Technical Report: Fine-tuning and Fusion of Embedding Models for AQA2024

Xingwu Hu
China Telecom, Shanghai, China
huxingwu@gmail.com

Xuantao Lu
Xiaohongshu Inc., Shanghai, China
luxuantao@xiaohongshu.com

June 24, 2024

1 Introduction

This report provides a detailed description of the methodology used to fine-tune and fuse embedding models for the AQA2024 competition. The primary objective was to leverage two pre-trained embedding models, Alibaba-NLP/gte-large-en-v1.5 and Snowflake/snowflake-arctic-embed-l, and improve their performance through fine-tuning and ensemble techniques.

2 Prerequisites

The following prerequisites are required for the implementation:

- Linux operating system.
- Python 3.9.
- PyTorch 2.3.0 with CUDA 12.0.
- Additional Python packages: transformers, datasets, FlagEmbedding.

3 Installation of Dependencies

To install the necessary dependencies, execute the following commands:

```
pip install transformers datasets
pip install -U FlagEmbedding
```

If any other packages are missing, they can be installed using `pip install`.

4 Data Processing

Place the training and development data in the `AQA` directory and the test data in the `AQA-test-public` directory. Run the following script to preprocess the data:

```
python clean_text/make_embedding_data.py
```

5 Fine-tuning Embedding Models

5.1 Fine-tuning Alibaba-NLP/gte-large-en-v1.5

To fine-tune the Alibaba-NLP/gte-large-en-v1.5 model and obtain inference scores for the top 200 candidate articles for each question, execute the following script:

```
sh first_try/gte_embedding_train.sh
```

If you prefer not to train the model, you can download the pre-trained checkpoint from this link. Extract the checkpoint and rename the folder to `embedding`$_o utput2, then run the inference script$:
`sh clean_text_test/clean_text_inference_probs.sh`
Rename the resulting file `AQA-test-public/result_test.jsonl` to `result_gte.jsonl`.

5.2 Fine-tuning Snowflake/snowflake-arctic-embed-l

Similarly, fine-tune the Snowflake/snowflake-arctic-embed-l model using the following script:

```
sh first_try/snowflake_embedding_train.sh
```

Alternatively, download the pre-trained checkpoint from this link. Extract the checkpoint and rename the folder to `embedding`$_o utput2, then run the inference script$:
`sh clean_text_test/clean_text_inference_probs.sh`
Rename the resulting file `AQA-test-public/result_test.jsonl` to `result_snowflake.jsonl`.

6 Model Fusion

To fuse the results from the Snowflake and GTE models, ensure that `result_snowflake.jsonl` and `result_gte.jsonl` are in the root directory. Then, run the following script:

```
python ensemble.py
```

7 Results on B Leaderboard

The performance of the individual and fused models on the B leaderboard is summarized in the table below:

Model	B Leaderboard Score
Snowflake/snowflake-arctic-embed-l	0.160779090207083
Alibaba-NLP/gte-large-en-v1.5	0.17240293828095
gte+snowflake	0.184657914972311

8 Methodology

8.1 Fine-tuning Embedding Models

Using the FlagEmbedding library, we fine-tuned the Alibaba-NLP and Snowflake models to adapt them to the specific task. This involved training the models on a curated dataset to improve their embedding representations.

8.2 Model Fusion

We employed an ensemble technique to combine the predictions from both fine-tuned models, leveraging the strengths of each model to improve overall performance.

8.3 Data Augmentation

We utilized special data construction techniques to enhance the training dataset, thereby improving the model's ability to generalize to unseen data.

9 Conclusion

The fine-tuning and fusion of the Alibaba-NLP and Snowflake embedding models showed significant improvements in performance on the AQA2024 competition. The ensemble method, in particular, yielded the best results, demonstrating the effectiveness of combining multiple models.