CSC490 A4: Creating an RL LLM as a Judge

Armaan Rehman Shah - 1009641309
Boaz Cheung - 1007673607
Alice Sedgwick - 1009301355
Yuan Yu - 1008782195

November 5, 2025

1 Part One — Background Research on GRPO and RLVR (20%)

1.1 Scope and Motivation

Our target is a judge model that scores immigration QA along factuality, coverage, relevance, and citation support. We study RL with verifiable rewards (RLVR) and closely related alignment methods to understand training signals that best promote reliable judgments.

1.2 Five Relevant Papers and Commentary

Paper (Year)	**Method**	**Key Contributions/Notes**	**Relevance to an Immigration RAG Judge**
Wen et al. (2025)	RLVR via GRPO	Verifiable, binary rewards on tasks (math/code) improve multi-step reasoning; introduces CoT-Pass@K; theory connects verifiable rewards and correct reasoning.	Use auto-checkable signals (e.g., exact/entailment matches against IRCC text) to shape judge toward faithful, step-aware verdicts.
Rafailov et al. (2023)	DPO (RL-free)	Closed-form objective that removes PPO instability; strong performance on preference tasks with simple supervised optimization.	Alternate path to train a judge from ranked pairs (correct vs. incorrect responses) without full RL stack.
Bai et al. (2022)	RLAIF (Constitutional AI)	AI-written principles and AI preference data reduce human labels; two-stage SFT+RL pipeline.	Encode "constitution" for judging (e.g., require citations to official sources, penalize unsupported claims) and bootstrap with AI feedback.
Nakano et al. (2022)	RLHF + Web browsing	Tool-use for retrieval with human-preference rewards; answers must include sources; beats human demonstrators on long-form QA.	Direct template for judge emphasis on evidence-supported answers in open-domain settings like immigration policy.

Glaese et al. (2022)	RLHF with rules (Sparrow)	Targeted rule-checking + evidence requirements; reduces unsafe/incorrect answers; improves trust.	Multi-criteria judgments (factuality, completeness, safety) mapped to learnable rewards; requires source evidence.

Takeaways. (1) Verifiable signals are powerful even when binary; (2) preference-based alignment can be done with or without RL (DPO); (3) rule- and evidence-conditioned reward models consistently improve factual QA; (4) requiring citations during training improves evaluator reliability.

2 Part Two — Metrics and Evaluations (30%)

We define five metrics relevant to high-stakes immigration QA. For each: why it matters, how we measure it (quant/qual), and scale challenges.

2.1 M1: Factual Consistency (Faithfulness)

Why. Answers must match official IRCC text; errors can mislead users.
How (Quant). Claim extraction → textual entailment vs. retrieved IRCC passages; or Question-Answer Generation/entailment scoring; sentence-level pass/fail aggregated to a score.
How (Qual). Human reviewers manually verify a sample of answers by comparing them against official IRCC documents, with relevant source passages highlighted to show which parts support each claim.
Scale Challenges. Subtle paraphrases; nuanced policy language; compute cost for per-claim verification at scale.

2.2 M2: Hallucination Rate

Why. Unsupported details erode trust. Dangerous if users believe specific details that are wrong.
How (Quant). Extract individual claims and search the retrieved documents for supporting evidence. Claims without any matching text spans are flagged as potential hallucinations. The hallucination rate is the percentage of claims lacking document support.
How (Qual). Human reviewers categorize each claim using a four-level rubric: (1) Fully Supported - direct evidence in documents, (2) Weakly Supported - reasonable inference from documents, (3) Unsupported - no basis in documents, (4) Contradicted - conflicts with documents.
Scale Challenges. Distinguishing acceptable inference vs. invention; avoiding string-match pitfalls.

2.3 M3: Coverage (Completeness)

Why. Immigration answers often require full criteria, steps, fees, or timelines.
How (Quant). Create gold-standard checklists of required facts for common question types (e.g., Express Entry requires: language test, ECA, work experience, funds, CRS score). Coverage is measured as the percentage of checklist items mentioned in the answer. Alternatively, we compute recall: the fraction of key facts from source documents that appear in the generated answer.
How (Qual). Human reviewers identify material omissions - important facts missing from the answer that could mislead users or cause application failures. Reviewers distinguish between critical omissions (missing requirements) and minor ones (missing helpful context). Answers are rated as

Complete, Mostly Complete, Incomplete, or Severely Incomplete.
Scale Challenges. Building gold checklists; balancing brevity vs. completeness.

2.4 M4: Relevance and Conciseness

Why. On-topic, clear answers improve usability.
How (Quant). Use a trained classifier to score how relevant each sentence in the answer is to the original question (0-1 scale). Eenforce length guidelines: answers should be 20-150 words for simple questions, with penalties for overly brief ($\leq$10 words) or excessively long ($\geq$ 300 words) responses.
How (Qual). Human reviewers (or LLM-as-judge) rate answers on four dimensions: (1) On-topic - directly addresses the question, (2) Clear - easy to understand without jargon, (3) Concise - appropriately brief without rambling, (4) Useful - provides actionable information. Each dimension scored 1-5, aggregated into an overall quality rating.
Scale Challenges. Subjectivity in what counts as "helpful context"; tuning thresholds.

2.5 M5: Source Attribution (Evidence Support)

Why. Verifiable answers require correct citations to IRCC or other official sources.
How (Quant). We compute two metrics: (1) Precision - what percentage of provided citations actually support their associated claims (verified by checking if the cited document contains the claimed information), and (2) Recall - what percentage of claims in the answer have citations. Additionally, we perform span-level verification: for each citation, we fetch the source page and search for the specific text that supports the claim, flagging citations that point to generic pages or contain outdated information.
How (Qual). Human reviewers see answers alongside their cited sources in a split-screen view. For each claim-citation pair, they verify: (1) the link works and leads to an official source, (2) the cited page actually contains the claimed information, (3) the source is current and not outdated, and (4) the citation is appropriately specific (not just linking to a homepage). Reviewers flag missing citations, incorrect citations, and citation spam (multiple links to the same generic source).
Scale Challenges. Preventing "citation spam"; mapping multi-source claims; entailment vs. exact overlap.

Table 2: Summary of metrics and practical measurement notes.

Metric	**Measurement (Quant/Qual)**	**Key Pitfalls/Challenges**
Factual Consistency	NLI/entailment on claims; QAG scorer; human spot-check	Nuance, paraphrase, compute cost
Hallucination Rate	Unsupported-claim fraction; adjudication rubric	Distinguish inference vs. invention
Coverage	Checklist/expected-fact recall	Checklist construction; brevity trade-off
Relevance/Conciseness	Sentence relevance; length rules; clarity rubric	Subjectivity of "helpful" context
Source Attribution	Evidence precision/recall; span checks	Citation spam; multi-source mapping

3 Part Three — Baseline Classifier for One Metric (15%)

3.1 Chosen Metric

We target **Factual Consistency** as our metric. Given a question and answer pair, the goal is to predict whether the answer is factually accurate in the context of Canadian immigration.

3.2 Dataset and Splits

We constructed a custom dataset of 200 Q&A pairs labeled as `faithful` (1) or `not faithful` (0). The data reflects questions asked by potential applicants across all IRCC-relevant categories (e.g., study permits, Express Entry, PR, visitor visas).

Splits:

- `train.jsonl` – 140 samples
- `val.jsonl` – 30 samples
- `test.jsonl` – 30 samples

3.3 Baselines

LLM Prompting. We use `gpt-3.5-turbo` in zero-shot setting with a deterministic prompt: "Is the following answer accurate and complete based on current Canadian immigration policy? Respond with YES or NO only."

Logistic Regression. A scikit-learn logistic regression model trained on sentence embeddings (from `all-MiniLM-L6-v2`) and cosine similarity features between question and answer pairs.

3.4 Evaluation Protocol

- **Primary metrics:** Accuracy, F1-score (with faithfulness as positive class)
- **Class-specific performance:** Precision/Recall for each label
- **Qualitative analysis:** Inspect disagreements and label/model failures

3.5 Results

Table 3: Model performance on test set of 30 samples.

Model	**Accuracy**	**F1 (Label 1)**	**Notes**
LLM Prompting (GPT-3.5)	76.67%	–	Minor misclassifications on nuanced policy
LogReg + Embeddings	63.33%	0.70	High recall, poor precision for label 1

3.6 Error Analysis

LLM Failures. LLM incorrectly flagged true answers as false for evolving policies (e.g., off-campus work rules, implied status) and borderline edge cases.

LogReg Failures. The classifier showed strong recall for label 1 but misclassified many label 0 examples as positive due to limited feature capacity (e.g., surface-level embedding similarity).

Shared Challenges.

- **Policy ambiguity:** Some real-world answers depend on exact program date or applicant stream, which models cannot fully infer from surface text.
- **Contradictions not explicit:** Some incorrect answers sound plausible unless matched against official policy text.
- **Data noise:** Some label-0 examples include partially correct content, increasing difficulty.

4 Part Four — RLVR Training Pipeline (35%)

4.1 Metric and Hypothesis

Target Metric: We measure **factual correctness** of immigration Q&A responses as binary classification (correct=1, incorrect=0).

Hypothesis: Parameter-efficient fine-tuning with LoRA will enable the judge model to achieve over 85% accuracy in identifying factually correct immigration answers.

4.2 Training Pipeline

Our training pipeline has four main steps:

1. **Input**: Question-answer pair with ground truth label
2. **Judge Model**: Generate prediction (YES or NO)
3. **Reward**: Compare prediction with label (+1 correct, -1 incorrect)
4. **Update**: Backpropagate to update LoRA adapters only

4.3 Base Model

Model: Qwen/Qwen2.5-1.5B-Instruct

Why this model:

- Small enough for 8GB VRAM (our hardware: 3070Ti)
- Pre-trained on instruction-following tasks
- Good baseline performance on reasoning

Parameter-Efficient Fine-Tuning with LoRA:

- Base model: 1.5B parameters (frozen, 4-bit quantized)
- LoRA adapters: 19M parameters (trainable)
- Only 1.2% of parameters are trainable
- Memory: 1.5GB instead of 6GB

4.4 RL Algorithm

Algorithm: Supervised Fine-Tuning with Verifiable Rewards

We use a simplified RLVR approach here instead of full GRPO:

- Ground truth labels provide verifiable rewards
- Model learns to generate ”YES” or ”NO” directly
- Loss computed only on target tokens (prompt masked)

Why simplified approach:

1. Small dataset (200 samples) - complex RL methods risk instability
2. Binary classification is simple - doesn’t need sophisticated exploration
3. Verifiable rewards eliminate need for learned reward models
4. More stable and faster to implement than full GRPO

4.5 Logging and Metrics

Logging Tools:

- TensorBoard for real-time visualization
- JSON logs for programmatic analysis

Tracked Metrics:

- Training loss (every 10 steps)
- Learning rate schedule
- Validation accuracy (every 50 steps)
- Precision, Recall, F1 score

4.6 Ablation Studies

We tested three hyperparameters: training epochs, learning rate, and LoRA rank.
Key Findings:

- **Epochs matter most:** 1 epoch gave 67% accuracy, 5 epochs gave 93% (26% improvement)
- **Higher learning rates work better:** 5×10^{-5} achieved 93% vs 87% with 1×10^{-5}
- **LoRA rank has no effect:** All three ranks (8, 16, 32) gave identical 90% accuracy

Table 4: Ablation study results

Hyperparameter	Value	Accuracy	F1
Epochs	1	66.7%	0.706
	3	90.0%	0.880
	5	**93.3%**	**0.923**
Learning Rate	1×10^{-5}	86.7%	0.857
	2×10^{-5}	90.0%	0.880
	5×10^{-5}	**93.3%**	**0.929**
LoRA Rank	8	90.0%	0.880
	16	90.0%	0.880
	32	90.0%	0.880

4.7 What Didn't Work

Challenges:

1. **Quantization without LoRA:** Initial attempts with 4-bit quantization failed - loss stayed at 0 and model generated gibberish. Fixed by adding LoRA adapters.
2. **Loss computation:** Early attempts to extract YES/NO from logits led to NaN losses. Fixed by switching to full sequence generation.
3. **Small dataset:** Only 200 samples limited model's ability to generalize. LoRA rank ablation showed no effect due to insufficient data.

Limitations:

- Not true GRPO - missing group-relative advantages and policy gradients
- Only evaluates factual correctness (not relevance, completeness, etc.)
- Small test set (30 samples) gives wide confidence intervals
- No confidence scores - only binary predictions
- Trained only on immigration domain

4.8 Next Steps

1. **Implement full GRPO:** Add group-based advantage computation and compare with current baseline
2. **Expand dataset:** Increase to 1000+ samples through data augmentation
3. **Add confidence scores:** Output probability instead of just YES/NO
4. **Multi-aspect evaluation:** Extend beyond correctness to assess relevance and completeness
5. **Cross-domain testing:** Test generalization to other domains (More complex legal cases)

References

[1] Shao, Z. et al. (2024). DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.

[2] Hu, E. et al. (2021). LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685.

[3] Dettmers, T. et al. (2023). QLoRA: Efficient Finetuning of Quantized LLMs. arXiv:2305.14314.

[4] Wen, Y. et al. (2025). RLVR/GRPO and CoT-Pass@K: Reinforcement Learning with Verifiable Rewards for Chain-of-Thought Reasoning.

[5] Rafailov, R. et al. (2023). Direct Preference Optimization: Your Language Model is Secretly a Reward Model. arXiv:2305.18290.

[6] Bai, Y. et al. (2022). Constitutional AI: Harmlessness from AI Feedback. arXiv:2212.08073.

[7] Nakano, R. et al. (2022). WebGPT: Browser-assisted Question-answering with Human Feedback. arXiv:2112.09332.

[8] Glaese, A. et al. (2022). Improving Alignment of Dialogue Agents via Targeted Human Judgements. arXiv:2209.14375.