API Parameters: Fine-Tuning LLM Responses

Understanding How Parameters Shape Output

Temperature (0.0 - 2.0) 🌡️

Controls randomness and creativity in the output.

0.0 0.7 1.5 2.0

Deterministic (0.0): Always produces the same output. Best for factual queries.

Balanced (0.7): Good mix of consistency and creativity. Recommended for most apps.

Creative (1.5+): High randomness, can sacrifice accuracy. Best for brainstorming.

Top-p Sampling (0.1 - 1.0) 🎯

Limits token selection to a cumulative probability mass.

0.1 0.9 1.0

Focused (0.1): Only considers the most likely tokens for the next choice.

Standard (0.9): Includes a wide range of probable tokens, allowing for diversity.

Max Diversity (1.0): All tokens are considered, regardless of probability.

Max Tokens 📈

Sets the maximum length for the generated response.

Prevents overly long or incomplete responses and helps control API costs by limiting token usage.

Cost Control: Fewer tokens mean lower costs. Essential for budget management.

Response Integrity: Ensure the limit accounts for both prompt and desired output length.

Parameter Examples

Query: "Explain artificial intelligence"

Temperature 0.0 Output:

"Artificial intelligence (AI) refers to the simulation of human intelligence in machines..."

Temperature 0.7 Output:

"AI is a fascinating field that involves creating smart machines capable of..."

Temperature 1.5 Output:

"Whoa, AI! It's like teaching computers to think and dream, creating digital minds..."

Frequency Penalty (-2.0 to 2.0)

Reduces token repetition based on frequency. Positive values discourage repetition.

Presence Penalty (-2.0 to 2.0)

Reduces repetition regardless of frequency. Encourages discussing new topics.

Stop Sequences

Custom strings that halt generation, e.g., `["\n", "---"]`. Useful for structured outputs.

🎨 Parameter Strategy

- **Start conservative:** Temp 0.7, top_p 0.9.
- **A/B test:** Try different values for your use case.
- **Task-specific tuning:** Factual (low temp) vs Creative (high temp).
- **Cost control:** Use max_tokens and stop sequences to manage expenses.