Fine-Tuning SpeechT5 for English Speech-to-Text on Technical Terminology Dataset

Model Overview

This project involves fine-tuning the `microsoft/speecht5_tts` model for speech-to-text tasks, with a specific focus on English technical terminology. The model was trained on the dataset `Yassmen/TTS_English_Technical_data` from Hugging Face to improve its ability to transcribe speech with precise technical terms.

Key Features:

- **Model**: SpeechT5 (Fine-tuned for speech-to-text)
- **Focus**: Accurate transcription of English technical terms
- **Loss after training**: 0.5056

Model Description

This project fine tunes the base model `microsoft/speecht5_tts by`, which is designed for automatic speech recognition (ASR), to improve its performance on domain-specific language, such as complex technical terms commonly used in fields like engineering, computer science, and data analytics.

This fine-tuned version aims to handle complex and domain-specific vocabulary typically found in technical contexts. The primary goal is to improve the transcription accuracy for technical terms and phrases that might otherwise be misinterpreted by a generic model.

Intended Uses & Limitations

Intended Uses:

- Speech-to-text tasks with an emphasis on technical terms.
- Suitable for technical presentations, conferences, or domain-specific discussions where accurate transcription of jargon is necessary.

Limitations:

- The model is fine-tuned on a specific dataset containing technical terminology, which may not generalize well for everyday conversational speech.
- Not tested on non-English languages or accents.

Dataset Description

Dataset Name: Yassmen/TTS_English_Technical_data

The dataset contains English audio recordings of technical terms and jargon. It consists of various technical domains such as engineering, computer science, data analytics, and other STEM-related fields. This dataset provides detailed vocabulary, aiming to challenge the model with domain-specific speech.

Key details of the dataset:

- **Language**: English
- **Content**: Focused on technical vocabulary from various fields
- **Format**: Audio files with corresponding text transcriptions

By leveraging this dataset, the model is better equipped to handle technical speech transcription in real-world applications.

Training Procedure

The model was fine-tuned on the above dataset using the following hyperparameters:

Training Hyperparameters:

- **Learning Rate**: 0.0001
- **Train Batch Size**: 4
- **Evaluation Batch Size**: 2
- **Seed**: 42
- **Gradient Accumulation Steps**: 8
- **Total Train Batch Size**: 32
- **Optimizer**: Adam with betas (0.9, 0.999) and epsilon 1e-08
- **Learning Rate Scheduler Type**: Linear
- **Scheduler Warmup Steps**: 100
- **Training Steps**: 600
- **Mixed Precision Training**: Native AMP (Automatic Mixed Precision)

Training Results:

The following results were recorded during training, with the final evaluation loss being 0.5056:

Training Loss	Epoch	Step	Validation Loss
0.5789	3.2258	100	0.5092
0.5046	6.4516	200	0.4970
0.4729	9.6774	300	0.4971
0.4598	12.9032	400	0.4942
0.4442	16.1290	500	0.5118
0.4421	19.3548	600	0.5056

Quantization for Efficiency

Quantization was applied to improve the model's efficiency and reduce the memory footprint. The following Python code was used for dynamic quantization. The following Python code was used for dynamic quantization:

```
def quantize_model(model):
    quantized_model = torch.quantization.quantize_dynamic(
        model, {torch.nn.Linear}, dtype=torch.qint8
    )
    return quantized_model
```

Benefits of Quantization:

- **Reduced Memory Usage**: Quantizing the model significantly reduces the model size by converting its weights to 8-bit integers (`qint8`).
- **Improved Inference Speed**: By using dynamic quantization, the model performs faster during inference without losing much accuracy.
- **Preserves Performance**: Quantization maintains the accuracy of the model in speech-to-text tasks, especially in handling technical terms.

Framework Versions

The following versions of libraries were used during model training and fine-tuning:

- **Transformers**: 4.44.2
- **PyTorch**: 2.4.1+cu121
- **Datasets**: 3.0.1
- **Tokenizers**: 0.19.1

Conclusion

This fine-tuned SpeechT5 model has been specifically designed for transcribing English technical speech with high accuracy, focusing on challenging technical terminology. By using the dataset `Yassmen/TTS_English_Technical_data` and applying dynamic quantization, the model delivers efficient and accurate results for technical transcription tasks. Further improvements and more information on its intended uses, limitations, and dataset-specific performance may be explored in future updates.