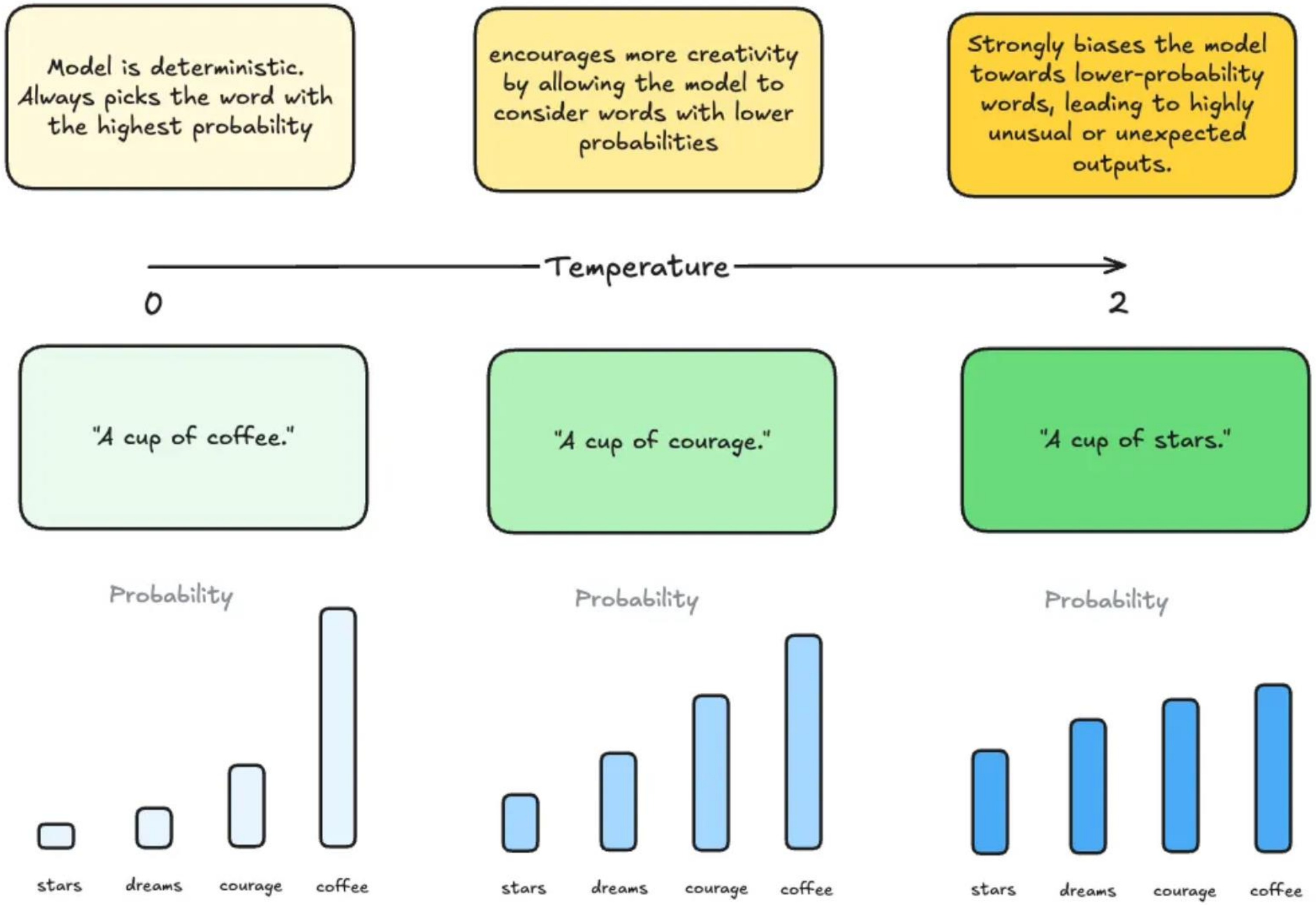
A visual explanation of LLM hyperparameters

DECODING

DATA SCIENCE

LLM Hyperparameters: Fine-tuning Language Models

- Understanding how hyperparameters influence the behavior of LLMs.
- Focus on: Temperature, Top-k, Top-p, and Frequency & Presence Penalties.



Model is deterministic. Always picks the word with the highest probability

encourages more creativity by allowing the model to consider words with lower probabilities

Strongly biases the model towards lower-probability words, leading to highly unusual or unexpected outputs.

"A cup of coffee."

"A cup of courage."

"A cup of stars."

*Image Source Medium Jenn J.

Temperature: Controlling Creativity

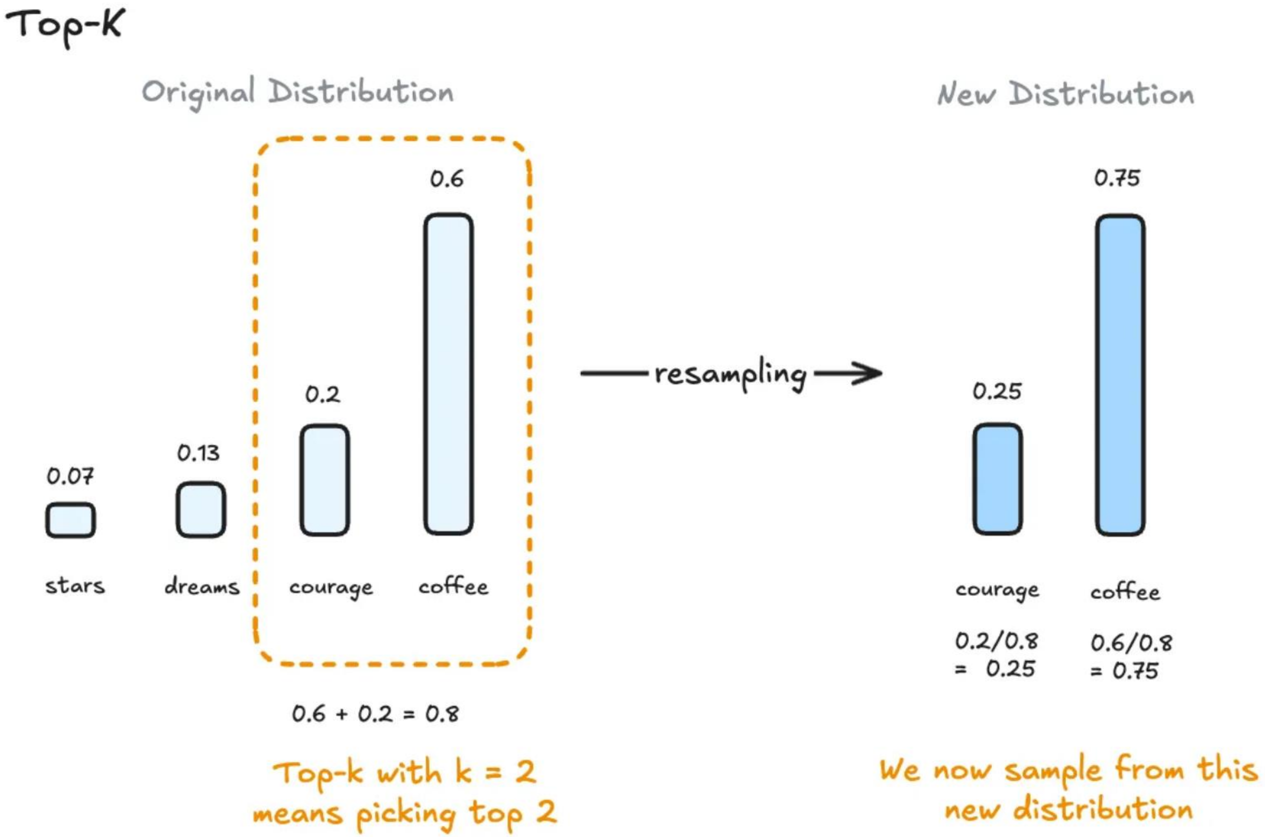
Temperature adjusts the randomness of predictions.

High Temperature (e.g., 0.8 or above):

- - Output: More creative and diverse text. The model considers less likely words.
- - Use Case: Storytelling, poetry, or brainstorming, where creativity is encouraged.

Low Temperature (e.g., 0.2):

- - Output: More deterministic and predictable results.
- - Use Case: Technical writing, formal documentation, or precise tasks where accuracy is required.



Top-k with k = 2 means picking top 2

We now sample from this new distribution

Image provided by the author.

*Image Source Medium Jenn J.

Top-k: Limiting the Next Word Choices

- Top-k restricts the model to only consider the top k most probable tokens.
- High Top-k (e.g., k=50):
- - Output: Adds more variability, leading to creative and sometimes unexpected results.
- - Use Case: Creative writing, dialogue generation, or open-ended tasks.

- Low Top-k (e.g., k=5):
- - Output: Focuses on coherence and relevance.
- - Use Case: Summarization, structured content, or tasks that need clarity and focus.

Top-p: Probability-based Sampling

- Top-p selects the smallest set of tokens where their combined probability exceeds the threshold p.
- High Top-p (e.g., p=0.95):
- - Output: More open-ended, allowing for a wider selection of probable words.
- - Use Case: Dialogue, creative content, where diversity is key.

- Low Top-p (e.g., p=0.5):
- - Output: Ensures the selection of the most probable words.
- - Use Case: News headlines, instructions, or formal summaries where precision is essential.

Frequency Penalty: Reducing Repetition

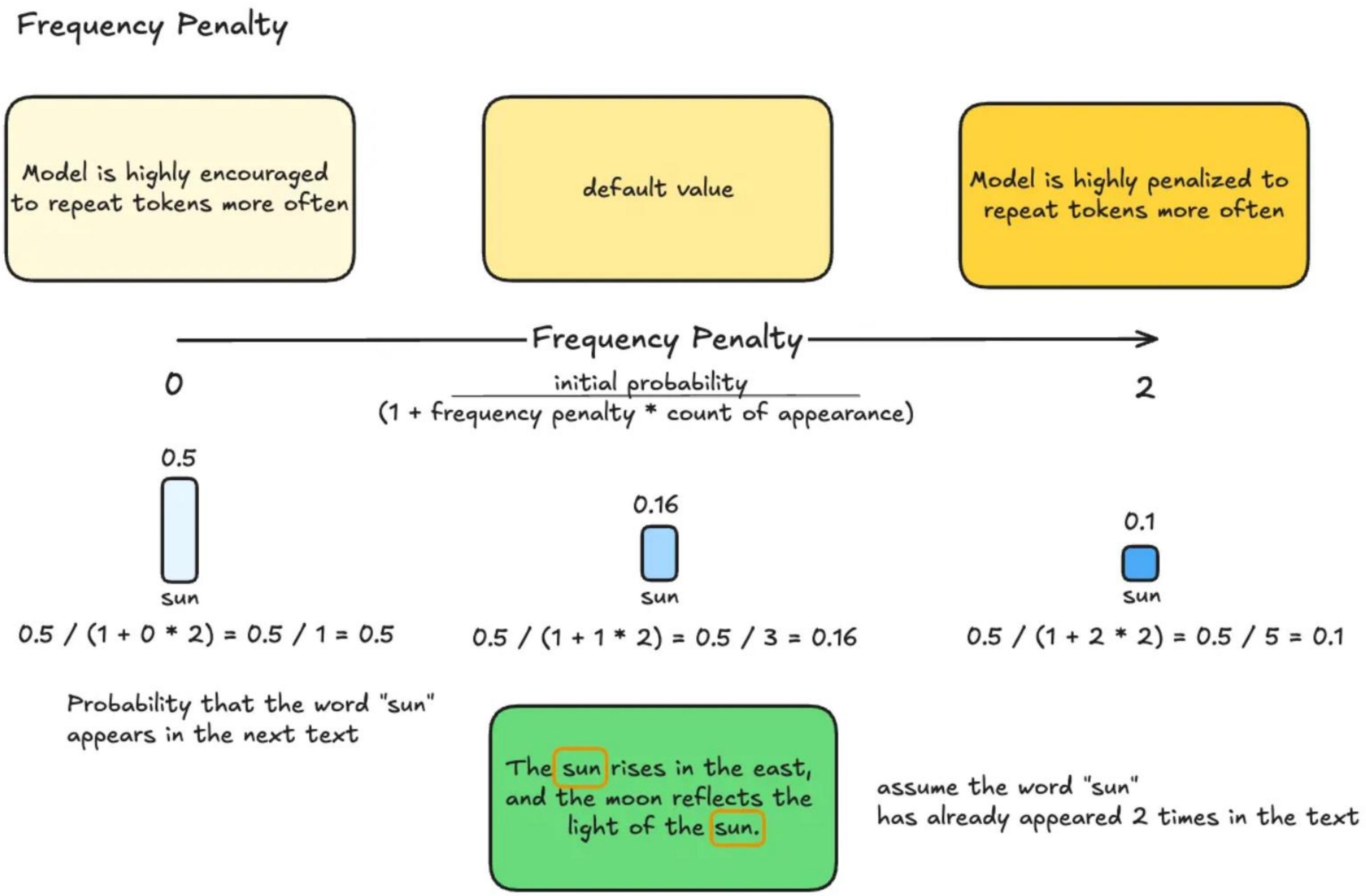
- Frequency Penalty reduces the likelihood of the model repeating the same word within the text.
- Formula: Adjusted probability = initial probability / (1 + frequency penalty * count of appearance).

High Frequency Penalty (e.g., 1.0):

- - Output: Minimizes word repetition, promoting diversity.
- - Use Case: Essays, research papers, and any task where repetition is undesirable.

Low Frequency Penalty (e.g., 0.0):

- - Output: Allows more frequent repetition of words.
- - Use Case: Poetry, marketing slogans, and tasks where repetition may be beneficial.



Frequency Penalty

Model is highly encouraged to repeat tokens more often

default value

Model is highly penalized to repeat tokens more often

$$\frac{\text{initial probability}}{(1 + \text{frequency penalty} * \text{count of appearance})}$$

$0.5 / (1 + 0 * 2) = 0.5 / 1 = 0.5$

$0.5 / (1 + 1 * 2) = 0.5 / 3 = 0.16$

$0.5 / (1 + 2 * 2) = 0.5 / 5 = 0.1$

Probability that the word "sun" appears in the next text

The sun rises in the east, and the moon reflects the light of the sun.

assume the word "sun" has already appeared 2 times in the text

*Image Source Medium Jenn J.

Presence Penalty(Topic Penalty): Penalizing Word Reuse

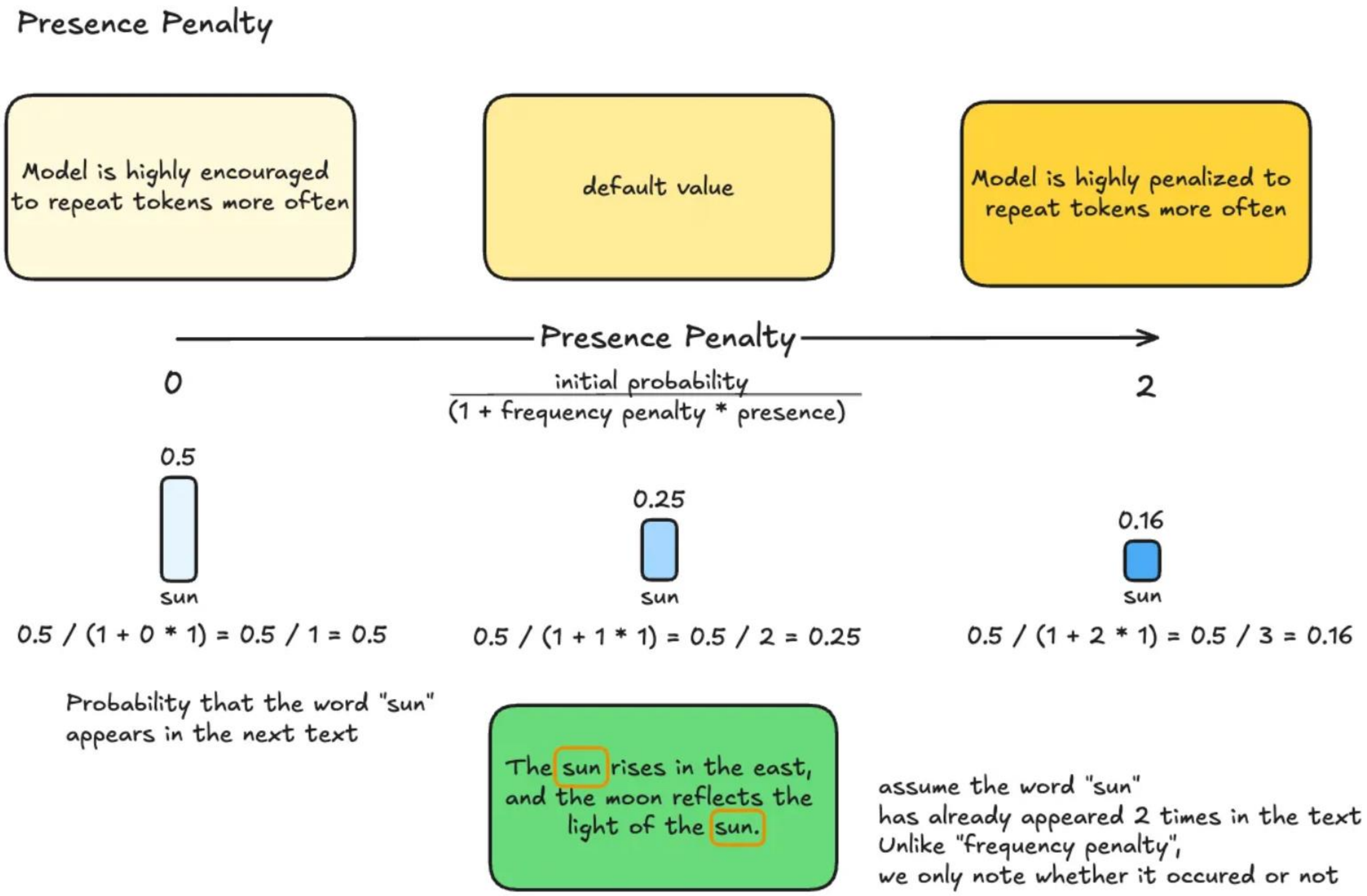
- Presence Penalty discourages the reuse of words that have already been mentioned, regardless of how many times they appear.
- Formula: Adjusted probability = initial probability / (1 + presence penalty * presence).

High Presence Penalty (e.g., 1.0):

- - Output: Encourages the use of new words and ideas instead of reusing old ones.
- - Use Case: Brainstorming sessions or exploratory content generation where fresh ideas are desired.

Low Presence Penalty (e.g., 0.0):

- - Output: Allows the reuse of words, useful when consistency is key.
- - Use Case: Technical writing, instructional material, or reinforcing key concepts.



Presence Penalty

Model is highly encouraged to repeat tokens more often

default value

Model is highly penalized to repeat tokens more often

Presence Penalty

0

2

$$\frac{\text{initial probability}}{(1 + \text{frequency penalty} * \text{presence})}$$

0.5

sun

0.5 / (1 + 0 * 1) = 0.5 / 1 = 0.5

0.25

sun

0.5 / (1 + 1 * 1) = 0.5 / 2 = 0.25

0.16

sun

0.5 / (1 + 2 * 1) = 0.5 / 3 = 0.16

Probability that the word "sun" appears in the next text

The sun rises in the east, and the moon reflects the light of the sun.

assume the word "sun"
has already appeared 2 times in the text
Unlike "frequency penalty",
we only note whether it occured or not

*Image Source Medium Jenn J.

Penalties: Reducing Redundancy

- Frequency Penalty: Penalizes repeated words based on frequency of appearance.
- Presence Penalty: Penalizes any reuse of words or phrases.

High Penalty (e.g., 1.0):

- - Output: Discourages repetition, promoting more diversity.
- - Use Case: Long-form essays, creative writing, or brainstorming.

Low Penalty (e.g., 0.0):

- - Output: Allows for repetition.
- - Use Case: Technical writing, brand messaging, or instructional content where consistency is key.

Choosing the Right Hyperparameters

Summary:

- Temperature: Controls creativity and randomness.
- Top-k & Top-p: Define the pool of next word choices based on probability.
- Penalties: Manage word repetition to balance diversity and consistency.

Optimization: Tailor hyperparameters based on the specific needs of your task.