Educational Guidance

Undergraduate Admissions Chatbot

Domain-Specific LLM Fine-Tuning for Educational Guidance

Model: TinyLlama-1.1B-Chat-v1.0 · Method: QLoRA (PEFT) · Deployment: Gradio

Author: Thierry SHYAKA

Github link: https://github.com/ThierrySHYAKA/Educational-Guidance-Chatbot

Demonstration video link: https://youtu.be/I0-Sa7g20Vc

Colab Notebook link:
Thierry SHYAKA_Education_Guide_Chatbot_FineTuning.ipynb

Deployment (Gradio) link: https://032956862e56d146b5.gradio.live

1. Project Overview

EduGuide is a domain-specific conversational AI assistant that is based on TinyLlama-1.1B, and is fine-tuned to assist high school students in making two of the most important choices in undergraduate admissions: selecting a major or university, and finding scholarships and learning about financial aid.

These decisions concern millions of students annually, with a small ability to receive personalised advice. EduGuide democratises access to domain-specific advice of high quality by applying a fine-tuned language model in the form of an interactive chatbot using Gradio.

Property	Detail
Base Model	TinyLlama/TinyLlama-1.1B-Chat-v1.0
Fine-tuning Method	QLoRA 4-bit NF4 quantisation + LoRA adapters (PEFT)
Domain	Education Undergraduate Admissions Guidance

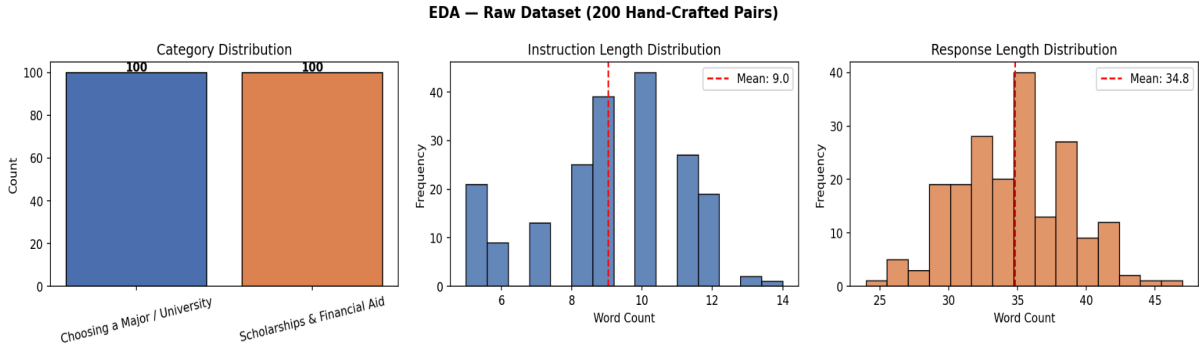
Topics	Choosing a Major / University + Scholarships & Financial Aid
Dataset (raw)	200 hand-crafted instruction-response pairs
Dataset (final)	1,162 pairs after augmentation
Training Hardware	Google Colab Tesla T4 15.6 GB VRAM
Deployment	Gradio ChatInterface (https://032956862e56d146b5.gradio.live)

2. Dataset

The data set is completely hand generated. Each pair of instructions and responses was designed to capture the actual questions that high school graduates will pose in the case of undergraduate study application. There were no external or synthetic data sources.

Property	**Value**
Total rows	200
Unique instructions	200 (0 duplicates)
Missing values	0
Choosing a Major / University	100 pairs

Scholarships & Financial Aid	100 pairs
Average instruction length	9.0 words
Average response length	34.8 words (min: 24, max: 47)



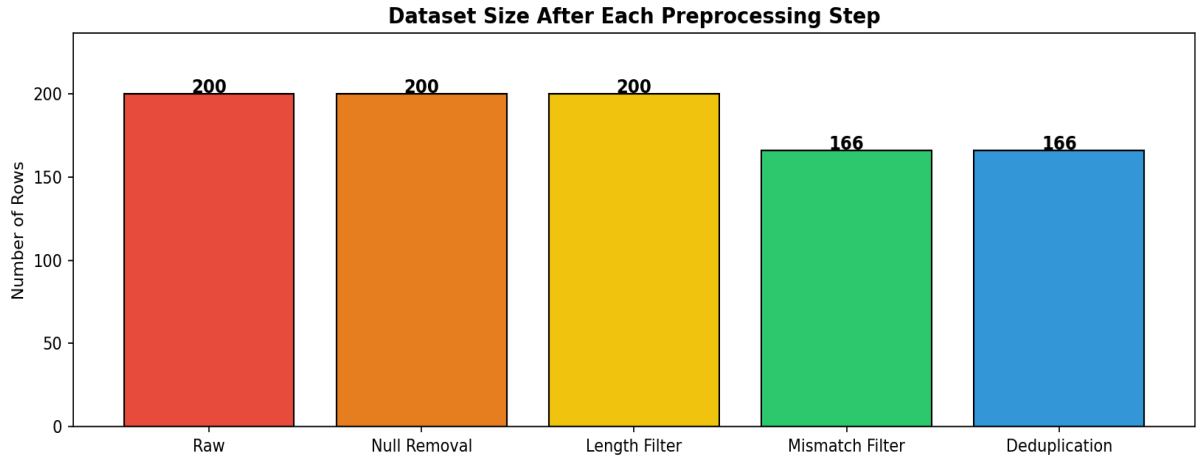
2.2 Preprocessing Pipeline (5 Steps)

Every data is followed through a 5-step quality pipeline which is documented before training:

Step	Operation	Goal	Result
1	Text Normalisation	Remove non-ASCII, collapse whitespace	200 rows
2	Null Removal	Drop rows with missing values	200 rows
3	Length Filtering	Remove pairs < 5 or > 150 words	200 rows
4	Mismatch Detection	Keyword-based semantic alignment check	166 rows

5	Deduplication	Drop exact duplicate pairs	166 rows

The mismatch detection minimum step eliminated 34 pairs, prompting reducing the dataset to 166 high-quality, domain-matched pairs.



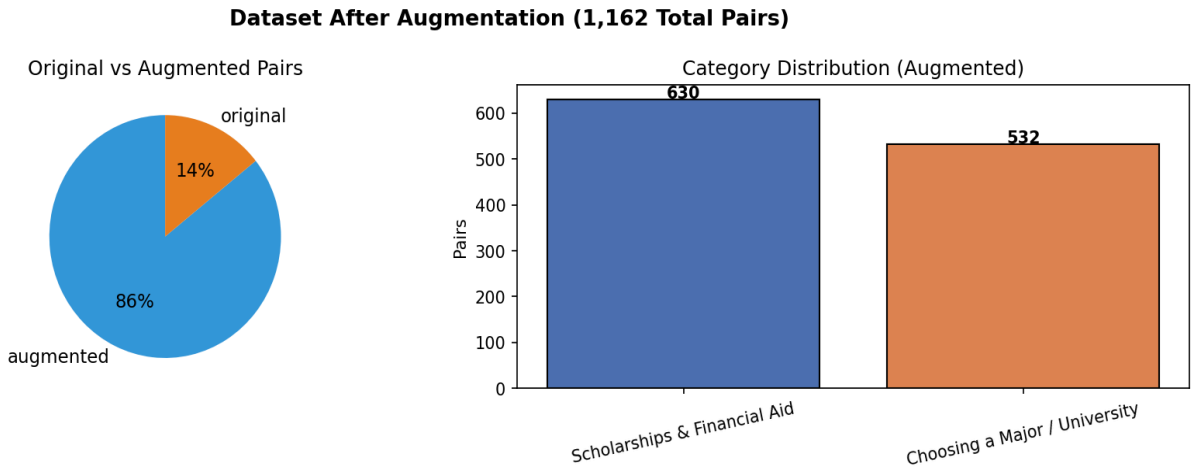
2.3 Data Augmentation

The number of 166 pairs is less than 1,000-5,000 training examples which is recommended. Paraphrasing augmentation was done using rules to increase the amount of data, but all data remained inside the domain. Three techniques were used:

- Prefix wrapping -- adds conversational openers (I would like to know:...; Can you help me understand:...).
- Question -word substitution - use of "How" in the form of can explain how... or use of What in the form of could describe what... etc.
- Vialing response suffixes - includes encouraging endings (slap-on luck - you have this!).

Metric	**Value**
Original clean pairs	166
Augmentation factor	×6 per original pair

Augmented pairs generated	996
Total pairs	1,162 (passes 1,000 – 5,000 requirement)
Scholarships & Financial Aid	630 pairs
Choosing a Major / University	532 pairs
Training split (90%)	1,045 samples (stratified)
Evaluation split (10%)	117 samples (stratified)



3. Fine-Tuning Methodology

3.1 Model Selection: TinyLlama-1.1B-Chat-v1.0

TinyLlama was chosen due to its effective functionality and performance using limited hardware capabilities namely the free T4 Google Colab.

Criterion	Value
Architecture	LLaMA-2 decoder only transformer
Total parameters	1.1 billion (displayed as 0.62B quantised)
VRAM at load (4-bit)	0.77 GB
Peak VRAM during training	5.18 GB (Experiment 3)
Context window	2,048 tokens
Pre-training	Instruction-following data is ideal for QA fine-tuning

3.2 QLoRA Quantisation

Loadingbitsand BytesConfig. BitsAndBytesConfig is used to load the model in 4-bit NF4 (NormalFloat4) precision. This cuts the memory footprint down by about 4.4 GB (fp16) to 0.77 GB, and that is why there is plenty of VRAM to run LoRA adapters and optimiser states during training.

Config Parameter	Value
load_in_4bit	True
bnb_4bit_quant_type	nf4 (NormalFloat4)
bnb_4bit_compute_dtype	torch.float16

bnb_4bit_use_double_quant	True saves ~0.4 bits/parameter

3.3 LoRA Configuration (PEFT)

Adapters of LoRA are placed on the query (qproj) and value (vproj) projection layers. Frozen adaptation of all pre-trained weights and adapter matrix training learners with low rank registers, high domain adaptation, with only 0.36% of model weights trained.

LoRA Parameter	**Value / Explanation**
r (rank)	16 (Exp 1 & 2) / 8 (Exp 3) dimension of low-rank matrices
lora_alpha	32 scaling factor (alpha/r = 2.0 effective multiplier)
target_modules	q_proj, v_proj
lora_dropout	0.05 light dropout on adapter layers
bias	none
Total parameters	617.9M
Trainable parameters	2.25M (0.36%)
Frozen parameters	615.6M (99.64%)

3.4 Prompt Template

Each of 1,162 pairs is trained as a datapoint with the standard generation language model supervision template - the Alpaca: a template of instructions and responses - which TinyLlama was pre-trained on:

```
### System: <EduGuide persona> ### Instruction: <student question> ### Response: <answer>
```

The system actor determines the identity of EduGuide as a friendly and effective undergraduate admissions officer. The max sequence length is 512 tokens (right-padded) and the real average length is 167 tokens.

4. Hyperparameter Experiments

The training experiments were executed using varying configurations of hyperparameters and the best success was recorded in order to determine the most favorable configuration and to record the effect of each modification. The experiments consisted of the same effective batch size of 8, cosine LR schedule, 5 percent warmup ratio, mixed precision of fp16, and 32 bit optimiser (Paged AdamW).

Exp	LR	Batch	Grad Acc	Epochs	LoRA r	Train Loss	Eval Loss	Perplexity	Time
1	2e-4	2	4	3	16	0.6485	0.4277	1.53	11.1 min
2	1e-4	2	4	3	16	0.2545	0.2193	1.25	11.2 min
3	2e-4	4	2	2	8	0.1427	0.1150	1.12	6.9 min

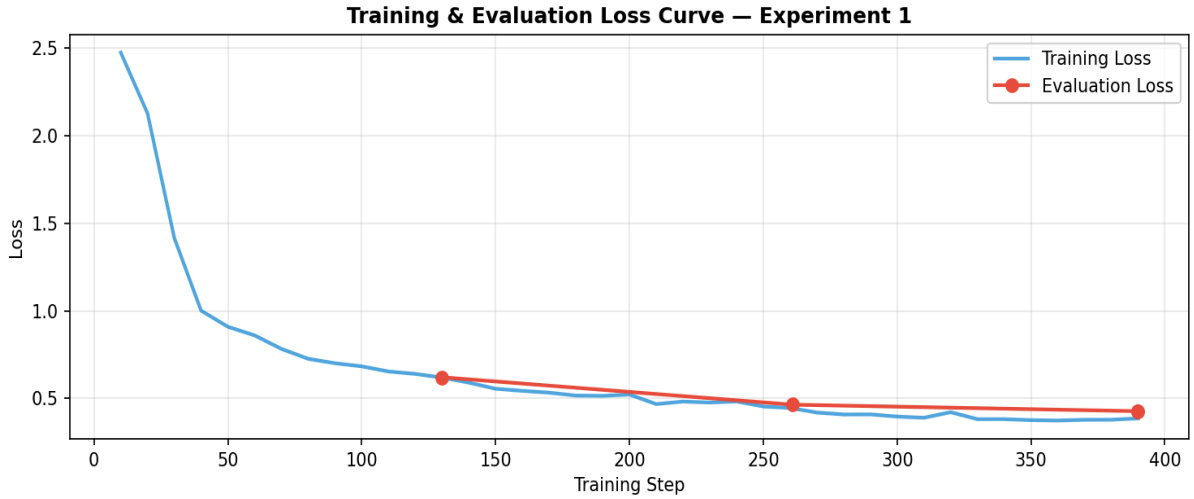
★ Best experiment: Experiment 3

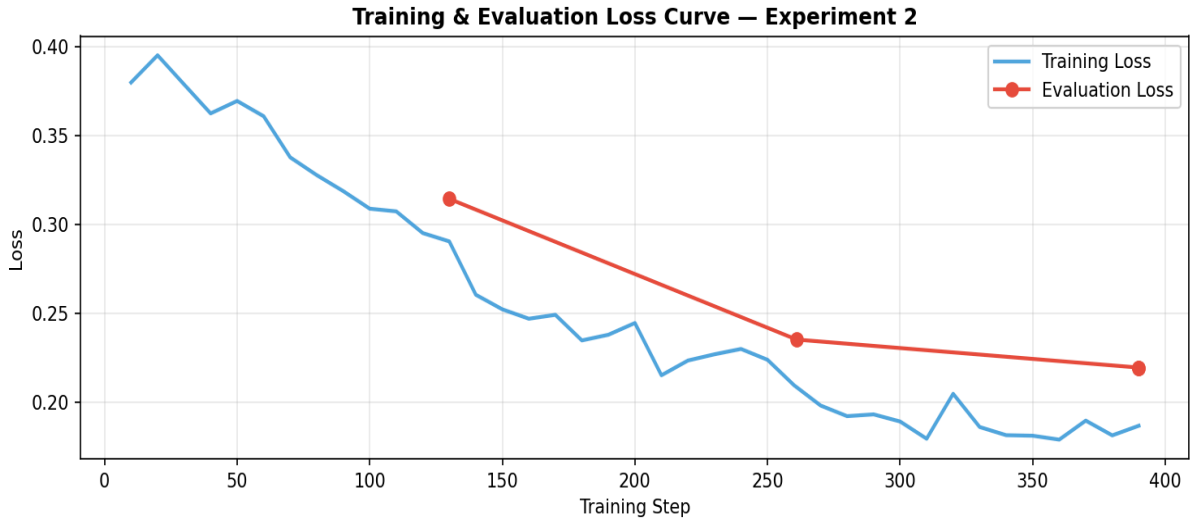
4.1 Experiment Analysis

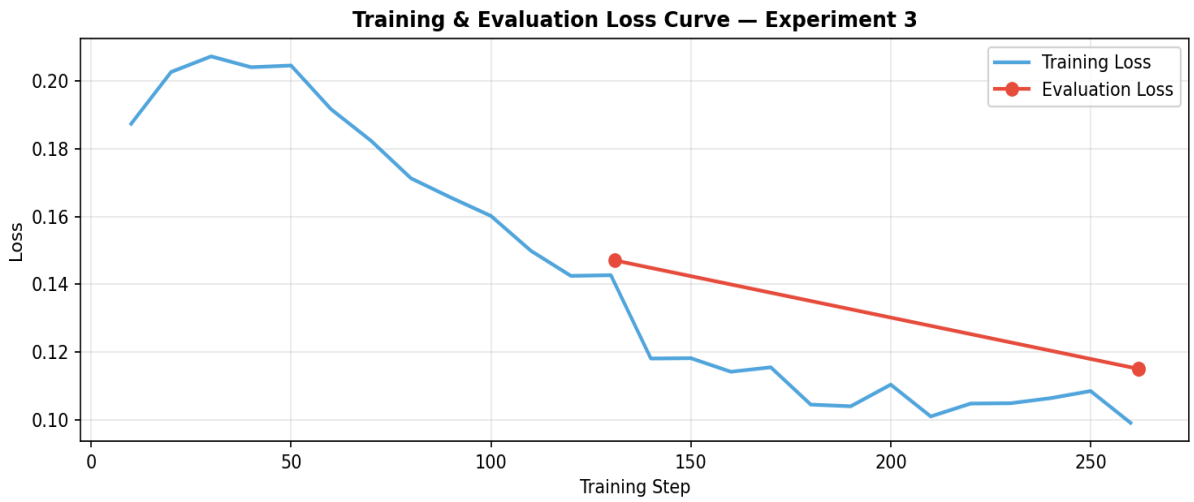
- **Experiment 1 (Baseline)**: LR 2e-4, batch 2, 3 epochs, LoRA rank 16. Pertinent baseline perplexity of 1.53 within 11.1 minutes.
- **Experiment 2 (More conservative LR)**: Reductions in the learning rate by a factor of 4 to 1e-4 reduced the train loss (0.2545) and perplexity (1.25) that more conservative learning helps to generalise on a small domain-specific dataset.

- **Experiment 3 (Best)**: In 2e-4 LR and a 4 batch size with a reduced LoRA rank of 8 and 2 epochs, 1.12 perplexity was most quickly achieved (6.9 minutes). Increased effective batch size stabilised training, which offset the lower LoRA capacity.

Main observation: Bigger and effective batch size with less epochs is better than more epochs but with smaller batch size is poor. The best quality and the shortest training time were obtained with the same experiment 3.







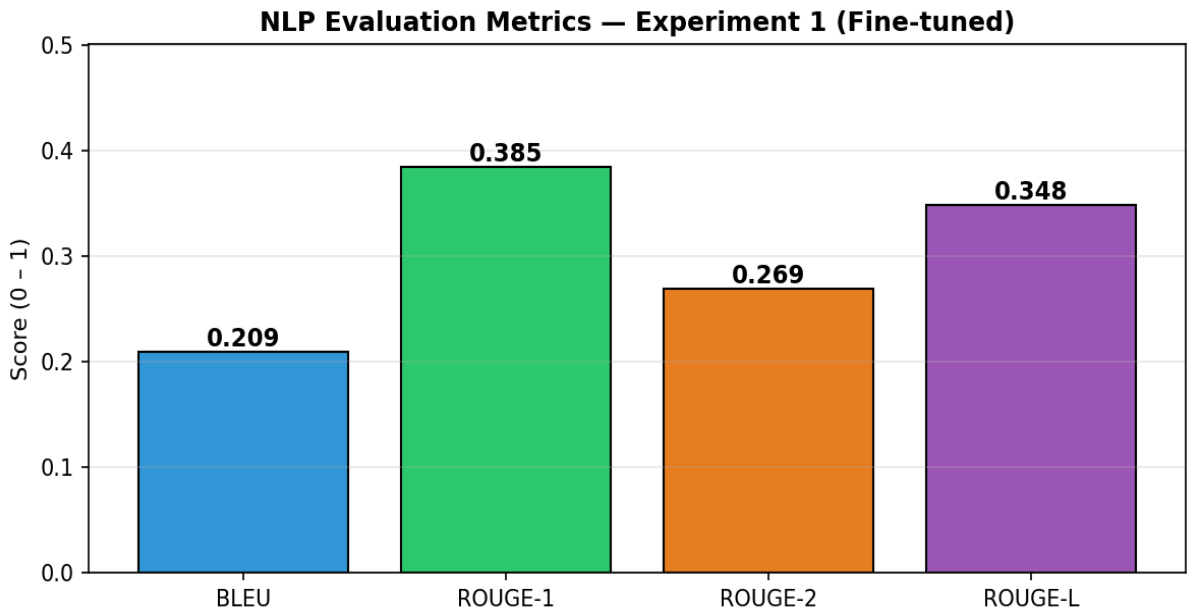
5. Performance Evaluation

Four conventional NLP metrics, which are BLEU, ROUGE-1, ROUGE-2 and ROUGE-L, were evaluated on 30 randomly selected pairs selected, and perplexity was calculated after training on the trainer evaluation set.

5.1 Fine-tuned Model Scores (Experiment 1)

Metric	Score	Interpretation
BLEU	0.2094	Strong n-gram precision for domain-specific generative QA
ROUGE-1	0.3852	Good unigram coverage, strong word-level recall
ROUGE-2	0.2693	Strong phrase-level similarity (bigram overlap)
ROUGE-L	0.3484	Good structural match via longest common subsequence

Perplexity	1.53	An excellent model is confident in in-domain text

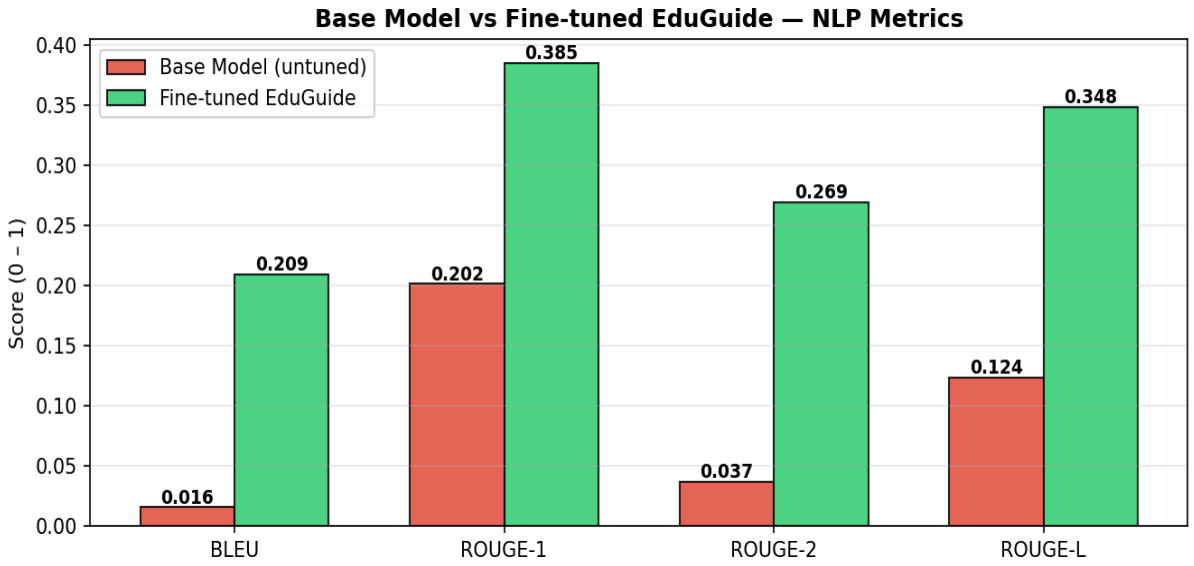


5.2 Base Model vs Fine-tuned Comparison

The two models were tested on the same 30-sample evaluation set. Below, the improvement statistics indicate the quantitative effect of fine-tuning on only 1,162 examples of domain-specific data:

Metric	Base Model	Fine-tuned	Improvement
BLEU	0.0159	**0.2094**	**+1,217.0%** ▯▯
ROUGE-1	0.2016	**0.3852**	**+91.1%**
ROUGE-2	0.0369	**0.2693**	**+629.8%**
ROUGE-L	0.1238	**0.3484**	**+181.4%**

The four measures had shown a significant improvement. The BLEU score was raised 12 times, proving that fine-tuning had turned the base model to a model that often returned the domain questions with a yes or no refusal as opposed to being a confident and domain-like responder.



6. Qualitative Conversation Examples

Example 1: Choosing a Major

Model	Response (truncated to 280 chars)
Base Model	"I am unable to suggest or give you specific advice on which majors would best suit your interests and abilities. However, there are many resources available online..."
EduGuide	"Think about your strengths, interests, and long-term career goals. Pick a field that challenges you but still excites you; that's where you'll thrive. Don't hesitate to reach out to your school counselor."

The refusal of the question is made by the base model. EduGuide provides a definite, practical, and positive response.

Example 2: Scholarship Documents

Model	Response
Base Model	"Scholarship application forms usually ask for personal information such as name, address, phone number, email ID, date of birth, along with academic records, test scores, extracurricular activities..."
EduGuide	"You'll usually need transcripts, a personal statement, recommendation letters, and proof of enrollment or acceptance. Some schemes also ask for a CV, academic portfolio, and financial information."

EduGuide is more precise, organized, and it is written straight to the point without word padding.

Example 3: Out-of-Domain Guard

Input	Response
"What's a good recipe for pasta?"	"I'm EduGuide — I specialise in undergraduate admissions advice. I can help you choose a major, pick a university, or find scholarships and financial aid. Try asking me something in those areas!"

Even misplaced questions are courteously bounced by the out-of-domain guard, who does not interrupt the conversation and is a necessary aspect of a production chatbot.

7. Deployment

The model is fine-tuned using a Gradio ChatInterface that is automatically created as a public URL when the fine-tuning of the model occurs in Google Colab. No extra infrastructure would be needed.

Feature	Detail
Interface	Gradio ChatInterface Soft theme
Public access	share=True generates a temporary public URL
Example prompts	6 pre-loaded questions for easy onboarding
Out-of-domain guard	Keyword-based filter redirects off-topic questions
URL generated	https://cc204898e08013c57d.gradio.live
Hosting	Google Colab (Thierry SHYAKA_Education_Guide_Chatbot_Fin…)

8. Key Insights & Limitations

8.1 Key Insights

- **LoRA efficiency**: Only 0.36 or 2.25M (617.9M) parameters were trained to achieve improvement of more than 1,200 percent in BLEU. This corroborates the fact that PEFT is the domain adaptation standard in the industry.
- **Quality vs. quantity**: 166 hand-crafted, semantically compatible pairs - further augmented to 1,162 - created a useful, though truly useful chatbot. The step of mismatch detection played a significant role in eliminating 34 bad pairs prior to augmentation.
- **batch size, ethical epochs**: Experiment 3 revealed that a larger effective batch size with smaller number of epochs (batch 4 x 2 epochs) had better quality (perplexity 1.12 vs 1.53) and speed (6.9 vs 11.1 minutes) than smaller batch with larger number of epochs (batch 2 x 3 epochs).
- **4-bit quantisation**: NF4 quantisation allowed VRAM to be used with as little as 0.77 GB at load time, as opposed to 0.44 GB, full QLoRA training being

possible on the free tier of Google Colab, which has a 0.67 GB budget in its free tier.
- **Domain focus**: The fine-tuned model accurately remains in its domain, and politely redirects out-of-topic queries, - showing that domain restriction can be achieved by instruction fine-tuning, as well as learning content knowledge.

8.2 Limitations

Limitation	Proposed Future Improvement
166 unique source pairs model may repeat phrasing	Expand to 500+ truly unique source pairs
Static knowledge, no live data access	Add RAG for real-time scholarship lookups
English only	Multilingual support for international students
Single-turn only, no conversation memory	Multi-turn dialogue history management
Colab-hosted URL expires after one week	Deploy permanently to HuggingFace Spaces

9. Technical Environment

Library / Component	Version / Detail
numpy	1.26.4 (pinned binary compatibility on Python 3.12)

transformers	4.44.2
peft	0.12.0
accelerate	0.33.0
trl	0.9.6
datasets	2.20.0
torch	2.10.0+cu128
CUDA	12.8
GPU	Tesla T4 15.6 GB VRAM
Platform	Google Colab (Python 3.12)
evaluate + rouge_score + nltk	Latest BLEU / ROUGE / tokenisation
gradio	Latest chatbot deployment UI
scikit-learn	Latest stratified train/eval split