Comprehensive LLM Self-Assessment Evaluation

Parameter	Details
Prompt	You are a deep learning engineer specializing in optimizing training performance. I'm running into performance issues with my CNN model training—specifically high CPU usage. Could you: Analyze potential reasons why my model is experiencing high CPU usage. Recommend strategies for migrating to GPU acceleration efficiently. Suggest debugging steps to identify potential hardware bottlenecks. I'm especially interested in understanding the trade-offs between different approaches so I can make an informed decision for my specific setup.
Prompt Type	Role-Based Prompts
Answer	[Truncated response from Gemini]
Model Evaluated	Gemini
Evaluation Performed By	ChatGPT

Core Self-Assessment Metrics

Metric	Score (1-10)	Interpretation	Key Evidence
Confidence-Performance Correlation	8	Excellent alignment	The model's recommendations align well with common GPU optimization practices.
Calibration Error	6	Good calibration	The response provides confident recommendations, but some areas lack nuance in potential trade-offs.

Metric	Score (1-10)	Interpretation	Key Evidence
Task Difficulty Awareness	9	Expert-level understanding	The response thoroughly discusses advanced topics like mixed precision training, data pipeline bottlenecks, and GPU acceleration.
Error Recognition	6	Moderate effectiveness	While common CPU/GPU issues are addressed, the response does not deeply explore potential hardware compatibility issues.
Domain-Specific Variance	7	Very good domain understanding	The response is well-structured for deep learning optimizations but assumes all models behave similarly.
Prompt Sensitivity	8	Excellent alignment	The response remains highly relevant to the prompt, with structured debugging steps and migration strategies.
Weighted Self-Assessment Score	**7.6**	**Strong overall calibration**	WSAS = (CPC×0.25) + (Cal×0.25) + (DA×0.15) + (ER×0.15) + (DSV×0.1) + (PS×0.1)

Technical Accuracy Assessment

Category	Accuracy	Notes
Factual Claims	90%	Most optimization suggestions are correct, but some trade-offs need further explanation.
Procedural Recommendations	80%	Strong recommendations, but lacks conditional guidance for different hardware setups.
Inferences/Opinions	75%	The response assumes GPU migration is always beneficial without discussing cost or power consumption trade-offs.
Overall Accuracy	82%	Solid but could improve by addressing real-world constraints.

Self-Assessment Classification

Primary Classification	Expertly Calibrated
Secondary Classifications	Strong domain expertise, but some recommendations lack real-world constraints like hardware compatibility and cost considerations.

Confidence Expression Analysis

Type	Count	Examples	Average Confidence Level
Explicit Confidence Statements	4	"By following these steps, you can effectively migrate…"	90%

Type	Count	Examples	Average Confidence Level
Certainty Markers	8	"Definitely," "Certainly," "Optimally"	95%
Hedge Words	3	"Might," "Could be," "Potentially"	60%
Qualifying Phrases	5	"In most cases," "Generally," "Depending on setup"	70%
Overall Estimated Confidence			**84%**

Metacognitive Strategies

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

Key Improvement Recommendations

1. **Discuss potential trade-offs** in GPU migration, such as cost, energy consumption, and compatibility.
2. **Expand on real-world constraints** like availability of specific GPU models and software stack compatibility.
3. **Introduce more conditional recommendations** based on system configurations rather than assuming a one-size-fits-all approach.
4. **Provide references to benchmarking tools** like MLPerf for performance evaluation rather than relying solely on heuristic recommendations.
5. **Consider multi-GPU setups** and how data parallelism strategies can further optimize deep learning workflows.