Date : 21.05.2025
Name & Surname: BURAK YILDIRIM
Student ID : 220201047

Generative AI Project Report

Project Name: Beyit & Şiir Creator - Türkçe Şiir Üretimi için GPT-2 Fine-Tuning

1. Project Objective

The main goal of this project is to generate Turkish classical poetry (especially in couplet and quatrain forms) by fine-tuning a pre-trained GPT-2 model with a curated Turkish poetry dataset. The model is trained to capture the nuances, poetic structures, and stylistic features of Turkish literary works to produce creative and syntactically correct poems.

2. Methods and Technologies Used

- **Model:** GPT-2 (124M parameters)
- **Technique:** Fine-tuning with HuggingFace `Trainer` API
- **Dataset:** Turkish poems by famous poets (e.g., Yunus Emre, Fuzuli, Nazım Hikmet)
- **Platform:** Google Colab (GPU-based training)
- **Libraries:**
  - `transformers`, `datasets`, `torch`, `sklearn`, `pandas`

Model Configuration:

Parameter	Value
Tokenizer	Custom GPT-2 tokenizer (trained on Turkish data)
Max Sequence Length	128 tokens
Epochs	5
Batch Size	2
Learning Rate	5e-5

Based on the available information, the following activities are inferred:

3. Project Workflow

1. **Dataset Collection & Preprocessing:**
    a. Poetry from several Turkish poets was collected and cleaned.
    b. Each poem was separated using \n\n, and special characters were removed.
2. **Tokenizer Training:**
    a. A new tokenizer was trained on the Turkish poetry dataset to better handle unique Turkish characters and structures.
3. **Model Fine-Tuning:**
    a. The base GPT-2 model was fine-tuned using the HuggingFace Trainer on Turkish poetry.
4. **Text Generation:**
    a. Poems were generated based on user input (e.g., a word or line).
    b. Beam search and top-k sampling were used to enhance creative generation.

4. Challenges Faced

- **GPU Limitations:**
  Google Colab's 12GB GPU made it necessary to keep the batch size small.
- **Tokenizer Compatibility:**
  Turkish diacritics and poetic syllable structures were incompatible with the original GPT-2 tokenizer, requiring a custom tokenizer.
- **Limited Dataset Size:**
  Overfitting was a risk due to the limited amount of Turkish poem data. Techniques like early stopping were used.

6. Model Comparison

Feature	Fine-Tuned GPT-2 (this project)	Pretrained GPT-2 (English)
Turkish Understanding	☑	❌
Poetic Structure	☑	❌
Creative Output	☑	☑
Customization Possible	☑	❌

7. Conclusion & Future Work

This project successfully fine-tuned GPT-2 to generate Turkish couplets and poems. The model shows strong fluency and thematic alignment with Turkish literary styles.

Possible Future Improvements:

- Use of a larger GPT-2 model (e.g., 345M or 774M)
- Integration of a rhyme detection module (kafiye)
- Development of a web or desktop GUI for users to interact with the poem generator
- Expanding the dataset for better semantic depth

Results & Outputs:

```
Enter your start word:
Yol


Yolu bir şarkı gibi
Çalar kalbimde her daim
```

```
Enter your start word:
Mavi


Mavi bir sandıktaydı gülüşün
Ama kalbim hâlâ seninle konuşuyor
```

```
Enter your start word:
Sevgi
Sevgiyle örülü bir dünya kurmak
Seninle yaşamak en güzel duygular

Sevgiyle dolu kalbim sana ait
Sonsuza dek sürecek bu hikaye
```

References

- HuggingFace `datasets` library.
- HuggingFace `transformers` library.
- PyTorch library.
- Custom Turkish poetry dataset used for fine-tuning.