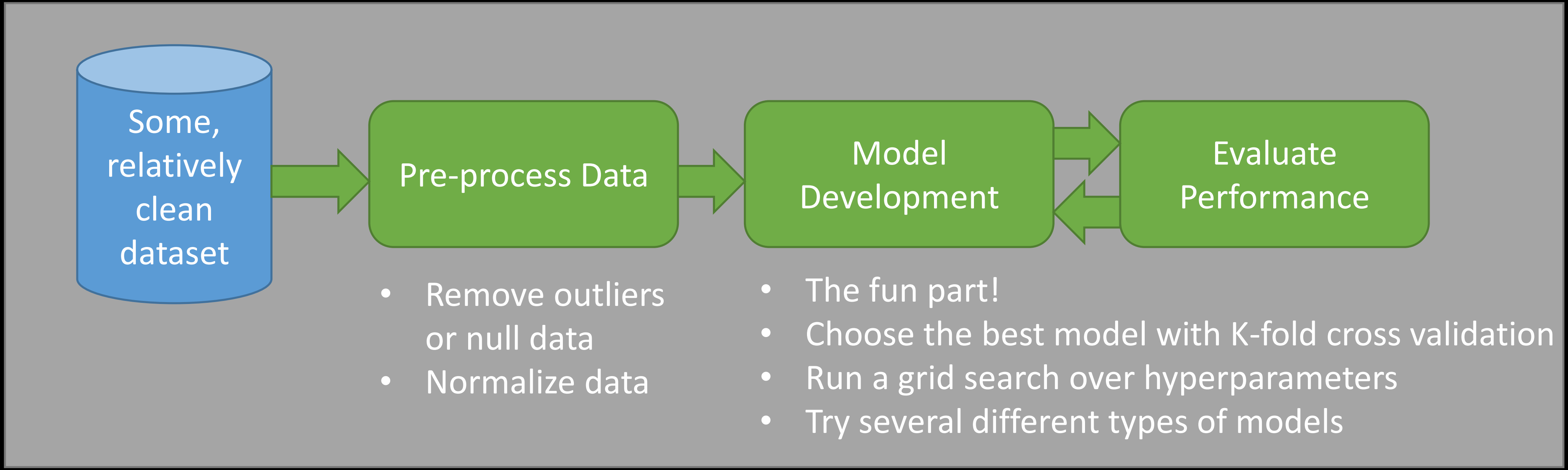
# The Challenges of Deploying AI Models

**Nathan Bosch**

# Key Takeaways

1. Models always perform worse in production than in development
2. Deployment standards are very young, we’re not mature yet and competence is missing
3. A successful ML deployment consists of ~20% model development

# Example ML Project




- Compile process, data exploration, training regiment, and final model performance into a report
  - E.g., a Jupyter notebook
- This is a common procedure in many company internships as well, although you are not guaranteed a clean dataset

# Next Steps

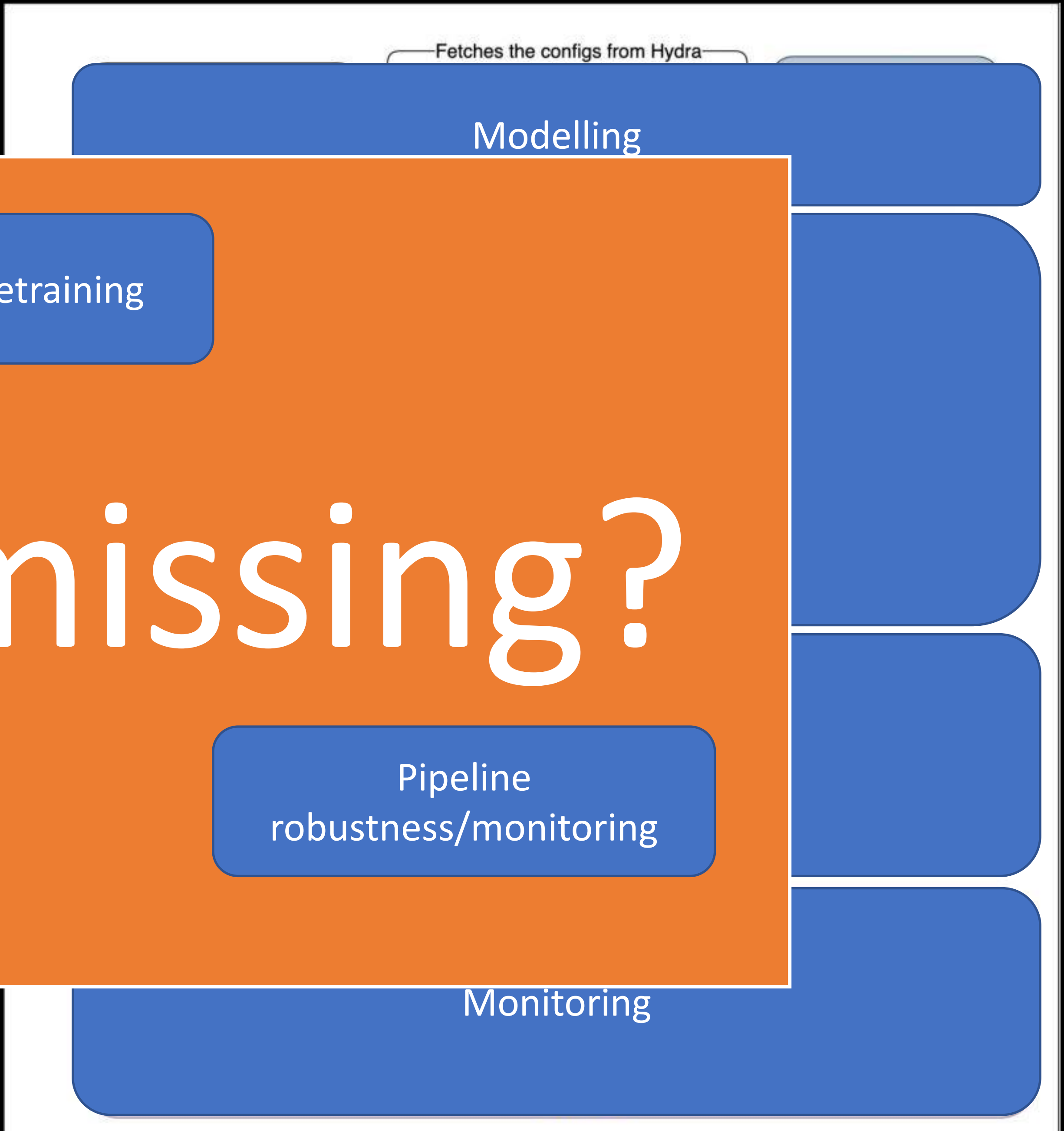
- Let’s say the model performance is great and we want to deploy the model in production. How should we do this?
- We’d need:
  - A service users can interact with
  - The model needs to be hosted somewhere
  - We want to monitor model performance
  - We’d need to be robust to failures
  - We might need to handle multiple requests at the same time
  - And more…

# Challenges

- Industry Issues
  - Lack of competence
  - Lack of standardization
- Technical Issues
  - Technical debt – the challenges with data
  - Data drift
  - Monitoring & alarms
  - Retraining poorly performing models
  - Etc…

# Industry Challenges

- Recent work in MLOps has addressed many modern concerns of
- The compet
- Standardiza

MLOps Basics Flow

Automated retraining

# What's missing?

Data storage/pipelines

Pipeline robustness/monitoring

# Technical Issues

- Technical debt
  - Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., ... & Dennison, D. (2015). Hidden technical debt in machine learning systems. *Advances in neural information processing systems*, *28*, 2503-2511.
  - **Key point:** Data Dependencies Cost More than Code Dependencies
  - Feedback loops
- Data drift:
  - A model trained on recent trends will perform very poorly on new data
  - Automated retraining needs to be set up for situations like this
    - What do we do if the retrained models lose performance?

# Technical Issues

- Performance of production models is always worse…
  - Outliers that are removed during training now contribute to either poor prediction quality or poor data coverage
  - Bias during model development (even with cross-validation) is very common
- Lack of interpretability
  - Poorly performing models which provide no explanation for their prediction leads to a lack of trust
- Scalability
  - Scenario: I need to query my 2GB language model 1000 times per second ☺
  - How can we achieve this? Often times simpler models are the easiest answer
- Baier, L., Jöhren, F., & Seebacher, S. (2019). Challenges in the deployment and operation of machine learning in practice.

# Unseen Difficulties in Machine Learning

- A high performing model does not indicate a valuable model
  - This is often lost in translation. Are you really solving a problem that people find valuable?
    - If so, what KPIs can you identify and optimize for?
  - Requires constant feedback with customers throughout development
- Designing user interaction with a machine learning model is not trivial
  - How should we present model output?
  - If requests are made implicitly (e.g., when loading a webpage), how is this handled on the front end?
  - What sort of language do you use?
- Model security

# Conclusions

- Modelling is only a small part of machine learning solutions
- Existing industry standards for ML deployment are very young
    - Very high competence required
- There are numerous technical issues to account for when deploying ML models
    - Data drift
    - Monitoring
    - Interpretability