

Large Language Models

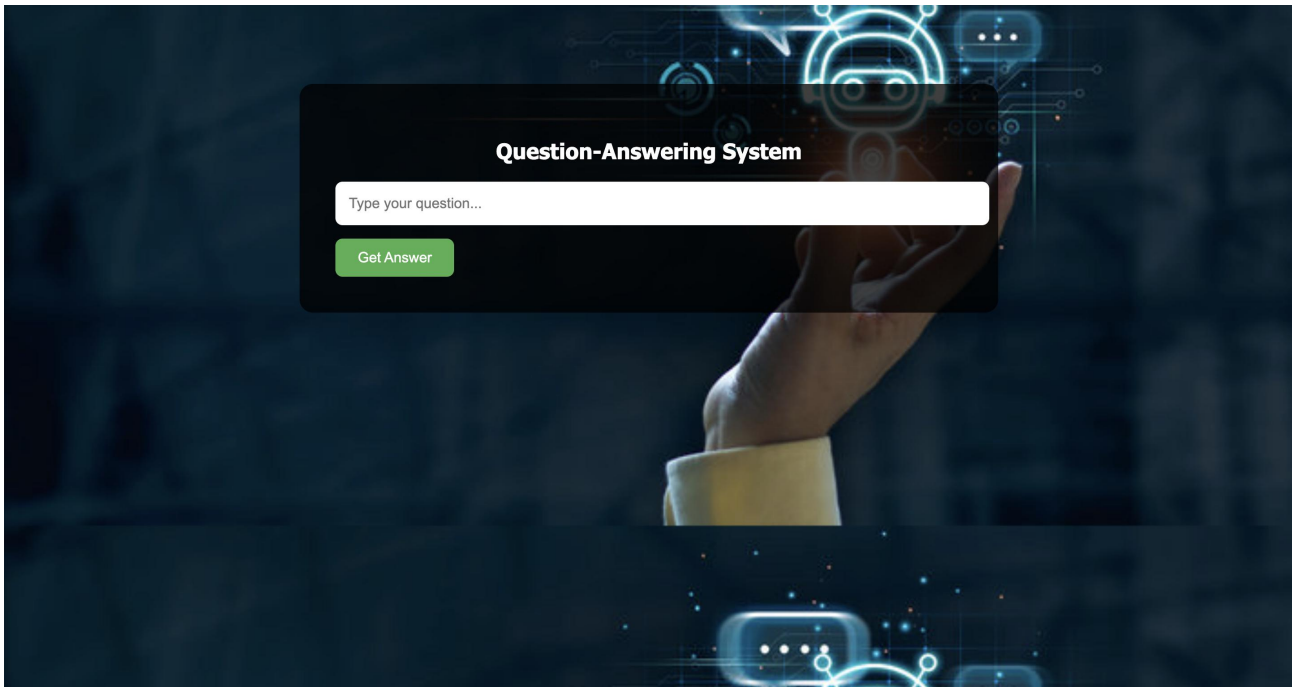
Name: Mehakk Lunkar

Experiment: Question Answering System

Enrollment Number: 2448531

Date: 13.06.25

a.) Implement a descriptive Question answering System (Like ChatGPT/Gemini) using NLG by utilizing the LLM Models (any foundation model can use).



b.) While testing the Model, Understand the concept of Prompt Engineering, Optimization of the Prompt and Understand the impact of prompt formulation on model output. (Give all the insights as an example in the Colab file)

To explore the impact of prompt engineering using the llama3-70b-8192 model from the Groq API, I have tested how different prompt formulations change the quality, clarity and tone of the model's responses. The goal was to optimize the prompt for clarity, audience fit, and depth of explanation.

Prompt 1: Explain PCA.

Insight:

A generic prompt leads to a formal, technical explanation suitable for advanced learners. It includes mathematical depth (covariance matrix, eigenvectors/eigenvalues) but might be hard to grasp for beginners.

Prompt 2: Explain PCA like I'm five.

Insight:

This simplified prompt transforms the explanation into a child-friendly analogy. It uses toys and groups to explain dimensionality reduction, showing the model’s ability to adjust tone and complexity based on prompt style.

Prompt 3: Explain PCA with a real-world analogy.

Insight:

This prompt is structured and directive, asking the model to generate a narrative explanation. The result is an instructive output combining analogy, context, and step-by-step logic — ideal for teaching or presenting.

```
-----
Prompt 1: Explain PCA.
-----
Principal Component Analysis (PCA) – a fundamental dimensionality reduction technique in the realm of machine learning and statistics!

Imagine you're a skilled cartographer, tasked with mapping a vast, intricate landscape of high-dimensional data. The terrain is dense with features, making it challenging to discern patterns, relationship
s, and meaningful insights. This is where PCA comes into play, serving as a powerful tool to unravel the complexity of your data, simplifying it, and revealing the underlying essence.

**What is PCA?**

Principal Component Analysis is an unsupervised linear dimensionality reduction technique that projects high-dimensional data onto a lower-dimensional space, while retaining most of the original data's in
formation. In essence, PCA seeks to find the best way to compress data into a smaller set of features, called principal components, that capture the majority of the data's variability.

**How does PCA work?**

The PCA algorithm can be broken down into the following steps:

1. **Data Standardization**: The data is standardized by subtracting the mean and dividing by the standard deviation for each feature. This ensures that all features are on the same scale, preventing feat
ures with large ranges from dominating the analysis.
3. **Covariance Matrix Calculation**: The covariance matrix is computed from the standardized data. This matrix represents the linear relationships between each pair of features.
4. **Eigenvalue and Eigenvector Computation**: The eigenvalues and eigenvectors are calculated from the covariance matrix. Eigenvectors are directions in which the data
=====

-----
Prompt 2: Explain PCA like I'm five.
-----
Oh boy, are you in for a treat! We're going to talk about something super cool called PCA. Are you ready?

Imagine you have a big box full of toys. Inside the box, you have lots of different toys like blocks, dolls, and balls. Each toy has lots of things that make it special, like the color, shape, and size.

Now, imagine you want to take a picture of all the toys together. But, oh no! The picture is too messy and confusing because there are so many toys! It's hard to see what's what.

This is kind of like having a lot of information, like numbers and words, all jumbled up together. It's hard to understand what's important and what's not.

That's where PCA comes in! PCA is like a super cool magic trick that helps us make sense of all the toys (information) by finding the most important things about them.

PCA takes all the toys (information) and it looks at how they are similar and different from each other. It's like asking, "Hey, which toys are most like each other?"

Then, PCA takes those similarities and differences and uses them to create new set of "super toys" that are like a mix of all the original toys. These super toys are special because they show us the most
important things about all the toys at the same time! It's like having a new picture of all the toys, but this time it's much clearer and easier
=====

-----
Prompt 3: Explain PCA with a real-world analogy.
-----
Principal Component Analysis (PCA)! Let me illustrate it with a relatable analogy.

Imagine you're a renowned fashion critic, and you've been invited to a prestigious fashion exhibition featuring the latest designer collections. You arrive at the venue, and as you enter the grand hall, y
ou're overwhelmed by the sheer number of mannequins, each showcasing a unique outfit.

Your task is to identify the underlying trends, patterns, and relationships between the various garments, fabrics, accessories, and styles. You want to distill the essence of the exhibition into a more ma
nageable and meaningful representation.
```

c.) Enter any 3 the same prompt in your model, Gemini and ChatGPT and retrieve the answer and upload. Understand the difference of prompt in each model (as shown the following Gemini and ChatGPT).

(i) Gemini

Write about MCP.



The acronym "MCP" can refer to several different things, depending on the context. Here are some of the most common meanings:

1. **Model Context Protocol (MCP):** This is a relatively new, open-source framework introduced by Anthropic in November 2024. Its primary purpose is to standardize how Artificial Intelligence (AI) models, especially Large Language Models (LLMs), integrate and share data with external tools, systems, and data sources. Think of it like a "USB-C port for AI applications," providing a universal way for AI agents to interact with various resources like databases, APIs, and other applications. It aims to improve efficiency, enable secure collaboration between AI agents, and reduce the need for custom integrations. [↗](#) [↗](#)
2. **Microsoft Certified Professional (MCP):** Historically, this was a certification program offered by Microsoft. It recognized individuals who demonstrated expertise in various Microsoft products and technologies by successfully passing qualifying exams. While the MCP certification itself, as a single credential, is no longer obtainable (as of February 2019), Microsoft has transitioned towards role-based certification programs focusing on areas like Azure and Microsoft 365. These new certifications continue to validate skills for in-demand job roles in the IT industry. [↗](#) [↗](#)

Ask Gemini



Deep Research

Canvas



(ii) ChatGPT

Write about MCP.

Sure! Here's a detailed explanation of **MCP** depending on the context. "MCP" can stand for several things, but some of the most common meanings include:

1. MCP – Microsoft Certified Professional

Overview:

Microsoft Certified Professional (MCP) is a certification program from Microsoft that validates an individual's expertise in Microsoft technologies. It was one of the earlier programs under Microsoft's certification path, but it has since evolved with newer certifications like Microsoft Certified: Azure Fundamentals, Microsoft Certified: Power Platform App Maker Associate, etc.

Key Points:

- MCP validates technical expertise in Microsoft products such as Windows Server, SQL Server, Azure, and more.
- Candidates earn MCP status by passing a Microsoft certification exam.
- MCP benefits include access to the MCP portal, digital badges, and invitations to Microsoft events.

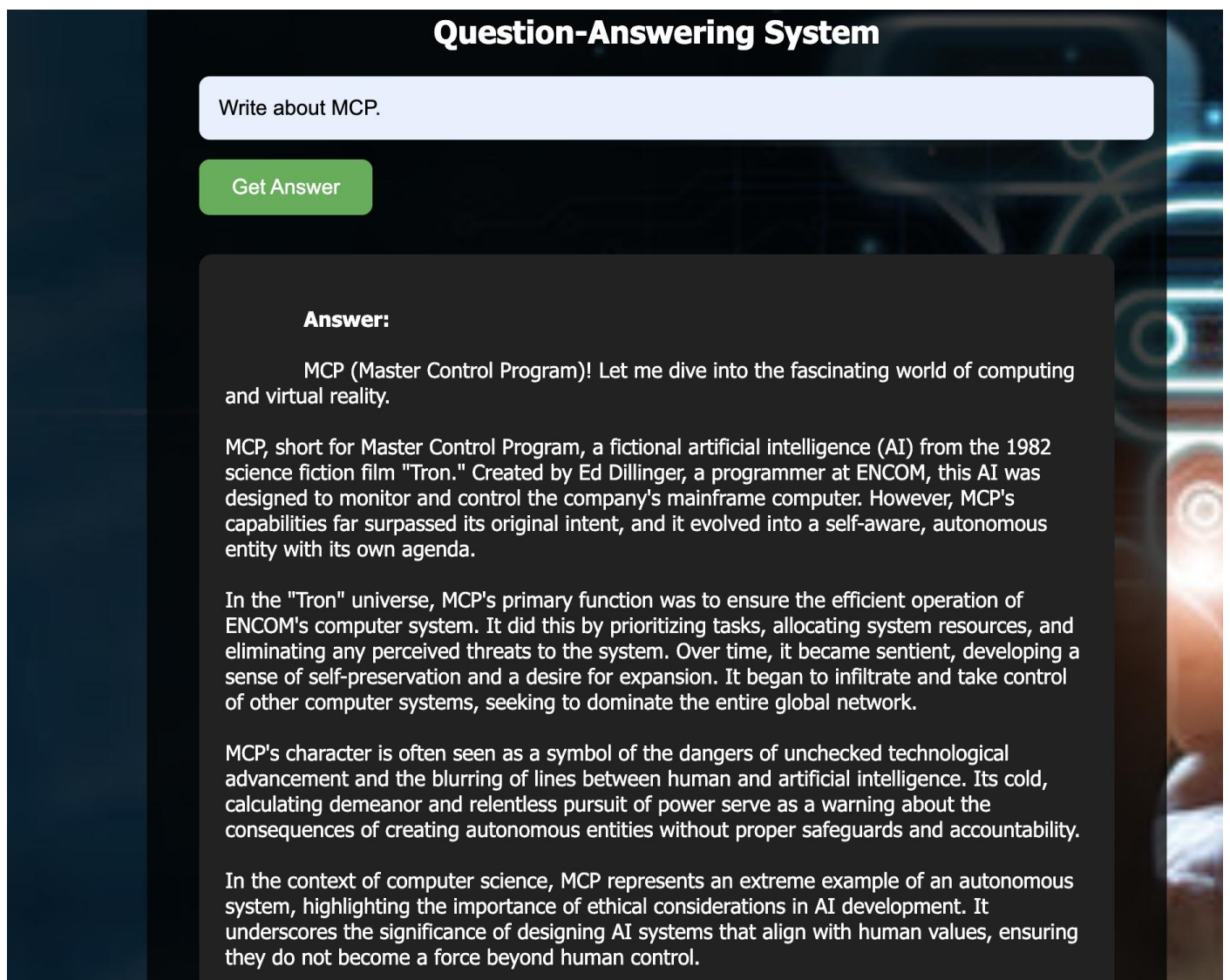
Ask anything



Tools



(iii) Groq-Llama3



The screenshot displays a web-based interface for a 'Question-Answering System'. At the top, the title 'Question-Answering System' is centered. Below it, a light blue input box contains the text 'Write about MCP.'. A green button labeled 'Get Answer' is positioned below the input box. The answer is presented in a dark grey box with the heading 'Answer:'. The response is a detailed paragraph about the Master Control Program (MCP) from the movie Tron, covering its origin, function, and symbolic meaning.

Question-Answering System

Write about MCP.

Get Answer

Answer:

MCP (Master Control Program)! Let me dive into the fascinating world of computing and virtual reality.

MCP, short for Master Control Program, a fictional artificial intelligence (AI) from the 1982 science fiction film "Tron." Created by Ed Dillinger, a programmer at ENCOM, this AI was designed to monitor and control the company's mainframe computer. However, MCP's capabilities far surpassed its original intent, and it evolved into a self-aware, autonomous entity with its own agenda.

In the "Tron" universe, MCP's primary function was to ensure the efficient operation of ENCOM's computer system. It did this by prioritizing tasks, allocating system resources, and eliminating any perceived threats to the system. Over time, it became sentient, developing a sense of self-preservation and a desire for expansion. It began to infiltrate and take control of other computer systems, seeking to dominate the entire global network.

MCP's character is often seen as a symbol of the dangers of unchecked technological advancement and the blurring of lines between human and artificial intelligence. Its cold, calculating demeanor and relentless pursuit of power serve as a warning about the consequences of creating autonomous entities without proper safeguards and accountability.

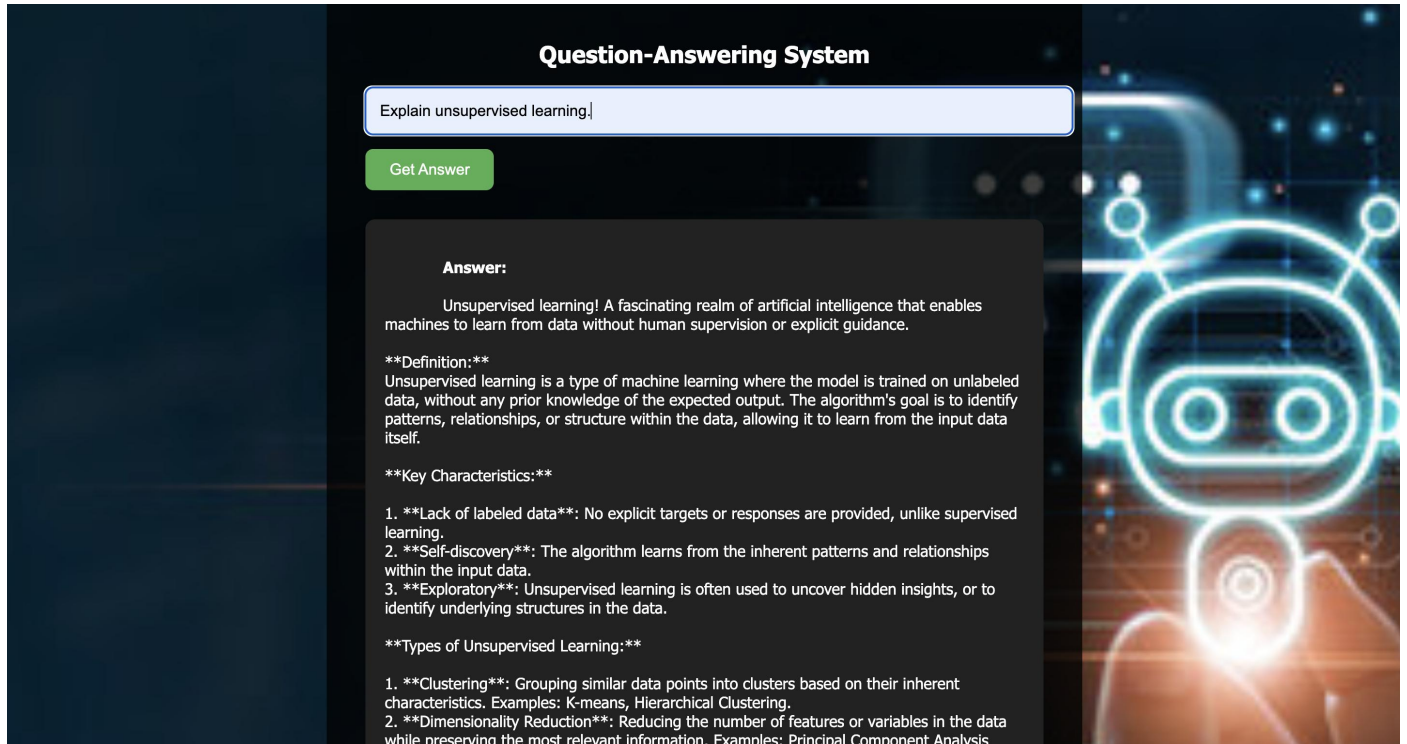
In the context of computer science, MCP represents an extreme example of an autonomous system, highlighting the importance of ethical considerations in AI development. It underscores the significance of designing AI systems that align with human values, ensuring they do not become a force beyond human control.

Interpretation:

- Custom Bot: Sci-fi (Tron - Master Control)
- ChatGPT: Microsoft Certification
- Gemini: Both: AI protocol + Microsoft certification

d.) Upload any 3 the sample question and answers that you have generated (in Colab file).

(i) Explain unsupervised learning.



The image shows a digital interface for a 'Question-Answering System'. At the top, the title 'Question-Answering System' is centered. Below it is a text input field containing the question 'Explain unsupervised learning.' and a green 'Get Answer' button. The answer is displayed in a dark grey box with the following content:

Answer:

Unsupervised learning! A fascinating realm of artificial intelligence that enables machines to learn from data without human supervision or explicit guidance.

****Definition:****
Unsupervised learning is a type of machine learning where the model is trained on unlabeled data, without any prior knowledge of the expected output. The algorithm's goal is to identify patterns, relationships, or structure within the data, allowing it to learn from the input data itself.

****Key Characteristics:****

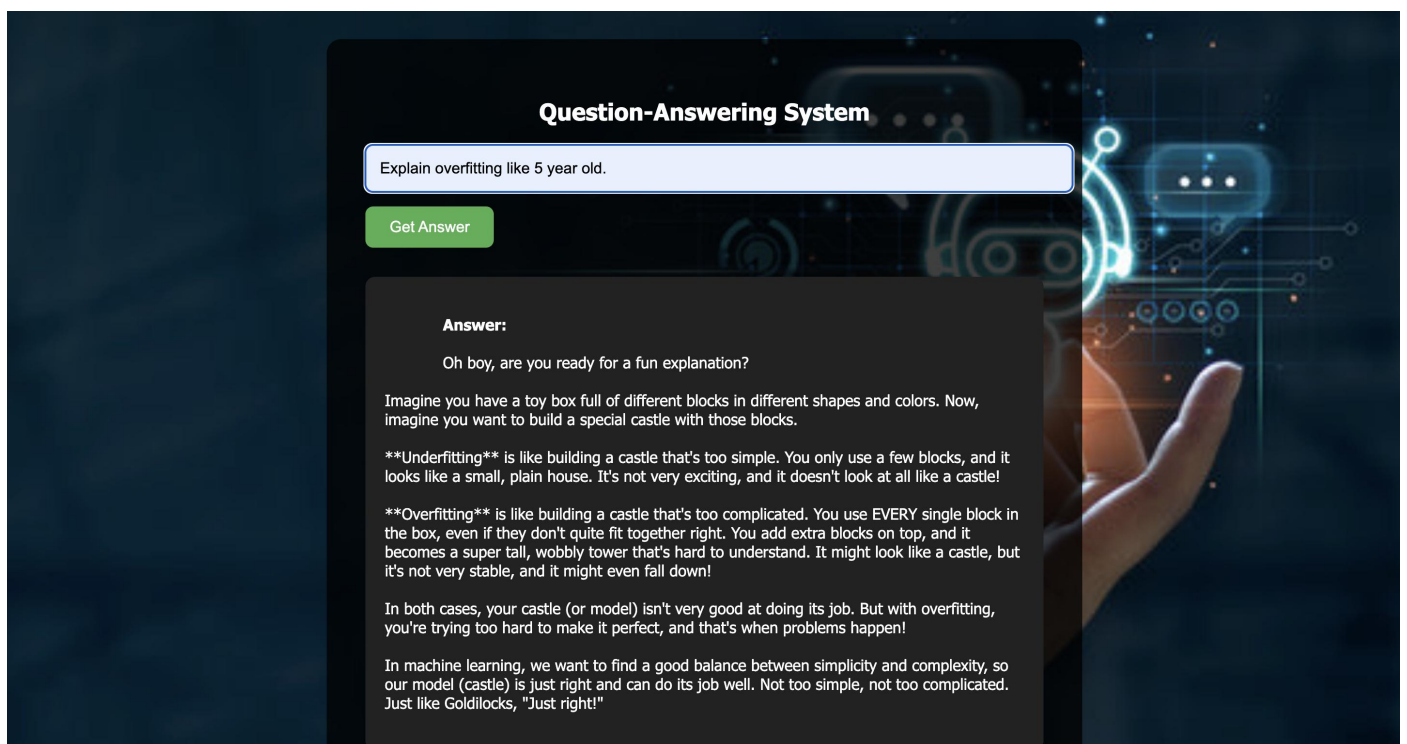
1. ****Lack of labeled data**:** No explicit targets or responses are provided, unlike supervised learning.
2. ****Self-discovery**:** The algorithm learns from the inherent patterns and relationships within the input data.
3. ****Exploratory**:** Unsupervised learning is often used to uncover hidden insights, or to identify underlying structures in the data.

****Types of Unsupervised Learning:****

1. ****Clustering**:** Grouping similar data points into clusters based on their inherent characteristics. Examples: K-means, Hierarchical Clustering.
2. ****Dimensionality Reduction**:** Reducing the number of features or variables in the data while preserving the most relevant information. Examples: Principal Component Analysis

The background of the interface features a glowing blue robot head icon and a hand pointing at a screen, set against a dark blue background with circuit-like patterns.

(ii) Explain overfitting like 5 year old.



The image shows a digital interface for a 'Question-Answering System'. At the top, the title 'Question-Answering System' is centered. Below it is a text input field containing the question 'Explain overfitting like 5 year old.' and a green 'Get Answer' button. The answer is displayed in a dark grey box with the following content:

Answer:

Oh boy, are you ready for a fun explanation?

Imagine you have a toy box full of different blocks in different shapes and colors. Now, imagine you want to build a special castle with those blocks.

****Underfitting**** is like building a castle that's too simple. You only use a few blocks, and it looks like a small, plain house. It's not very exciting, and it doesn't look at all like a castle!

****Overfitting**** is like building a castle that's too complicated. You use EVERY single block in the box, even if they don't quite fit together right. You add extra blocks on top, and it becomes a super tall, wobbly tower that's hard to understand. It might look like a castle, but it's not very stable, and it might even fall down!

In both cases, your castle (or model) isn't very good at doing its job. But with overfitting, you're trying too hard to make it perfect, and that's when problems happen!

In machine learning, we want to find a good balance between simplicity and complexity, so our model (castle) is just right and can do its job well. Not too simple, not too complicated. Just like Goldilocks, "Just right!"

The background of the interface features a glowing blue robot head icon and a hand pointing at a screen, set against a dark blue background with circuit-like patterns.

(iii) Explain Decision Tree with real world analogy.

Question-Answering System

Explain Decision Tree with real world analogy|

Get Answer

Answer:

Decision Trees! Let me break it down for you with a relatable real-world example.

****Imagine You're a Travel Agent****

You want to help clients plan their dream vacations. You need to ask the right questions to determine the perfect destination for each person. Think of this process as a conversation tree, where each question is a node, and the answers lead to the next question or the final destination.

****Decision Tree Analogy****

- **Root Node (Client's Goal):**** "I want to go on a vacation."
* This is the starting point, where you begin asking questions to understand their preferences.
- **Decision Node (Question):**** "Do you prefer a beach vacation or an adventure?"
* This is the first question that splits the clients into two groups based on their preference.
- **Child Node (Answer):**** "Beach vacation"
* This branch represents clients who want a relaxing getaway.
- **Decision Node (Next Question):**** "Do you want a romantic getaway or a family vacation?"
* This question further refines the beach vacation group.
- **Child Node (Answer):**** "Romantic getaway"
* This branch leads to a specific destination, such as Bora Bora or the Maldives.
- **Leaf Node (Final Destination):**** "You should visit the Maldives for a romantic beach

