LLMs for Optimization

Modeling, Solving, and Validating with Generative AI

Connor Lawless, Ellen Vitercik, Leonard Boussioux, Madeleine Udell

AAAI 2026 | January 20th, 2026

Utilizing Operations Research and Analytics to Bring an End to Global Hunger

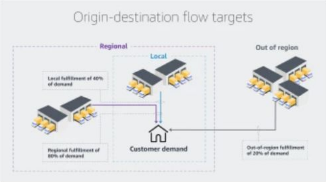
2024 INFORMS Annual Meeting Kicks Off with Plenary from World Food Programme

By Ashley Smith

SHARE: PRINT ARTICLE: https://doi.org/10.1287/orms.2024.03.21n

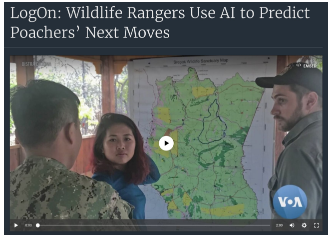
Maximizing the efficiency of Amazon's own delivery networks

Staff writer | October 14, 2022



INFORMS talk explores techniques Amazon's Supply Chain Optimization Technologies organization is testing to fulfill customer orders more efficiently.

OPERATIONS RESEARCH AND OPTIMIZATION



The unsung engineering behind Cornell's fall 2020 schedule

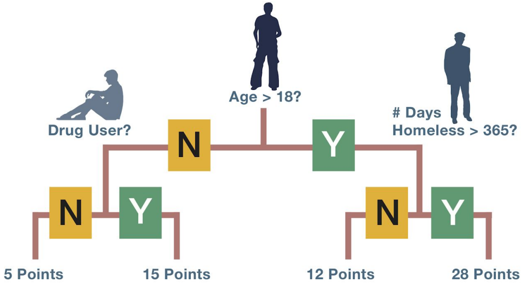
Optimizing the Path Towards Plastic-Free Oceans

Dick den Hertog, Jean Pauphilet, Yannick Pham, Bruno Sainte-Rose, Baizhi Song

Bus Routing Optimization Helps Boston Public Schools Design Better Policies

Dimitris Bertsimas, Arthur Delarue, William Eger, John Hanlon, Sebastien Martin

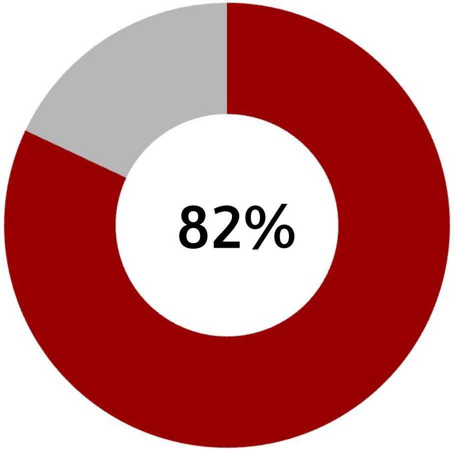
Published Online: 24 Jan 2020 | https://doi.org/10.1287/inte.2019.1015



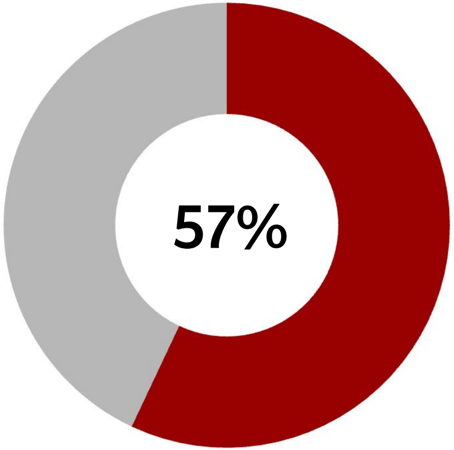
Optimization is inaccessible!

Most of Gurobi's users are optimization experts!

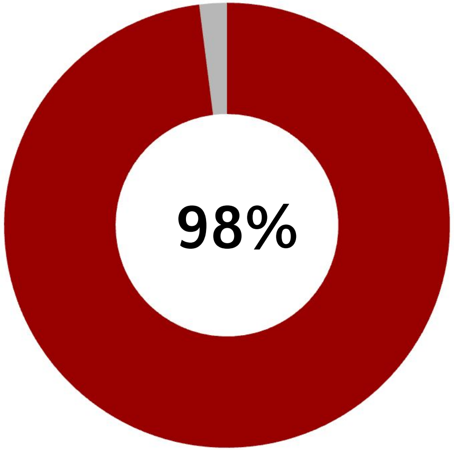
Have a Graduate Degree



Have a Degree in OR



Growing Demand for OR



State of Mathematical Optimization. *Gurobi (2024)*

Optimization is inaccessible!

The accessibility of optimization tools is becoming an existential problem in OR:

Making Operations Research More Accessible: Insights from the Rise of Machine Learning

Tho V. Le,[a,*] Laura A. Albert,[b] Thibaut Vidal[c]

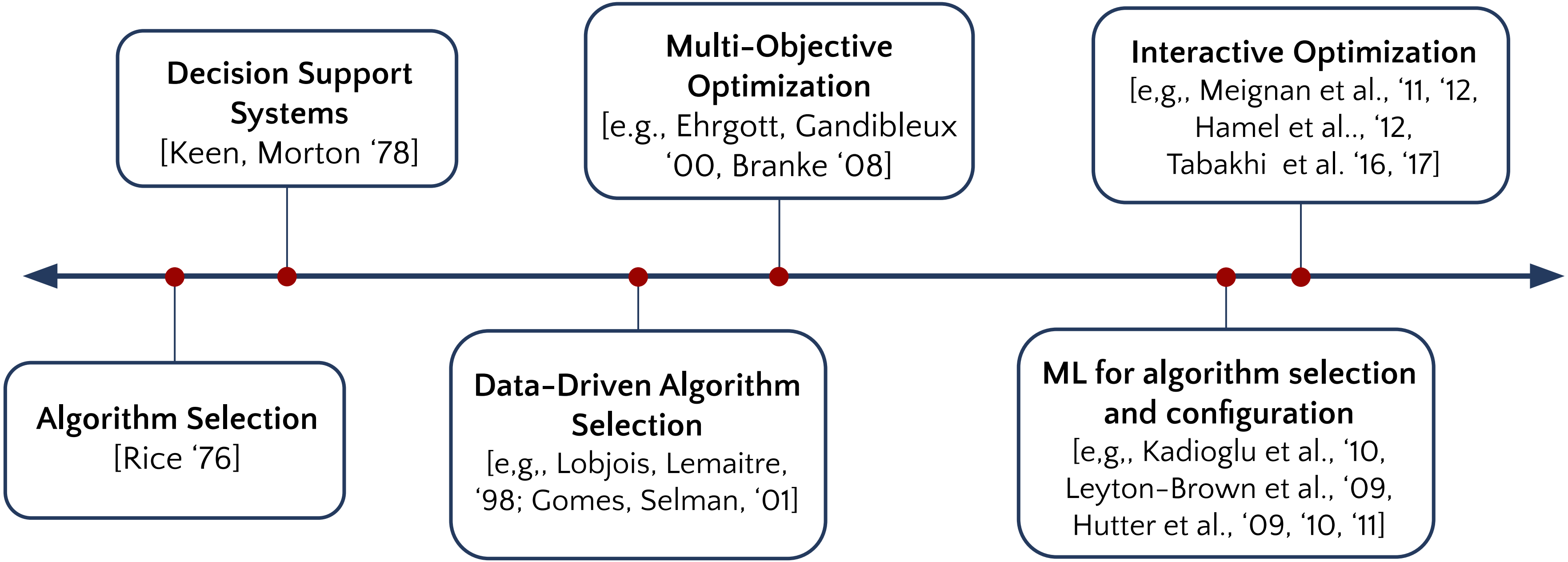
[a] School of Engineering Technology, Purdue University, West Lafayette, Indiana 47906; [b] Department of Industrial & Systems Engineering, University of Wisconsin-Madison, Madison, Wisconsin 53706; [c] Department of Mathematics & Industrial Engineering, Polytechnique Montréal, Montréal, Quebec H3T 0A3, Canada

3.3. Research and Technology

Leveraging technology has the potential to emphasize how OR utilize big data and advanced analytics to solve complex problems as well as to position it as a complementary field to ML in data-driven decision making (Action 7). Developing and promoting user-friendly OR software tools can empower practitioners to apply optimization techniques without requiring deep technical expertise, thereby democratizing the use of decision-making tools and broadening its user base. Key to success is in lowering the barrier of entry for users without advanced degrees in OR to engage with AI. Examples that have achieved this include the INFORMS Insights webinar series (INFORMS 2024b) and courses designed to make AI accessible to undergraduates in OR (see Albert 2025).

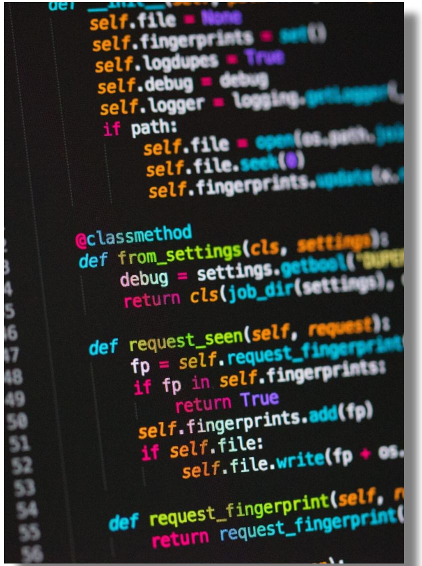
Rich History

Traditional approaches to improving the accessibility of optimization has focused on automating or personalizing small structured aspects of optimization.



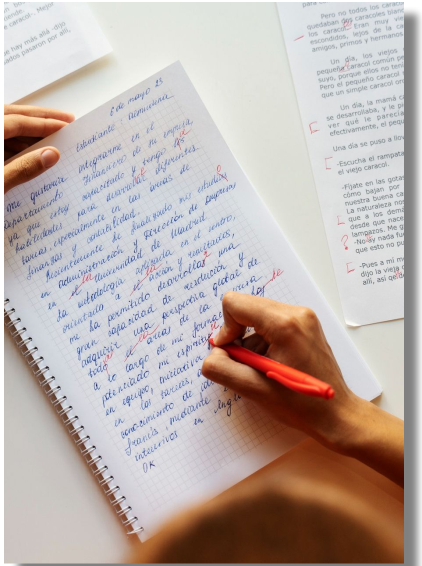
Generative AI: A New Hope

Generative AI has been used to help bridge expertise gaps in a number of domains:



Coding
(e.g., Chen et al., 2021)

SQL
(e.g., Pourreza et al., 2025)

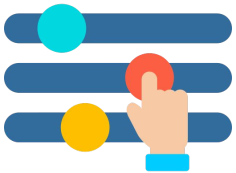


Writing
(e.g., Yeh et al., 2024)

Can we leverage **generative AI** to **democratize access to optimization**?

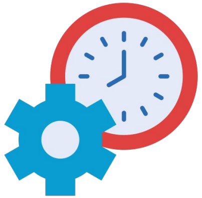
Democratizing Optimization

Reducing barriers to accessing optimization tools has the potential for big impact:



Personalization
Enable general purpose optimization models to be customized for individual users.

Democratization
Enable non-expert users to construct and run optimization models without OR Expertise



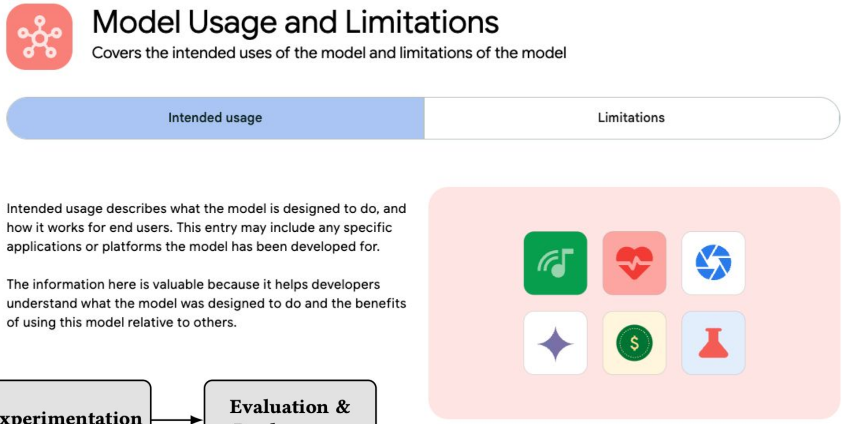
Efficiency
Support OR experts in developing (and tuning) models quickly.

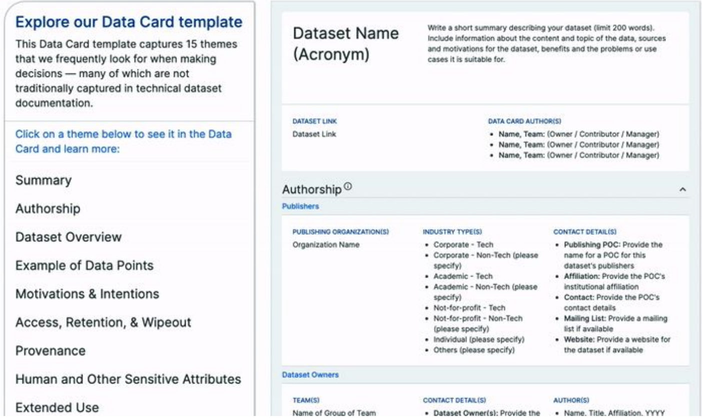
Can we leverage generative AI to democratize **access to optimization**?

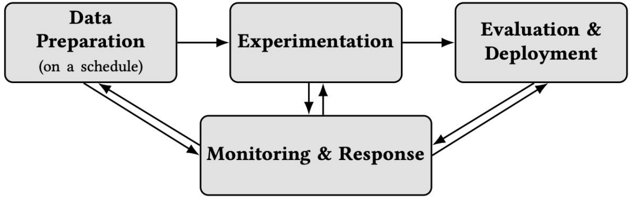
How do OR Practitioners actually solve real-world problems?

Understanding Machine Learning Practitioners' Data Documentation Perceptions, Needs, Challenges, and Desiderata

AMY K. HEGER, Microsoft, USA
LIZ B. MARQUIS, University of Michigan, School of Information, USA
MIHAELA VORVOREANU, Microsoft, USA
HANNA WALLACH, Microsoft, USA
JENNIFER WORTMAN VAUGHAN, Microsoft, USA







Assessing the Fairness of AI Systems: AI Practitioners' Processes, Challenges, and Needs for Support

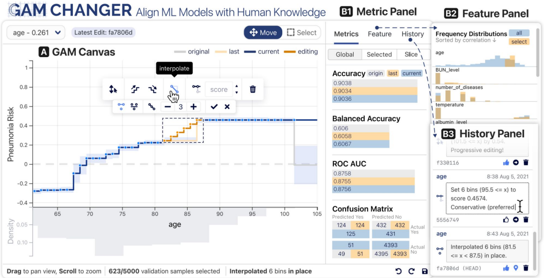
MICHAEL MADAIO, Microsoft Research, USA
LISA EGEDE, Carnegie Mellon University, USA
HARIHARAN SUBRAMONYAM, Stanford University, USA
JENNIFER WORTMAN VAUGHAN, Microsoft Research, USA
HANNA WALLACH, Microsoft Research, USA

Investigating statistical machine learning as a tool for software development

Authors: Kayur Patel, James Fogarty, James A. Landay, Beverly Harrison | Authors Info & Claims

How Data Science Workers Work with Data: Discovery, Capture, Curation, Design, Creation

Authors: Michael Muller, Ingrid Lange, Dakuo Wang, David Piorkowski, Jason Tsay, Q. Vera Liao, Casey Dugan, Thomas Erickson | Authors Info & Claims



Qualitative Study of Practitioners

We ran the **first qualitative study** of optimization practitioners to understand their workflows and pain points.

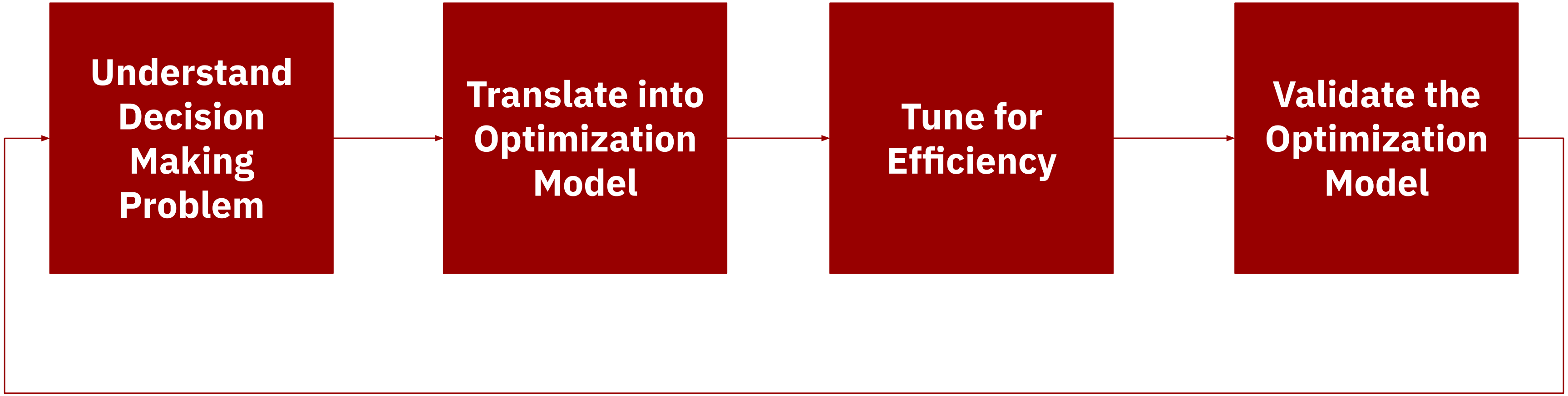
15
Participants

7
Industries

650+
Minutes

Optimization Workflow

Practitioners described a slow, iterative process that required sustained dialogue with domain experts, integrating messy sparse data, and pragmatic trade-offs.



Tutorial Gameplan

1. Background:
    a. Mathematical Optimization
    b. Large Language Models (LLMs)
2. Model Formulation

Coffee Break

3. Model Solving
4. Model Validation
5. Open Questions

Tutorial Gameplan

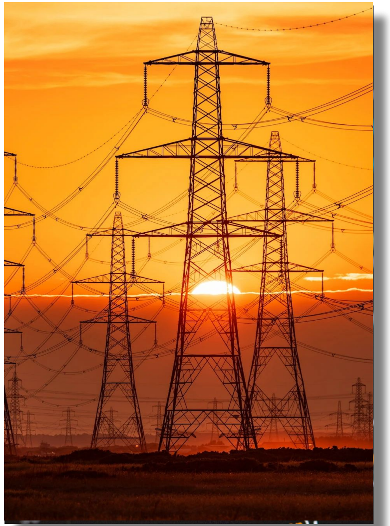
1. **Background**:
   a. **Mathematical Optimization**
   b. Large Language Models (LLMs)
2. Model Formulation

Coffee Break

3. Model Solving
4. Model Validation
5. Open Questions

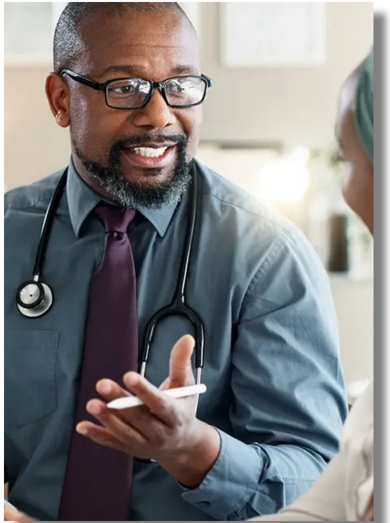
Mathematical Optimization

Optimization is a framework for selecting the best element from a set of alternatives.



Power Flow in a Grid

Planning Bus Routes



Scheduling a OR

We group optimization problems by the class of functions/constraints we optimize over.

E.g., Quadratic Programming – optimize a quadratic function over linear constraints

This tutorial will focus (mostly) on
Mixed-Integer Linear Programming (MILP)

Why MILP?

Flexible and Powerful Modeling Language
MILP can capture a number of important industrial applications including scheduling, routing, and production planning.

Modeling is hard!
As we will see later, a MILP problem can be modeled different ways – each with vastly different computational performance.

Well-Developed Solvers
Commercial solvers like Gurobi and CPLEX can handle large industrial scale problems out of the box!

Building Blocks of a MILP

MILP gives us a common language for discussing optimization problems.
At a high-level there are three basic components:

$$
\begin{aligned}
\text{maximize} \quad & \boldsymbol{c}^T \boldsymbol{x} \\
\text{subject to} \quad & A\boldsymbol{x} \leq \boldsymbol{b} \\
& \boldsymbol{x} \geq \boldsymbol{0} \\
& x_j \in \mathbb{Z} \qquad \text{for some or all } j \in [n].
\end{aligned}
$$

Building Blocks of a MILP

MILP gives us a common language for discussing optimization problems.
At a high-level there are three basic components:

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} \quad & A\boldsymbol{x} \leq \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0} \\ & x_j \in \mathbb{Z} \qquad \text{for some or all } j \in [n]. \end{aligned}$$

Decision Variables: Describe choices under our control

Building Blocks of a MILP

MILP gives us a common language for discussing optimization problems.
At a high-level there are three basic components:

$$\begin{aligned} \text{maximize} \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} \quad & A\boldsymbol{x} \leq \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \\ & x_j \in \mathbb{Z} \quad \text{for some or all } j \in [n]. \end{aligned}$$

Constraints: Limitations restricting our choices for the decision variables

Building Blocks of a MILP

MILP gives us a common language for discussing optimization problems.
At a high-level there are three basic components:

$$
\begin{aligned}
\text{maximize} \quad & \boldsymbol{c}^T \boldsymbol{x} \\
\text{subject to} \quad & A\boldsymbol{x} \leq \boldsymbol{b} \\
& \boldsymbol{x} \geq \boldsymbol{0} \\
& x_j \in \mathbb{Z} \qquad \text{for some or all } j \in [n].
\end{aligned}
$$

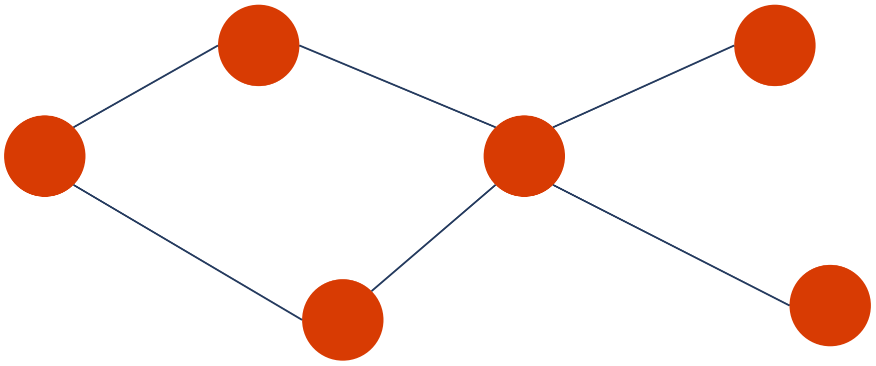
Objective: Criterion we want to maximize/minimize

Example: Maximum Independent Set

Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.



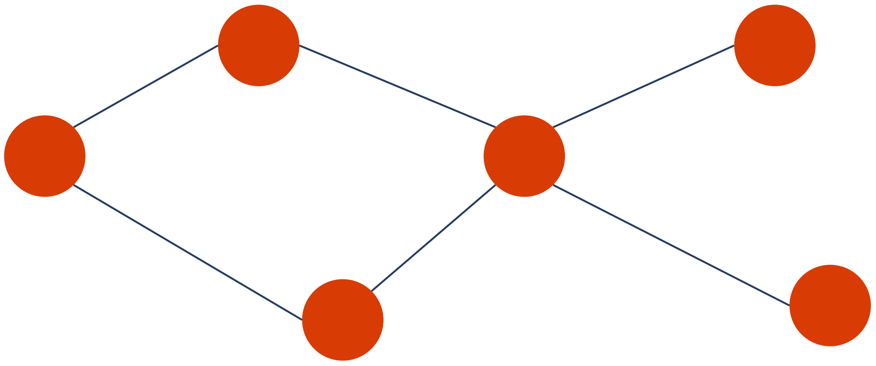
$G = (V, E)$

Example: Maximum Independent Set

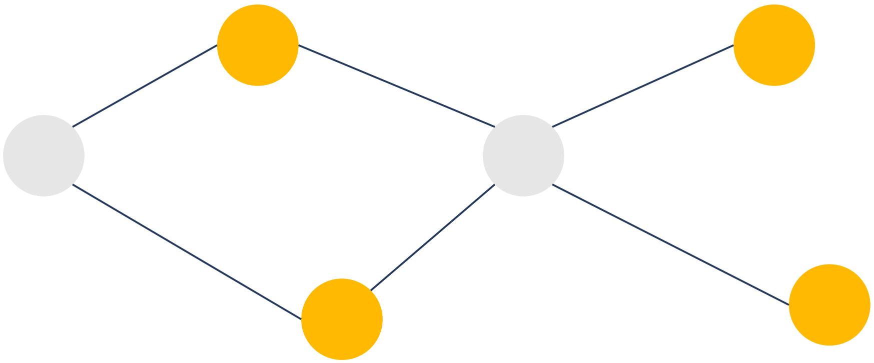
Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.



$G = (V, E)$



Maximum Independent Set

Example: Maximum Independent Set

Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.

Decision Variables:
For each vertex $i \in V$ we define the decision variable:

$$x_i = \begin{cases} 1 & \text{if } i \text{ is in the independent set} \\ 0 & \text{else.} \end{cases}$$

Example: Maximum Independent Set

Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.

Constraints:

We need to ensure our set is independent, we can do so with:

$$x_i + x_j \leq 1 \text{ for all } (i, j) \in E$$

Example: Maximum Independent Set

Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.

Objective:
We want to maximize the size of our independent set:

$$\sum_{i \in V} x_i$$

Example: Maximum Independent Set

Consider a graph $G = (V, E)$.

A subset of nodes $S \subseteq V$ is an independent set if no vertices in S are connected.

The *Maximum Independent Set (MIS) problem* is to find the largest independent set.

MILP Formulation:

$$\begin{aligned} \text{maximize} \quad & \sum_{i \in V} x_i \\ \text{subject to} \quad & x_i + x_j \leq 1 \quad \text{for all } (i,j) \in E \\ & x_i \in \{0,1\} \quad \text{for all } i \in V. \end{aligned}$$

Strength of a Formulation

There are often **many ways to model the same optimization problem** (it's an art)!

We can compare formulations by looking at the ***strength of their linear relaxation*** (*how well does the linear relaxation capture the integer points)*

$$\max \sum_{i \in \mathcal{V}} x_v$$

$$x_i + x_j \leq 1 \quad \forall (i,j) \in E$$

$$\sum_{i \in k} x_i \leq 1 \quad \forall k \in \mathcal{K}$$

$$x_i \in \{0,1\} \quad \forall i \in \mathcal{V}$$

Ex: Clique Constraints

Preserves all integer solutions, but cuts out parts of the linear relaxation!

Scaling MILP Solvers

Modern solvers for MILP are built on an algorithm called branch-and-cut.
To scale to large problems there is a common toolbox:

Cutting Planes
Algorithms to develop inequalities that strengthen a linear relaxation

Primal Heuristics
Algorithms to find good feasible solutions (can help warm-start solver).

Decomposition Algorithms
Divide and conquer approach for tackling large-scale problems.

Tutorial Gameplan

1. **Background**:
   a. Mathematical Optimization
   b. **Large Language Models (LLMs)**
2. Model Formulation

Coffee Break

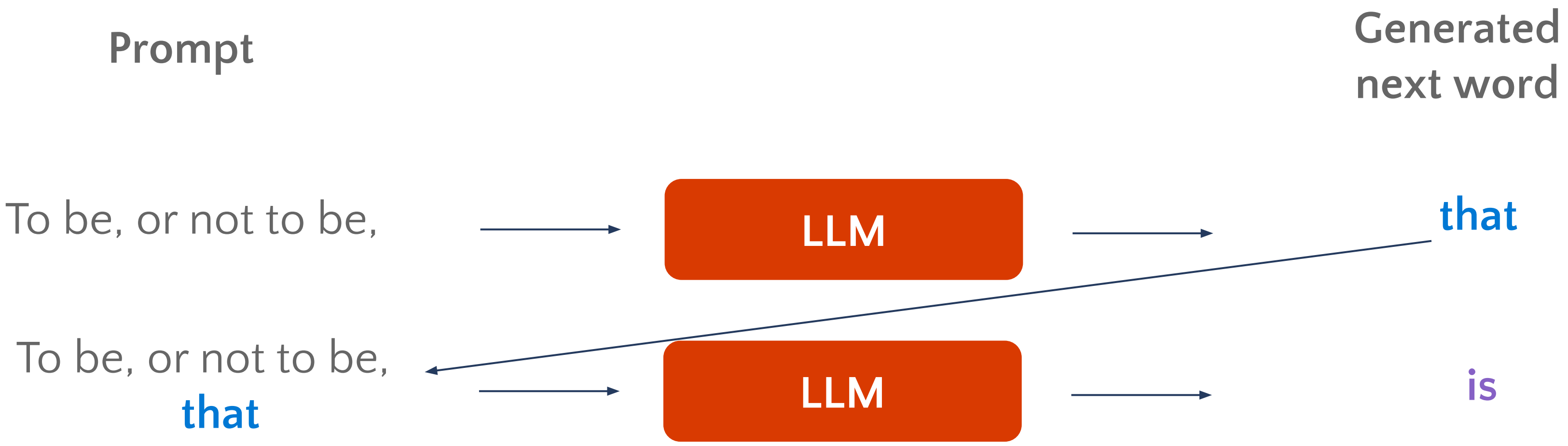
3. Model Solving
4. Model Validation
5. Open Questions

Large Language Models (LLMs)

Given an input (i.e., a “prompt”), **LLMs can pick the most likely next word or sample a word from the output distribution**.

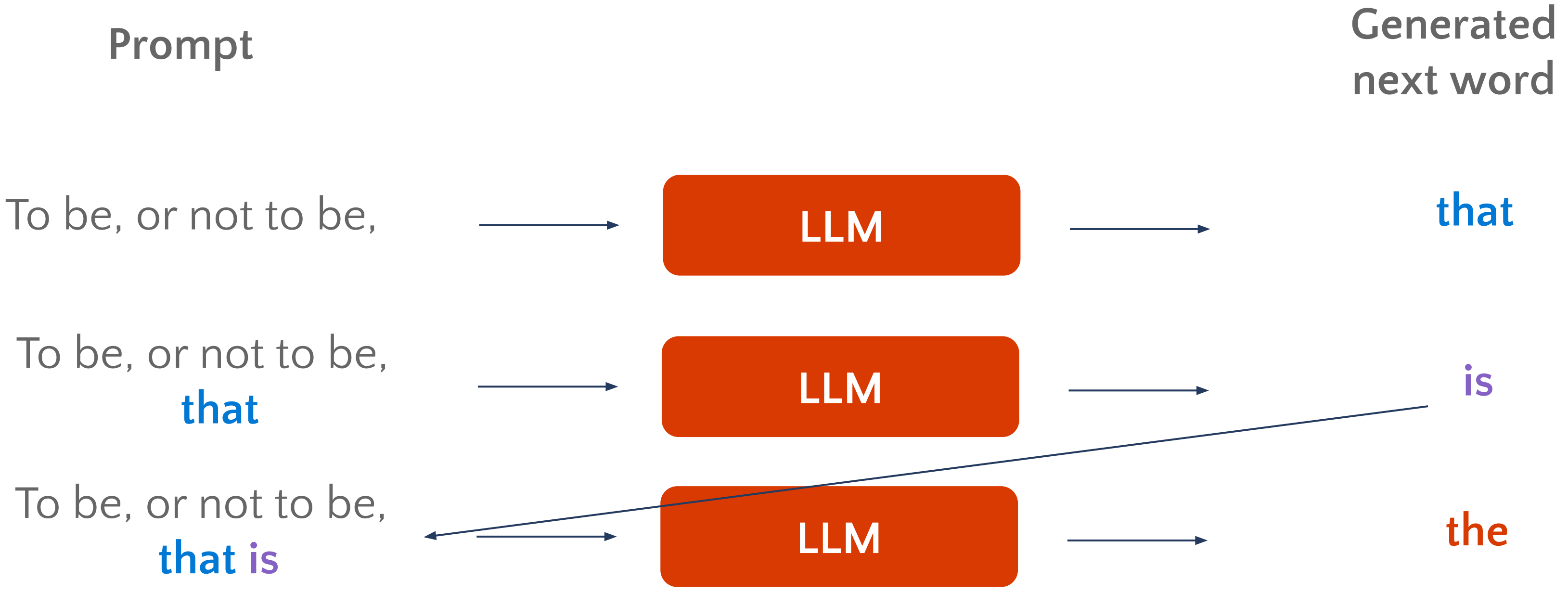
Large Language Models (LLMs)

Given an input (i.e., a "prompt"), **LLMs can pick the most likely next word or sample a word from the output distribution**.



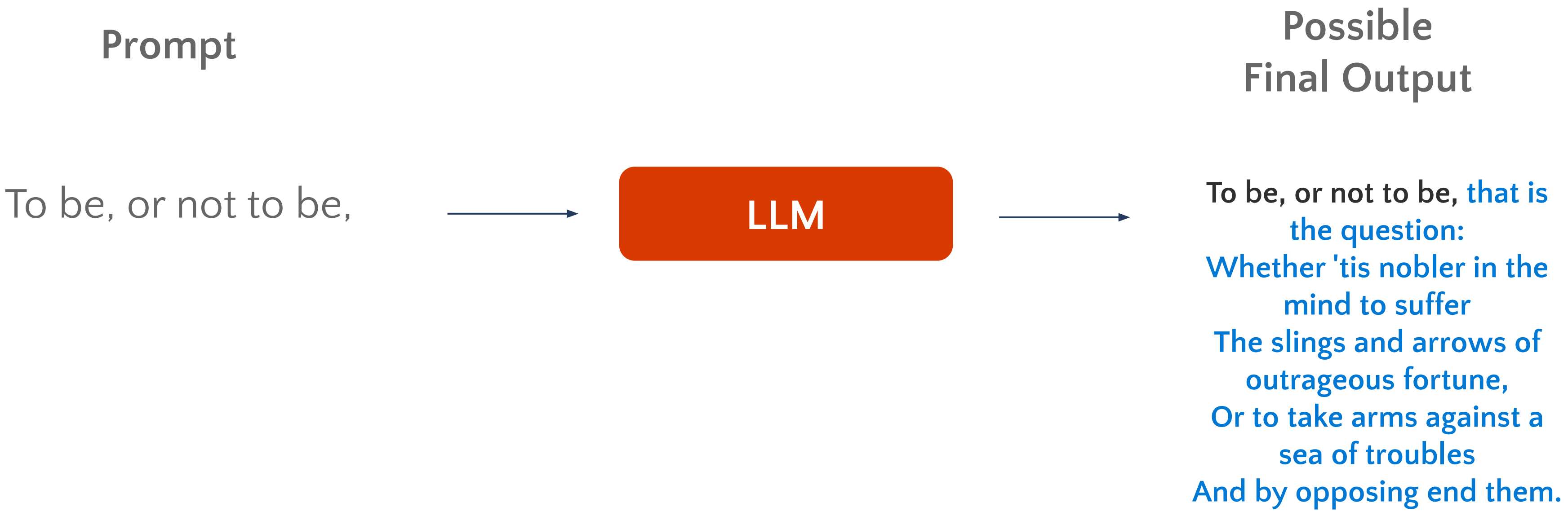
Large Language Models (LLMs)

Given an input (i.e., a "prompt"), **LLMs can pick the most likely next word or sample a word from the output distribution**.



Large Language Models (LLMs)

… the (auto-regressive) process repeats until some stopping condition is met.



What makes LLMs special?

What makes LLMs different from traditional machine learning techniques?

Special architecture: Transformers
LLMs use a special type of deep neural network that includes features such as: self-attention, contextual understanding, and parallel processing.

A very large number of parameters (weights)
Often more than 1B weights!

Very large training data
LLMs are trained on gigantic datasets of text and code.

Disclaimer: We will mostly be using LLMs as a black-box in this tutorial!*

Disclaimer: We will mostly be using LLMs as a black-box in this tutorial!*

*Check out this link for more resources on generative AI

Prompt Engineering

Craft input prompts to achieve desired outputs from language models effectively.

Example: Chain-of-Thought (Think Step-By-Step) *Wei et al. (2023)*

Model Input

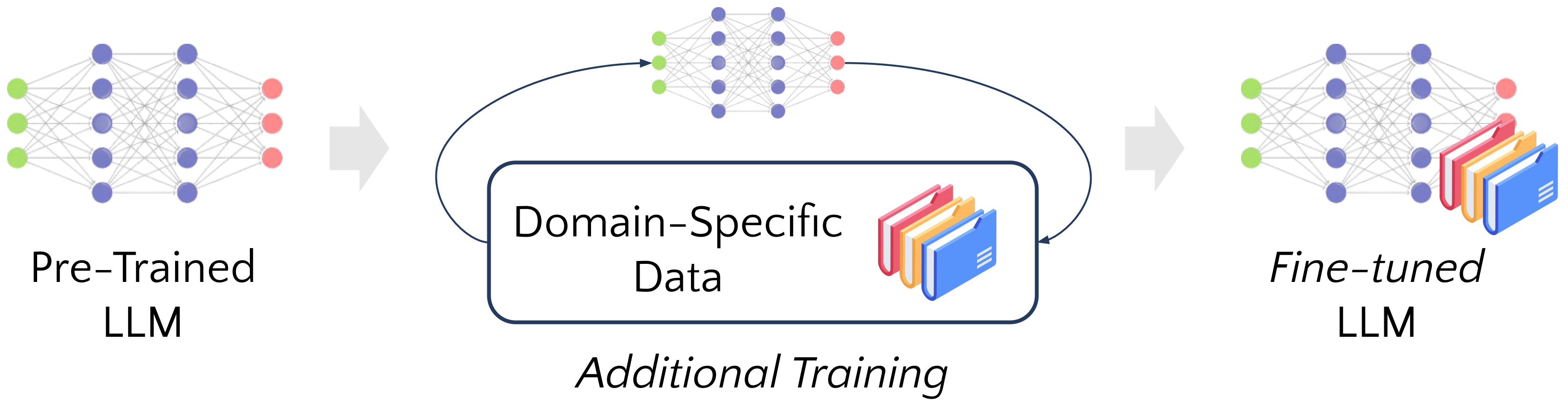
Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.
Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 – 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9.

Supervised Fine-Tuning (SFT)

The goal of fine-tuning is to adapt a pre-trained LLM to a specific task or format:



Note: This requires collecting enough data to do the new training!

Thanks! Questions?