in more human-like ways.

Human-AI Collaboration

New frameworks will emerge for effective collaboration between humans and AI systems, leveraging the strengths of both.

Specialized Platforms

Purpose-built operational platforms will develop to manage the unique requirements of agentic systems at scale.

This evolution parallels how we've moved from basic feature phones to smartphones to now anticipating ambient computing environments - each step bringing more capability and integration into our daily lives.

The Three Heroes of Our Story

Hero 1: MLOps

Superpower: Bringing order to ML chaos

Weapon of choice: CI/CD pipelines, versioning, monitoring

Mission: Bridge the gap between data science and engineering

Hero 2: LLMOps

Superpower: Harnessing the power of language models

Weapon of choice: Prompt engineering, vector stores, evaluation

Mission: Make large language models reliable and governable

Hero 3: AgenticAIOps

Superpower: Autonomous problem-solving

Weapon of choice: Planning, tools, memory systems

Mission: Create AI systems that can operate independently

Our journey will follow these three heroes and their quest for operational excellence.

Key Takeaways

1

Evolutionary Journey

Machine learning has evolved from academic theory to transformative technology impacting virtually every industry.

2

Operational Challenges

Each evolutionary stage (ML → LLM → Agents) brings new operational challenges requiring innovative solutions.

3

Framework Evolution

MLOps → LLMOps → AgenticAIOps represents the parallel evolution of operational frameworks to manage increasing complexity.

4

Growing Autonomy

The complexity and autonomy of AI systems continues to increase, demanding more sophisticated management approaches.

Mastering these operational frameworks is essential for successful AI implementation in any organization. As systems become more powerful, the operational practices become even more critical to ensure reliability, safety, and effectiveness.

Thank You!

We hope this presentation has provided valuable insights into the evolution of AI systems and their operational frameworks. The School of DevOps is committed to helping professionals master these critical skills for the future.

Let's begin our journey into mastering MLOps, LLMOps, and AgenticAIOps

> "The best way to predict the future is to create it."

We invite you to join us in creating the future of AI operations, building the frameworks and practices that will enable the next generation of intelligent systems.