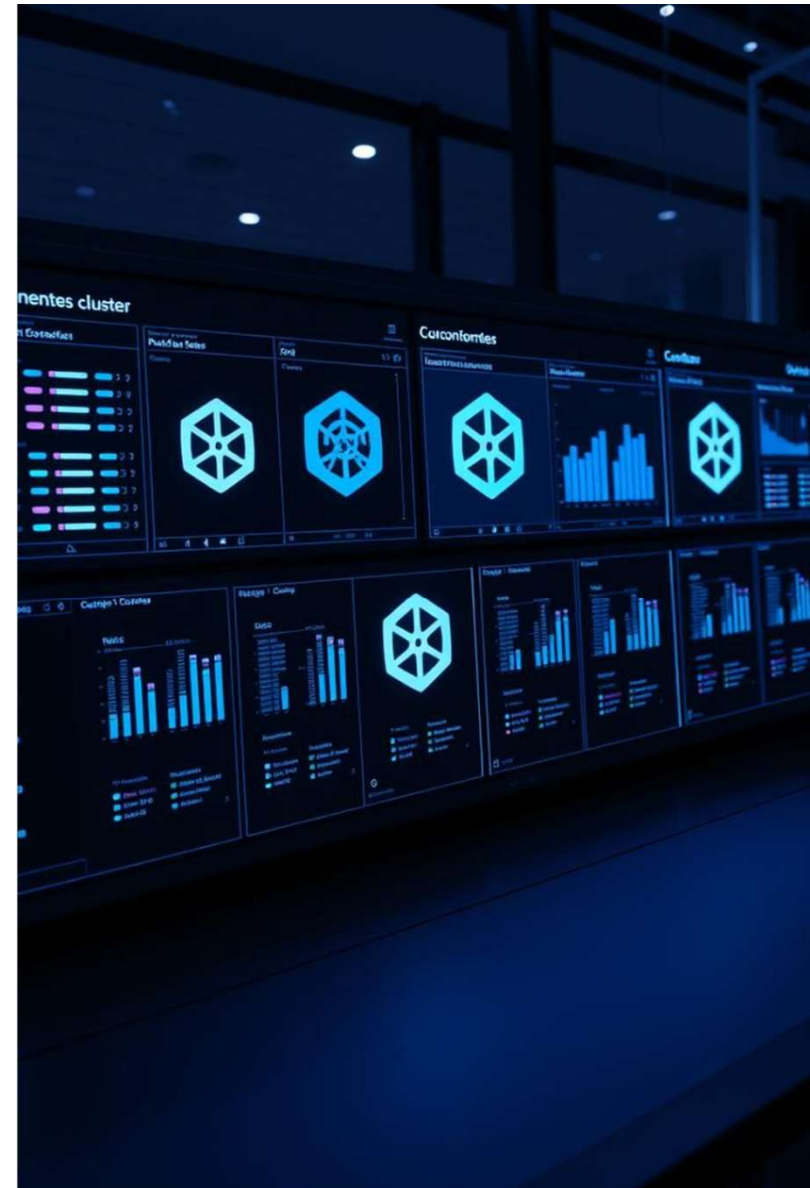


Phase 1: AI/ML Model for Predicting Kubernetes Issues

This project focuses on proactive issue detection within Kubernetes clusters. The goal is to create an AI/ML model for predicting potential failures. We aim for reduced downtime, improved resource use, and enhanced stability.





Problem Statement: Kubernetes Failure Prediction

Kubernetes failures cause downtime and degrade performance. Manual fixes are slow and reactive. Current tools lack predictive power. The average power. The average downtime is 4 hours monthly, costing \$50,000.



Downtime Impact

Application downtime and performance performance loss



Manual Monitoring

Reactive and time-consuming problem solving



Tool Limitations

Lack of predictive capabilities

AI/ML Solution: Predictive Model Overview

We propose an AI/ML model using cluster metrics. The model will be trained on historical and real-time data. It may use LSTM or time series forecasting.



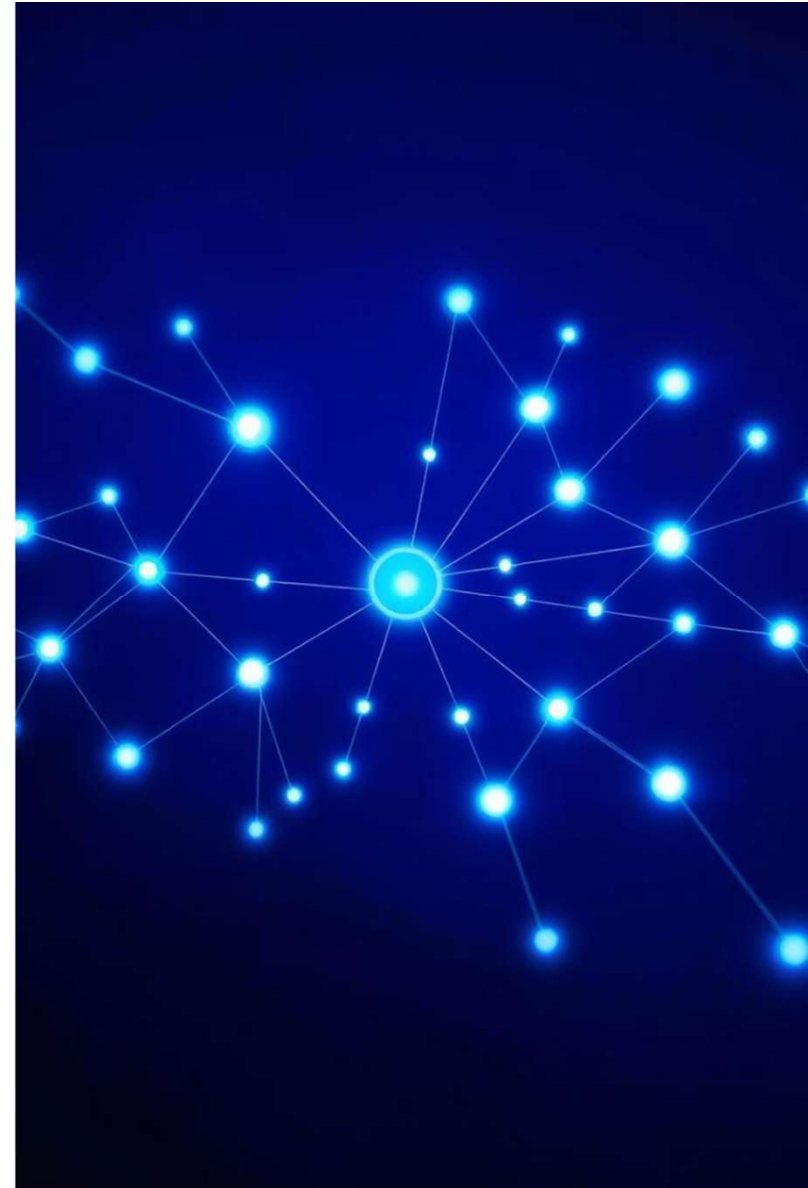
Model Architecture



Key Metrics



Prediction



Data Collection and Preparation

Data comes from Kubernetes API, Prometheus, Grafana, and logs. We will ingest 500GB monthly. Data will be cleaned, transformed, and engineered. and engineered.

Data Sources

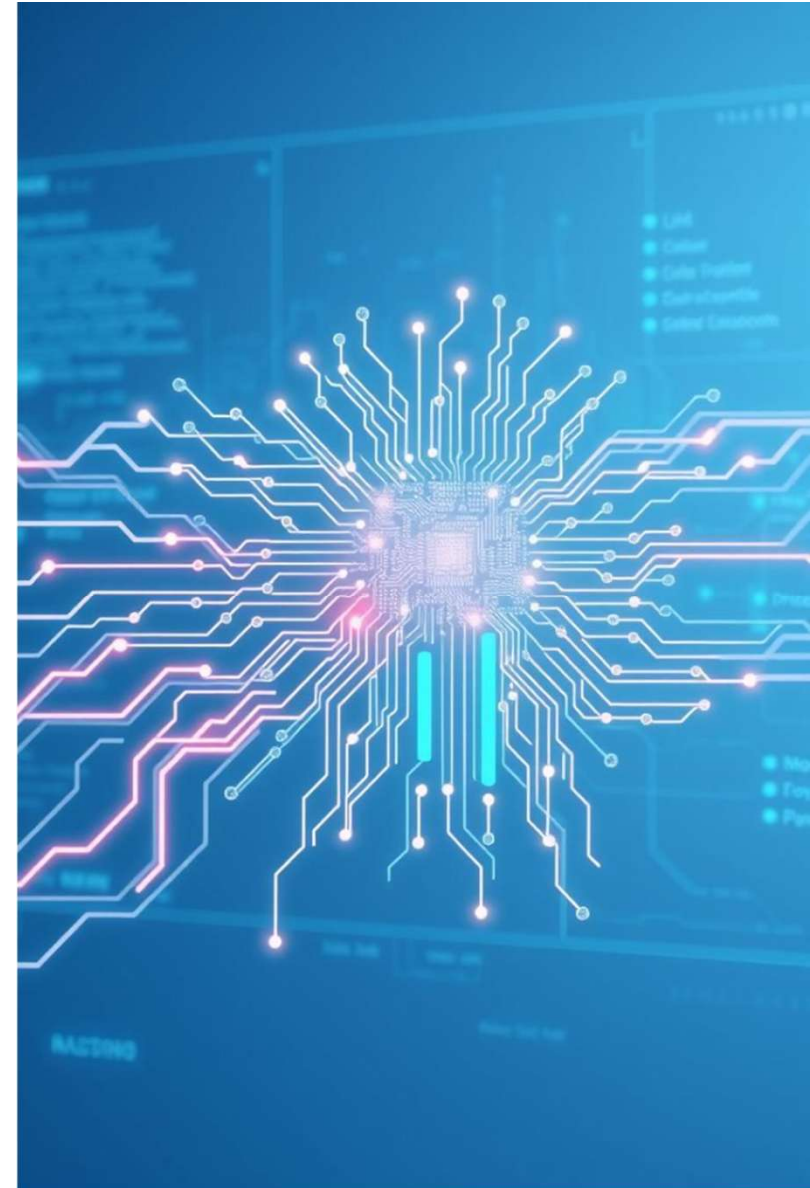
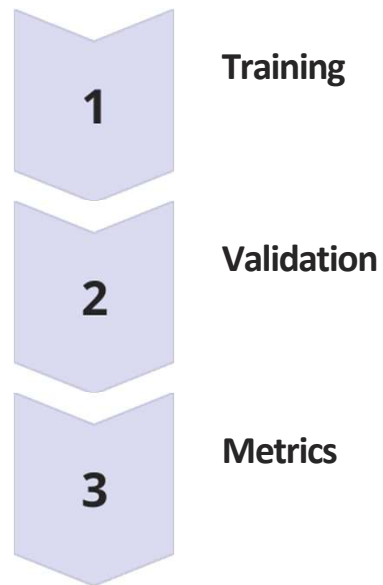
- Kubernetes API
- Prometheus
- Grafana

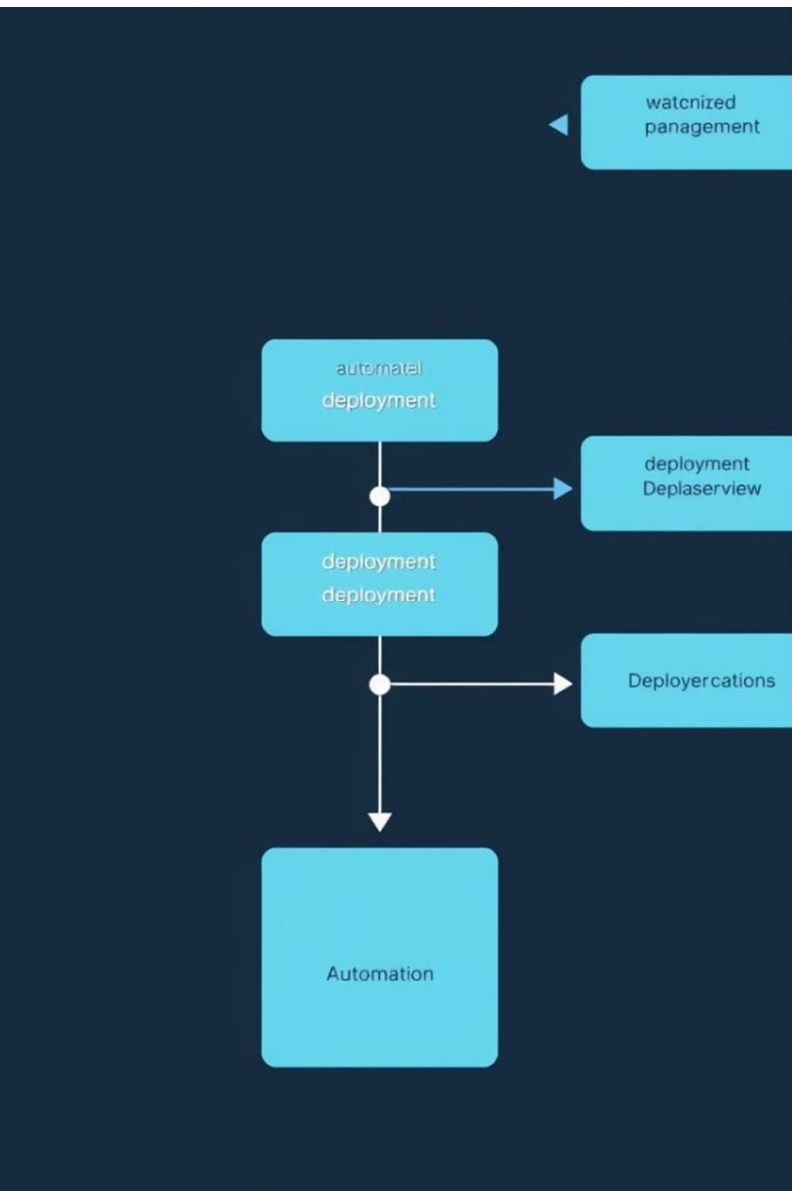
Tools

- Python (Pandas, NumPy)
- Apache Spark

Model Training and Evaluation

The training set is 80% of data from Jan 2023 to June 2024. 20% of historical data will be data will be used for validation. Performance will be measured using standard metrics.





Deployment and Integration

The model will be containerized with Docker and Kubernetes. We will integrate the model with monitoring and alerting systems. Alerts will be sent to Slack/PagerDuty.

Containerized

Using Docker and Kubernetes

Integrated

Monitoring dashboards & alerts
alerts

Alerting

Via Slack/PagerDuty

Expected Benefits and Impact

We expect a 50% downtime reduction through proactive fixes. Resources will be better allocated. This will cut costs and improve system stability.

- 1 Reduced Downtime
- 2 Resource Use
- 3 Cost Savings
- 4 System Stability

Next Steps and Future Considerations

Once issues are predicted, the next step is to automate or recommend actions for remediation. The challenge in Phase 2 is to create an agent or create an agent or system capable of responding to these predicted issues by suggesting or implementing actions to mitigate potential failures in the potential failures in the Kubernetes cluster.

