Multilingual Text Summarization and Translation

Motivation

The goal is to develop a high-quality English to Telugu translation system. With increasing digital content, accessibility in regional languages like Telugu is crucial for education, healthcare, government, media, and cultural preservation. By training on 3B English tokens and 1.5B Telugu tokens, the system aims to achieve accurate, fluent, and natural translations.

Process (Pipeline)

1. **Requirements**
    - **Hardware**: GPUs/TPUs, large storage, high RAM.
    - **Software**: Python, PyTorch/TensorFlow, tokenizers.
2. **Data Collection**
    - **Parallel corpora**: subtitles, translated articles, aligned datasets.
    - **Monolingual corpora**: English (3B), Telugu (1.5B), including news, Wikipedia, books.
3. **Data Preparation**
    - Clean, normalize, deduplicate, and align text.
    - Split into training, validation, and test sets.
4. **Tokenization**
    - Use **subword tokenization** (SentencePiece/BPE).
    - Create a **shared vocabulary** for English and Telugu.
5. **Dataset Splitting**
    - **Training**: learn translation patterns.
    - **Validation/Test**: evaluate and prevent overfitting.
6. **Model Training**
    - Train a **Transformer encoder–decoder** architecture.
    - Optimize with large-scale parallel data and multiple epochs.
7. **Evaluation**
    - **Automated metrics**: BLEU, chrF.
    - **Human evaluation**: naturalness and fluency.
8. **Scaling**
    - Ensure training on **3B English + 1.5B Telugu tokens**.
    - Use **back-translation** for data augmentation.
9. **Deployment**
    - Export model in **ONNX/TorchScript**.
    - Deploy via **API or web interface**.

Learning Modules

- **Data Engineering**: Large-scale text collection, cleaning, and preparation.
- **Tokenization**: Subword-based methods for multilingual handling.
- **Neural Machine Translation (NMT)**: Transformer encoder–decoder training.
- **Evaluation Techniques**: BLEU, chrF, and human evaluations.

- **Scaling & Augmentation**: Back-translation and big data integration.
- **Deployment**: Model conversion and API integration for real-world use.

Conclusion

The English → Telugu translation project will deliver:

- A **clean dataset** with 3B English and 1.5B Telugu tokens.
- A **trained and evaluated Transformer-based model**.
- A **deployment-ready API/demo** for practical use cases.

This system not only supports education, government, and healthcare but also contributes to **cultural preservation** and **knowledge accessibility** by enabling seamless translation between English and Telugu.