Report on

Email Subject Line Generation using Fine-Tuned LLM

Submitted by-

Md Sayed Delowar

sayeddelowar.buet@gmail.com | +880 1300-684644 | linkedin.com/in/s-delowar | github.com/s-delowar

Objective

The goal of this project was to fine-tune a Large Language Model (LLM) to generate concise, relevant subject lines from email bodies using a processed subset of a public dataset, followed by training and evaluation using standard text generation metrics.

Approach

Dataset

I have used the **Yale-LILY/aeslc** dataset from Hugging Face, which contains email bodies and corresponding subject lines from the Enron email corpus. From the original dataset:

- Train set: 14,436 examples (cleaned to 12,794)
- Validation set: 1,960 (cleaned to 1,734)
- Test set: 1,906 (cleaned to 1,718)

Preprocessing

1. **Cleaning**: Removed empty entries, duplicates, and filtered subject lines by length (6–80 characters).
2. **Formatting**: Transformed email bodies into instructive prompts using randomized instruction templates like: “Generate a concise subject line for this email:”
3. **Final Format**:
    - Input: Instruction + Email Body
    - Output: Subject Line

Processed data was saved in CSV format to Google Drive for use in training.

Model and Training Strategy

- Model used: `google/flan-t5-small` – a lightweight instruction-tuned encoder-decoder model.
- Processed CSV files were loaded and converted to Hugging Face Datasets.
- Tokenization: Inputs were tokenized with a max input length of 512 tokens and output length of 32 tokens.
- Fine-tuning framework: Hugging Face Transformers with `Seq2SeqTrainer`.
- Fine-tuning was performed on Google Colab using T4 GPU.
- Training Configuration:
    - Epochs: 6
    - Batch Size: 8 (both training and evaluation)
    - Learning Rate: 3e-5
    - Weight Decay: 0.01 (to prevent overfitting)
    - Evaluation Strategy: Performed at the end of each epoch
    - Save Strategy: Checkpoints saved after each epoch, retaining the latest 2
    - Text Generation: `predict_with_generate=True` enabled generation-based evaluation
    - Model Hub Integration: Automatically pushed the model to the Hugging Face Hub after training
    - `DataCollatorForSeq2Seq`: automatically pad with the maximum length within a batch.
    - Metrics: `ROUGE (ROUGE-1, ROUGE-2, ROUGE-L, ROUGE-Lsum)`

The model was trained and pushed to the Hugging Face Hub at: sdelowar2/flan-t5-email-subjectline

Performance Metrics and Results

The model was trained for six epochs with steady improvements observed in both loss and ROUGE scores. Training loss decreased from 2.99 to 2.73, while validation loss remained stable around 3.66–3.69, indicating consistent generalization.

Table-1: Key performance metrics

Metric	Validation Set		Test Set
	Epoch - 1	Epoch - 6	
ROUGE-1	0.298	0.317	0.297
ROUGE-2	0.151	0.164	0.159
ROUGE-L	0.293	0.311	0.292
ROUGE-Lsum	0.292	0.312	0.292

The ROUGE scores showed steady improvement, peaking at Epoch 6, with consistent but modest gains. Test set scores closely align with validation results, confirming the model's robustness.

Qualitatively, the model was able to generate short, contextually appropriate, and relevant subject lines.

Table-2: Sample Qualitative Results

Actual Subject Line	Generated Subject Line
Response to Redrock Air Permit Delay	Redrock Air Permit Issue
Pollution Prevention Training	Environmental Training
Update to Virus Scanning Software	Computer Virus Update

Observations & Challenges

Observations:

- The model consistently produced subject lines that were concise, contextually relevant, and professional in tone.
- Generated outputs often captured the core message of the email, showing good semantic understanding.
- Even with limited model size and data, the model generalized well to unseen samples.
- The minimal change in validation loss suggests that the model reached a performance plateau, possibly due to data complexity, or limited model capacity.

Challenges:

- Token length limits: Some long email bodies required aggressive truncation, potentially omitting useful context.
- Using a small LLM (flan-t5-small) restricted performance gains compared to larger models.
- The small dataset size (~14k samples) limited the model’s exposure to diverse email contexts and phrasing.

Source Code & Test Output on GitHub: https://github.com/S-Delowar/LLM-Email-Subjector