

Exam 1

Artifacts and Abstraction (Sessions 0 - 2)

There are two ways in which artifacts can contain political properties, where we define political as an arrangement of power and authority in human association as well as the activities that take place within those arrangements:

1. Technical arrangements as a form of order

- a. Precedes use of the actual thing
- b. Intended
- c. Can also transcend simple category of “intended” and “unintended”
- d. Example: Low hanging overpasses on Long Island designed to enforce social segregation; blocks off beach access to low-income people
- e. Example: Park benches and homeless; no handrests makes it harder for them to lie down on it.

2. Inherently political technologies

- a. Adoption of a certain technology *requires* or is strongly *compatible* with certain social conditions.
- b. Example: Nuclear power plants encourage centralized, hierarchical control structures due to the complexity and potential hazards associated with nuclear technology. This need for centralized control and specialized expertise leads to a form of social organization that is highly structured and authoritarian, reflecting and reinforcing certain power dynamics within society.
- c. Example: Cotton mills; people had to work a very specific way and in a certain timeframe; sequence is super important (do exactly what you are told to do)

False Example: Energy sources have politics because the impact they have on climate change is an inherently political property.

Main AI Ingredient = Data

We define AI as the “use of technologies to build machines and computers that have the ability to mimic cognitive functions associated with human intelligence.”

We define ML as a subset of AI, and it is the “use of algorithms to automatically learn insights and recognize patterns from data.”

We define supervised learning as a subset of ML, and it is the “use of labeled datasets to train algorithms that to classify data or predict outcomes accurately.”

Values Embedded in ML Pipeline (Sessions 3 - 6)

Basics of supervised learning: We take input data, X , and using a function $f(X)$, predict known labels, Y .

Machine Learning Pipeline:

1. Problem formulation

- a. What is the overarching goal of the system?
 - i. Different stakeholders can have different goals.
- b. What is the mechanism of entry into the population subjected to the algorithm?
 - i. Need to consider who will encounter the algorithm and if they have a choice.
 - ii. Not determined by the sampling approach used to collect training data! It is who will encounter the algorithm!
- c. What is the space of possible decisions?
 - i. Need to consider the decisions the algorithm is informing and alternative decision spaces. This refers to the different decisions that could be made with the algorithm.
- d. Algorithms that solve seemingly the same task can be embedded within entirely different problem formulations, which directly impacts fairness considerations.
 - i. Need to consider what burdens and benefits are being allocated.
- e. Some tasks may be inherently biased and grounded on discriminatory assumptions.
- f. It can be misleading to talk about “fairness”; can be narrowly defined as disparities in performance.
- g. Accuracy and good AI performance can also be harmful!

2. Data collection and representation

- a. Garbage in = Garbage out.
- b. ML algorithms are often trained with convenient, inexpensive data.
- c. Big data \neq Good data.
- d. Is the data collection ethical?
 - i. Need to consider if the people that are represented have given consent, and if that consent is meaningful.
 1. People can be mislead into giving consent or have no alternative.
 - ii. The data market may be noxious in that our personal data may be commoditized (along with our privacy right), which threatens equal fundamental political rights and freedoms, and equal rights to a fair share of social welfare and equality of opportunity; morally objectionable; harmful outcomes to individuals or society.
- e. Sampling Bias: Who is represented?
 - i. Occurs when the distribution of the sampled data does not match the distribution of the population that will encounter the algorithm
 - ii. This matters because the resulting bias can have severe consequences, such as in responding to disasters (among other tasks).
 - iii. Sources:
 1. Access to technology and resources.
 2. Previously served/underserved communities.
 3. Trust in authorities (reporting).
 - iv. Invisibilization in sampling bias often *compounds* previous injustices.
- f. Differential Subgroup Validity: How are people represented?
 - i. Predictive power of features collected may differ across subpopulations.

1. Example: In healthcare, symptoms that are studied, taught, and recorded may only hold diagnostic power for some (skintones).
- ii. Choice may be informed by, and only hold predictive power, in some cultural contexts.

1. Number of credit cards as a positive signal for “creditworthiness.”

- iii. Ability to adapt to a certain choice of features may also differ across groups. Moreover, strategic adaptation to incentives is not possible for everyone.

1. Example: standardized tests - we can incentivize students to invest in tutoring and retake tests, but not everyone can do this.

- iv. Can we measure what we care about: mind the gap

1. $Y \rightarrow$ Quantifiable, imperfect proxy (substitute for ideal target variable)

2. $Y_c \rightarrow$ Complex, multi-faceted (and sometimes contested) outcomes; what we aim for, but even this can be debated (think of capstone)

3. Misleading comparisons:

- a. Human vs Machine is not necessarily a fair comparison; calling one of them “better” can be misleading because it depends on the context; a human and machine may not engage in the same predictive task; it depends on context.

4. Self-fulfilling prophecy: When you assume the proxy is a good target variable substitute, this effect overall outcome prediction evaluations

- v. Human assessments encoded as labels:

1. Target labels are often termed “ground truth”, but they may also encode biases. We have to consider whose views are encoded and valued when humans determine the target labels.

- g. Example Question: An organization is developing a disaster relief system that can support rescue efforts during natural disasters. The algorithm identifies anomalous patterns in social media activity. However, a team inside the organization is concerned that this tool may not provide equal support to all communities. The salient type of bias that this is likely to affect this system is: sampling bias.

3. Optimization objectives and evaluation (can go back to 2)

- a. How do we define good? What does “good performance” mean for an AI system? What is the optimization objective? How do we evaluate performance? We need to consider all of these questions.
- b. Optimizing for overall performance is not ideal because it does not implicitly optimize performance for all subgroups. It can actually lead to disparate performance across subgroups.
- c. Common predictive performance metric: ROC AUC (Area under the ROC Curve).
- d. There is no one-size-fits-all metric of success:
 - i. In healthcare, we often care about performance at low false negative rates (diabetes example) as opposed to false positive rates. This differs for other business contexts.
- e. In many cases accuracy does not capture our business needs.
 - i. Example: A third party platform like Zillow, who could be making an algorithm to price houses, has to maintain both sides between buyer and the seller.

Balancing between overestimating and underestimating in predictions can be tricky, and accuracy does not capture this balance.

- f. Accuracy can be misleading.
 - i. Example: For a system with 90% accuracy, it can be useless if you classify everything as one class and this accuracy is a result of the data distribution.
- g. In ML we often take for granted that “overall performance” is the desired goal. Is something that works “for most” enough? We do not think so for many instances in our society.
 - i. Example: Stairs work for 90% of people, but what about the 10% in wheelchairs?

4. Algorithmic adoption (can go back to 2 or 1)

- a. Incentives for deployment (and overarching goals); need to consider the following
 - i. How is a certain technology integrated into a broader social technical system?
 - 1. In the case of AI-assisted decisions, how are algorithmic recommendations integrated into decisions?
 - ii. Ethical implications for *future of work*; need to consider the following:
 - 1. Who gains from this efficiency? Who owns the labor?
 - 2. Different possibilities under different social/power organizations?
 - 3. Are there trade-offs?
 - 4. Efficiency at what cost?
 - iii. How do algorithmic recommendations interact with human beliefs, values, and abilities?; need to consider the following
 - 1. AI technologies may exacerbate ethical concerns in a socialtechnical system (and humans may exacerbate ethical concerns of algorithm)
 - iv. Feedback loops and algorithmic manipulation:
 - 1. Once deployed, algorithms are *actors* that *alter* the state of the world
 - 2. There are two key ways in which this affects the ethical properties of the algorithm:
 - a. Algorithms can manipulate us: is an algorithm that predicts what you are likely to buy addressing your needs/wants or is it *altering* what you think you want?
 - 3. Once deployed, algorithms create *incentives*; people and organizations may adapt to it
 - a. Different people may have different resources/abilities to adapt (recall standardized tests)
 - 4. Goodhart’s Law: When a measure becomes a target, it ceases to be a good measure.
 - a. In other words, when we use a measure to reward performance, we provide an incentive to manipulate the measure in order to receive the reward.

Algorithmic Bias Detection (Sessions 7 - 9)

Algorithmic Bias:

- One concrete type of harm that may stem from the use of ML

- Defined as algorithmic outputs that may result in disparate outcomes or other forms of injustices for subgroups of the population, especially those who have been historically marginalized
- Important on the basis of legal compliance, social responsibility, and utility

Why is discrimination wrong?

- Argument can be made that the whole point of using ML algorithms to discriminate
- Systematic relative disadvantage: treatment that systematically imposes a disadvantage on one social group relative to others.
- Different normative underpinning for why it is wrong:
 - Relevance: Relying on characteristics that bear little or no relevance to the outcome or quality that decision makers might be trying to predict or assess
 - Generalizations: Needlessly coarse groupings
 - Is there additional info that could provide a more granular view?
 - Example: Provide a fitness test to all firefighter applicants instead of banning women.
 - Prejudice: When decision makers hold entire groups in lesser regard than others
 - Disrespect: Casting certain groups as categorically inferior to others and thus not worthy of equal respect
 - Different from prejudice because it is about the message that the actions of the decision makers is sending.
 - Immutability: Treating people differently according to characteristics over which they possess no control.
 - Compounding injustice: People cannot be morally culpable for certain facts about themselves that are not the result of their own actions.
 - This is especially the case if these facts are the result of some past injustice.

Can it be unjust if it is accurate?

- It may depend on your normative grounding.
- Grounded on a compounding injustice view, there may be *non-accuracy affecting injustices*:
 - Accuracy-affecting injustices: Either the data or output of a model inaccurately estimate a fact about people. (Accuracy is inaccurate)
 - EX: Gaps in FP or FN Rates - emphasis on whether prediction is correct or rate we are incurring errors
 - Non-accuracy-affecting injustices: Data and models accurately estimate a fact about people, but these traits themselves result from injustice.
 - EX: Demographic Parity (gives us different grounds for why we want a certain rate to be the same; only looking at the predictions, not the true labels; just equality between two ratios)

What constitutes a discriminatory algorithm?

- How the law (in the US) conceives of discrimination (predating algorithms)
 - Disparate treatment (procedural justice): The decision was made *because* of a protected attribute

- But-for causation: if the protected attribute was different, the decision would have been different.
- Disparate impact (distributive justice): Disproportionate effect on a protected class, even if unintentional
 - Must be both *unjustified* and *avoidable*

Avoiding disparate treatment does not always ensure we also prevent disparate impact, and vice versa.

Algorithmic fairness deals with algorithmic outputs that may result in disparate outcomes or other forms of injustices for subgroups of the population, especially those who have been historically marginalized.

How do we know if there is bias? To measure bias, we need:

- A definition of bias
- A method to measure it
- Four types of mathematical approaches to quantitatively measure bias:
 - Statistical/group measures
 - Most popular given the practical feasibility of implementation.
 - Quantify statistical disparities across groups.
 - Measures based on confusion matrix.
 - Multiple possible measures, which disparity do we care about? Depends on the business context.
 - Different measures correspond to different philosophical/political notions of fairness, and their relevance may also vary across contexts.
 - Not possible to simultaneously satisfy all the different measures
 - Similarity measures
 - Causal measures
 - Utility-based measures

In training an ML model, assume g represents a sensitive attribute (e.g gender), and is an additional piece of info required that may or may not be part of input X . How can we compare the true labels, y , and the predicted values, \hat{y} , and are the predictions biased when considering g ?

1. Confusion Matrix

- a. TP: Predicted is True, Actual is True
- b. FP: Predicted is True, Actual is False
- c. TN: Predicted is False, Actual is False
- d. FN: Predicted is False, Actual is True

For all measures below, we compare between two different groups, and see if the values between the three are equal or what the difference is between them.

2. Demographic Parity:

- a. Is the rate of people of different groups predicted to have a certain outcome the same? In other words, it is: $\# \text{ predicted} / \text{total population}$.

- b. Somewhat grounded on legal notion of the 80% rule: if selection rates for a job are sufficiently large across groups, it is considered a presumption of adverse impact.
- c. Does not consider the observed/true outcome
 - i. Advantage: Sometimes observed outcome is biased
 - ii. Disadvantage: Sometimes the *true rates* are different, so a perfect prediction give different rates for different groups.

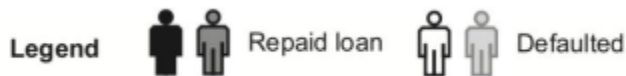
3. Equalized Odds (Recall):

- a. $TP / TP + FN$
- b. Are the predictive errors biased against a certain group?
- c. Considers what the observed outcome is.
- d. Out of all of the true “positive” class, how many did we correctly predict?

4. Predictive Parity (Precision):

- a. $TP / TP + FP$
- b. Are the precision rates biased against a certain group?
- c. Probability that a prediction is correct does not depend on the sensitive attribute.
- d. Out of all the predicted “positive” class, how many did we correctly predict?

Examples:

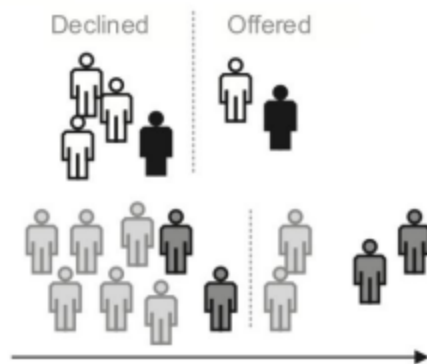


DP: $4/8 = 4/8$

EQ-Odds: $1/1 = 3/3$ (or from FP perspective, $3/7 \neq 1/5$)

Pred-Par: $1/4 \neq 3/4$

(b)

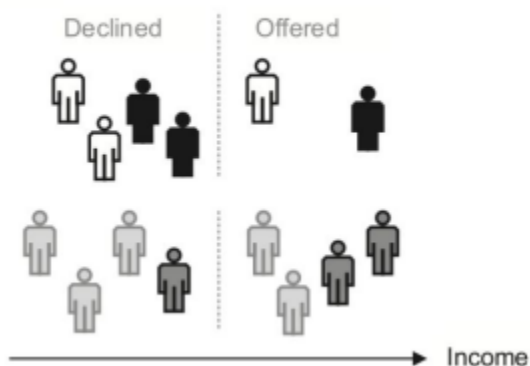


DP: $2/6 = 4/12$

EQ-Odds: $1/2 = 2/4$ (or from FP perspective, $1/4 = 2/8$)

Pred-Par: $1/2 = 2/4$

(c)



DP: $2/6 \neq 4/8$

EQ-Odds: $1/3 \neq 2/3$ (or from FP perspective, $1/3 \neq 2/5$)

Pred-Par: $1/2 = 2/4$

We cannot satisfy everything at once!

- If the **base rates (rate of positives)** are different, i.e. $P(Y=1 | g = 1) \neq P(Y=1 | g = 0)$, then we cannot simultaneously satisfy the three different fairness metrics we have studied. (In other words, **if DP is not met, at least one of EQ-Odds or Pred-Parity will also not be met**)
 - If the base rates (rate of positives) are the same, algorithmic fairness metrics can be simultaneously satisfied. (In other words, **if DP met, it is possible for both EQ-Odds and Pred-Par to be met simultaneously**)
- How do we choose which metric(s) are relevant?
 - It may be useful to report on multiple metrics, and reason over what these mean for a given case.

- When we are going to enforce a certain metric, or use as criteria to choose among multiple algorithms, we may need to choose which metric(s) we care about.
- Guiding criteria:
 - What are the goods and burdens allocated by the possible decisions?
 - What are the previous harms and injustices that the algorithm risks compounding?
 - How will the choice of metric affect other **desiderata**?
- Different metrics may be more relevant to some stakeholders than others

Post Exam 1

Inclusion of Sensitive Attributes

Grounded on the notion of disparate treatment, we may be tempted to think that omitting sensitive attributes will prevent algorithmic bias.

However, omitting sensitive attributes is neither necessary nor sufficient.

- In some cases, including sensitive attributes may reduce group fairness gaps
- In other cases, removing sensitive attributes may only partially reduce group fairness gaps

Compounding Injustice: If initial imbalance constitutes injustice: Model's prediction is informed by, and compounds, previous injustice.

- Essentially, what this is saying is that if the starting point already has unfair differences among people, it's considered wrong or unjust. From there, if a model (like an algorithm or software) makes predictions based on this initial unjust situation, it will make the unfairness worse. Essentially, the model uses the existing unfair data and then acts in a way that adds to or intensifies that injustice, rather than correcting it.

Different Types of Approaches to Mitigate Bias

Pre-Processing Bias Mitigation (Data): Modifications to data prior to training algorithm

Resampling

Oversample statistical minority so that there are same number of instances from each group.

Undersample statistical majority so that there are same number of instances from each group.

There are different “flavors” of resampling - such as a mix of over and undersampling.

Reweighting

Many ML algorithms allow you to provide weights for each instance, which are used during the training process.

- The default is that each instance has a weight of 1.
- If some instances are less/more important, we can reweight.
- Commonly used approach when one type of error is more costly than others, or where there is class imbalance.

When using reweighting for group fairness, the weight corresponds to the group imbalance

Weights can be updated so that sum of weights is the same across groups

$$\sum_{i \in G_1} w_i = \sum_{i \in G_2} w_i$$

Formula Breakdown:

- Assume each instance (e.g., a person or a data point) in the dataset has a weight, which we can denote as w_i
- When you add up the weights of all instances in one group (Group 1), it equals the sum of the weights of all instances in another group (Group 2).
- The formula is simply saying that the total weight for Group 1 is the same as for Group 2 after reweighting.

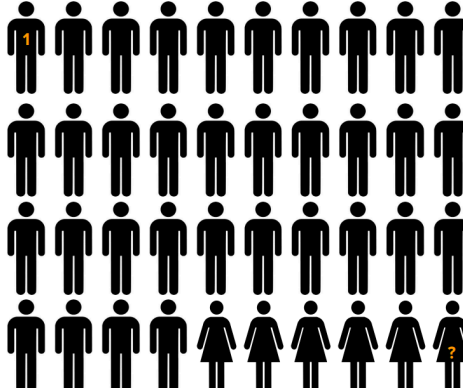
How do we **reweight** to mitigate bias?



Surgeons
women in data:
14.6%

How much weight should we give to women instances, if we are upweighting them to address statistical imbalance?

- ☐ 6.85
- ☐ 14.6
- ☐ 5.8
- ☐ 0.146



To solve this question, we have the % of women in the dataset, so we can get the number of men in the dataset as 100% - 14.6%. This results in 85.4%. To have

equal total weight, a woman's instance weight multiplied by 14.6% should equal a man's instance weight multiplied by 85.4%. So, we can do the % of men in the dataset divided by the % of women in the dataset. So, $85.4/14.6$. This results in 5.8.

Training-Time Constraints (Training): Modifications to the machine learning algorithm

Decoupled Classifiers

Each decoupled classifier is trained with a portion of data that belongs to that subgroup.

- So from an X dataset, you would split/filter X into different subsets based on the different classes the sensitive variable, g, can take.
- Each subset of X would get its own unique function applied to it to come up with predictions.

While there are different considerations/approaches for merging predictions, typically, we use the output probabilities of each of the unique functions per subset of X without modification.

This is motivated by differential subgroup validity.

Training a separate classifier allows us to capture group-specific patterns.

Regularized Loss

Typically, when thinking about how we apply a function to take input data X to predict y, we are solving some sort of optimization problem.

- For instance, we could be trying to maximize some sort of loss function based on parameters we are learning. This loss function depends on what algorithm we are training.

We can add a penalty to the function we are trying to optimize, such as a gap in TPR.

Post-Processing (Output): Modifications to the output of a machine learning algorithm

Group specific classification thresholds

Many ML algorithms provide an output that is a score between 0 and 1. If calibrated, this corresponds to a probability

If scores are calibrated and we want to maximize accuracy, threshold $\tau = 0.5$

- Other goals can lead to other thresholds, e.g. low FPR tolerance

We can choose group specific classification thresholds such that a specific fairness measure (such as the equalized FPR) is satisfied.

This pertains to what probability threshold is set for each group such that if they are above that threshold, they will be classified as the positive class, otherwise they are the negative class. We essentially make this different across groups.

Training time constraints, such as regularized loss, are not appropriate when source of bias is a label bias or there is no data available for one subgroup. It is appropriate when there is differential subgroup validity.

- This is because if you have no data for a group, regularized loss won't help; you can't adjust for bias in a group you have no information about. You need data to understand and adjust for biases using techniques like regularized loss. If there's no data for a subgroup, the model can't learn about it, regardless of the training constraints you apply. Additionally, regularized loss helps address overfitting but doesn't directly correct biases in the labels themselves. If the problem is that the data you're using to train the model is biased, simply regularizing won't fix the issue; you need to correct the labels or get better data.

In settings in which disparate treatment is undesirable, decoupled classifiers are not appropriate and therefore it is desirable to use post-processing approaches, such as group-specific thresholds. This is false.

- This is because if disparate treatment is undesirable, you wouldn't want to use different thresholds for different groups, as that would be a form of disparate treatment. Instead, you'd want to apply the same standards to everyone, adjusting the model as a whole to be fairer, rather than adjusting the outcomes for different groups separately.

Utility-Fairness Relationship

If fairness is a constant, we don't always pay a price; there is not always a tradeoff between utility and fairness.

An optimization problem is not necessarily convex, so there may be more than one solution. Therefore, introducing a fairness constraint does not necessarily lead to reduced utility.

- Many solutions are equivalent in terms of utility but different in terms of fairness.

When considering accuracy affecting injustices, many interventions may improve both utility and fairness.

- More data
- Different data
 - Different features
 - Different target label

When considering non-accuracy affecting injustices, there is an inherent trade-off.

Appropriate Use of Mitigation Strategies

TABLE 3 Guidance for choosing a bias mitigation strategy

Source of bias	Mitigation recommendations
<i>Data collection and representation</i>	
Sampling bias (Subsection 3.1.1)	<ul style="list-style-type: none"> ✓ Revised strategies for data collection to address sampling bias may be most effective ✓ Training models with fairness constraints or post-processing the outcome may be helpful if sampled data has enough signal for inference for undersampled group(s) ⚠ Additional data collection should be done ethically and with informed consent
Differential subgroup validity (Subsection 3.1.2)	<ul style="list-style-type: none"> ✓ Modifications to the algorithm may be most effective to capture cross-group heterogeneity ⚠ Using different predictive patterns for different groups may constitute disparate treatment and be at odds with some regulation
Biased observed outcomes (Subsection 3.1.3)	<ul style="list-style-type: none"> ✓ Revisiting problem formulation may be most effective, especially by acknowledging gap between desired outcome and observed outcome ✓ Data collection may be effective if another (less biased) outcome can be collected ⚠ Training constraints and post-processing approaches may be ineffective and yield misleading results because most of these rely on the assumption that observed outcome is the true outcome
<i>Model estimation</i>	
Optimization objectives and evaluation bias (Subsection 3.2.1)	<ul style="list-style-type: none"> ✓ Controlling bias through modifications to the algorithm likely most effective ⚠ The ability to appropriately diagnose this may be complicated by an over-reliance on chosen performance metrics
Bias ingrained in the assumptions of the objective (Subsection 3.2.2)	<ul style="list-style-type: none"> ✓ Revisiting the problem formulation is likely necessary ⚠ Flaws in problem formulation may imply a “nonsolution” and require a substantial restructuring, which may include deciding that a technology should not be built
<i>Deployment</i>	
Bias in BA adoption (Subsection 3.3)	<ul style="list-style-type: none"> ✓ Modifications to the algorithm may be effective if they improve human-AI complementarity ✓ Implement post-deployment monitoring to routinely probe for bias and potential feedback loops ⚠ Appropriately identifying and mitigating bias requires assessing the human-AI team; considering the algorithm in isolation will not suffice

Basics of LLMs

Natural language processing (NLP) poses machine learning tasks that are a little different from the supervised learning paradigm we have focused on. As a result, the machine learning models are also different.

A core task in NLP is learning how to represent text in a structured format. We need to go from unstructured data to structured data.

History of Text Representations:

1. Bag of Words: Take each unique word as a feature/independent variable in order to represent the content holistically.
 - a. Fancier versions (td-idf: term frequency/inverse document frequency): how important a term is within a document relative to a collection of documents
2. Pre-Trained Embeddings
 - a. Learned representations using lots of data
 - b. Cosine similarity → meaning similarity

- c. Use as a representation for new tasks
- d. What do we do with words that have multiple meanings?
 - i. Contextual Word Embeddings: Embedding is not fixed, but depends on the context in which the word appears.
 - 1. Ex: I crossed the river to get to the bank vs I crossed the street to get to the bank
- e. A core task is predicting the likely sequence of words: what is statistically plausible?
- f. Transformers: Core building block of LLMs: predicting what is plausible sequence of words based on enormous amounts of data,
 - i. Enabled parallelizing the process: faster computation
 - ii. More data led to better performance (no learning curve plateau)

LLMs we know today have two core building blocks:

1. Pretrained language model (enabled by transformers)
2. RLHF: Reinforcement Learning from Human Feedback
 - a. Human feedback used to improve the language model: Human annotators rank outputs of LLMs, then that feedback is used to improve the LLM.

The general flow of an LLM is from a prompts dataset, we sample many prompts which go into an initial LLM. From there, the output of the initial LLM is scored and ranked by a human. This initial output and the corresponding score/rank is then used to train a reward (preference) model, which is used to update the overall language model, subject to a constraint of not changing the model too much.

LLMs predict the most likely sequence of text (based on what is statistically likely in a huge corpus) that will be rewarded by human annotators.

Issues to keep in mind:

1. Data rights: Whose data is used to train the system?
2. What are the environmental costs of training the system?
3. Is what is statistically frequent what we always want?
4. Who are the humans doing the scoring and shaping the system? Whose views are encoded?
5. What are the conditions of workers employed to do this work?

When an LLM outputs a text, we can be certain the same text appears at least once in the training data. This is false. This is because LLMs are not simple lookup tables that output text directly from the training data. Instead, they are trained to understand patterns and relationships between words, phrases, and sentences so that they can generate text that is similar in structure and meaning to the training data but not necessarily an exact copy. The training process involves adjusting internal parameters (weights) based on the vast amount of text the model is exposed to. This training enables the model to predict the next word in a sequence in a way that is statistically likely, given the context provided by the words that come before it.