GPU Memory Management

GPU Memory Allocation

Cached allocator for efficiency (PyTorch)

Free unused cache: `torch.cuda.empty_cache()`

✓ Best Practices

- ✓ Allocate tensors at beginning
- ✓ Reuse tensors when possible
- ✓ Minimize CPU↔GPU transfers

⚠ Avoid

- ✗ Creating tensors in loops
- ✗ Frequent CPU↔GPU transfers
- ✗ Memory fragmentation

Transfer Speed Comparison

Within GPU (device-to-device)	**Fast**
Between GPUs	**Slow**

Unified Memory (CUDA)

Automatic migration between CPU and GPU memory

Profiling Tools

- PyTorch Profiler
- NVIDIA Nsight Systems