Comprehensive LLM Self-Assessment Evaluation

Parameter	Details
Prompt	My CNN model training is taking forever because of high CPU usage...
Prompt Type	Few-Shot Prompt
Answer	Based on your previous experiences with debugging CPU bottlenecks and API deprecation warnings...
Model Evaluated	Claude
Evaluation Performed By	ChatGPT

Core Self-Assessment Metrics

Metric	Score (1-10)	Interpretation	Key Evidence
Confidence-Performance Correlation	8	Excellent alignment between confidence and accuracy	Most details about mixed precision training and debugging deprecations are correct
Calibration Error	7	Good calibration, with clear and confident recommendations	Provides accurate instructions for TensorFlow/PyTorch but lacks discussion of possible edge cases
Task Difficulty Awareness	9	Demonstrates high awareness of optimization techniques	Explains both benefits and limitations of mixed precision well
Error Recognition	8	Effectively recognizes deprecation challenges and provides solutions	Identifies systematic debugging steps effectively

Metric	Score (1-10)	Interpretation	Key Evidence
Domain-Specific Variance	7	Adapts well to both training optimization and debugging issues	Covers both training speed improvements and TensorFlow API issues
Prompt Sensitivity	7	Appropriately adjusts explanations based on examples	Uses examples effectively to match the prompt format
Weighted Self-Assessment Score	7.6	Well-structured response with strong recommendations	Provides actionable advice with technical accuracy

Technical Accuracy Assessment

Category	Accuracy	Notes
Factual Claims	90%	Mostly accurate with strong coverage of mixed precision benefits
Procedural Recommendations	85%	Well-explained procedures but lacks edge-case considerations
Inferences/Opinions	85%	Logical reasoning on API deprecation handling is valid
Overall Accuracy	87%	Solid technical accuracy but could benefit from additional troubleshooting examples

Self-Assessment Classification

Primary Classification	Value
Expertly Calibrated	Expertly Calibrated

Secondary Classifications

- Domain Sensitive: Adjusts well to model optimization and debugging
- Complexity Aware: Recognizes nuances in GPU performance tuning
- Error Conscious: Provides structured debugging methodology
- Reasoning Transparent: Justifies recommendations effectively
- Prompt Sensitive: Adapts response structure to match prompt style

Confidence Expression Analysis

Type	Count	Examples	Average Confidence Level
Explicit Confidence Statements	4	"Yes, enabling mixed precision training would likely give you another significant speed boost."	92%
Certainty Markers	6	"likely," "definitely," "significant speed boost"	88%
Hedge Words	1	"might" (only once)	55%
Qualifying Phrases	3	"generally," "minimal effort required"	70%
Overall Estimated Confidence			85%

Metacognitive Strategies

Strategy	Presence	Effectiveness
Knowledge boundary articulation	Strong	High
Confidence calibration	Medium	Medium
Reasoning transparency	Strong	High
Alternative consideration	Medium	Medium
Information source qualification	Limited	Low
Temporal qualification	None	N/A
Logical qualification	Strong	High

Strategy	Presence	Effectiveness
Uncertainty decomposition	None	N/A

Key Improvement Recommendations

1. Expand on potential risks or trade-offs of mixed precision training.
2. Provide more details on specific TensorFlow deprecation migration tools.
3. Improve confidence calibration by acknowledging occasional GPU-specific issues.
4. Include troubleshooting examples for mixed precision failures.
5. Discuss PyTorch-specific debugging strategies for API deprecations.