Euro-Par 2025 Artifact Overview Document

Title: *A Sparsity Predicting Approach for Large Language Models via Activation Pattern Clustering*

1. Getting Started Guide

Platform Requirements

- **OS:** Ubuntu 20.04 or later
- **Python:** 3.9+
- **CUDA:** 11.8+
- **GPU:** 8× NVIDIA A100 (80GB each)
- **RAM:** 512 GB
- **Environment Setup:** Use the env.yml inside llm-awq/ (provided)

Directory Structure

```
artifact/
 ├── llm-awq/
 │   ├── awq/entry.py
 │   ├── env.yml
 │   └── ...
 ├── clustering/
 │   ├── clustering_results_50_mistral_weighted/
 │   ├── activation_aware_clustering.py
 │   ├── new_select_centroids.py
 │   ├── penalized_brbkmeans.py
 │   ├── penalized_brbkmeans_parallel.py
 │   ├── replace_centroids.py
 │   ├── replace_centroid_based_on_sparsity.py
 │   ├── select_centroids.py
 │   ├── test_penalty.py
 │   ├── error_checking.py
 │   ├── testing_all_chunks.py
 ├── thresholds/
 │   └── thresholds_50_percent_sparsity.json
```

Quick Test (≤30 Minutes)

```
cd llm-awq
conda env create -f env.yml
conda activate llm-awq
python -m awq.entry --model_path models/Mistral-7B-v0.1/ --tasks wikitext
```

This command:

- Applies 50% sparsity using the thresholds in ffn_thresholds_50.json
- Uses stored centroids for sparse computation
- Extracts activations and evaluates model performance
- Reports perplexity (PPL) score

2. Step-by-Step Instructions to Reproduce Results

Step 1: Activation Extraction + Sparse Inference

Run the following command inside llm-awq/:

```
python -m awq.entry --model_path models/Mistral-7B-v0.1/ --tasks wikitext
```

This:

- Applies FFN layer thresholds (50% sparsity)

- Loads centroids from clustering_results_50_mistral_weighted/
- Executes inference and reports final perplexity

Step 2: Clustering Centroid Generation

Run the following command inside the clustering/ directory:

```
python activation_aware_clustering.py
```

This script:

- Processes activation values extracted from the model
- Performs clustering separately for gate_proj, up_proj, and down_proj layers
- Saves centroids under clustering_results_50_mistral_weighted/

No additional configuration is needed.

3. Platform Used for Experiments

- **Server OS:** Ubuntu 22.04
- **GPUs:** 8× NVIDIA A100 (80 GB each)
- **CPU:** AMD EPYC
- **RAM:** 512 GB
- **CUDA Version:** 11.8
- **Framework:** PyTorch
- **Base Codebase:** Modified llm-awq repository (included in artifact)

4. Included Files and Outputs

- All .pt centroids for multiple cluster sizes and sparsity levels
- Threshold JSON file used for enforcing 50% FFN sparsity
- Clustering scripts for centroid generation and selection

- Modified llm-awq repo to support activation extraction, thresholding, and PPL evaluation using centroids
- Complete instructions to reproduce and validate results
- All the plots.