

Machine Learning Systems Design

Introduction to ML System Design

Lecture 2: Scoping the ML System Design Problem



CE 40959 Spring 2023

Ali Zarezade

[SharifMLSD.github.io](https://github.com/SharifMLSD)

Agenda

1. When to Use ML?
2. Business and ML Objectives
3. ML System Requirements and Constraints
4. Framing ML Problem

1. When to Use ML?

Clearly define the problem

- Outline use cases
- Understand the assumptions



To be or not to be: “ML”

- GMR#1: Don't be afraid to launch a product without machine learning
- GMR#3: Choose machine learning over a complex heuristic

When to use ML?

- The problem is too complex for coding (lots of rules)
- The problem is constantly changing (rules change)
- It is a perceptive problem (hard to find rules)
- It is an unstudied phenomenon (no clues for any rule)
- The problem has a simple objective (yes/no decisions)



When not to use ML?

- Every action of the system must be explainable
- The cost of an error made by the system is too high (healthcare)
- You can solve the problem using a heuristic at a lower cost
- Getting the right data is too hard or impossible
- The phenomenon is unpredictable (stock prices?!)

Determine the priority of ML problem

- Impact of ML
 - business impact
 - benefit in getting inexpensive (but probably imperfect) predictions
 - replace a complex (rules based) part in your engineering project
- Cost of ML
 - the difficulty of the problem
 - the cost of data, and infra
 - the need for accuracy

Determine the priority of ML problem

- Estimate of complexity
 - comparison with other projects (speech recognition)
 - simplifying the problem (chatbot)
- Estimate of return on investment (ROI)

2. Business and ML Objectives

The importance of KPIs

- GMR#2: First, design and implement metrics

Project objectives

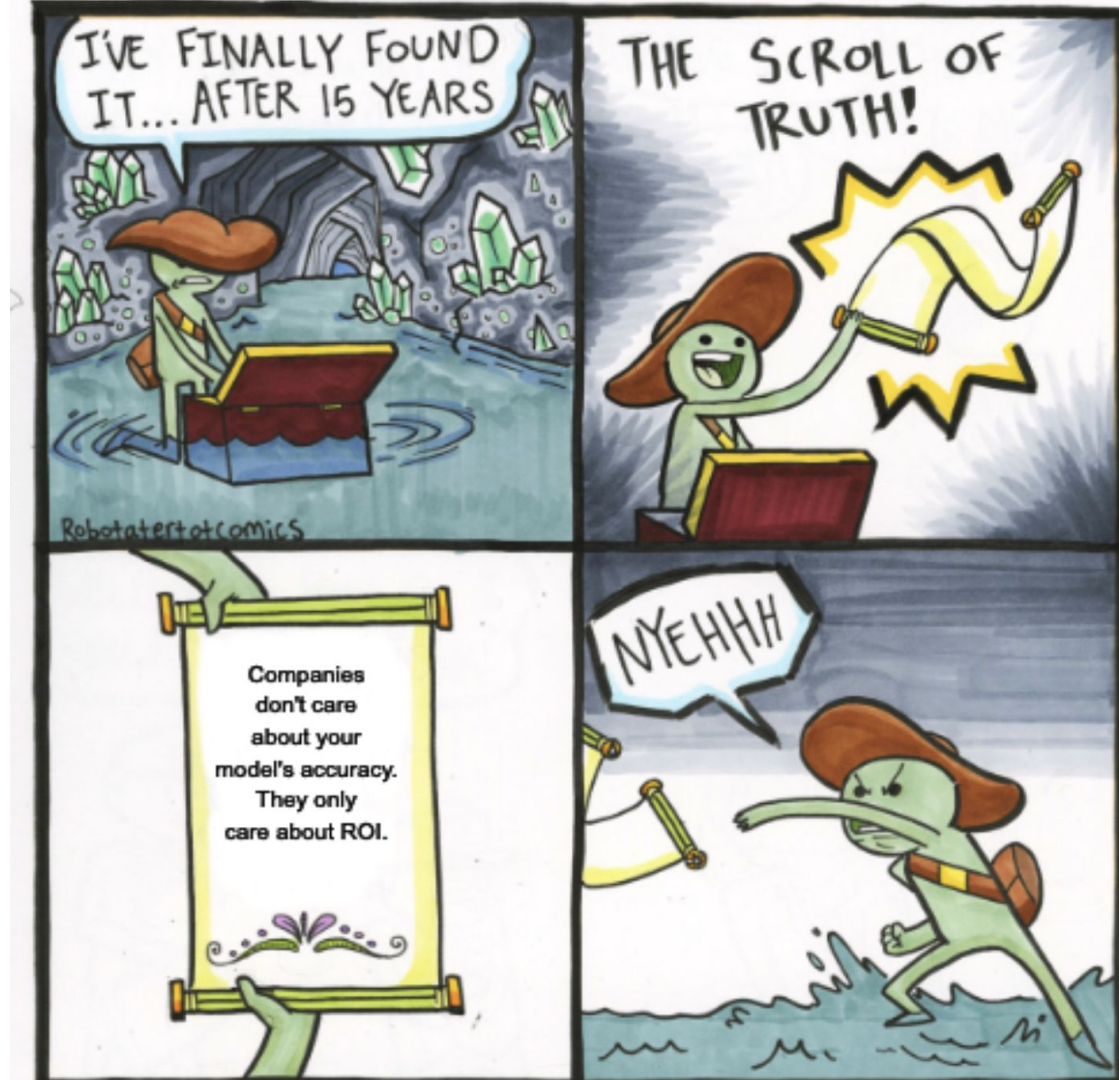
- ML objectives
 - Performance
 - Latency

Project objectives

- ML objectives
 - Performance
 - Latency
- Business objectives
 - Cost
 - ROI
 - Regulation & compliance

Project objectives

- ML objectives
 - Performance
 - Latency
- Business objectives
 - Cost
 - ROI
 - Regulation & compliance

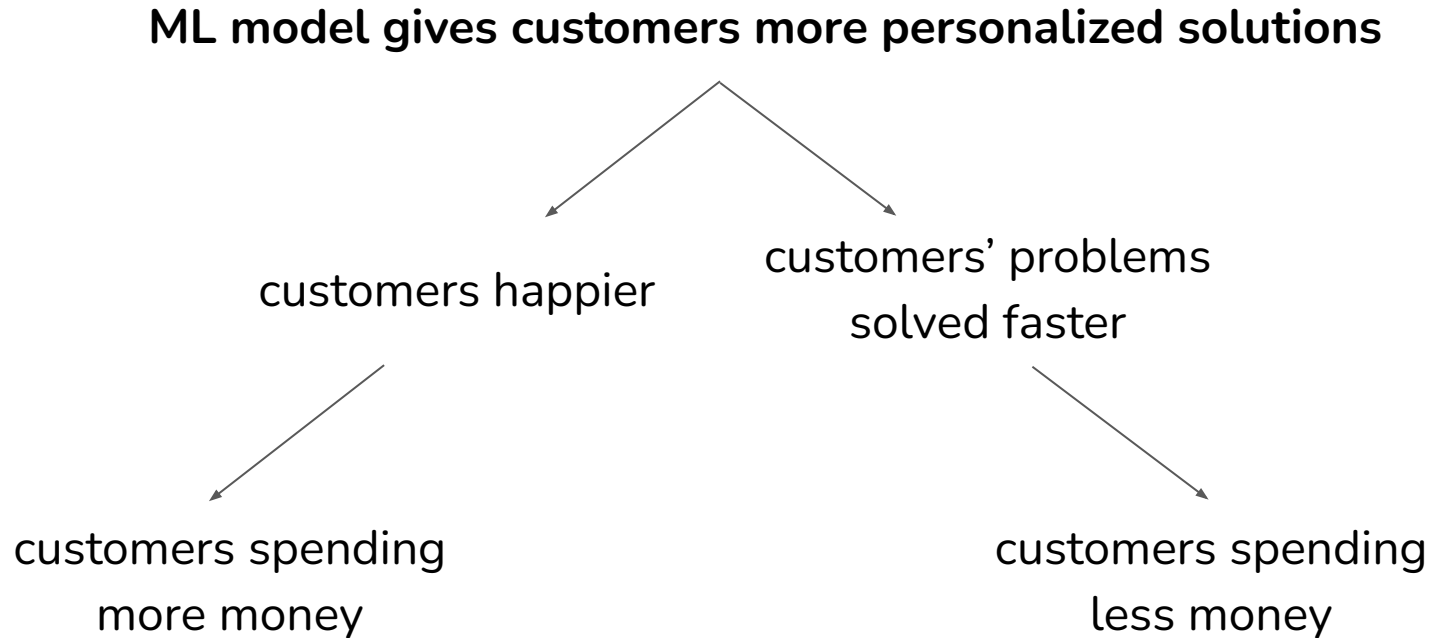


Business objectives

How can this ML project increase profits directly or indirectly?

- Directly: increasing sales (ads, conversion rates), cutting costs
- Indirectly: increasing customer satisfaction, increasing time spent on a website

ML <-> Business: can be tricky



ML <-> Business: mapping

- There are many cases without any clear mapping from business to ML KPIs
 - RecSys, Ads, search
 - Customer satisfaction
 - Segmentation

ML <-> Business: mapping

- **Baselines**
 - Existing solutions, simple solutions, human experts, competitors solutions, etc.
- **Usefulness threshold**
 - Self-driving needs human-level performance. Predictive texting doesn't.
- **False negatives vs. false positives**
 - Covid screening: no false negative (patients with covid shouldn't be classified as no covid)
 - Fingerprint unlocking: no false positive (unauthorized people shouldn't be given access)
- **Interpretability**
 - Does it need to be interpretable? If yes, to whom?
- **Confidence measurement (how confident it is about a prediction)**
 - Does it need confidence measurement?
 - Is there a confidence threshold? What to do with predictions below that threshold—discard it, loop in humans, or ask for more information from users?

Decoupling objectives

Possible high-level goals when building a ranking system for newsfeed?

1. minimize the spread of misinformation
2. maximize revenue from sponsored content
3. maximize engagement

Side note: ethics of maximizing engagement

Several current and former YouTube employees, who would speak only on the condition of anonymity because they had signed confidentiality agreements, said company leaders were obsessed with increasing engagement during those years. The executives, the people said, rarely considered whether the company's algorithms were fueling the spread of extreme and hateful political content.

Employee raises at Facebook depend on engagement, and newly leaked private Zuckerberg recordings show the Groups algorithm prioritizes engagement.

In data terms, anti-vaxx groups and QAnon hysteria are going to get far better engagement than your average drag queen or 'Vote Yes on Proposition Z' groups. Moreover, the group recommendations tool prioritizes the angriest and most out-to-lunch groups, because those tend to get more clicks when they appear in the recommended field

Goal: maximize engagement

Step-by-step objectives:

1. Filter out spam
2. Filter out NSFW content
3. Rank posts by engagement: how likely users will click on them

Wholesome newsfeed

Goal: maximize users' engagement while **minimizing the spread of extreme views and misinformation**

Step-by-step objectives:

1. Filter out spam
2. Filter out NSFW content
3. **Filter out misinformation**
4. **Rank posts by quality**
5. Rank posts by engagement: how likely users will click on them

Decoupling objectives

Goal: maximize users' engagement while minimizing the spread of extreme views and misinformation

Step-by-step objectives:

1. Filter out spam
2. Filter out NSFW content
3. Filter out misinformation
4. Rank posts by quality
5. Rank posts by engagement: how likely users will click on it


How to rank posts by both
quality & engagement?

Multiple objective optimization (MOO)

- Rank posts by quality
 - Predict posts' quality
 - Minimize **quality_loss**: difference between predicted quality and true quality
- Rank posts by how likely users will click on it
 - Predict posts' engagement
 - Minimize **engagement_loss**: difference between predicted clicks and true clicks

One model optimizing combined loss

- Rank posts by quality
 - Predict posts' quality
 - Minimize **quality_loss**: difference between predicted quality and true quality
- Rank posts by how likely users will click on it
 - Predict posts' engagement
 - Minimize **engagement_loss**: difference between predicted clicks and true clicks

$$\text{loss} = \alpha \text{ quality_loss} + \beta \text{ engagement_loss}$$


Train one model to minimize this combined loss
Tune α and β to meet your need

Side note 1: check out Pareto optimization if you
want to learn about how to choose α and β

One model optimizing combined loss

- Rank posts by quality
 - Predict posts' quality
 - Minimize **quality_loss**: difference between predicted quality and true quality
- Rank posts by how likely users will click on it
 - Predict posts' engagement
 - Minimize **engagement_loss**: difference between predicted clicks and true clicks

$$\text{loss} = \alpha \text{ quality_loss} + \beta \text{ engagement_loss}$$

Train one model to minimize this combined loss

Side note 2: this is quite common, e.g. style transfer

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x})$$

One model optimizing combined loss

- Rank posts by quality
 - Predict posts' quality
 - Minimize **quality_loss**: difference between predicted quality and true quality
- Rank posts by how likely users will click on it
 - Predict posts' engagement
 - Minimize **engagement_loss**: difference between predicted clicks and true clicks

$$\text{loss} = \alpha \text{ quality_loss} + \beta \text{ engagement_loss}$$

Train one model to minimize this combined loss

 Every time you want to tweak α and β , you have to retrain your model!



Multiple models: each optimizing one objective

- Rank posts by quality
 - Predict posts' quality
 - Minimize **quality_loss**: difference between predicted quality and true quality
- Rank posts by how likely users will click on it
 - Predict posts' engagement
 - Minimize **engagement_loss**: difference between predicted clicks and true clicks

M_q : optimizes **quality_loss**
 M_e : optimizes **engagement_loss**

Rank posts by $\alpha M_q(\text{post}) + \beta M_e(\text{post})$

Now you can tweak α and β without retraining models

Decouple different objectives

- Easier for training:
 - Optimizing for one objective is easier than optimizing for multiple objectives
- Easier to tweak your system:
 - E.g. α % model optimized for quality + β % model optimized for engagement
- Easier for maintenance:
 - Different objectives might need different maintenance schedules
 - **Spamming techniques** evolve much faster than the way **post quality** is perceived
 - **Spam filtering systems** need updates more frequently than **quality ranking systems**

Establish a single evaluation metric to optimize

- Classification
 - precision
 - recall
 - FPR
 - TPR

Establish a single evaluation metric to optimize

- Classification
 - precision
 - recall
 - FPR
 - TPR
 - F1

Establish a single evaluation metric to optimize

- Classification
 - precision
 - recall
 - FPR
 - TPR
 - micro/macro/weighted F1

Establish a single evaluation metric to optimize

- Classification
 - precision
 - recall
 - FPR
 - TPR
 - micro/macro/weighted F1
 - kappa

Establish a single evaluation metric to optimize

- Consider a learning algorithm with accuracy and latency (inference time)
 - accuracy?
 - latency?

Establish a single evaluation metric to optimize

- Consider a learning algorithm with accuracy and latency (inference time)
 - accuracy?
 - latency?
 - $\text{accuracy} - 0.5 * \text{latency}$?

Establish a single evaluation metric to optimize

- Consider a learning algorithm with accuracy and latency (inference time)
 - accuracy?
 - latency?
 - accuracy - 0.5*latency?
 - max accuracy in models with latency < 0.01s

Establish a single evaluation metric to optimize

- Consider a learning algorithm with accuracy and latency (inference time)
 - accuracy?
 - latency?
 - accuracy - 0.5*latency?
 - max accuracy in models with latency < 0.01s
- If you are trading off N different criteria, consider N-1 metrics as satisficing and optimize only in one metric

When to change or add a metric

- ?

When to change or add a new metric

- The metric is measuring something other than what the project needs to optimize
- Notice a problem? Add a metric to track it!
- Excited about some quantitative change on the last release? Add a metric to track it!

3. ML System Requirements and Constraints

Outline your system requirements and constraints

- *Requirements*: what we want to happen
- *Constraints*: real-world limits around what we want to happen

ML system requirements

They vary from use case to use case, but, most systems should have these four characteristics:

- Reliability
- Scalability
- Maintainability
- Adaptability

ML system requirements

- Reliability
 - continuing to work correctly, even when things go wrong (fault-tolerant)
- Scalability
 - system's ability to cope with increased load
- Maintainability
 - operability, simplicity, and evolvability
- Adaptability
 - adapt to shifting data distributions and business requirements

ML system constraints

- Time
 - Rule of thumb: 20% time to get initial working system, 80% on iterative development
- Budget
 - Data, resources, talent

Constraints: Time/budget tradeoffs

- Use more (powerful) machines
- Hire more people to label data faster
- Run more experiments in parallel
- Buy existing solutions

ML system constraints

- Time
 - Rule of thumb: 20% time to get initial working system, 80% on iterative development
- Budget
 - Data, resources, talent
- Privacy
 - Data, model

Constraints: privacy

- Annotation
 - Can data be shipped outside organizations for annotation?
- Storage
 - What kind of data are you allowed to store? How long can you store it?
- Third-party solutions
 - Can you share your data with a 3rd party (e.g. managed service)?
- Regulations
 - What regulations do you have to conform to?

ML system constraints

- Time
 - Rule of thumb: 20% time to get initial working system, 80% on iterative development
- Budget
 - Data, resources, talent
- Privacy
 - Data, model
- Technical
 - Competitors
 - Legacy systems

Constraints: technical

- Competitors
- Legacy systems



Chip Huyen @chipro · Dec 3, 2020

I'm of the increasing belief that the main technical challenge for companies to successfully adopt ML isn't the lack of functionality, but legacy systems.

The bigger a company is, the more existing tools it uses, and the slower it will be in adopting new tools.

8:23 PM · Dec 3, 2020 · Twitter Web App

Jeremy Kun @jeremyjkun · Dec 3, 2020
Replying to @chipro

Hell even Google has this problem

1 5

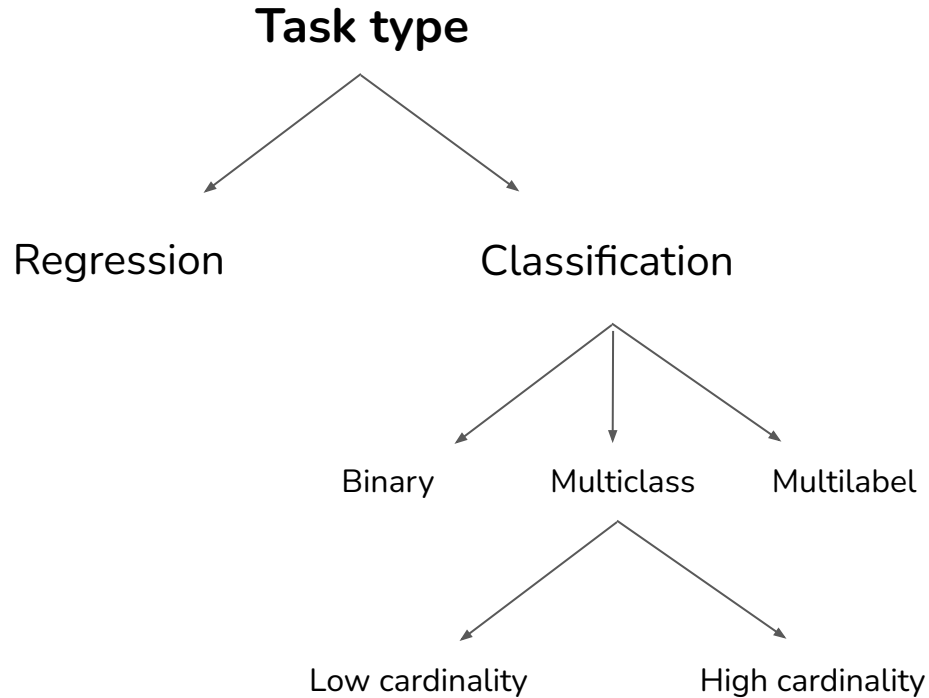
Jeremy Kun @jeremyjkun · Dec 3, 2020

I'd you've got no legacy system you can start fresh with ML, if you start with any existing system you have to prove the ML is better, a hurdle the original system never had to overcome.

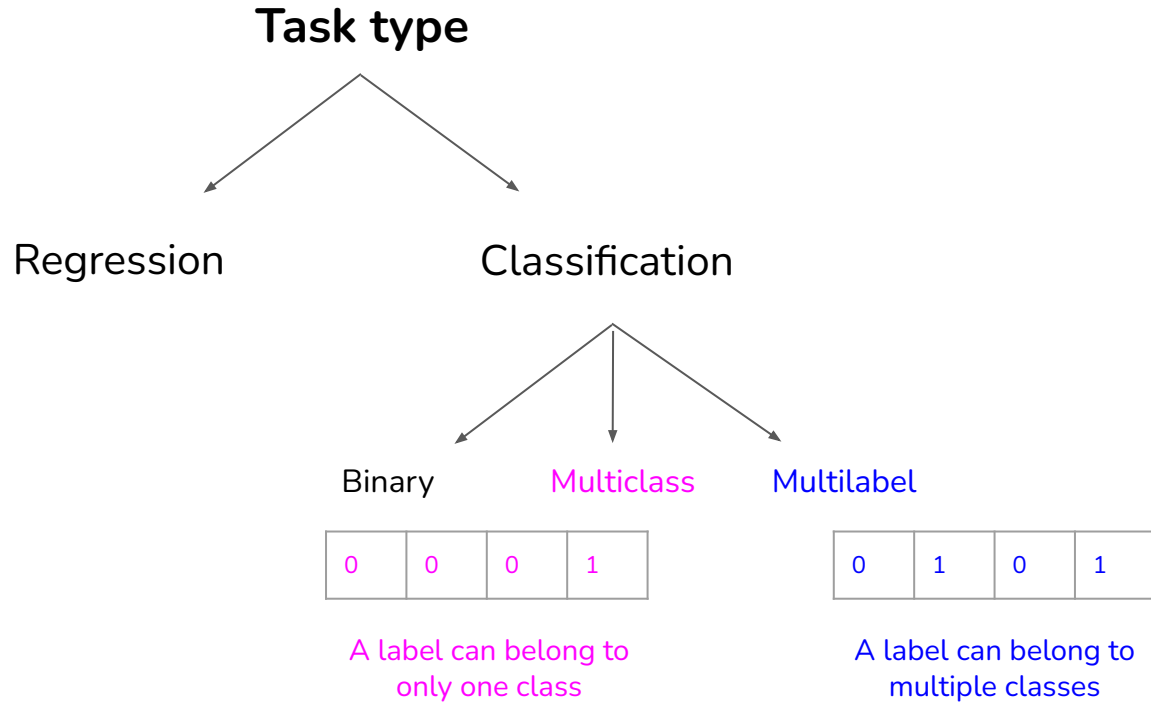
1 8

4. Framing ML Problem

Types of ML tasks



Multiclass vs. multilabel



How to handle multilabel tasks

Multilabel problem solution

A multiclass problem

A set of multiple binary problems

0	1	0	1
---	---	---	---

Model 1:
Does this
belong to
class 1?

Model 2:
Does this
belong to
class 2?

...

Multilabel is harder than multiclass

Multilabel problem solution

A multiclass problem A set of multiple binary problems

0	1	0	1
---	---	---	---

Model 1:
Does this
belong to
class 1?

Model 2:
Does this
belong to
class 2?

...

1. How to create ground truth labels?
2. How to decide decision boundaries?

Multilabel: decision boundaries

Multilabel problem solution

A multiclass problem

0	1	2	3
0.45	0.33	0.2	0.02

Poll:
Which classes should this
example belong to?

1. 0
2. 0, 1
3. 0, 1, 2

A set of multiple binary
problems

Model 1:
Does this
belong to
class 1?

Model 2:
Does this
belong to
class 2?

...

Framing can make the problem easier/harder

Problem: predict the app users will most likely open next

Regression

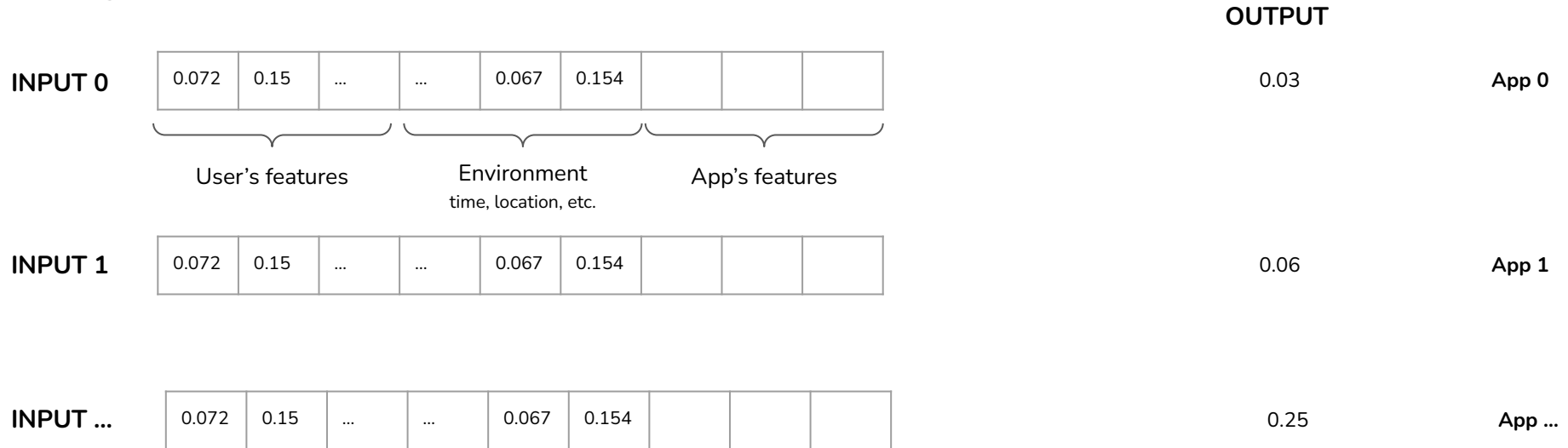
		OUTPUT										
INPUT 0	<table border="1"><tr><td>0.072</td><td>0.15</td><td>...</td><td>...</td><td>0.067</td><td>0.154</td><td></td><td></td><td></td></tr></table> <p>User's features Environment time, location, etc. App's features</p>	0.072	0.15	0.067	0.154				0.03	App 0
0.072	0.15	0.067	0.154							
INPUT 1	<table border="1"><tr><td>0.072</td><td>0.15</td><td>...</td><td>...</td><td>0.067</td><td>0.154</td><td></td><td></td><td></td></tr></table>	0.072	0.15	0.067	0.154				0.06	App 1
0.072	0.15	0.067	0.154							
INPUT ...	<table border="1"><tr><td>0.072</td><td>0.15</td><td>...</td><td>...</td><td>0.067</td><td>0.154</td><td></td><td></td><td></td></tr></table>	0.072	0.15	0.067	0.154				0.25	App ...
0.072	0.15	0.067	0.154							

Framing can make the problem easier/harder

Problem: predict the app users will most likely open next

Very common framing for recommendations / ads CTR

Regression



Data requirements

- Labeled or unlabeled data
- Minimum required data
- Public or private dataset

Baseline and HPL

- Is there any baseline, what's its performance?
- What's the performance of state-of-the-art?
- What's the human performance level?

Machine Learning Systems Design

Introduction to ML System Design

Next Lecture: ML System Development Life Cycle



CE 40959 Spring 2023

Ali Zarezade

[SharifMLSD.github.io](https://github.com/SharifMLSD)