Conclusion

Applied Machine Learning
CS 441

Derek Hoiem

This class: Conclusion

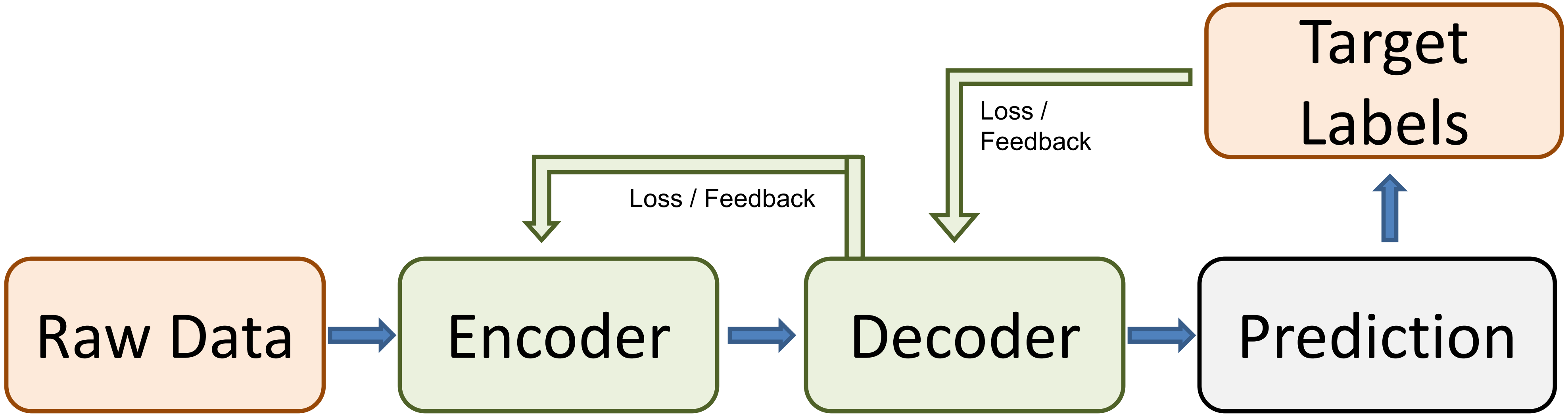
- Birds-eye view of machine learning, recap
- Where to learn more
- Trends and future of machine learning
- Feedback and closing remarks

Any questions or things you want me to talk about?

Put into the form (at top) at any time, and I will answer some of them later

https://tinyurl.com/AML441-L26

We've learned a lot



Text
Images
Audio
Discrete/continuous values
Structured/unstructured
Few/many examples
Few/many features
Clean/noisy labels

Manual feature design
Deep networks
Trees
Clustering
Feature selection
Kernels
Density estimation

Linear regressor
SVM
Logistic regressor
Nearest Neighbor
Probabilistic model

Clusters
Low dimensional embedding
Category
Pixel labels
Generated text, image, audio
Positions
Yes/No

We’ve covered what you need to get started for most ML applications

- Solid foundation in core ML concepts
  - Classification, clustering, dimensionality reduction, objectives, losses, regularization, generalization, experimental setups
- Intuition, some math, and application of classic methods
  - Logistic regression, Linear Regression, SVM, KNN, Neural Nets, Naïve Bayes, Boosting, Trees, Ensembles, PCA, K-means, Kernel Density Estimation, EM algorithm, PCA
- Exposure to state of the art recent methods
  - Deep convolutional networks, transformers, CLIP, GPT, UMAP
- Data representation and application domains
  - Images, text, audio, general data
- Practical considerations for deployment of ML applications
  - Societal impact, bias and mitigation, example of business application

Predictors: Three basic kinds

1. Linear (Regression, Logistic Regression, SVM)
   – General purpose for classification/regression
   – Benefits from feature learning

2. Instance-based (KNN)
   – Zero training time, flexible
   – Benefits from feature learning

3. Probabilistic:
   – Works well if a parametric form is known, or with tree ensembles

Feature learning: Two basic kinds

- Tree
  - Learn how to partition input space to group examples together in a discriminative way
  - Followed by instance/linear/probabilistic prediction
  - Easily combine unnormalized continuous and discrete features
  - Especially powerful with ensembles, as in Random Forests or Boosted Trees

- Neural network
  - Jointly train feature representation with linear predictor
  - Major advantages for structured data with convolutional or transformer architectures, or use for retrieval/similarity
  - Representations can be learned on one problem and tuned or applied for another problem in the same domain

Go-to approaches for **classification**

- K-NN
  - Super easy and sometimes surprisingly effective, good sanity check
- Linear SVM or linear logistic regression
  - Easy to optimize, often works well
- Random forest or boosted trees
  - Highly effective with minimal fuss, very flexible
- Linear probe or fine-tune deep network
  - Best for common structured data like images, text, audio

Go-to approaches for **regression**

- Linear regression
  - Interpretable and often works well
  - Sometimes important to transform features or targets so they are better captured by a linear model

- Random forest
  - Highly effective with minimal fuss

- K-NN
  - Especially helpful when predicting multiple correlated values, e.g. which patch can be used to fill a hole in the image

In supervised learning, always remember

- **Clean experiments**
  - training set to learn model parameters
  - validation set to select method and hyperparameters
  - test set for final performance evaluation

- **Bias/variance trade-off**
  - Avoidable error is due to challenge in fitting parameters (variance) and inability to perfectly fit the data (bias)
  - Model designs and hyperparameters often trade off between these, e.g. increasing model complexity can increase variance but reduce bias
  - Ensembles work around this trade-off, and modern deep networks often act like ensembles

- People and algorithms use mental shortcuts that lead to a kind of **bias that can be harmful to society** – transparency and fairness require conscientious sourcing, development, and evaluation

Data organization

- PCA is used to compress a vector into fewer values in a way that can be decompressed with minimal mean squared difference

- Clustering reveals common modes of data
  - K-means is an essential algorithm

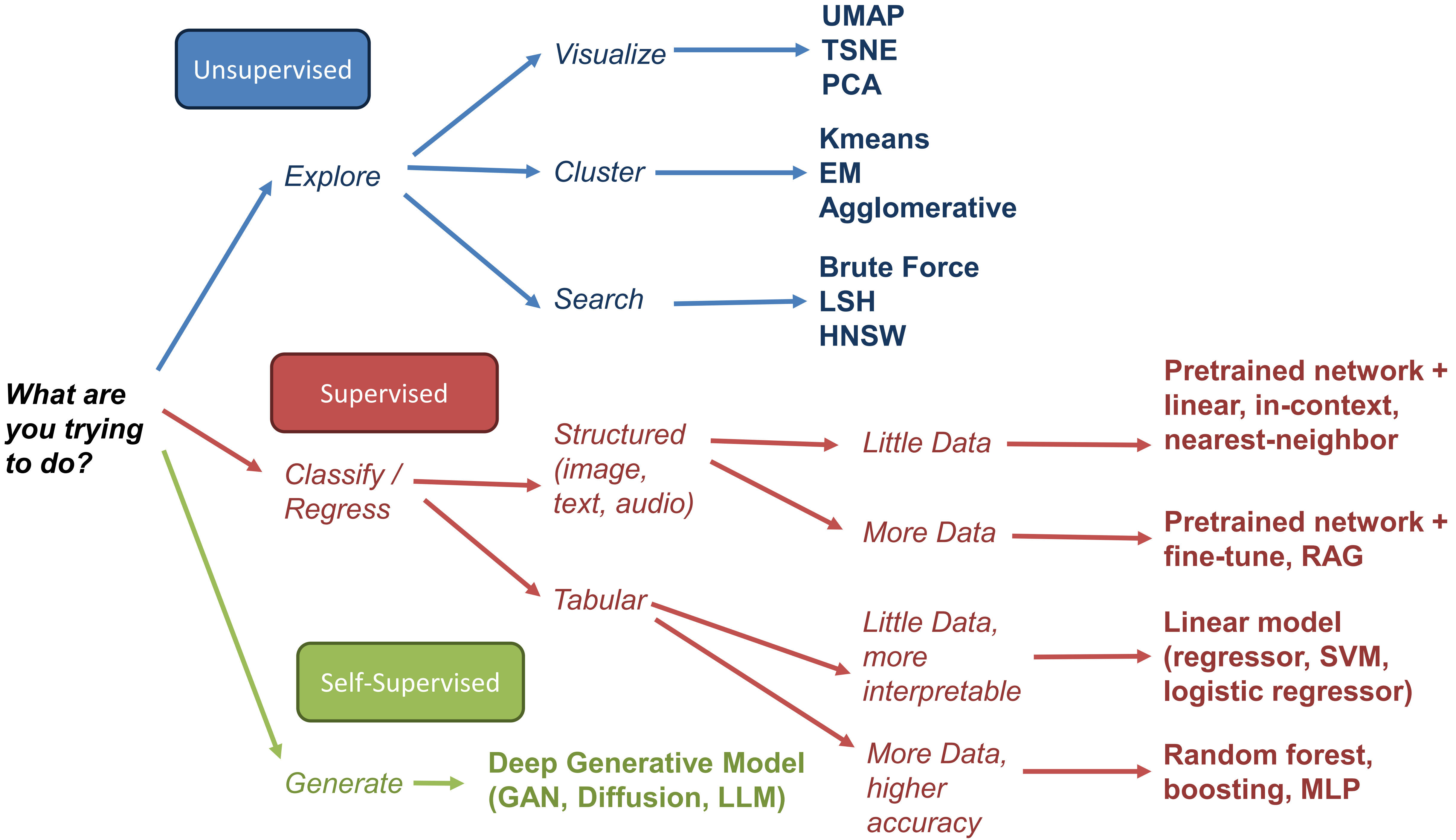
- Search for similar data items is an important application and is the computational foundation for clustering
  - FAISS is a very useful library for efficient search
  - Approximate search, e.g. trees and LSH, are needed when speed is a priority
  - Representation learning (e.g. with deep networks) is needed to make similarity meaningful

It’s all about the data

- Model architectures and computational techniques get all the attention, but there are often many reasonable choices that perform similarly

- To get best performance, data requires more thought and effort than algorithms

- Need creative ways to get freely supervised data, as well as careful curation of evaluation sets


Unsupervised
Supervised
Self-Supervised
What are you trying to do?
Explore
Visualize
UMAP
TSNE
PCA
Cluster
Kmeans
EM
Agglomerative
Search
Brute Force
LSH
HNSW
Classify / Regress
Structured (image, text, audio)
Little Data
Pretrained network + linear, in-context, nearest-neighbor
More Data
Pretrained network + fine-tune, RAG
Tabular
Little Data, more interpretable
Linear model (regressor, SVM, logistic regressor)
More Data, higher accuracy
Random forest, boosting, MLP
Generate
Deep Generative Model (GAN, Diffusion, LLM)

Choosing the right tool for the job

https://tinyurl.com/AML441-L26

Choosing the right tool for the job: answers

Predict the sale price ($) of a house given location, square footage, year built, and other information typically included in real estate listings. Training data has ~50K samples.

- ○ Nearest neighbor classifier
- ○ Naive Bayes Classifier
- ○ Logistic regression
- ○ Linear regression
- ◉ Random forest regressor
- ○ Random forest classifier
- ○ Convolutional network

Random forest regressor
- Regression problem
- Mixed continuous and categorical features
- Lots of training data available

Identify the plant type from a photo of a seed on a white background. You have 100 examples per plant type in training data.

- ○ Nearest neighbor classifier
- ○ Naive Bayes Classifier
- ○ Logistic regression
- ○ Random forest classifier
- ○ Convolutional network, trained from scratch
- ◉ Convolutional network, fine-tuned based on an ImageNet pretrained model

Fine-tuned ConvNet
- Image classification (use deep network to take advantage of structured image data)
- Limited data, so fine-tuning from pre-trained is much better than training from scratch

Choosing the right tool for the job: answers

Given the text of a review, assign a score from 1 to 5 indicating how positive the review is. You have 1M training samples.

- ○ Use Naive Bayes Classifier on word count
- ○ Use GPT-4
- ◉ Fine-tune a BERT model
- ○ Boosted decision tree on word count

Fine-tune BERT

- Using a deep network language model gives more effective features than word count
- Plenty of data for fine-tuning
- GPT-4 is slow/expensive and not easy to tune

Automatically identify which students are present from a photo of the classroom. Training data is one headshot per student. Face detector is available to localize faces.

- ○ Use nearest neighbor on an ImageNet pretrained model
- ◉ Train a face classifier on Labeled Faces in the Wild (LWF), and then use nearest neighbor on the trained encoder features
- ○ Train a face classifier on LFW and then fine-tune on the training set
- ○ Train a MLP from scratch
- ○ Use PCA on the patch of face pixels and then nearest neighbor

NN w/ LWF-pretrained model

- Training a deep network face classifier on a large dataset can provide good features
- Only one example per face available, so use nearest neighbor (fine-tuning will overfit)

Choosing the right tool for the job: answers

Problem that involves a combination of image and text analysis, with lots of training data available.

- ○ Nearest neighbor
- ○ Logistic regression
- ○ Convolutional network model
- ◉ Transformer model

Transformer

- Both images and text can be represented as tokens that can be processed together with the transformer
- Plenty of data for training

Image classification problem with 10 classes and ~10 examples per class available for training.

- ○ Deep network, trained from scratch
- ◉ Deep network with pre-trained model and linear classifier ("linear probe") on features
- ○ Fine-tuned deep network

Pre-trained model w/ linear classifier

- Use good features from pre-trained model
- Probably not enough examples to effectively fine-tune and validate

What questions do you have?

Other related courses

AI Core

Machine Learning
CS 440 Artificial Intelligence
CS 441 Applied Machine learning
CS 442 Trustworthy Machine Learning
CS 443 Reinforcement Learning
CS 446 Machine Learning
CS 498 Introduction to Generative AI
CS 498 Agentic AI
CS 540 Deep Learning Theory
CS 542 Statistical Reinforcement Learning
CS 547 Deep Learning
CS 598 Learning to Learn
CS 598 Deep Generative Models
CS 598 Topic in LLM Agents

Computer Vision
CS 444 Deep Learning for Computer Vision
CS 445 Computational Photography
CS 543 Computer Vision
CS 544 Optimization in Computer Vision
CS 598 Computer Vision for Healthcare
CS 598 3D Vision

Language
CS 410 Text Information Systems
CS 447 Natural Language Processing
CS 546 Advanced Topics in NLP

Audio and Signals
CS 448 Audio Computing Laboratory
CS 545 Machine Learning for Signal Processing

Robotics
CS 452 Topics in Robotics
CS 588 Autonomous Vehicle System Engineering

AI Breadth and Other Electives

Data Workflows
CS 412 Introduction to Data Mining (DAIS)
CS 470 Social & Information Networks (DAIS)
CS 510 Advanced Information Retrieval (DAIS)
CS 512 Data Mining Principles (DAIS)
CS 513 Theory & Practice of Data Cleaning (DAIS)
CS 514 Advanced Social & Information Networks (DAIS)

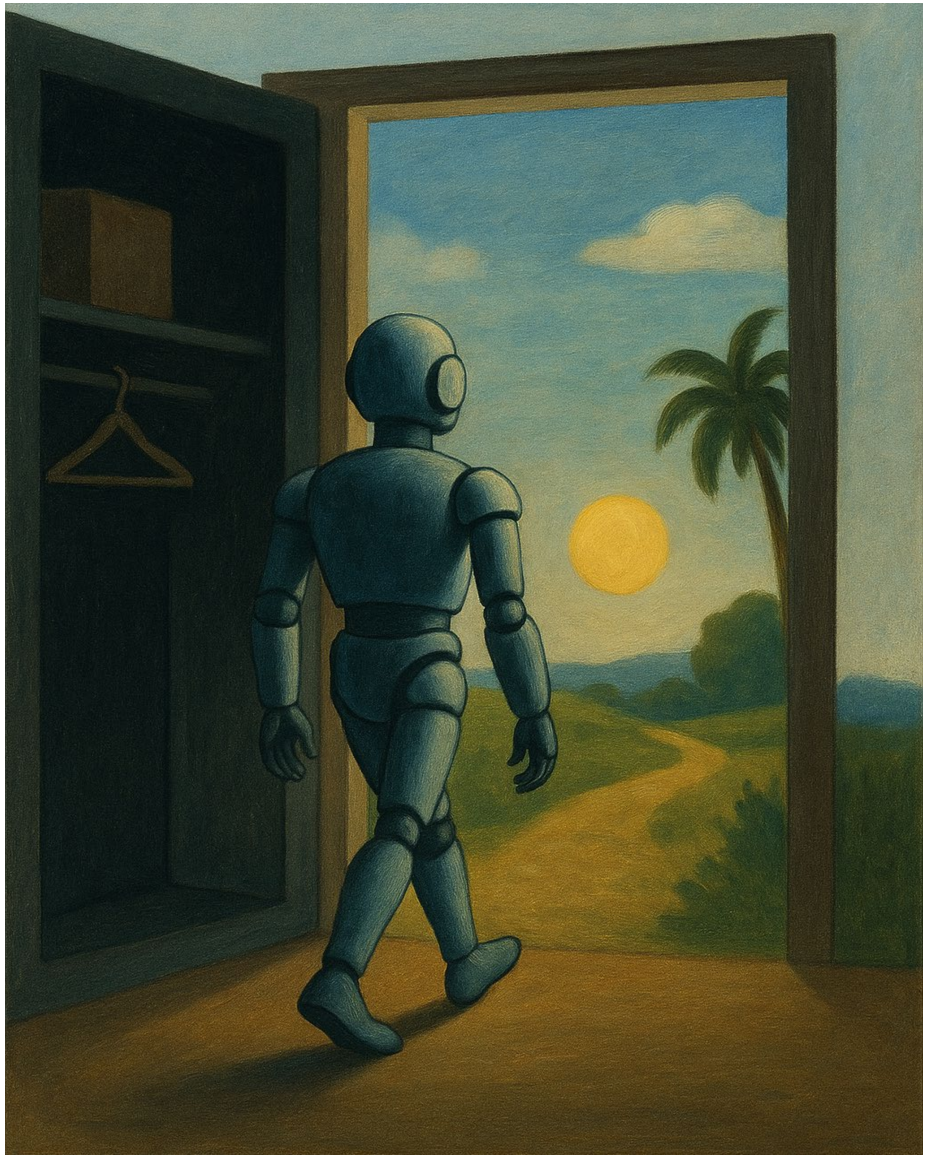
People & AI
CS 464 Spec Top in Societal Impacts / Cyber Dystopia (IC)
CS 565 Ethics in AI (IC)
CS 568 User-Centered Machine Learning (IC)
CS 442 Trustworthy Machine Learning (SP)
CS 562 Advanced Topics in Security, Privacy and ML (SP)
CS 598 Language, Interfaces, and Communication (IC)
CS 598 CSS: Computational Social Science (IC)

ML Systems
CS 498 Machine Learning Systems (SN)
CS 598 Systems for GenAI (SN)
CS 521 ML and Compilers (PLFMSE)
CS 521 Trustworthy AI Systems (PLFMSE)
CS 598 ML for Code (PLFMSE)
CS 598 Software Quality Assurance with Generative AI (PLFMSE)
CS 598 AI Efficiency: Systems and Algorithms (SN)
CS 598 APE - Advanced Perform Engineering (ACP)

Other AI Electives
CS 417 Introduction to Virtual Reality (IC)
CS 466 Introduction to Bioinformatics (BCB)
CS 469 Computational Advertising Infrastructure (IC)
CS 463 Computer Security II (SP)
CS 473 Algorithms (Th)
CS 563 Advanced Computer Security (SP)
CS 582 Machine Learning for Bioinformatics (BCB)
CS 598 Deep Learning for Healthcare (BCB)

AI MCS (preview)
- 3 AI Core, including 1 ML
- 1 each of Data Workflows, People & AI, ML Systems
- 1 more of anything

Where is AI going?

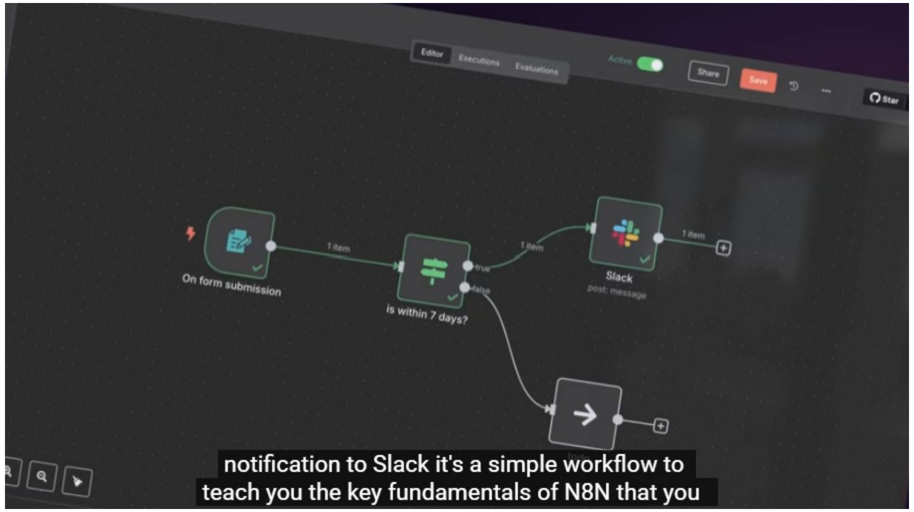


Agentic AI Workflows – multi-decision processes involving humans and models working together

- Think about which of your daily tasks involve multiple software steps, e.g.
  - Checking multiple news feeds
  - Paying bills
  - Scheduling appointments

- In business, e.g. sales, customer help, billing, reporting, there are many more of these

- These can be automated using AI agents, and more complex processes can be built by stacking multiple workflows and human steps



https://www.youtube.com/watch?v=4cQWJViybAQ

Automating work with computers

1. Model watches you perform your daily tasks to learn how you use software

2. You can ask it to do something for you, e.g. draft your emails which requires looking up information in various systems

3. You can work with the agent, talking through what you’re trying to do so that it solves problems for you in parallel

Robots

- A lot of jobs require physical presence and action

- Robots can run, do dishes, stack blocks, and more

- What's missing?

Barriers to more complete AI solutions

- Memory
  - AI has no episodic memory – it doesn't remember where or when it learned something, and acquired memory is only accessible through simplistic retrieval mechanisms
  - Weights are trained and frozen, limiting adaptability
- Real-time interaction
  - We don't know to make AI work fluidly with people (everything is framed as input → processing → output
- Physicality
  - Robots are physically brittle, with little agility, and limited sensors (e.g. humans have 4M touch sensors, but most robots have 0)
  - Very limited generalization – new objects, new environments, new variations on tasks tend to completely break a robot's ability. Most demos are narrowly scripted.

Finishing up the semester

- Dec 14: Final project due at 11:59pm

- Dec 4-9: Final exam
  - Covers all material for semester
  - See CampusWire pinned post for details

- Dec 9 – I will be here in lecture period for help with final project (not exam prep), or possibly general questions beyond class

- Last office hour is Dec 12
  - We will still monitor and respond to CampusWire until Dec 14

FLEX Feedback

- We put much time and energy into this course
- Please take some time to provide ratings and feedback
  - go.illinois.edu/flex
- We'll use it to keep making the course better

Thank you to the course staff for all their hard work to answer questions and grade.

Thank you for your hard work and engagement!