Cloud-Native MLOps Framework

Data Fest 2021

Artem Koval

Big Data and Machine Learning Practice Lead at ClearScale

About Speaker

- Hey all!
- Name: Artem Koval
- Position: Big Data and Machine Learning Practice Lead
- Company: ClearScale

Agenda

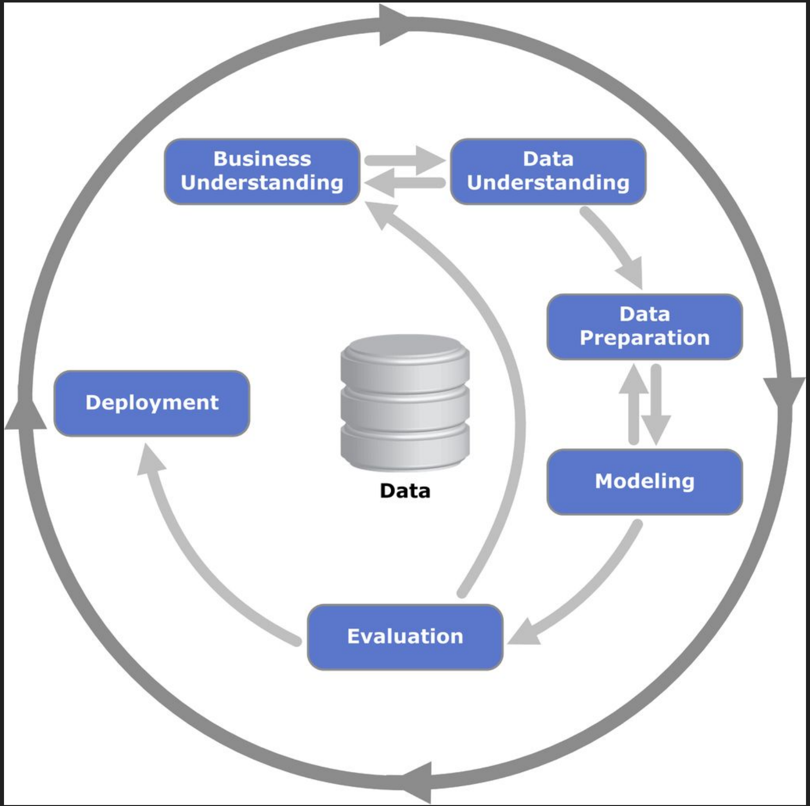
- What is modern MLOps
- Why the shift towards Human-Centered AI
- Fairness, Explainability, Model Monitoring
- Human Augmented AI
- How much MLOps do you need in your organization
- The future

What is MLOps?

- https://en.wikipedia.org/wiki/MLOps
- https://ml-ops.org/
- A process of deploying ML models in CI/CD manner into production, establishing *model monitoring*, *explainability*, *fairness*, and providing tools for *human intervention*

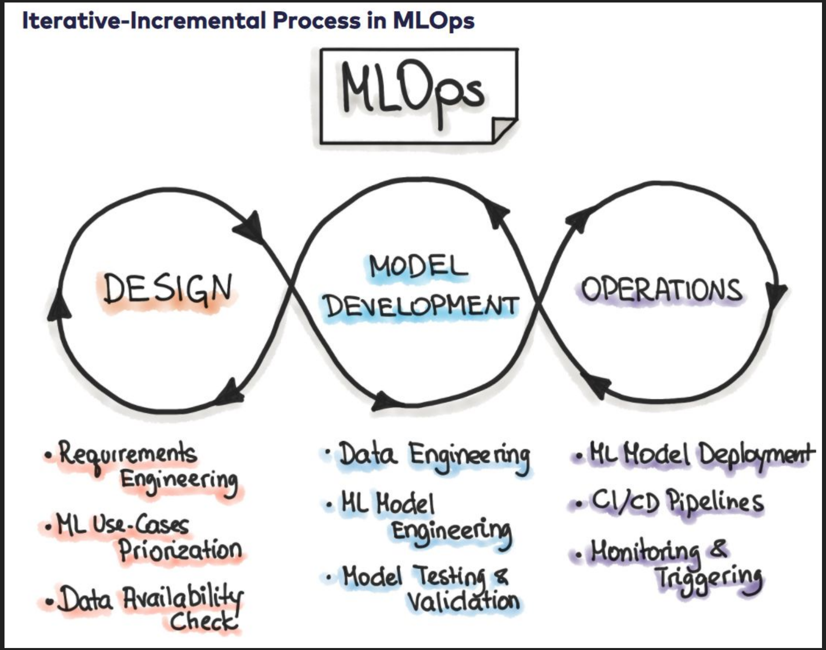
CRISP-DM

- https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining
- Too generic



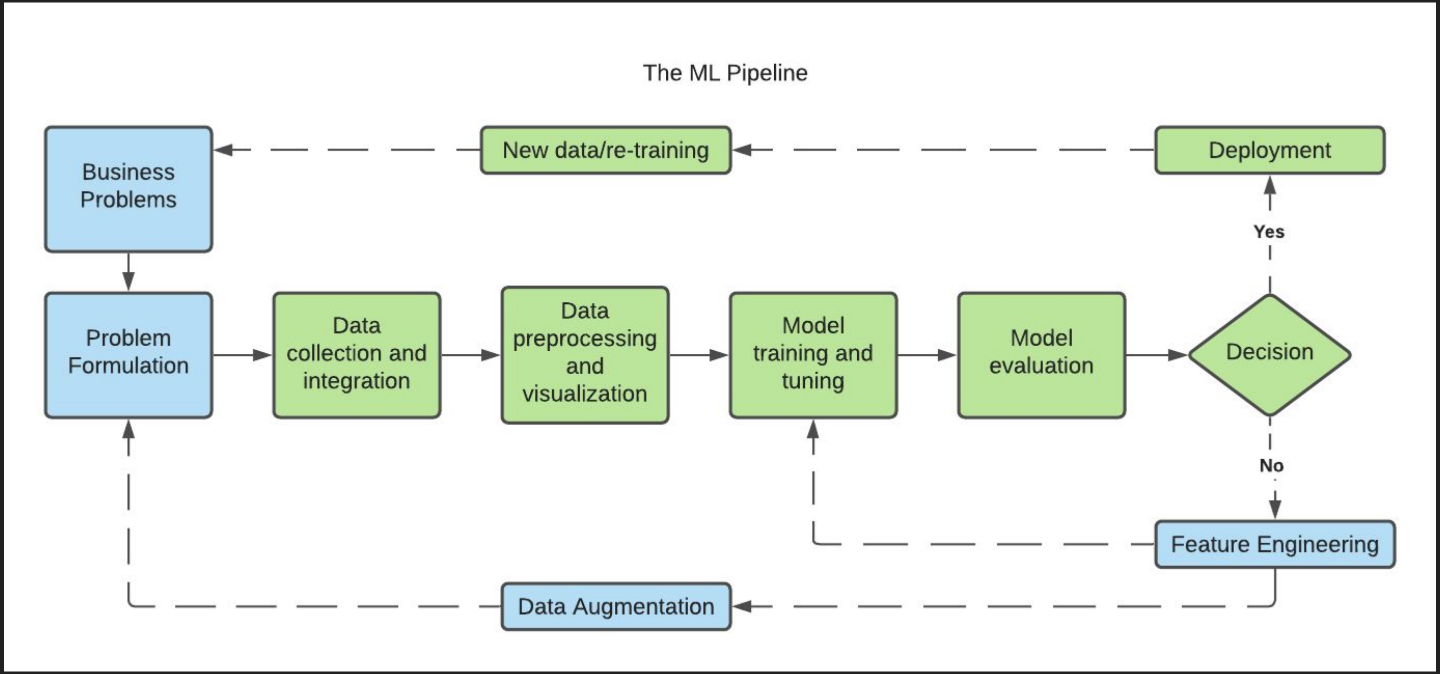
Why a Framework

- A need for the end-to-end solution, from the data ingestion to the model monitoring, data labeling, algorithms explainability



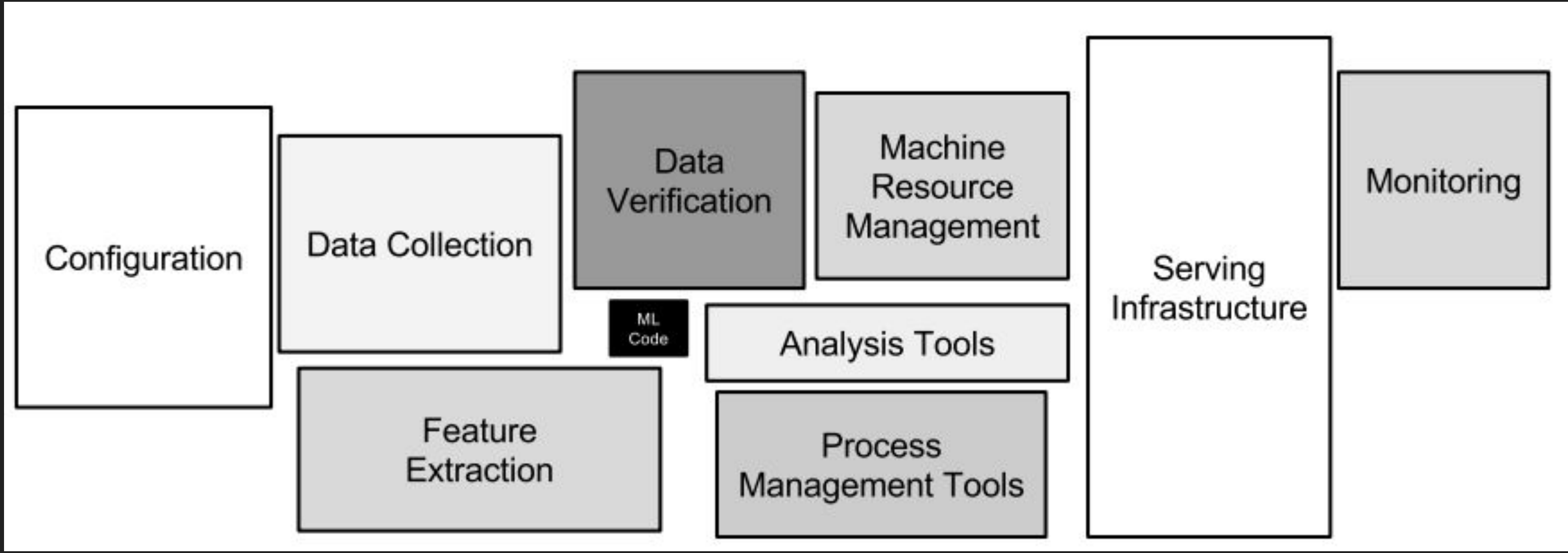
An elegant weapon for a more civilized age (c)

- Your father's ML Pipeline

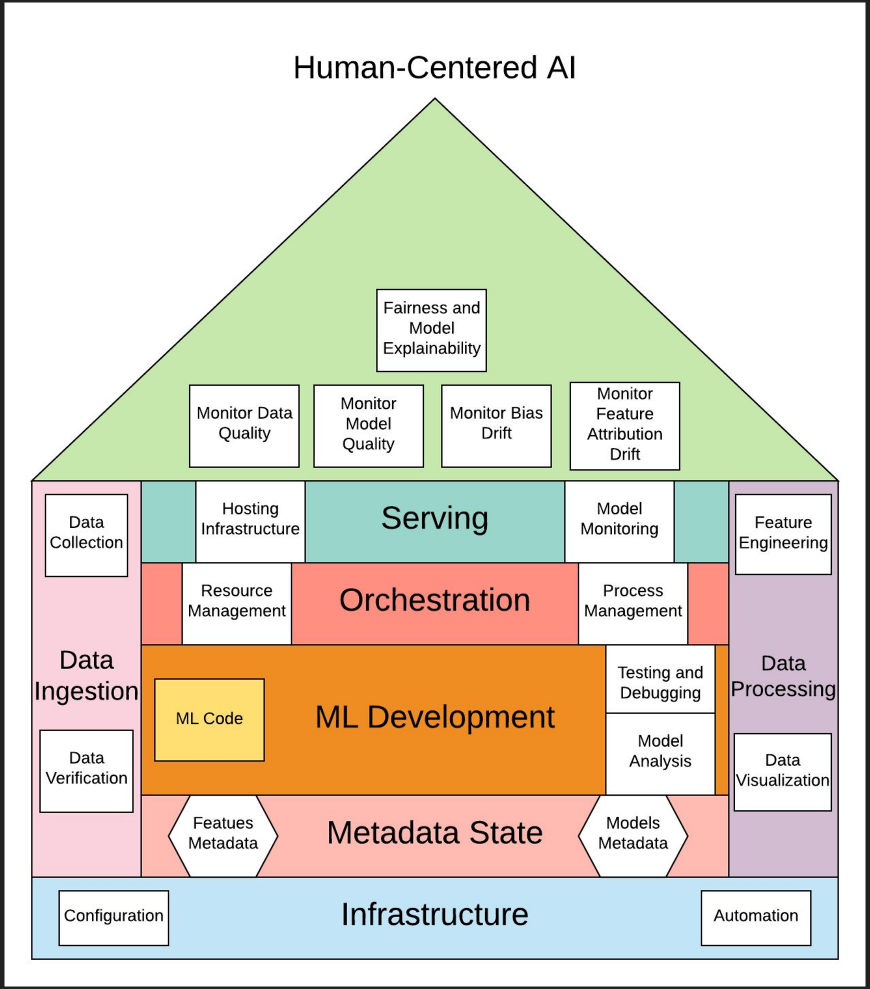


ML has Technical Debt?

- [Hidden Debt in Machine Learning Systems]()



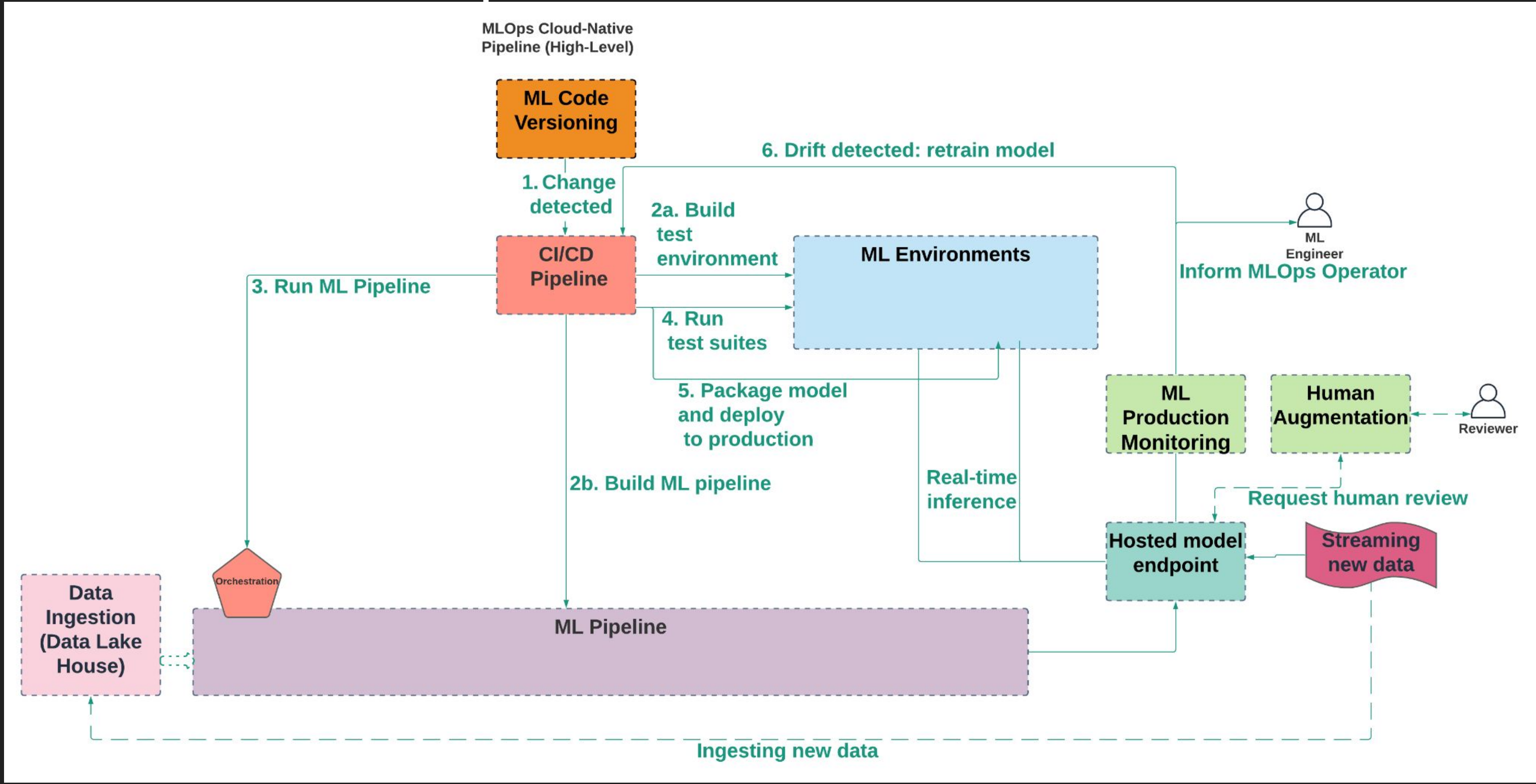
The House of MLOps



Human-Centered AI

- https://hai.stanford.edu/
- https://plato.stanford.edu/entries/ethics-ai/
- https://ethical.institute/
- Humans must control AI end-to-end solutions

Cloud-Native MLOps



Modern MLOps Framework Drivers

- Not only CI/CD and ML code anymore
- *Fairness and Explainability*
- *Observability (Monitoring)*
- Scalability (Training and Inference)
- *Data Labeling*
- A/B Testing, Acceptance Testing
- *Human Review*
- Legacy Migration
- Multi-tenant Multi-model

Fairness

- https://github.com/slundberg/shap
- Regulatory requirements
- Business trust

Explainability

- https://github.com/Trusted-AI/AIF360
- No bias in data, no bias in inference (gender, racial, religious, ageism etc.)
- Fairness and Explainability by Design as a Process

Monitor Data Quality

- Monitors ML models in production and notifies when data quality issue arise
- Enable data capture (inference input & output, historical data)
- Create a baseline (https://github.com/awslabs/deequ)
- Define and schedule data quality monitoring jobs
- View data quality metrics/violations
- Integrate data quality monitoring with a Notification Service
- Interpret the results of a monitoring job
- Visualize results

Data Quality Violations/Metrics

- data_type_check
- completeness_check
- baseline_drift_check
- missing_column_check
- extra_column_check
- categorical_values_check
- Max, Min, Sum, SampleCount, Average, Distribution, StdDev, Mean
- ...

Monitor Model Quality

- Monitors the performance of a model by comparing the live predictions with the actual ground truth labels
- Enable Data Capture
- Create a baseline
- Define and schedule model quality monitoring jobs
- Ingest ground truth labels that model monitor merges with captured prediction data from real-time/batch inference endpoints
- Integrate model quality monitoring with a Notification Service
- Interpret the results of a monitoring job
- Visualize the results

Model Quality Metrics

- Regression: mae, mse, rmse, r2, ...
- Binary classification: confusion matrix, recall, precision, accuracy, recall_best_constant_classifier, precision_best_constant_classifier, accuracy_best_constant_classifier, true_positive_rate, …
- Multiclass classification: weighted_recall, weighted_f1, weighted_f2_best_constant_classifier, ...

Monitor Bias Drift

- https://github.com/aws/amazon-sagemaker-clarify
- https://github.com/anodot/MLWatcher
- Training data differs from the live inference data
- Pre-training/post-training/common

Bias Metrics

- Class Imbalance (CI)
- Difference in Positive Proportions in Labels (DPL)
- Kullback-Liebler Divergence (KL)
- Jensen-Shannon Divergence (JS)
- Total Variation Distance (TVD)
- Kolmogorov-Smirnov Distance (KS)
- Conditional Demographic Disparity in Labels (CDDL)
- Difference in Conditional Outcomes (DCO)
- Difference in Label Rates (DLR)
- ...

Monitor Feature Attribution Drift

- https://github.com/slundberg/shap
- A drift in the distribution of live data for models in production can result in a corresponding drift in the feature attribution values

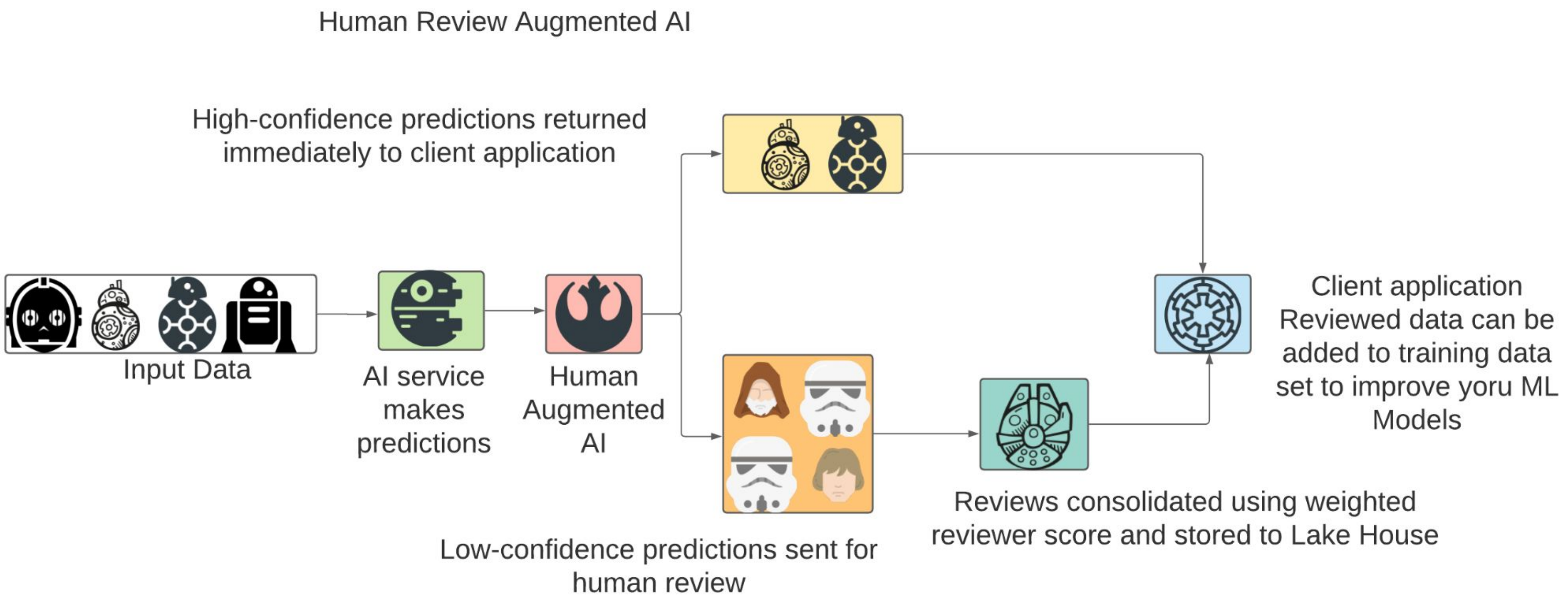
Feature Attribution Drift Monitoring Methods

- LIME
- Shapley sampling values
- DeepLIFT
- QII
- Layer-wise relevance propagation
- Shapley regression values
- Tree interpreter

Human Augmented AI Drivers

- Need human oversight to ensure accuracy with sensitive data (healthcare, finance)
- Implement human review of ML predictions
- Integrate human oversight with any application
- Flexibility to work with inside and outside reviewers
- Easy instructions for reviewers
- Workflows to simplify the human review process
- Improve results with multiple reviews

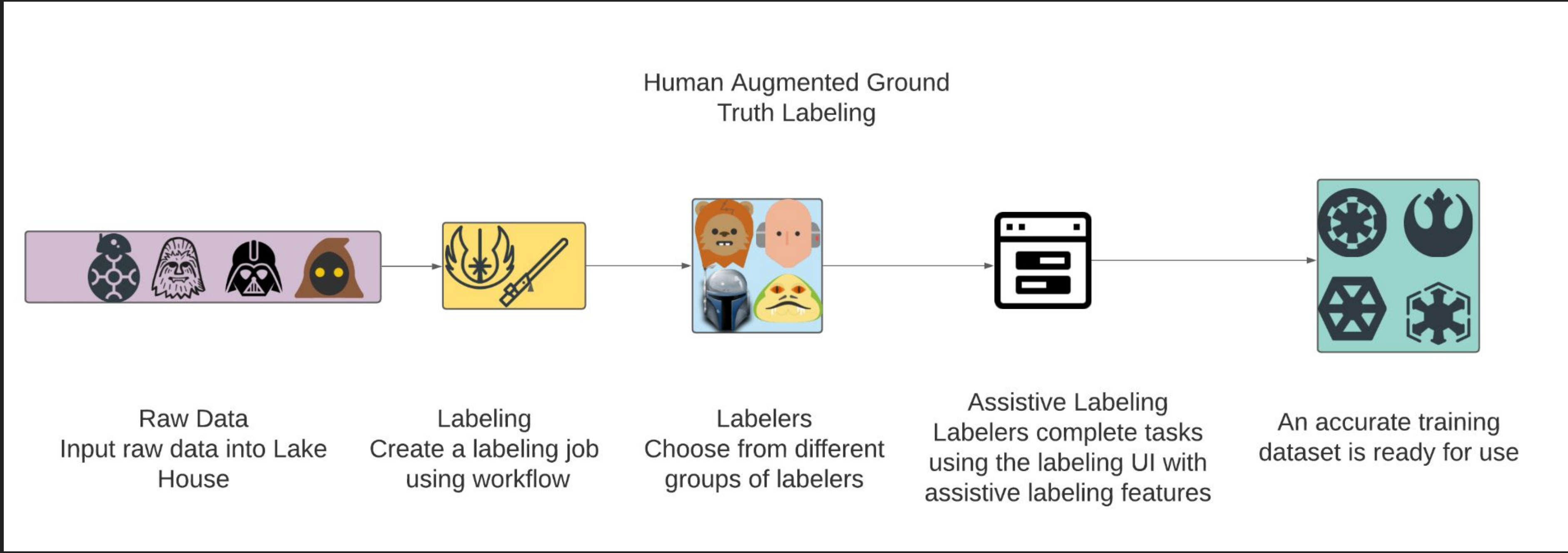
Human Augmented AI



Human Augmented Ground Truth Labeling Drivers

- Improve data label accuracy
- Easy to use (automatic snapping, image denoising, pre-selecting object contour, etc.)
- Reduce costs
- Distribute workload over varying workforce

Human Augmented Ground Truth Labeling



MLOps Levels

- Lightweight MLOps
- Cloud-Native Greenfield SMB MLOps
- Enterprise MLOps
- Human-Centered AI

Lightweight MLOps Capacity

- One-person data science shop
- Small number of models (1-3)
- ML system is a greenfield
- Need to run time-critical demo for a small audience
- Models are custom, lightweight, don’t require compute-intensive model training/HPO
- Low traffic is expected

Lightweight MLOps Solution Blueprint

- Convert models with TensorFlow Lite (or other framework-specific strip-down)
- Write a simple API microservice (e.g., Flask)
- Deploy as is, with no containerization, as a web app calling ML layer
- Use CPU-based commodity cloud instances
- *Minimal model monitoring to at least capture drift*
- Data analysis, feature engineering, orchestration, CI/CD, acceptance/AB testing might be omitted
- Bootstrapping is highly needed to organize the process (e.g. Metaflow)

Cloud-Native SMB MLOps Capacity

- Have engineering resource
- Custom proprietary algorithms
- ML system is a greenfield
- Model development requires advance comput for training/HPO and inference
- Multi-model, multi tenant setup is needed

Cloud-Native SMB MLOps Solution Blueprint

- Containerize models (Docker)
- Utilize framework/cloud vendor specific HPO approaches
- Use GPU-based commodity cloud instances when needed
- Use cloud vendor specific elastic inference approaches
- Abstract and isolate data analysis, feature engineering, model training and other steps
- Orchestrate with Apache Airflow or similar technology agnostic tools
- Ensure multi-tenancy by logical isolation of ML Workflows
- *Implement model monitoring at least partially (bias drift, model quality)*

Enterprise MLOps Capacity

- Have legacy ML system with a lot of microservices, models, orchestration flows
- Have highly custom proprietary libraries requiring complex make
- Have advanced tenant isolation requirements
- Have a lot of models (>10)
- Have advanced needs for a large data science team to collaborate

Enterprise MLOps Blueprint

- Serve dockerized models with Kubeflow in a Kubernetes cluster
- Use Kubeflow tenancy isolation
- Use KFServing to deploy multiple variants of multiple models
- Use Katib for HPO
- Use Prometheus + Grafana, ELK for the full model monitoring, consuming metrics with the open-source empowered microservices (SHAP, etc.)
- *Implement advanced production acceptance testing (e.g., Differential Testing, Shadow Deployments, Integration Testing etc.)*
- *Built custom human augmented Review/Labeling tools*

Human-Centered AI Blueprint

- Can be added at any size/project configuration
- Ideally should be incorporated as a process touch all steps (data analysis, training, deployment, monitoring)
- *Remember: the moment your model is deployed to production it's already obsolete. Build with the CI/CD and human operations review in mind*

The future

- Privacy-Preserving Machine Learning (differential, compressive, etc.)
- Models interpretability (global, local, saliency mapping, semantic similarity etc.)
- Model Monitoring in AutoML (AutoKeras/Keras Tuner + SHAP, etc.)
- Measuring human augmentation (uncertainty/diversity sampling, active learning, quality control, annotation/augmentation quality metrics, etc.)

Thanks everyone!

Questions?

The End

You could reach me via mail@artemkoval.com or LinkedIn

May the MLOps be with you!

Extra Resources

- https://github.com/EthicalML/awesome-production-machine-learning
- https://aws.amazon.com/solutions/implementations/aws-mlops-framework/
- https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning
- https://azure.microsoft.com/en-us/services/machine-learning/mlops/
- https://docs.aws.amazon.com/sagemaker/latest/dg/whatis.html
- https://github.com/visenger/awesome-mlops#mlops-books