

LEC 15

CSE 123

Machine Learning

Questions during Class?
Raise hand or send here

sli.do #cse123



BEFORE WE START

Talk to your neighbors:

What is your favorite beverage?

Respond on [sli.do](#)!

Instructor: Brett Wortzman

TAs:

Arohan	Jonah	Kavya	Eeshani	Trien
Ashar	Brice	Misha	Aidan	Evan
Sean	Chris	Kieran	Cora	Rena
Chloe	Elden	Sahana	Dixon	Katharine
Jenny	Ishita	Anirudh	Nhan	Anyia
Nate	Kuhu	Crystal		

Now playing:  [CSE 123 26wi Lecture Tunes](#) 

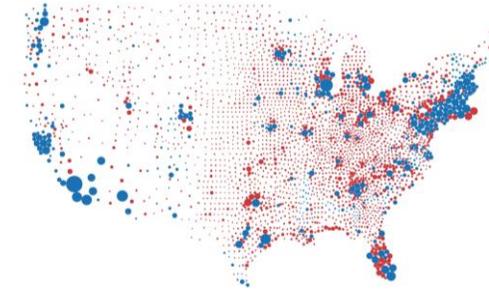
Announcements

- *Programming Assignment 3* out, due next Friday! (3/6)
- Resubmission 5 closes tonight
- Quiz 2 next Tuesday (3/3)

Applications of ML

Estimation

- *Opinion Polls*
 - How does a population feel about an issue?



Prediction

- *Content Recommendation*
 - Can we predict how much someone will like a movie based on past ratings?
- *Object Recognition*
 - Identify {Car, Road, Plane, Bird, Person} within an image?



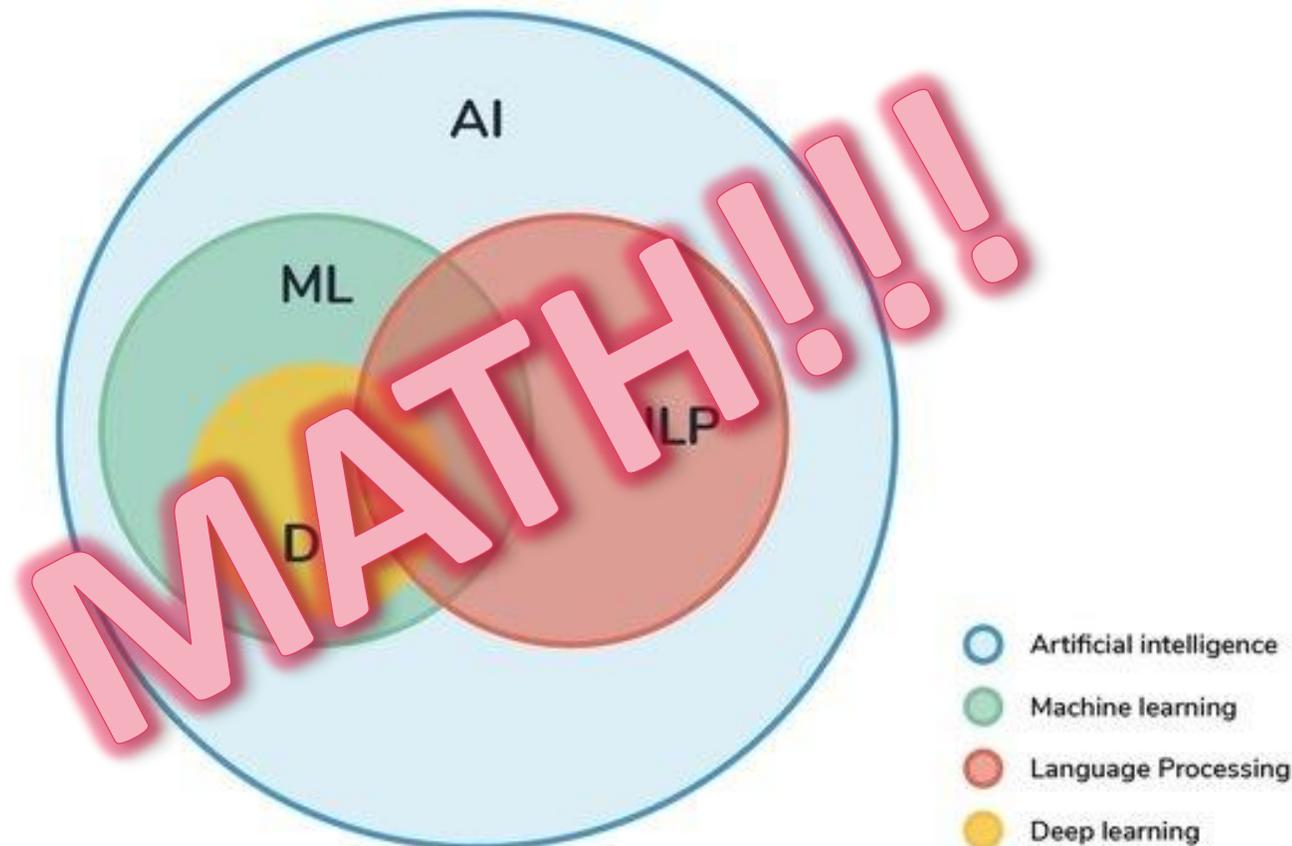
Generation

- *Text Generation*
 - Can computers generate text written like a human?
- *Image Generation*
 - Can computers generate images from a prompt



What is Machine Learning (ML)?

- Subset of Computer Science concerned with “learning” data trends
 - (Today’s lecture will not be tested on quizzes/exams.)



What is Machine Learning (ML)?

- Subset of Computer Science concerned with “learning” data trends
- Simple example: maximum likelihood estimation (MLE)

$$D = (HHTHT)$$

Is this coin biased or not?

What's the best guess for “how biased” it is?

$\theta = \text{coin bias}, n = \text{flips seen}, k = \text{heads seen}$

$$P(D|\theta) = \theta^k (1 - \theta)^{n-k}$$

Goal: find $\hat{\theta}_{MLE}$, value that maximizes probability of what we saw

Maximum Likelihood Estimation

$$P(D|\theta) = \theta^k (1 - \theta)^{n-k}$$

$$\hat{\theta}_{MLE} = \operatorname{argmax}_{\theta} P(D|\theta)$$

$$\frac{\partial}{\partial \theta} (\theta^k (1 - \theta)^{n-k}) = 0$$

$$k\theta^{k-1}(1 - \theta)^{n-k} - (n - k)\theta^k(1 - \theta)^{n-k-1} = 0$$

Maximum Likelihood Estimation

$$k\theta^{k-1}(1-\theta)^{n-k} = (n-k)\theta^k(1-\theta)^{n-k-1}$$

$$k(1-\theta) = (n-k)\theta$$

$$k = n\theta$$

$$\hat{\theta}_{MLE} = k/n$$

*Takeaway: There are formal, mathematical ways to verify intuition!
+ We can perform this process with more complicated distributions!*

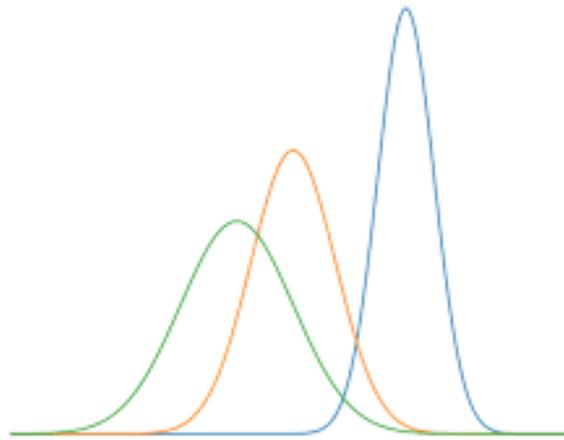
What is Machine Learning (ML)?

- Subset of Computer Science concerned with “learning” data trends
- Simple example: maximum likelihood estimation (MLE)
 - As $n \rightarrow \infty$, we know that $\hat{\theta}_{MLE} \rightarrow \theta^*$ (true distribution)
 - With enough data points, we can estimate any statistical distribution!
 - Central limit theorem: sample mean is normally distributed on true mean...

$$P(x | \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

What is Machine Learning (ML)?

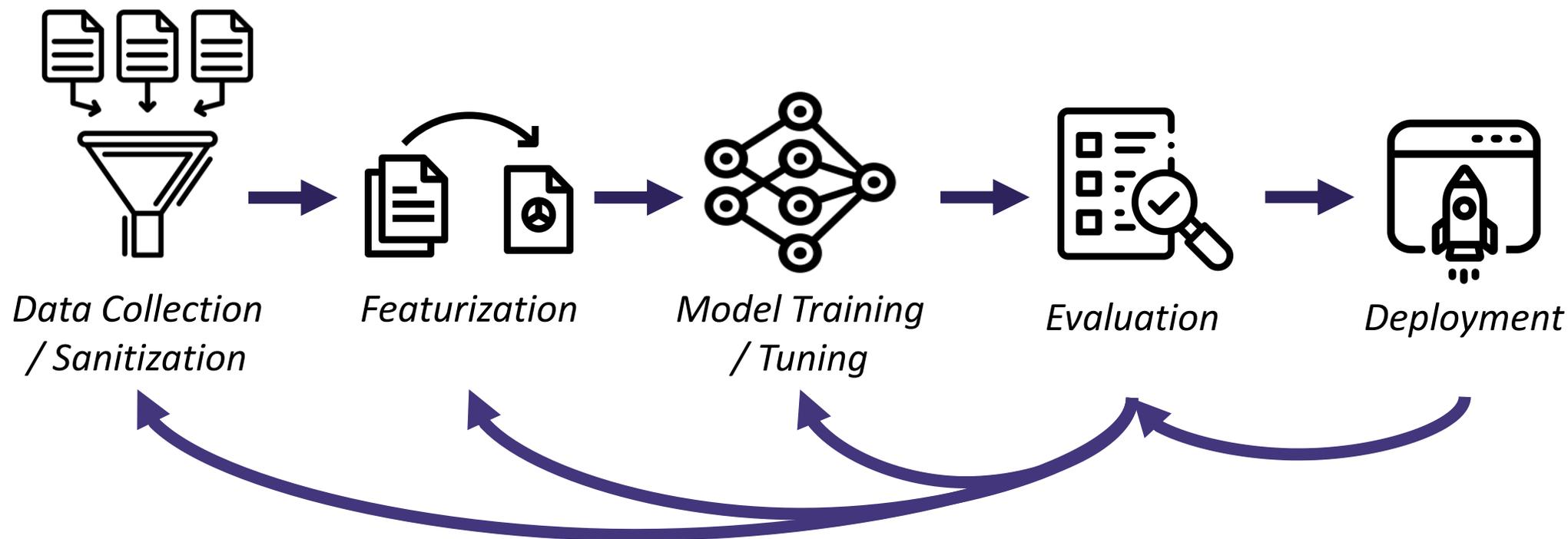
- Subset of Computer Science concerned with “learning” data trends
- Simple example: maximum likelihood estimation (MLE)
 - As $n \rightarrow \infty$, we know that $\hat{\theta}_{MLE} \rightarrow \theta^*$ (true distribution)
 - With enough data points, we can estimate any statistical distribution!
 - Central limit theorem: sample mean is normally distributed on true mean...



Given enough previous examples, we can estimate the underlying distribution and make predictions about... anything!

ML Pipeline

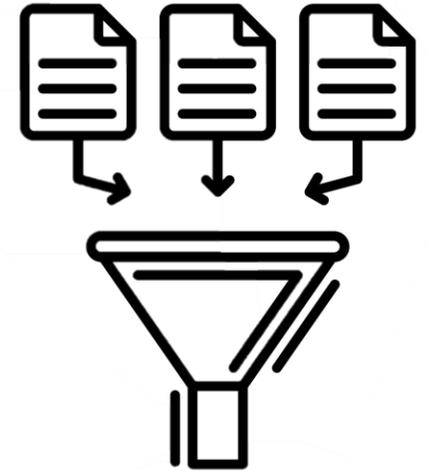
- Generally, building an ML model involves the following steps:



- Notice that you can step backwards!
 - ML in particular is an applied science, it's all an experiment!

1. Data Collection

- We *need* example data to understand a distribution
 - Lots and lots of it too ($n \rightarrow \infty$)
- Where does this data come from?
 - Language: Reddit, Twitter, Facebook, Wikipedia, Blogs, etc.
 - Images: Google, Twitter, Websites
 - Code: Github
 - Really, anywhere publicly (or not) accessible on the Internet
- Who determines what data is used? $_ \backslash (\text{ツ}) _ /$
 - Often companies buy preprocessed data from others
 - Let's say that you accidentally post your phone number on your twitter
 - A model could scrape that info, memorize it, and regurgitate it when prompted
- **Data carries PII / bias that we need to account for**



Data Bias

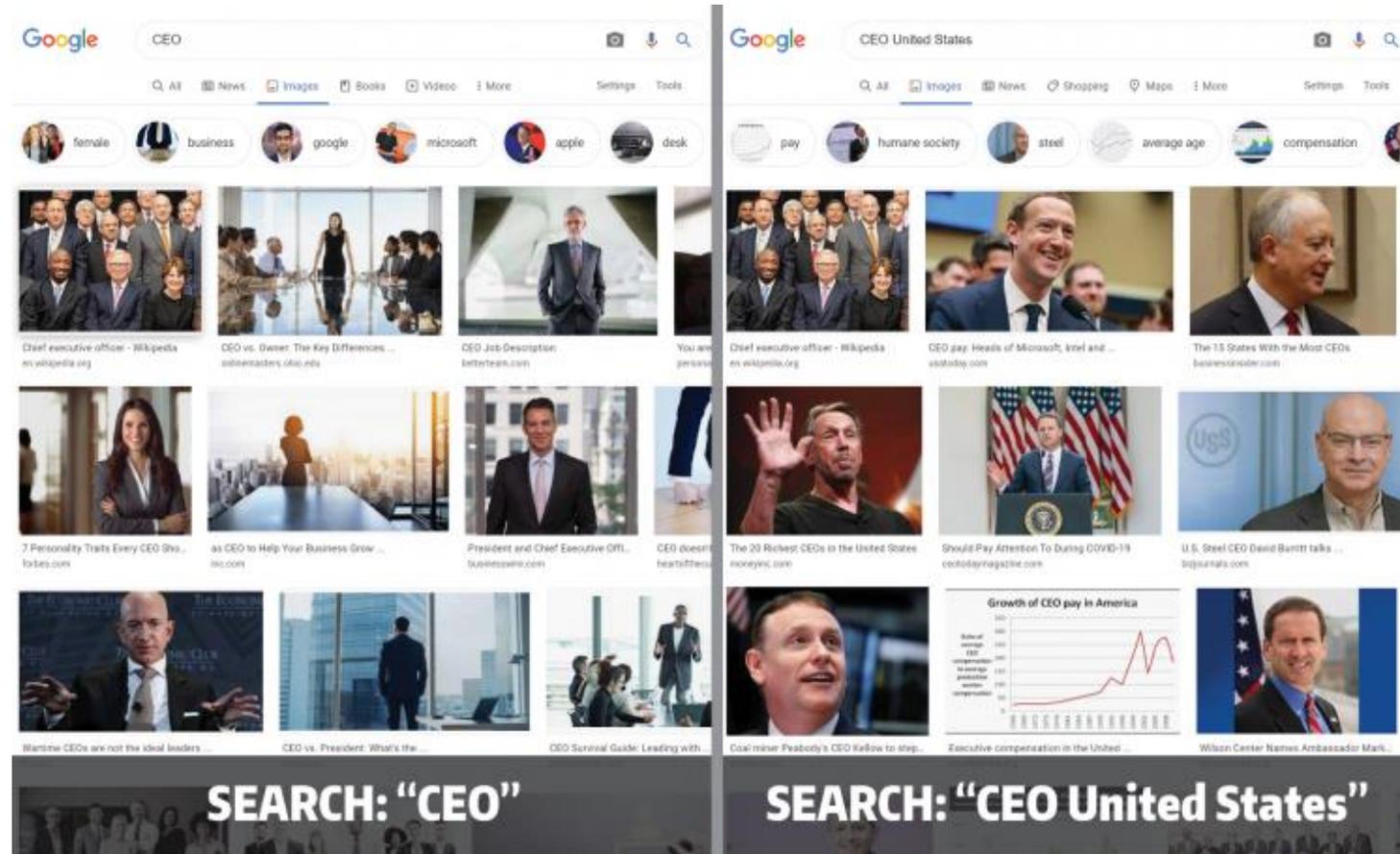
- Image results for searching the term “CEO” on Google (2015)
 - Notice anything about the results?



<https://www.washington.edu/news/2015/04/09/whos-a-ceo-google-image-results-can-shift-gender-biases/>

Data Bias

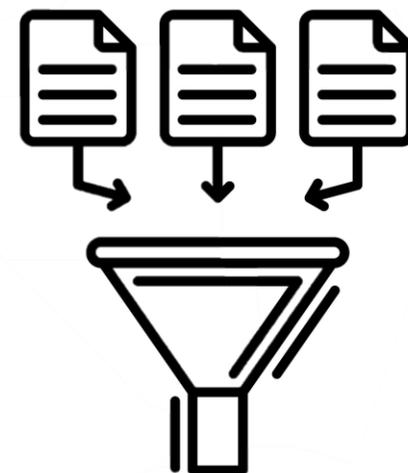
- Fix: Image results for searching “CEO” and “CEO United States” (2022)



<https://www.washington.edu/news/2022/02/16/googles-ceo-image-search-gender-bias-hasnt-really-been-fixed>

Data Sanitization

- **Data carries PII / bias that we need to account for**
- We don't want our model to memorize a phone number
 - Let's just remove all phone numbers from our inputs!
 - Is this an effective solution?
- Sanitization can be ethically gray – does it disproportionately affect subpopulations?
 - Correlated features

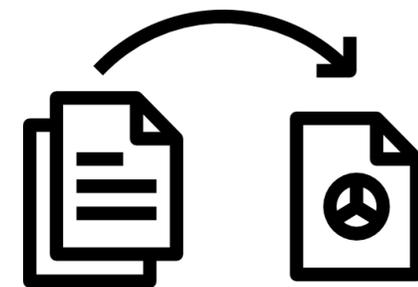


Our models are only as strong as the data they're built upon.

Garbage in, garbage out.

2. Featurization

- Now that we have all our data, we need to convert it into something a computer can understand (numbers)
 - How can we convert text / images into numbers?
- Determine what aspects of the data interest you (features)
- Words can be “vectorized”
 - Converted into n -dimensional vectors $n \in \{50, 200, 500, \dots\}$
 - Determined from the word2vec algorithm
- Images are already numbers... (2d array of RGB values)

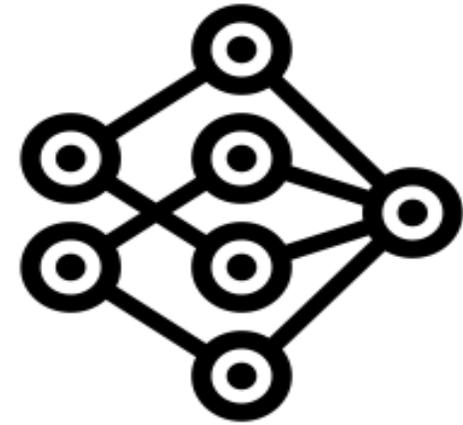


Word Embeddings

- We call these word vectors “embeddings” and they’re pretty interesting to mess around with
- Can perform mathematical operations on them
 - Find the nearest vectors to any given word (synonyms)
 - Compute comparisons (dog is to puppy as cat is to ____)
 - Take the difference between puppy and dog (age vector) and add it to cat
 - Find the nearest vectors to the result and you’ll likely see “kitten”
- These operations can further reveal bias
 - man is to doctor as woman is to _____
 - **Any model trained from biased data points will estimate a biased distribution**

3. Model Training

- Pick some way of using data to estimate
- Lots of different flavors of this
 - Regression (linear, logistic)
 - Neural Networks (CNNs, RNNs, Transformer, etc.)
 - Nearest neighbors
 - **Decision trees**
- Provide additional computation (memory / GPUs / time) until desired result is achieved



It's all one big experiment – try options until something sticks.

This should feel somewhat concerning...

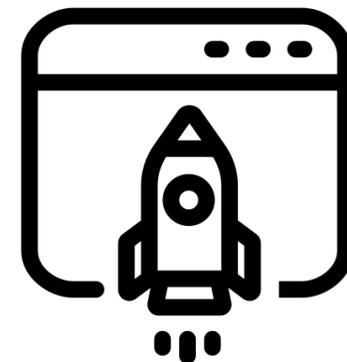
4. Evaluation



- Does your model actually work?
- Typically we split our initial dataset into 3 different subsets:
 - Train (provided to the model during training)
 - Validation (used after a model has trained to compare to previous iterations)
 - Test (used once a model has been chosen to see how it performs)
- Determine whether or not your model is over / underfitting
- Most ML applications go no further than this step
 - No attempt to determine *why* a particular model is working well

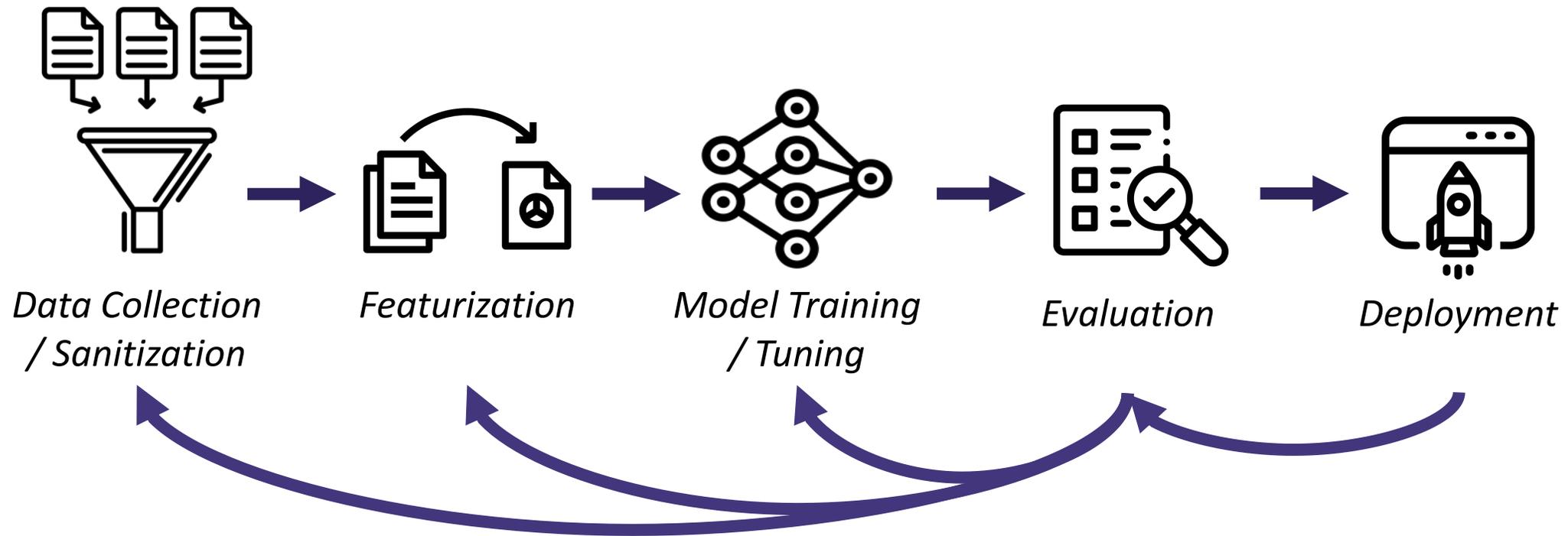
5. Deployment

- Put your model out into the real world and see what happens
 - Does it perform the job as expected? Should further work be put into development?
- At this point, often the next iteration of refinement takes place
 - GPT 2.0 -> 3.0 -> 3.5 -> 4.0
 - Options include:
 - Collect more data, use more compute, discover better tuning, discover better model
- Often, not much effort is put into understanding negative impacts
 - Case in point: ChatGPT and the education system



ML Pipeline

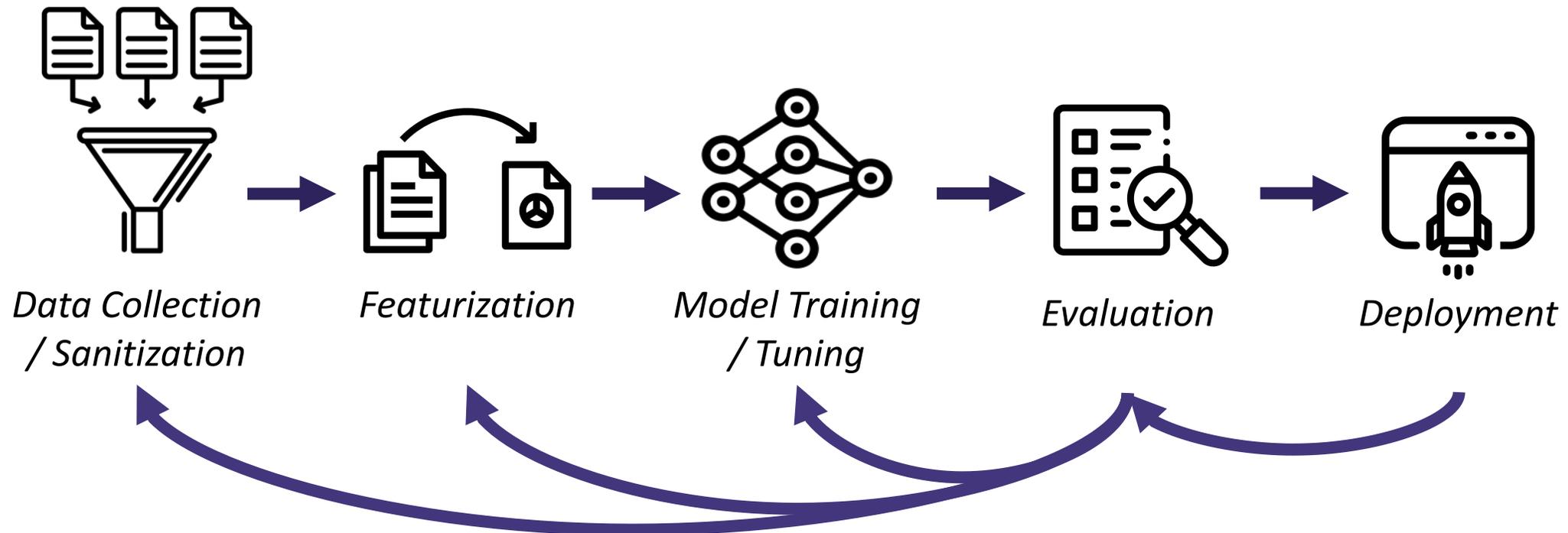
- That's it – in essence, that's how every ML model is created



Does this knowledge change your perspective on ML / AI?

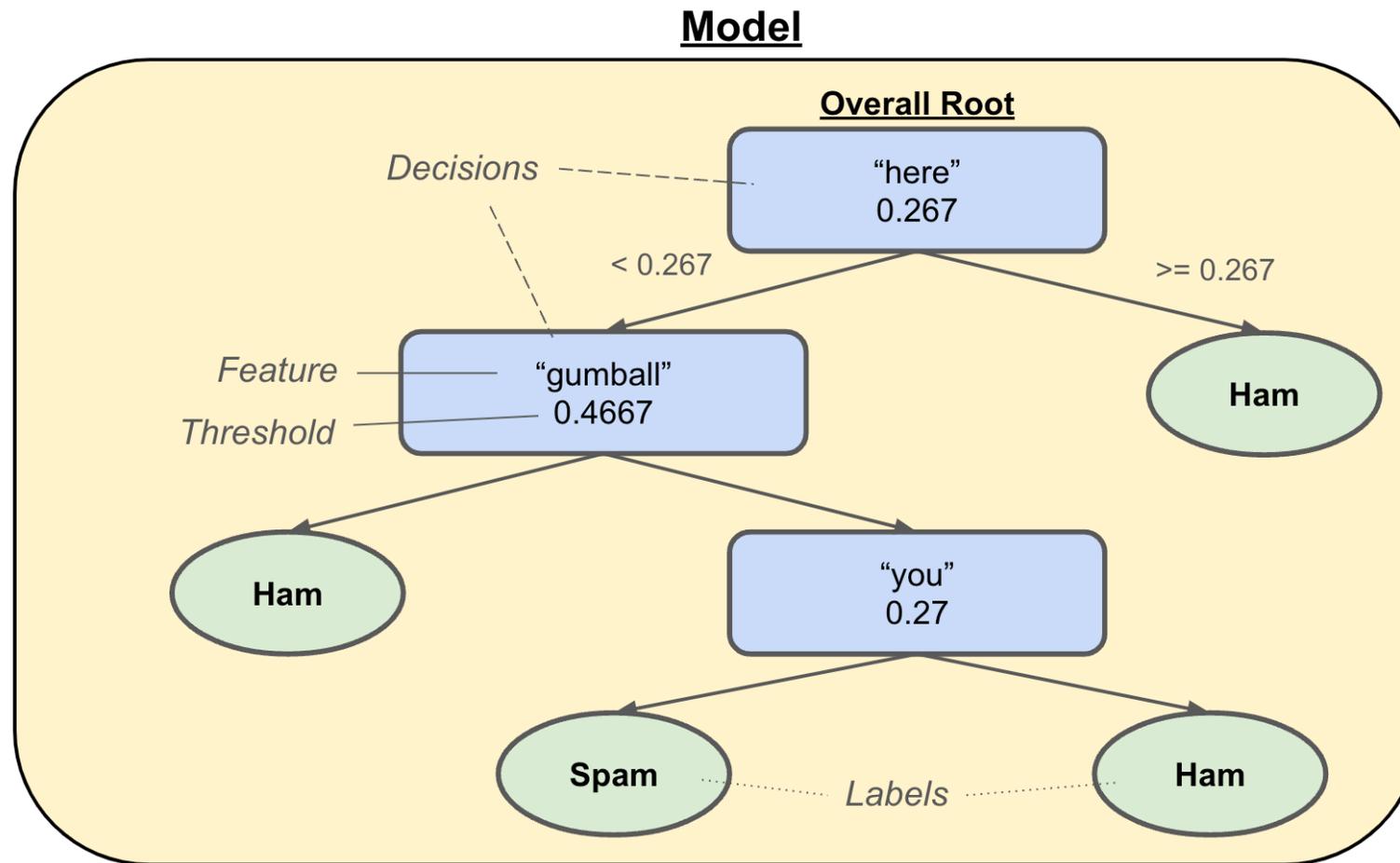
Cornbear's Classifier

- Programming assignment will involve part 3 of this pipeline
 - You'll implement a *decision tree* capable of classifying text



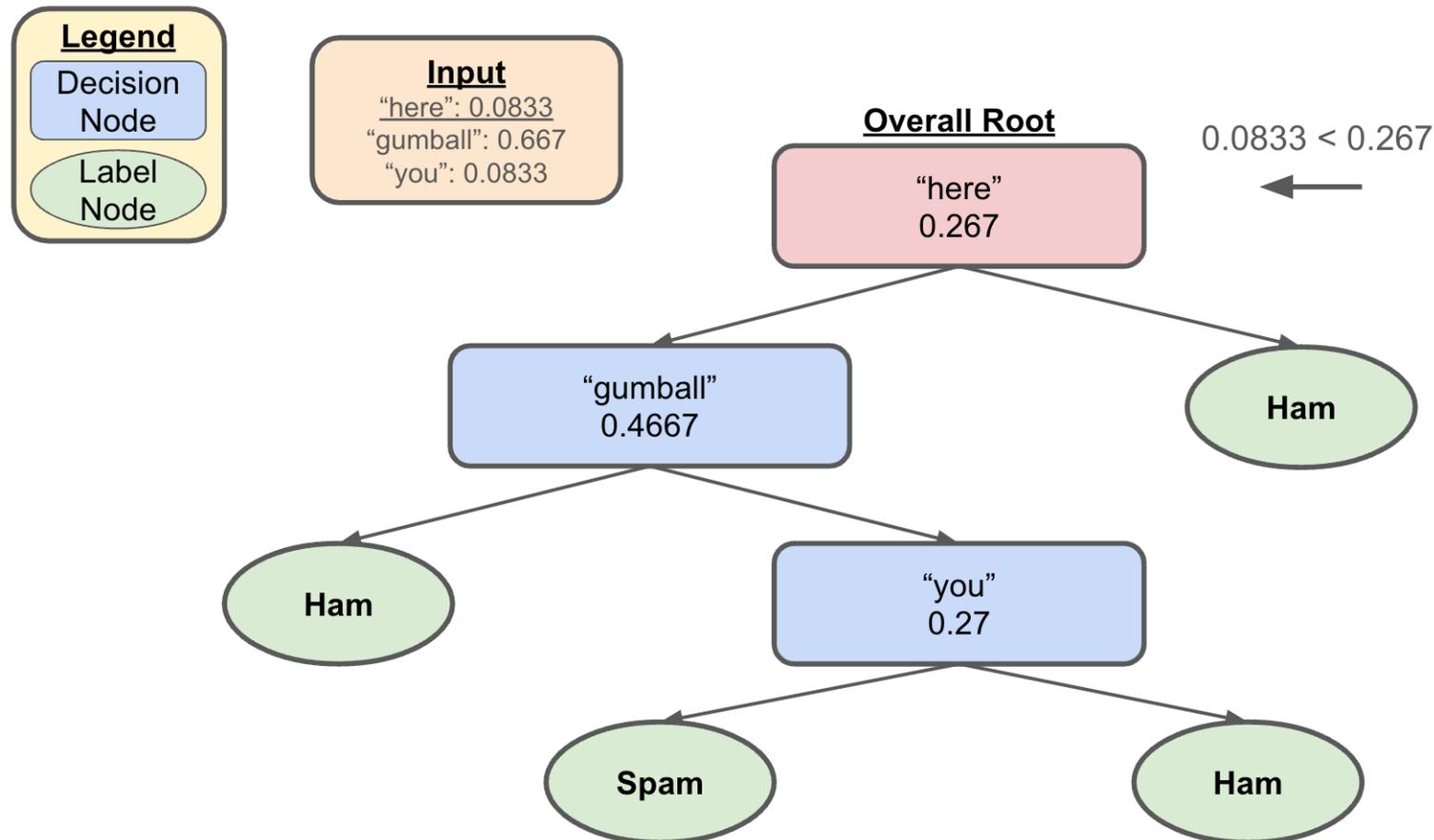
Decision Trees

- Tree structure where each intermediary node contains a feature / threshold pair (decision) and leaf nodes are labels



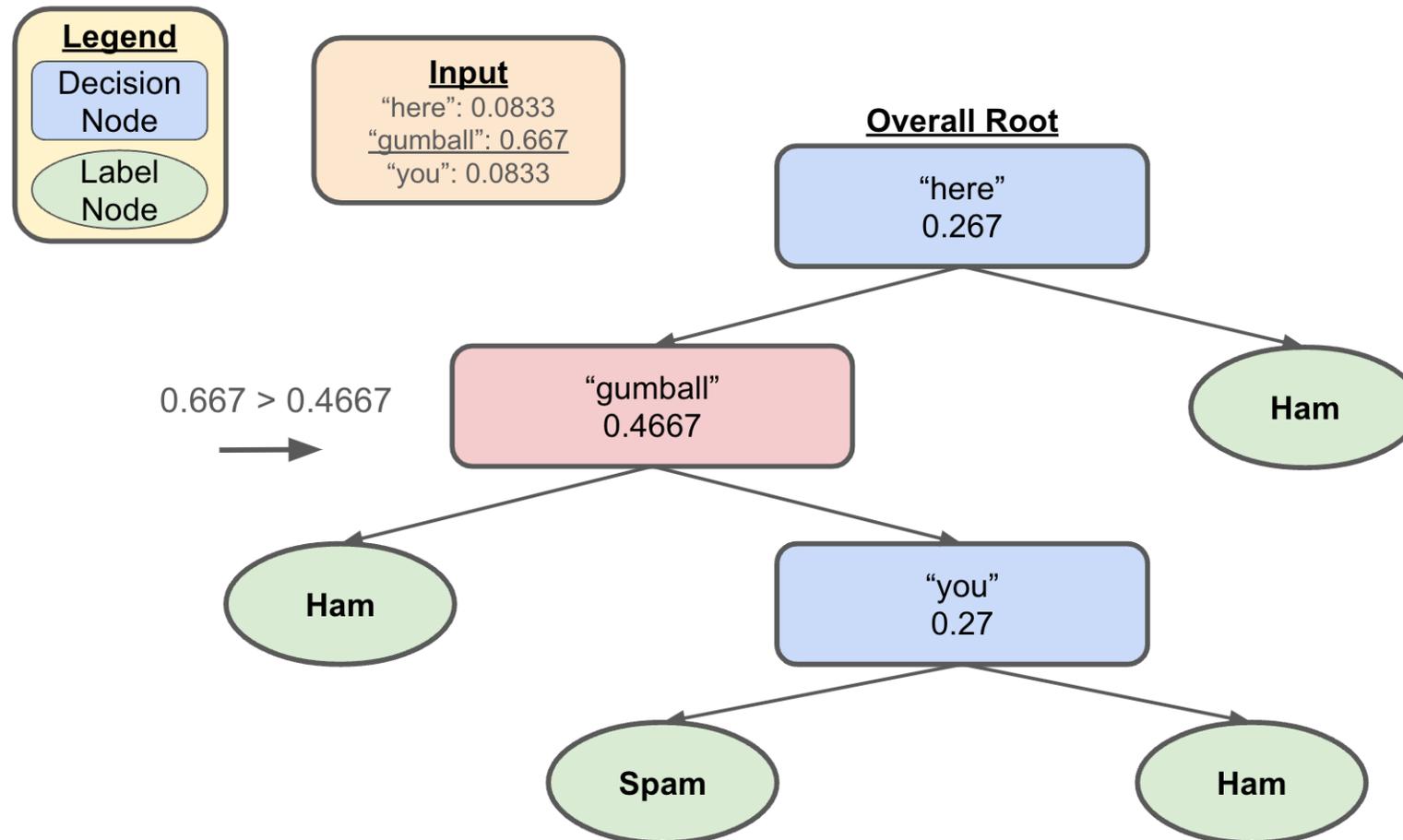
Decision Trees

- Let's say we wanted to classify the following
 - here gumball gumball gumball gumball you silly gumball gumball gumball gumball doggo



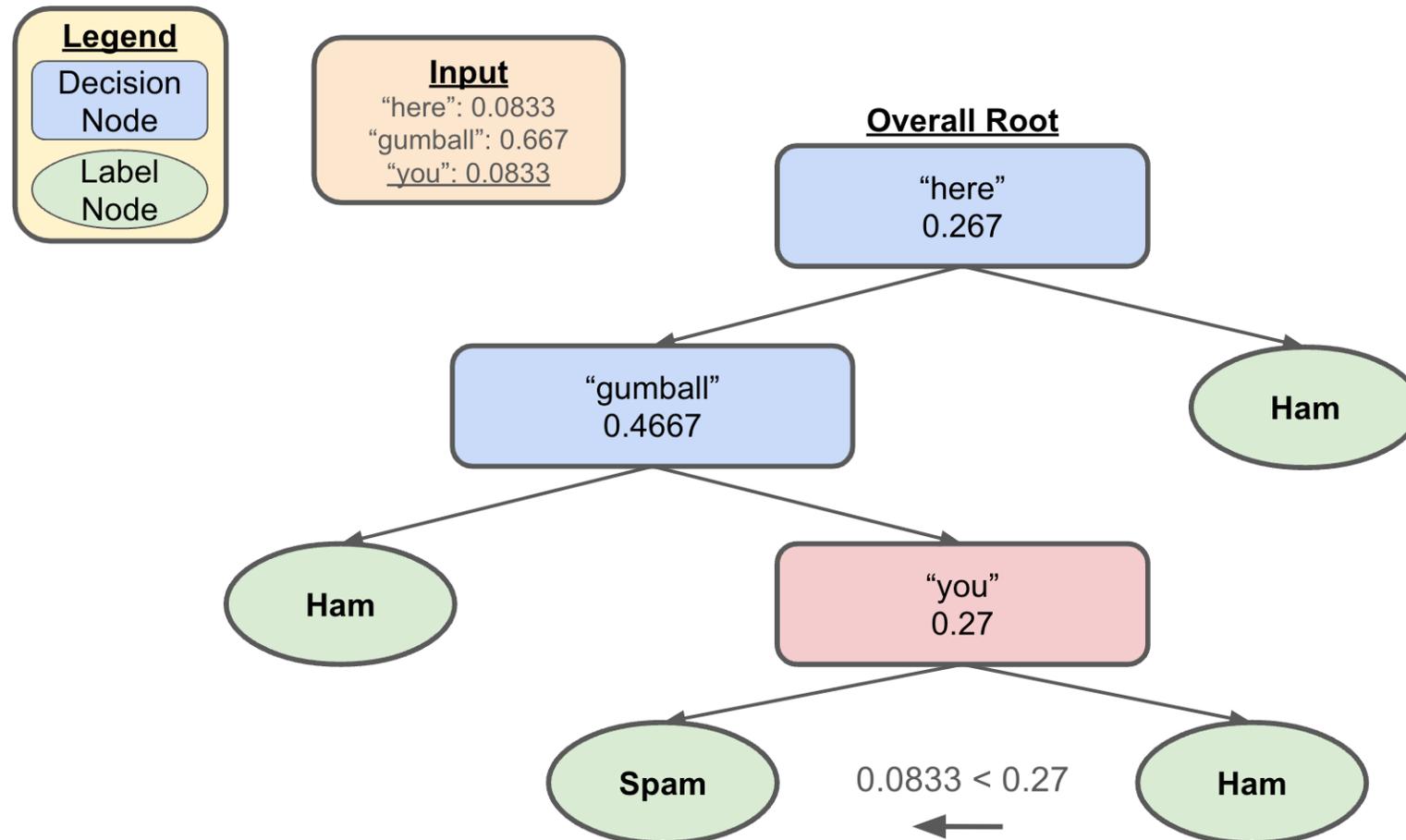
Decision Trees

- Let's say we wanted to classify the following
 - here gumball gumball gumball gumball you silly gumball gumball gumball gumball doggo



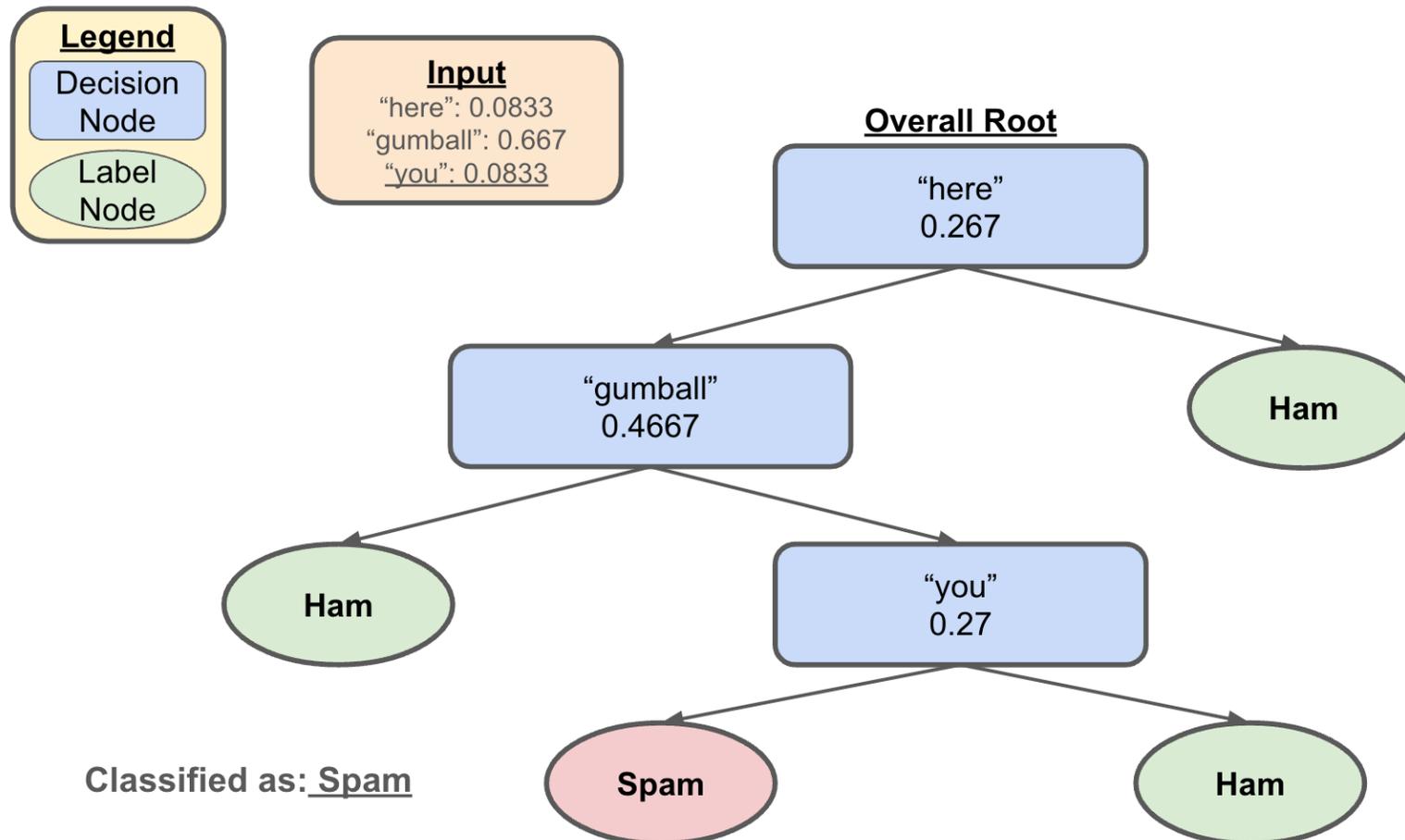
Decision Trees

- Let's say we wanted to classify the following
 - here gumball gumball gumball gumball you silly gumball gumball gumball gumball doggo



Decision Trees

- Let's say we wanted to classify the following
 - here gumball gumball gumball gumball you silly gumball gumball gumball gumball doggo



Questions to Consider

- Are ML models capable of “learning”?
 - i.e. is it possible to “learn” just by observing / memorizing?
 - Does ChatGPT actually “understand” language?
- If all output from ML models is based on previous examples, who gets credit / takes responsibility for generation?
 - Think AI art and your C2 / P2 reflection responses
- What harm could come from deploying ML models we don’t fully understand?
- If society itself is biased, how much should we worry about the bias present in data / ML models?
 - To what extent should concern about bias hinder further advancements?