LLaMA Inference Environment Setup

Complete environment setup for running LLaMA model inference with vLLM, FastAPI, and PromptBench.

📋 Prerequisites

1. **Conda/Miniconda**: Install from https://docs.conda.io/en/latest/miniconda.html
2. **NVIDIA GPU** (recommended): With CUDA 12.1+ for optimal performance
3. **System RAM**: At least 32GB recommended
4. **GPU VRAM**: At least 80GB for 70B models (8x A100 or similar)

🚀 Quick Start

Option 1: Automated Setup (Recommended)

```bash
# Make setup script executable
chmod +x setup_environment.sh

# Run automated setup
./setup_environment.sh

# Activate the environment
source activate_inference.sh
```

Option 2: Using Make Commands

```bash
# Install full environment
make install

# Or install CPU-only version
make install-cpu

# Start the model server
make server

# Run inference
make inference
```

Option 3: Manual Conda Setup

```bash
# Create environment from YAML file
conda env create -f environment.yml

# Activate environment
conda activate llama-inference

# Verify installation
python -c "import torch; print(f'CUDA available: {torch.cuda.is_available()}')"
```

📦 Environment Variants

File	Purpose	Use Case
`environment.yml`	Full installation	Production with GPU support
`environment-cpu.yml`	CPU-only	Development/testing without GPU
`environment-minimal.yml`	Minimal setup	Quick testing, reduced dependencies
`environment-dev.yml`	Development	Includes testing and debugging tools
`environment-docker.yml`	Docker deployment	Containerized applications

🛠️ Common Operations

Check GPU Configuration

```bash
make check-gpu
# Or manually:
nvidia-smi
python -c "import torch; print(torch.cuda.is_available())"
```

Start Model Server

```bash
make server
# Or manually:
python model_server_inference.py
```

Run Bulk Inference

```bash
# Using make
make inference

# Or manually with options
python bulk_inference.py \
  --dataset sst2 \
  --mode parallel \
  --batch-size 32 \
  --use-sampled-data
```

Create Sampled Datasets

```bash
make sample-datasets
# Or manually:
python dataset_sampler_inference.py
```

Monitor GPU Usage

```bash
make monitor
# Or manually:
watch -n 1 nvidia-smi
```

🔧 Troubleshooting

CUDA Version Mismatch

```bash
# Check CUDA version
nvidia-smi | grep "CUDA Version"

# Reinstall PyTorch with correct CUDA version
pip install torch --index-url https://download.pytorch.org/whl/cu121  # for CUDA 12.1
```

vLLM Installation Issues

```bash
# Install with specific CUDA version
pip install vllm --extra-index-url https://download.pytorch.org/whl/cu121

# Or build from source
git clone https://github.com/vllm-project/vllm.git
cd vllm
pip install -e .
```

Out of Memory Errors

```bash
# Set memory allocation strategy
export PYTORCH_CUDA_ALLOC_CONF=max_split_size_mb:512

# Reduce batch size in inference
python bulk_inference.py --batch-size 8  # Smaller batch size
```

Environment Conflicts

```bash
# Clean and reinstall
make clean
make install

# Or manually
conda env remove -n llama-inference
conda env create -f environment.yml
```

📊 Performance Optimization

Environment Variables

```bash
```

```
# Optimize memory allocation
export PYTORCH_CUDA_ALLOC_CONF=max_split_size_mb:512

# Disable tokenizer parallelism (prevents warnings)
export TOKENIZERS_PARALLELISM=false

# Set visible GPUs
export CUDA_VISIBLE_DEVICES=0,1,2,3,4,5,6,7
```

Server Configuration

```
python

# In model_server_inference.py, adjust:
engine_config = AsyncEngineArgs(
    model=MODEL_PATH,
    tensor_parallel_size=8,  # Number of GPUs
    max_num_batched_tokens=4096,
    max_num_seqs=256,
)
```

Inference Optimization

```
python

# Adjust sampling parameters
sampling_params = {
    "temperature": 0.1,  # Lower for deterministic outputs
    "top_p": 0.9,
    "max_tokens": 256,  # Reduce for faster inference
}
```

📁 Project Structure

```
.
├── environment.yml            # Main conda environment
├── setup_environment.sh       # Automated setup script
├── activate_inference.sh      # Quick activation script
├── Makefile                   # Make commands
├── model_server_inference.py  # FastAPI model server
├── bulk_inference.py          # Bulk inference runner
├── dataset_processor_inference.py  # Dataset processing
├── dataset_sampler_inference.py    # Dataset sampling
├── dataset_loader_inference.py     # Dataset loading hooks
├── dataset_cache/             # Cached datasets
├── sampled_datasets/          # Sampled dataset files
└── results/                   # Inference results
```

🧪 Testing

Run Tests

```bash
make test
```

Validate Configuration

```bash
make validate
```

Test Model Server

```bash
# Start server
make server

# In another terminal, test endpoints
curl http://localhost:8000/health
curl -X POST http://localhost:8000/generate \
  -H "Content-Type: application/json" \
  -d '{"prompt": "Hello, world!", "sampling_params": {"max_tokens": 50}}'
```

🐳 Docker Deployment

Build Docker Image

```bash
make docker-build
```

Run Docker Container

```bash
make docker-run
```

Docker Compose (create docker-compose.yml)

```yaml
version: '3.8'
services:
  llama-inference:
    build: .
    runtime: nvidia
    environment:
      - CUDA_VISIBLE_DEVICES=all
    ports:
      - "8000:8000"
    volumes:
      - ./models:/models
      - ./results:/results
```

📚 Additional Resources

- vLLM Documentation
- FastAPI Documentation
- PromptBench Documentation
- PyTorch CUDA Setup

🤝 Support

For issues or questions:

1. Check the troubleshooting section above
2. Verify all prerequisites are installed

3. Ensure CUDA versions match between system and packages
4. Check logs in `bulk_inference.log`

License

This setup is provided as-is for research and development purposes.