

Performance Improvement Report

Task 1: Mountains Named Entity Recognition

Current Architecture

A **DistilBERT**-based model, fine-tuned on a custom dataset. The current implementation uses a standard token classification setup with a **CrossEntropyLoss** function.

Improvements and Justifications

1. **Architecture Enhancements:** Switch to **BERT-large** or **RoBERTa** for more powerful contextual understanding as BERT-large offers a larger number of parameters, potentially improving the quality of predictions for complex or rare entities like mountain names.
2. **More and Better Data:** Use **more data** in the process of fine-tuning. One can combine NER datasets into one and use the bigger corpus for training. Use model using **domain-related corpus** (e.g. geological or topographical texts)
3. **Hyperparameter Tuning:** Increase **learning rate warmup steps** to stabilize training and **employ learning rate schedulers** with cosine annealing, as those improve convergence and help avoid overfitting. Try different **weight-decay** values for better regularization.
4. **Using Label Smoothing for CrossEntropyLoss** to handle noisy annotations and reduce overconfidence.

Changes to the validation score calculations:

- Implement **token-level precision, recall, and F1 scores per class** to assess performance specifically on **MOUNTAIN** labels.
 - Implement **perplexity** to monitor general model learning.
-

Task 2: Sentinel-2 Image Matching

Current Approach

A feature-based matching system using DISK and LightGlue.

Improvements and Justifications

1. **Alternative Architectures:** Integrate **SuperPoint** and **SuperGlue**. SuperPoint is a learned feature detector and descriptor, while SuperGlue optimally matches these features.
2. **Data Augmentation for Seasonality:** Apply **color jittering** synthetically to improve robustness across seasons by mimicking varying environmental conditions.
3. **Hyperparameter Optimization:** Adjust **feature detection thresholds** for scale and rotation invariance and tune **descriptor dimensions** and **matching thresholds**.
4. **Inference Speed Optimizations:** **Quantize models** and apply **pruning** and implement **TorchScript compilation** for reduced latency..