From-Scratch Transformer for Extractive Question Answering

Wong Tsz Yat

Project Overview & Motivation

- Implementing a Transformer-based model from scratch for extractive QA
- Building core components: Multi-head attention, positional embeddings, feed-forward layers
- Evaluating performance on the SQuAD dataset
- Gain in-depth knowledge of Transformer architecture

Dataset: SQuAD

•Format: Context paragraphs, questions, answer spans

•**Size**:

•Training examples: 80000

•Validation examples: 10000

•**Example:**

•**Context**: "The Denver Broncos defeated the Carolina Panthers 24-10 to win Super Bowl 50..."

•**Question**: "Which NFL team won Super Bowl 50?"

•**Answer**: "Denver Broncos"

Architecture & Pipeline

- **Data Preprocessing:** JSON parsing, tokenization with BERT tokenizer, span mapping
- **Embeddings**: GloVe or trained from scratch
- **Encoder**: 2-4 Transformer blocks Multi-head self-attention Position-wise FFN with GELU; Position-wise FFN with GELU; Layer normalization & residuals
- **QA Head**: Linear layers for start/end position prediction

Training Details

- **Loss**: Sum of cross-entropies for start/end positions
- **Optimizer**: AdamW with cosine scheduling
- **Metrics**: Exact Match (EM) and F1 score
- **Augmentation**: Question rephrasing for robustness

Expected vs. Actual Results

- **Expected Performance**:
- Initial goal: 50-60% F1 score
- With tuning: 60-70% F1 score
- **Current Progress**:
- Custom model: ~15% F1 score
- Pre-trained BERT baseline: 50-60% F1

Challenges & Lessons

- **Fine-tuning difficulty**: Harder than expected to optimize the transformers from scratch
- **Tokenization issues**: Critical importance of proper answer span mapping
- **Training dynamics**: Self-attention models require careful initialization
- **Next steps**: Investigate pre-training and more advanced optimization