Project Proposal: Enhancing Knowledge Distillation for GPT-2 with Distributional Losses

1. Problem Selection and Rationale

Large pre-trained language models like GPT-2 have shown remarkable capabilities, but their size often hinders deployment in resource-constrained environments. Knowledge distillation (KD) offers a promising approach to compress these models into smaller, faster student models while retaining performance. However, standard KD methods often rely on simple loss functions like Mean Squared Error (MSE) or Kullback-Leibler (KL) divergence between teacher and student outputs or hidden states, which may not fully capture the complex distributions learned by the teacher.

This project proposes to investigate the effectiveness of using distribution-matching losses, specifically Wasserstein Distance (WD), for distilling knowledge from a pre-trained GPT-2 model to a smaller student GPT-2 model. The hypothesis is that by directly minimizing the distance between the distributions of teacher and student hidden states (or output logits), WD might enable the student to learn a richer representation and potentially achieve better performance compared to traditional MSE-based distillation. We aim to explore if this approach leads to student models that better mimic the internal feature distributions of the teacher across different layers.

2. Database/Dataset

To train and evaluate the distilled models, we plan to utilize publicly available text corpora suitable for language modeling tasks. Potential candidates include standard benchmarks like WikiText-103 or OpenWebText. The final choice will depend on computational resources and initial experiments regarding dataset size and complexity. We will start with a subset to facilitate faster iteration during development. Data preprocessing will involve standard tokenization appropriate for GPT-2 models.

3. NLP Methods

The core NLP methods involved will be:

- **Pre-trained Language Models (Transformers):** We will use a pre-trained GPT-2 model (e.g., `gpt2` or `gpt2-medium` from Hugging Face) as the teacher model and initialize a smaller GPT-2 architecture as the student.
- **Knowledge Distillation:** We will implement a distillation framework where the student learns from the teacher's intermediate hidden states and potentially output logits.
- **Loss Functions:** The central part of the investigation involves comparing:
    - Wasserstein Distance (WD): Implemented possibly via a learned Critic network (as in WGANs) to estimate the distance between teacher and student hidden state distributions layer-wise.
    - Mean Squared Error (MSE): As a baseline, calculated between teacher and student hidden states.
- We will explore different strategies for applying these losses across layers. This approach involves customizing standard distillation techniques by integrating the WD loss calculation.

4. Planned Packages

We anticipate using the following Python packages:

- `transformers`: To access pre-trained GPT-2 models and tokenizers (Hugging Face). Rationale: Provides easy access to state-of-the-art models and tools.
- `torch`: As the primary deep learning framework for model definition, training, and loss computation. Rationale: Widely used, flexible, good support for custom modules.
- `deepspeed`: To potentially accelerate training and handle larger models/batches using distributed training and memory optimization techniques. Rationale: Essential for efficiently training transformer models.
- `datasets`: For loading and processing the chosen text dataset (Hugging Face). Rationale: Integrates well with `transformers`.
- `numpy`, `scipy`: For numerical operations and potentially metric calculations. Rationale: Standard numerical libraries.
- `tqdm`: For displaying progress bars during training. Rationale: Improves user experience during long runs.

5. NLP Tasks

The primary NLP tasks involved are:

- **Language Modeling:** Both teacher and student models are fundamentally language models. Their ability to predict the next token will be a key evaluation aspect.
- **Knowledge Distillation:** The overarching task of transferring knowledge from the teacher to the student model.

6. Performance Judgment Metrics

We plan to evaluate the performance of the distilled student models using a combination of metrics:

- **Perplexity (PPL):** A standard intrinsic metric for language model quality on a held-out test set. Lower PPL indicates better language modeling performance.
- **Loss Curves:** Monitoring the distillation loss (MSE or WD/Critic loss) and the student's own language modeling loss during training.
- **Computational Cost:** Comparing model size (number of parameters) and inference speed between the teacher and student models.
- **Qualitative Analysis (Optional):** Potentially examining text generated by the student models.
- **Distributional Similarity (Exploratory):** We may explore visualization techniques (e.g., t-SNE) on hidden states to qualitatively assess how well the student matches the teacher's internal representations under different loss functions.

7. Rough Schedule

- **Weeks 1-2:** Setup environment, implement baseline MSE distillation framework, finalize dataset choice and preprocessing pipeline. Initial experiments with baseline.
- **Weeks 3-4:** Implement Wasserstein Distance loss component, including the Critic network. Integrate WD loss into the distillation framework. Debugging and initial WD training runs.

- **Weeks 5-6:** Conduct main comparison experiments between MSE and WD distillation. Systematic training runs with different hyperparameters (learning rates, layer matching strategies).
- **Weeks 7-8:** Evaluate models using defined metrics (PPL, loss analysis, potentially visualizations). Analyze results and draw conclusions.
- **Weeks 9-10:** Prepare final report, code documentation (READMEs), and presentation. Finalize code and repository structure.