DADS7305: MLOPs

Northeastern University

Instructor: Ramin Mohammadi

September 7, 2025

These materials have been prepared and sourced for the course **MLOPs** at Northeastern University. Every effort has been made to provide proper citations and credit for all referenced works.

If you believe any material has been inadequately cited or requires correction, please contact me at:

`r.mohammadi@northeastern.edu`

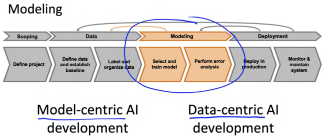
Thank you for your understanding and collaboration.

Select and train model

Modeling overview

Model-centric vs Data-centric AI Development

- **Model-centric AI development**
  - Focus on selecting and training models
  - Iterative error analysis to improve accuracy
- **Data-centric AI development**
  - Emphasis on improving data quality
  - Labeling, organizing, and refining datasets
- Both approaches interact across the ML lifecycle stages



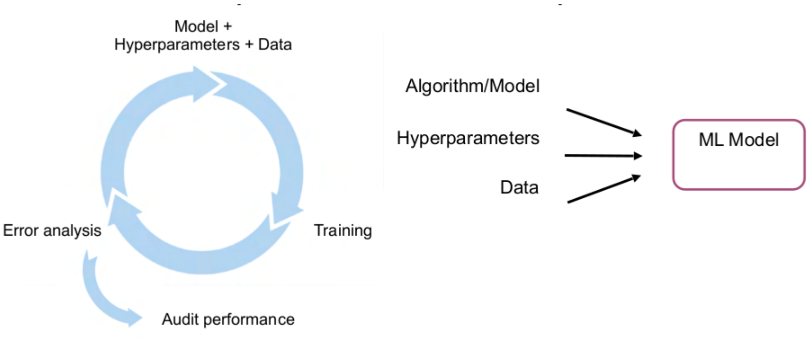
Select and train model

Key challenges

AI system = Code + Data

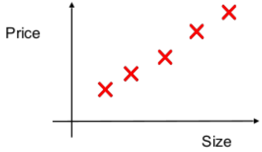
(algorithm/model)

Model development is an iterative process



Challenges in model development

- 1. Doing well on training set (usually measured by average training error).
- 2. Doing well on dev/test sets.
- 3. Doing well on business metrics/project goals.



Select and train model

Why low average test error isn't good enough

Performance on disproportionately important examples

Web Search example

"Apple pie recipe" "Latest movies"
"Wireless data plan" "Diwali festival"

Informational and Transactional queries

"Stanford" "Reddit" "Youtube"

Navigational queries

Performance on key slices of the dataset

Example: ML for loan approval

- Make sure not to discriminate by ethnicity, gender, location, language or other
- protected attributes.

Example: Product recommendations from retailers

- Be careful to treat fairly all major user, retailer, and product categories.

Rare classes

Skewed data distribution

99% negative 1% positive

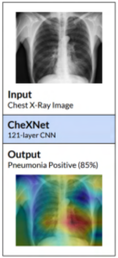
print("0")

Accuracy in rare classes

Condition	Performance
Effusion	0.901
Edema	0.924
Mass	0.909
Hernia	0.851

10,000 → Effusion

~100 → Hernia



Unfortunate conversation in many companies

MLE: "I did well on the test set!"

Product Owner: "But this doesn't work for my application"

MLE: "But... I did well on the test set!"

Select and train model

Establish a baseline

Establishing a baseline level of performance

Speech recognition example:

Type	Accuracy	Human level performance	HLP
Clear Speech	94%	95%	1%
Car Noise	89%	93%	4%
People Noise	87%	89%	2%
Low Bandwidth	70%	70%	~0%

Structured and unstructured data

Unstructured data

Image

Audio

Text

This restaurant was great!

Structured Data

User Id	Purchase	Number	Price
3421	Blue shirt	5	$20
612	Brown shoes	1	$35

Price	Product
3421	Red skirt

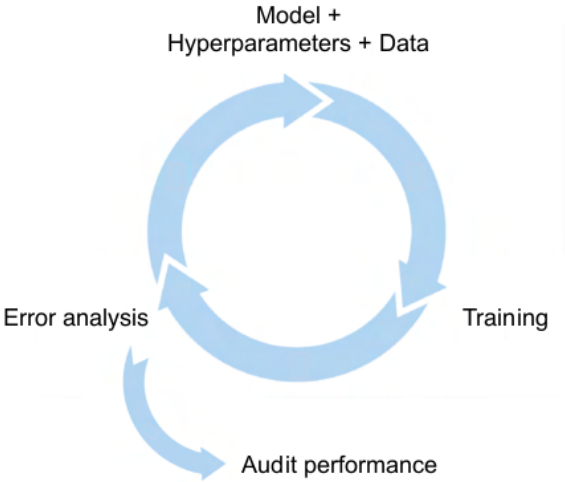
Ways to establish a baseline

- Baseline gives an estimate of the irreducible error / Bayes error and indicates
- what might be possible."Human level performance (HLP)
- "Older system"
- Literature search for state-of-the-art/open source

Select and train model

Tips for getting started

ML is an iterative process



Getting started on modeling

- Literature search to see what's possible.
- Find open-source implementations if available.
- A reasonable algorithm with good data will often outperform a great algorithm with not so good data.

Deployment constraints when picking a model

Should you take into account deployment constraints when picking a model?

- Yes, if baseline is already established and goal is to build and deploy.
- No, if purpose is to establish a baseline and determine what is possible and might be worth pursuing.

Sanity-check for code and algorithm

Try to overfit a small training dataset before training on a large one.

- "Example #1: Speech recognition
- "Example #2: Image segmentation
- "Example #3: Image classification

Error analysis and performance auditing

Error analysis example

Speech recognition example

Example	Label	Prediction	Car Noise	People Noise	Low Bandwidth
1	"Stir fried lettuce recipe"	"Stir fry lettuce recipe"	✓		
2	"Sweetened coffee"	"Swedish coffee"		✓	✓
3	"Sail away song"	"Sell away some"		✓	
4	"Let's catch up"	"Let's ketchup"	✓	✓	✓

Iterative process of error analysis

Visual inspection:

- Specific class labels (scratch, dent, etc.)
- Image properties (blurry, dark background, light background, reflection….)
- Other meta-data: phone model, factory

Product recommendations:

- User demographics
- Product features

Useful metrics for each tag

- "What fraction of errors has that tag?
- "Of all data with that tag, what fraction is misclassified?
- "What fraction of all the data has that tag?
- "How much room of improvement is there in that tag?

Error analysis and performance auditing

Prioritizing what to work on

Prioritizing what to work on

Type	Accuracy	Human level performance	Gap to HLP	% of data
Clean Speech	94%	95%	1%	60% → 0.6%
Car Noise	89%	93%	4%	4% → 0.16%
People Noise	87%	89%	2%	30% → 0.6%
Low Bandwidth	70%	70%	0%	6% → ~0%

Prioritizing what to work on

Decide on most important categories to work on based on:

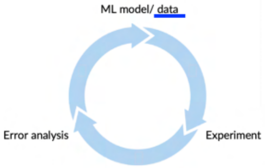
- "How much room for improvement there is.
- "How frequently that category appears.
- "How easy is to improve accuracy in that category.
- "How important it is to improve in that category.

Adding data

For categories you want to prioritize:

- "Collect more data (or improve label accuracy)
- "Use data augmentation to get more data

Type	Accuracy	Human level performance	Gap to HLP	% of data
Clean Speech	94%	95%	1%	60%
Car Noise	84%	93%	4%	40%
People Noise	87%	84%	2%	30%
Low Bandwidth	70%	70%	0%	6%



Error analysis and performance auditing

Skewed datasets

Examples of skewed datasets

Manufacturing example

99.7% no defect $y=0$

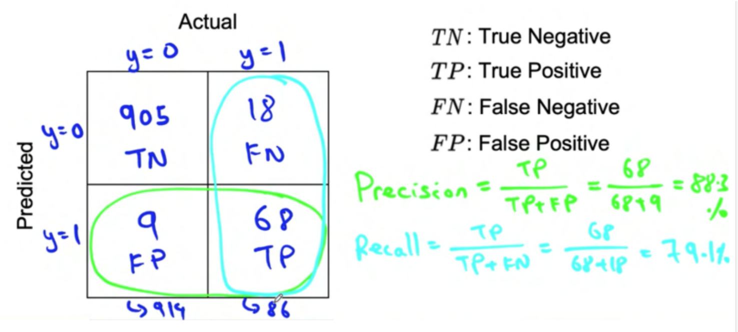
0.3% defect $y=1$

print("0")
99.7%

Medical Diagnosis example**:** 98% of patients don't have a disease

Speech Recognition example**:** In wake word detection, 96.7% of the time wake word doesn't occur

Confusion matrix: precision and recall



What happens with print(“0”)?

Predicted \ Actual	y = 0	y = 1
y = 0	914	86 FN
y = 1	0 FP	0 TP

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{0}{0+0}$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{0}{0+86} = 0\%$$

Combining precision and recall – F1 score

	Precision (P)	Recall (R)	F_1
Model 1	88.3	79.1	83.4 %
Model 2	97.0	7.3	13.6 %

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}}$$

Multi-class metrics

Classes: Scratch, Dent, Pit mark, Discoloration

Defect Type	Precision	Recall	F_1
Scratch	82.1%	99.2%	89.8%
Dent	92.1%	99.5%	95.7%
Pit mark	85.3%	98.7%	91.5%
Discoloration	72.1%	97%	82.7%

Error analysis and performance auditing

Performance auditing

Auditing framework

Check for accuracy, fairness and bias.

- 1. Brainstorm the ways the system might go wrong.
  - Prevalence of specific errors/outputs (e.g., FP, FN).
  - Performance on rare classes.
  - Performance on subsets of data (e.g., ethnicity, gender).
- 2. Establish metrics to assess performance against these issues on appropriate slices of data.
- 3. Get business/product owner buy-in.

Speech recognition example

- 1. Brainstorm the ways the system might go wrong.
  - Accuracy on different genders and ethnicities.
  - Accuracy on different devices.
  - Prevalence of rude mistranscriptions.

- 2. Establish metrics to assess performance against these issues on appropriate slices of data.
  - Mean accuracy for different genders and major accents.
  - Mean accuracy on different devices.
  - Check for prevalence of offensive words in the output.

Data iteration

Data-centric AI development

Data-centric AI development

Model-centric view	Data-centric view
Collect what data you can, and develop a model good enough to deal with the noise in the data.	The consistency of the data is paramount. Use tools to improve the data quality; this will allow multiple models to do well.
Hold the data fixed and iteratively improve the code/ model.	*Hold the code fixed and iteratively improve the data.*

Neural Architecture Search

Hyperparameter tuning

Neural Architecture Search

- Neural architecture search (NAS) is is a technique for automating the design of artificial neural networks
- It helps finding the optimal architecture
- This is a search over a huge space
- AutoML is an algorithm to automate this search

Types of parameters in ML Models

- Trainable parameters:
  - Learned by the algorithm during training
  - e.g. weights of a neural network
- Hyperparameters:
  - set before launching the learning process
  - not updated in each training step
  - e.g: learning rate or the number of units in a dense layer

Manual hyperparameter tuning is not scalable

- Hyperparameters can be numerous even for small models
- e.g shallow DNN:
  - Architecture choices
  - activation functions
  - Weight initialization strategy
  - Optimization hyperparameters such as learning rate, stop condition
- Tuning them manually can be a real brain teaser
- Tuning helps with model performance

Automating hyperparameter tuning with Keras Tuner

- Automation is key: open source resources to the rescue
- Keras Tuner:
  - Hyperparameter tuning with Tensorflow 2.0.
  - Many methods available

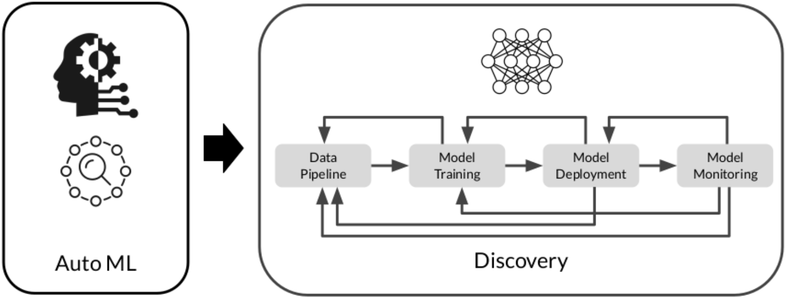
AutoML

Intro to AutoML

Outline

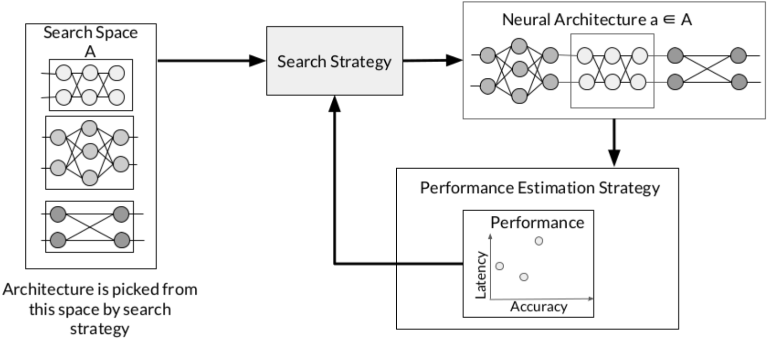
- Introduction to AutoML
- Neural Architecture Search
- Search Space and Search Strategies
- Performance Estimation
- AutoML on the Cloud

Automated Machine Learning (AutoML)



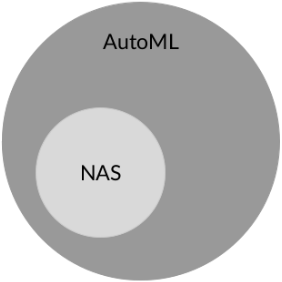
AutoML automates the entire ML workflow

Neural Architecture Search



Neural Architecture Search

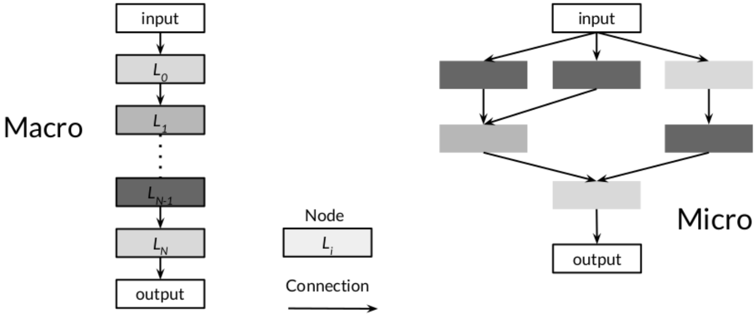
- **AutoML** automates the development of ML models
- **AutoML** is not specific to a particular type of model.
- Neural Architecture Search (**NAS**) is a subfield of AutoML
- NAS is a technique for automating the design of artificial neural networks (ANN).



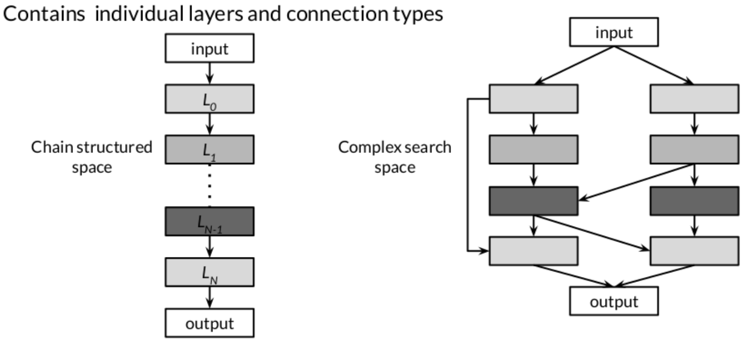
AutoML

Understanding Search Spaces

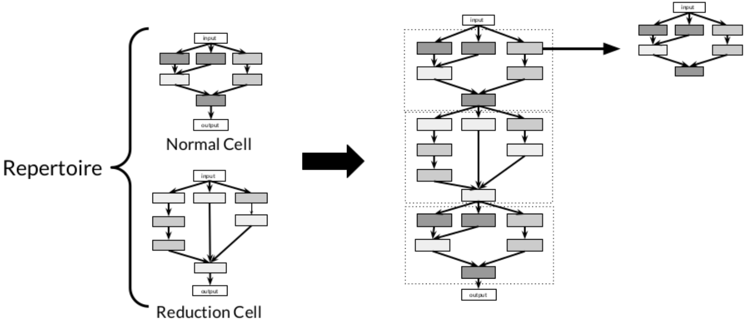
Types of Search Spaces



Macro Architecture Search Space



Micro Architecture Search Space

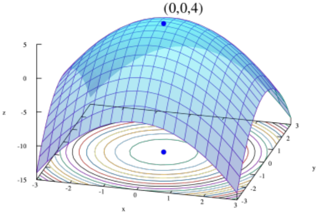


AutoML

Search Strategies

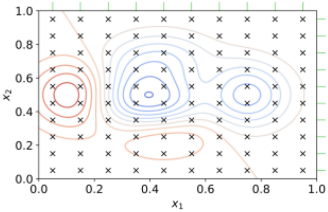
A Few Search Strategies

- 1. Grid Search
- 2. Random Search
- 3. Bayesian Optimization
- 4. Evolutionary Algorithms
- 5. Reinforcement Learning



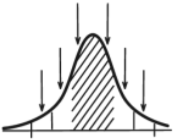
Grid Search and Random Search

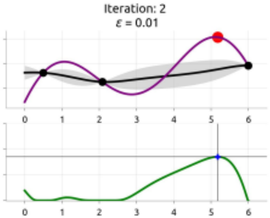
- Grid Search
  - Exhaustive search approach on fixed grid values
- Random Search
- Both suited for smaller search spaces.
- Both quickly fail with growing size of search space.



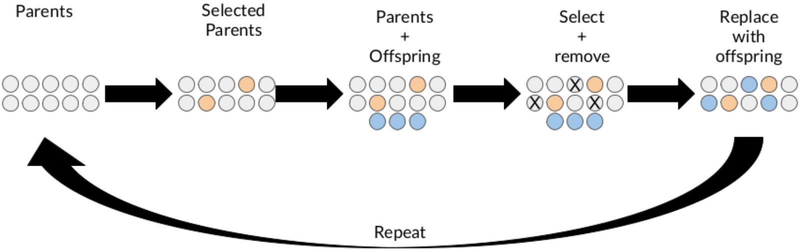
Bayesian Optimization

- Assumes that a specific probability distribution, is underlying the performance.
- Tested architectures constrain the probability distribution and guide the selection of the next option.
- In this way, promising architectures can be stochastically determined and tested.



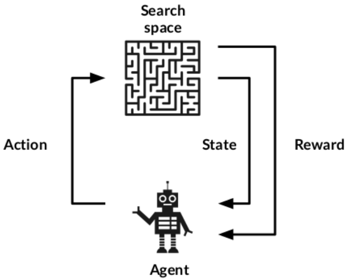


Evolutionary Methods

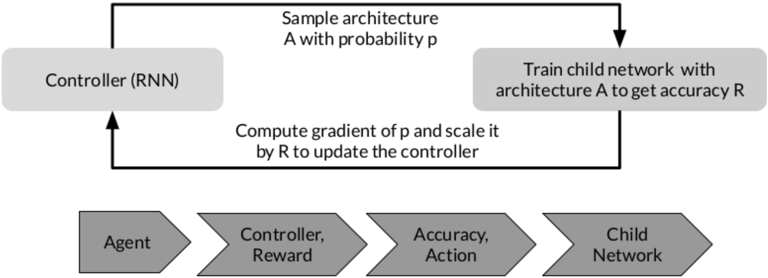


Reinforcement Learning

- Agents goal is to maximize a reward
- The available options are selected from the search space
- The performance estimation strategy determines the reward

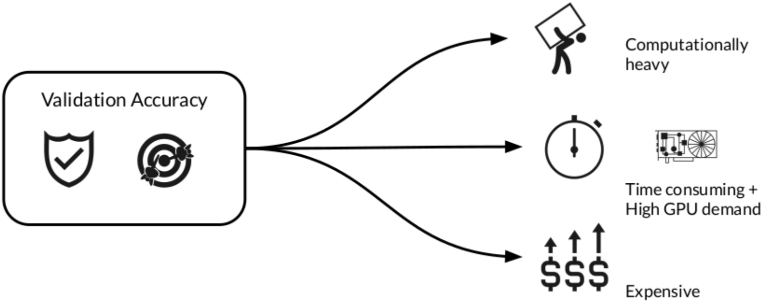


Reinforcement Learning for NAS



Measuring AutoML Efficacy

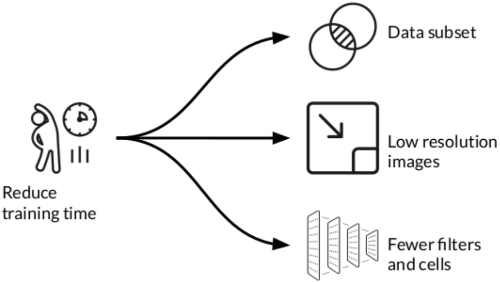
Performance Estimation Strategy



Strategies to Reduce the Cost

- 1. Lower fidelity estimates
- 2. Learning Curve Extrapolation
- 3. Weight Inheritance/ Network Morphisms

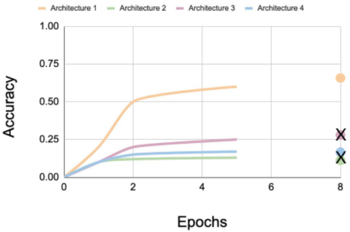
Lower Fidelity Estimates



- Reduce cost but underestimates performance
- Works if **relative ranking** of architectures does not change due to lower fidelity estimates
- Recent research shows this is not the case

Learning Curve Extrapolation

- Requires predicting the learning curve reliably
- Extrapolates based on initial learning.
- Removes poor performers



Weight Inheritance/Network Morphisms

- Initialize weights of new architectures based on previously trained architectures
  - Similar to transfer learning
- Uses Network Morphism
- Underlying function unchanged
  - New network inherits knowledge from parent network.
  - Computational speed up: only a few days of GPU usage
  - Network size not inherently bounded

Labs for This Week

Objective

Briefly describe the learning goal for this week's lab(s).

Lab Activities:

- Lab 9: [Docker] — [Docker Tutorial]
- Lab 9: [Kerat Tuner] — [Keras Tuner Tutorial]
- Lab 9: [LLMs] - [Fine-tuning]

Submission Deadline: [Before the next class]

- Assignment 9: [Docker] — [Create a experiment of your choice]
- Assignment 9: [LLMs] - [Fine-tune a model of your choice]

Reading Materials

This Week's Theme

Topic focus: [People + AI Guidebook - Data Collection + Evaluation.pdf]
You should use the worksheet related to this pdf to your project and submit it when its requested.

Required Readings:

- [On the Reliable Detection of Concept Drift from Streaming Unlabeled Data]

Be prepared to discuss highlights and open questions in class.

- DeepLearning.AI
- The People + AI Guidebook