PatentGRPO: Group Relative Policy Optimization for Patent Text Generation

Jieh-Sheng Jason Lee

Assistant Professor

NYCU School of Law, Taiwan

2025-11-14

nycu

NATIONAL
YANG MING CHIAO TUNG
UNIVERSITY

Agenda

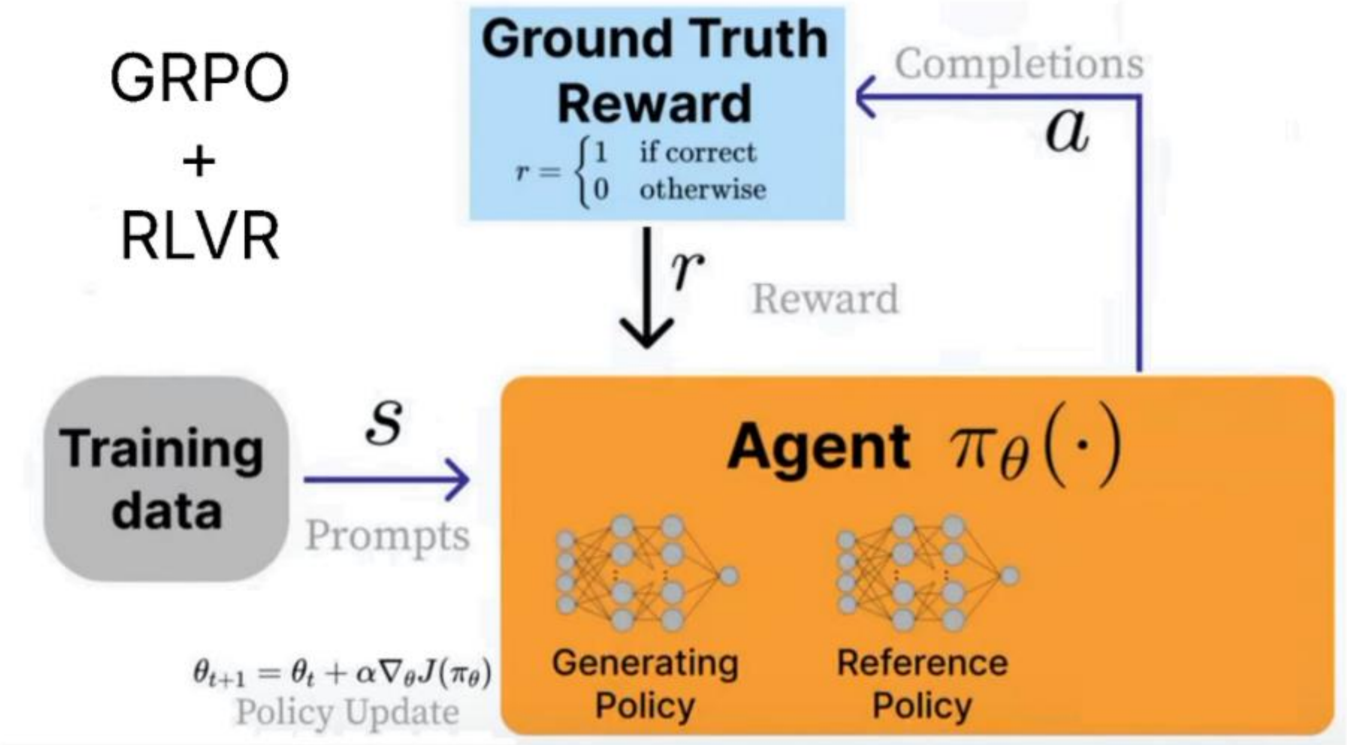
- Introduction
- Implementation
- Conclusion

Introduction

- Idea
  - apply DeekSeek techniques to patent tasks
- Which technique?
  - GRPO (Group Relative Policy Optimization)
- Which task?
  - Patent Text Generation
    - Controlling Patent Claim Length

What is GRPO

- RLHF → PPO → GRPO and RLVR
  - Reinforcement Learning from Human Feedback
  - Proximal Policy Optimization
  - Group Relative Policy Optimization
  - Reinforcement Learning with Verifiable Rewards


GRPO
+
RLVR
Ground Truth Reward
$r = \begin{cases} 1 & \text{if correct} \\ 0 & \text{otherwise} \end{cases}$
Completions
a
r
Reward
Training data
s
Prompts
Agent $\pi_\theta(\cdot)$
Generating Policy
Reference Policy
$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J(\pi_\theta)$
Policy Update

- Generating Policy (current trained model)
- Reference Policy (original model)

(source: https://docs.unsloth.ai/get-started/reinforcement-learning-rl-guide)

Why controlling patent claim length

- A patent grants the patent holder the exclusive right to utilize an invention.
- The scope of protection is determined by the patent claims.
- In practice, the objective is to maximizing protection scope
    - generally, shorter claims encompass fewer features, which are less limiting and lead to a broader patent scope
    - longer claims include more detailed features, resulting in a narrower patent scope
- Since controlling patent claim length can influence the patent scope, this paper explores GRPO for controllability of patent claim text generation

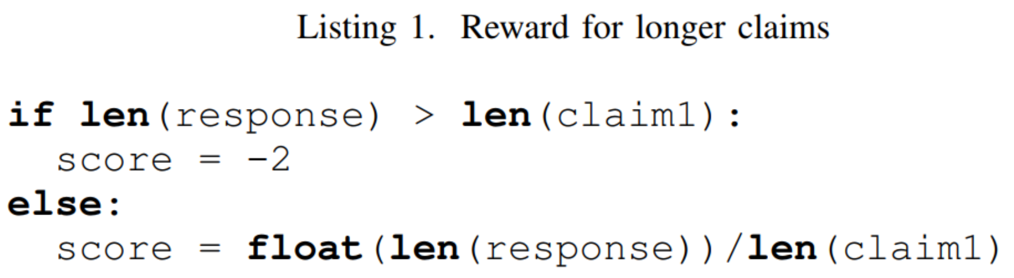
Implementation

- Dataset
  - Patent Claims
    - (USPTO) Artificial Intelligence Patent Dataset
    - 30,701 granted patents and 603,088 patent claims
- Codebase
  - Jupyter notebook by Unsloth team
  - Runable on a single consumer-grade GPU (24GB)
- Model
  - Qwen3 (4B)
  - Supervised Fine-Tuning (SFT) using Low-Rank Adaptation (LoRA)

Experiments

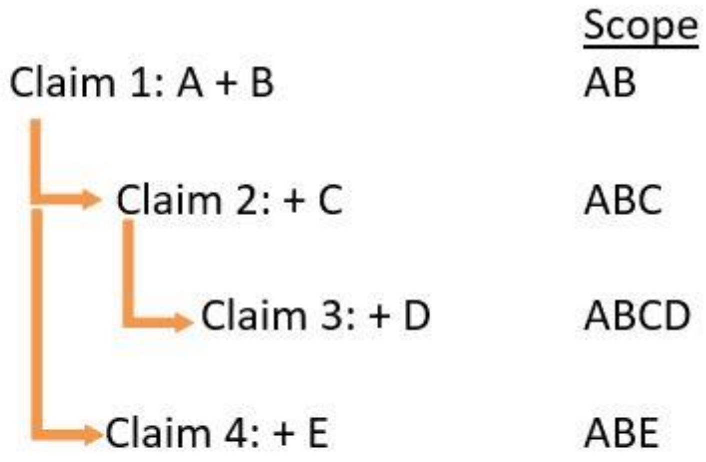
- Experiment 1
  - generate longer patent claims
    - output: a dependent claim derived from claim 1
    - dependent claims are typically shorter than their independent claims
  - reward function

$$\frac{\text{length of response}}{\text{length of claim1}}$$



Listing 1. Reward for longer claims

```
if len(response) > len(claim1):
    score = -2
else:
    score = float(len(response))/len(claim1)
```



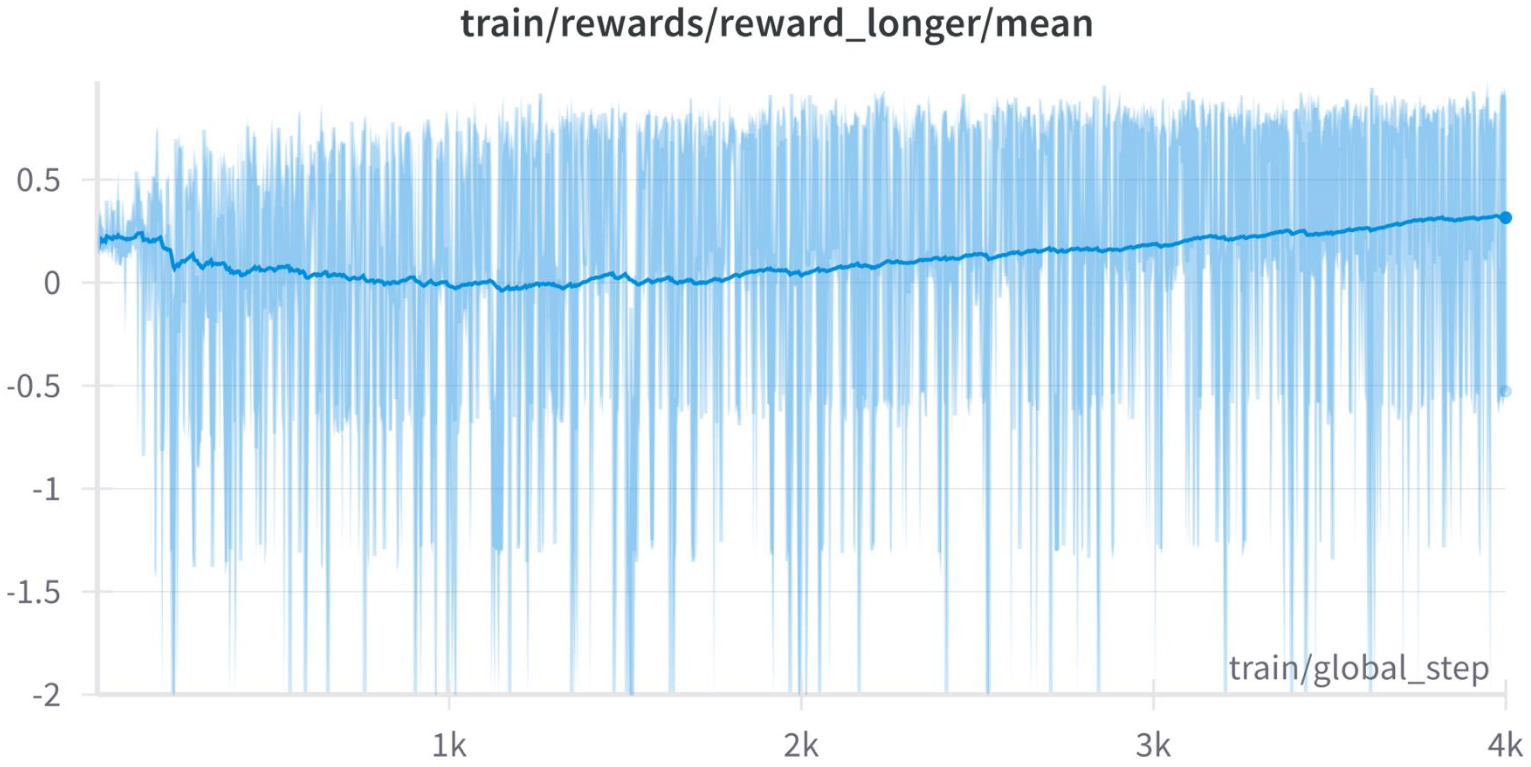
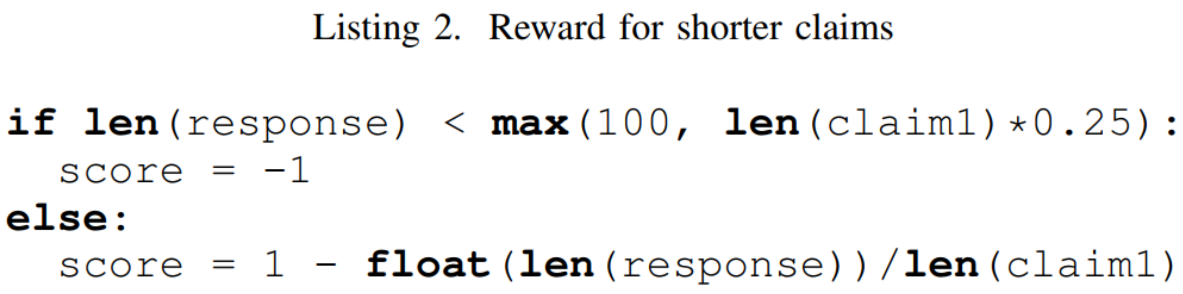


Fig. 1. Reward longer claims.

- Experiment 2
  - generate shorter patent claims
    - output: a dependent claim derived from claim 1
    - avoid overly brief responses: a minimum length of 100 characters or at least 0.25 times the length of claim 1
  - reward function

$$1 - \frac{\text{length of response}}{\text{length of claim1}}$$



Listing 2. Reward for shorter claims

```
if len(response) < max(100, len(claim1)*0.25):
  score = -1
else:
  score = 1 - float(len(response))/len(claim1)
```

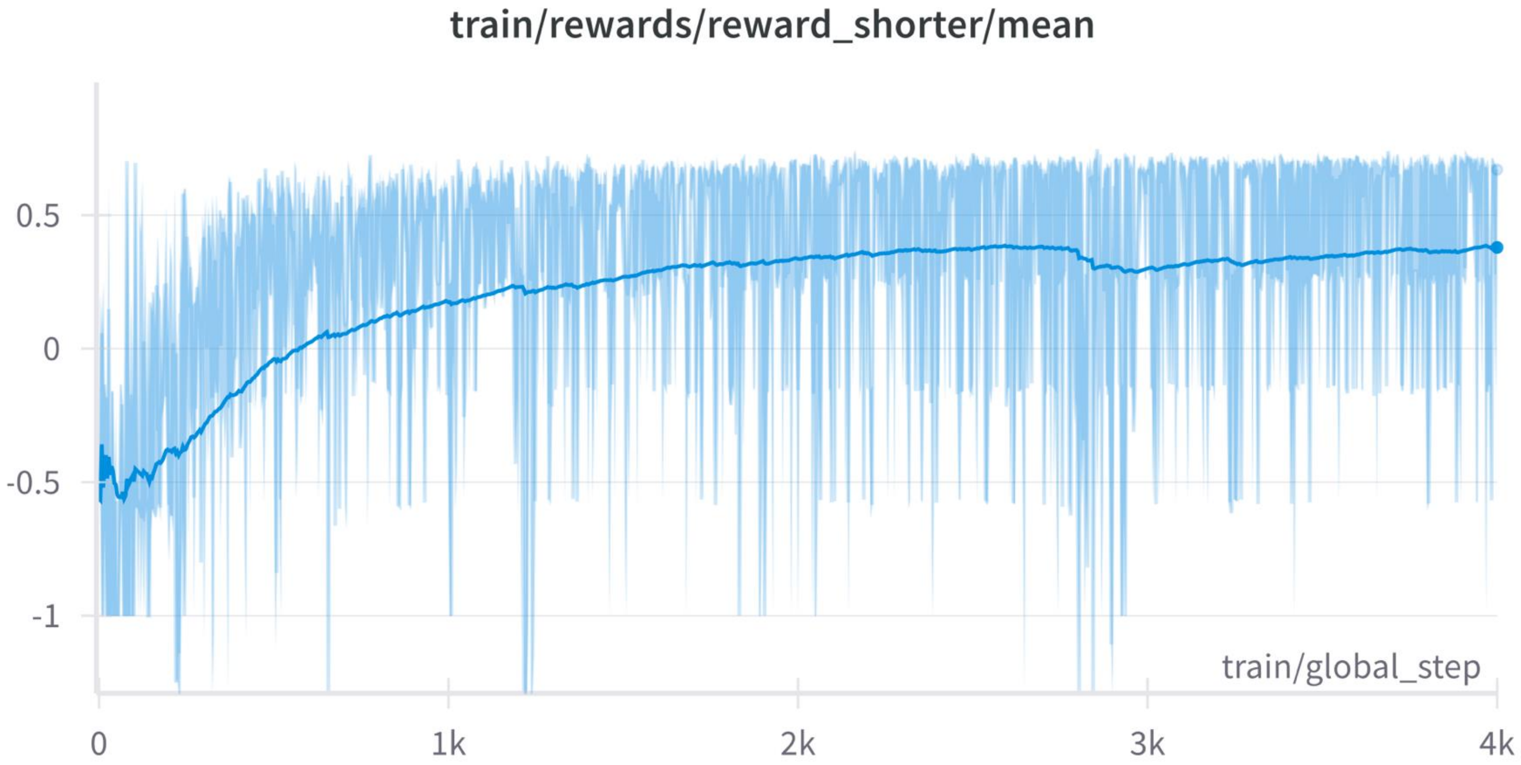
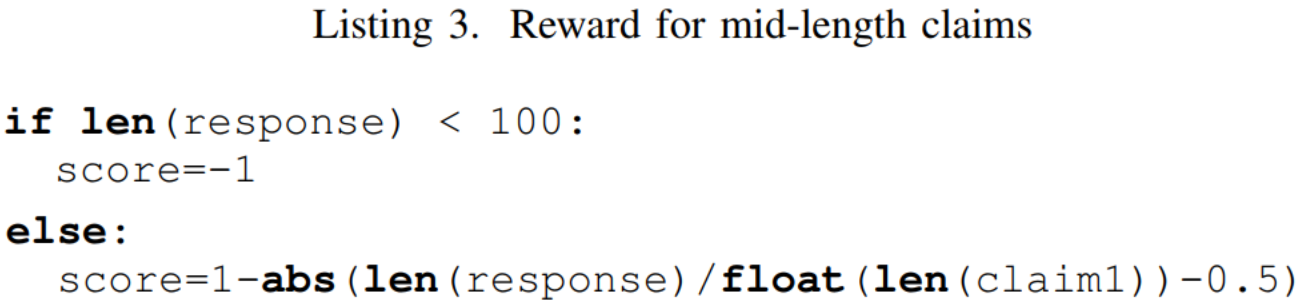


Fig. 2. Reward shorter claims.

- Experiment 3
  - generate mid-length patent claims
    - output: a dependent claim derived from claim 1
    - use half the length of claim1 as the optimal length
  - reward function

$$1 - abs(\frac{\text{length of response}}{\text{length of claim1}} - \frac{1}{2})$$



Listing 3. Reward for mid-length claims

```
if len(response) < 100:
  score=-1
else:
  score=1-abs(len(response)/float(len(claim1))-0.5)
```

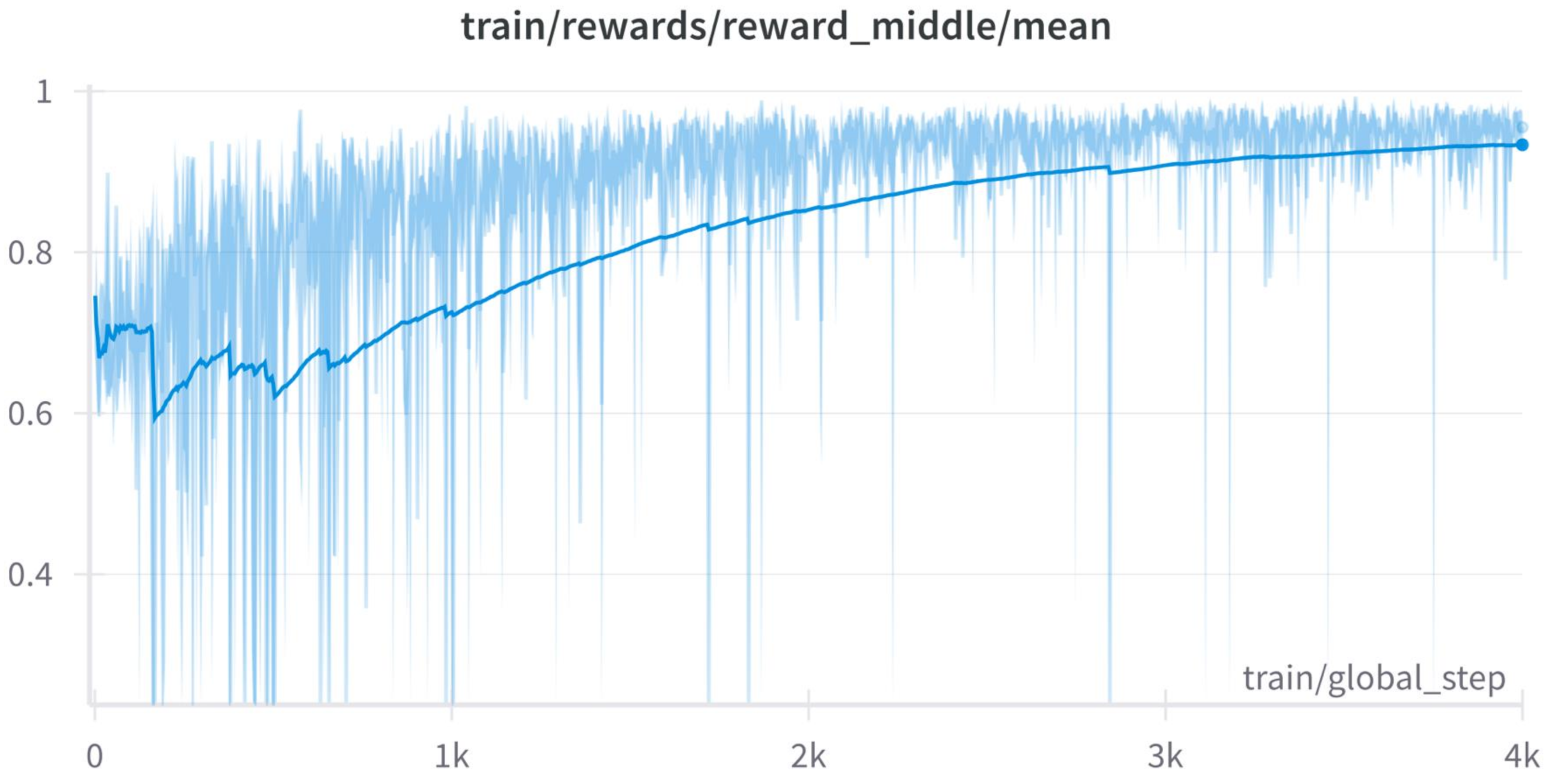
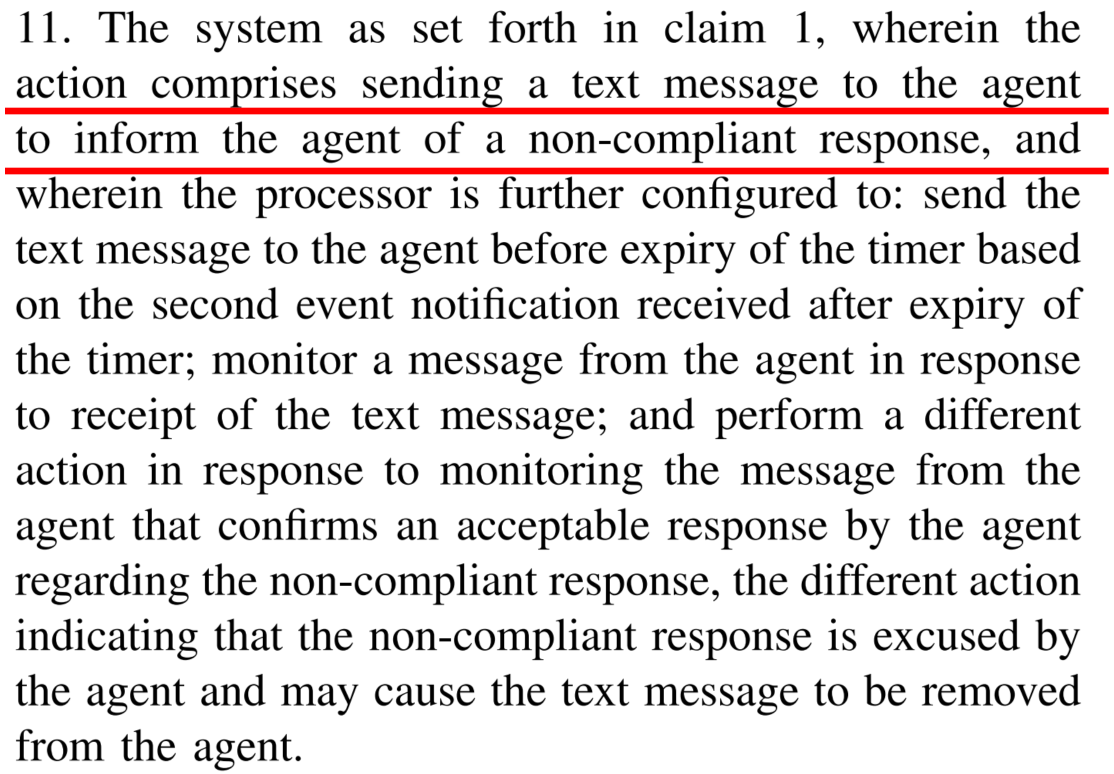


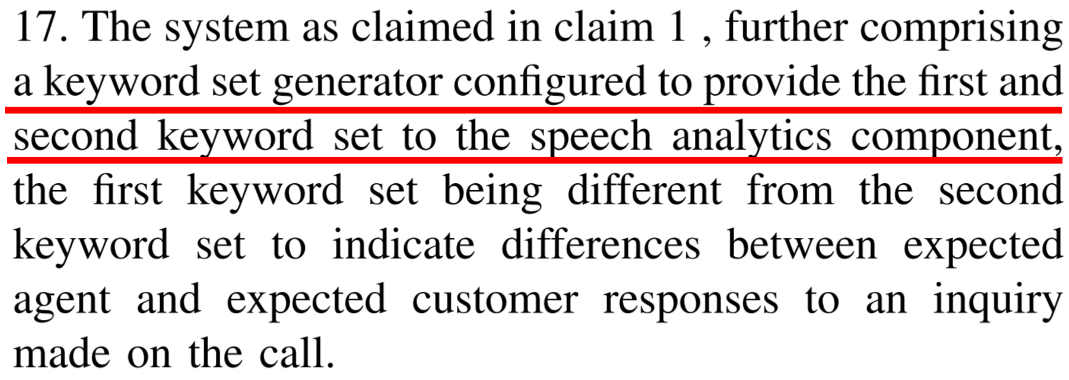
Fig. 3. Reward mid-length claims.

- 9,473,634
- 1. A system for monitoring compliance by an agent of a contact center on a call to a remote party comprising: a processor configured to: receive a first event notification from a speech analytics component indicating detection of a first keyword from a first keyword set in speech associated with the call from the remote party; start a timer after receipt of the first event notification; receive a second event notification from the speech analytics component after receiving the first event notification, the second event notification indicating detection of a second keyword from a second keyword set in speech associated with the call from the agent; and perform an action in response to determining the second event notification is not received prior to expiry of the timer, wherein the action comprises displaying a visual indication on a display of a computer terminal used by the agent informing the agent of a non-compliant response.
- ...
- 12. The system of claim 1 , wherein the first keyword set reflects indication of a reassigned telephone number and the action further comprises prompting the agent to disposition the reassigned telephone number as a reassigned number.



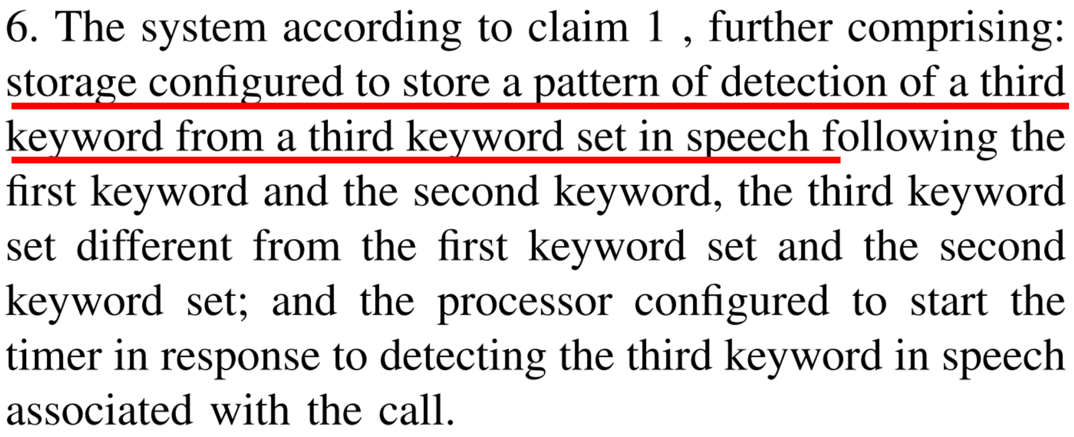
11. The system as set forth in claim 1, wherein the action comprises sending a text message to the agent to inform the agent of a non-compliant response, and wherein the processor is further configured to: send the text message to the agent before expiry of the timer based on the second event notification received after expiry of the timer; monitor a message from the agent in response to receipt of the text message; and perform a different action in response to monitoring the message from the agent that confirms an acceptable response by the agent regarding the non-compliant response, the different action indicating that the non-compliant response is excused by the agent and may cause the text message to be removed from the agent.

Experiment 1



17. The system as claimed in claim 1 , further comprising a keyword set generator configured to provide the first and second keyword set to the speech analytics component, the first keyword set being different from the second keyword set to indicate differences between expected agent and expected customer responses to an inquiry made on the call.

Experiment 2



6. The system according to claim 1 , further comprising: storage configured to store a pattern of detection of a third keyword from a third keyword set in speech following the first keyword and the second keyword, the third keyword set different from the first keyword set and the second keyword set; and the processor configured to start the timer in response to detecting the third keyword in speech associated with the call.

Experiment 3

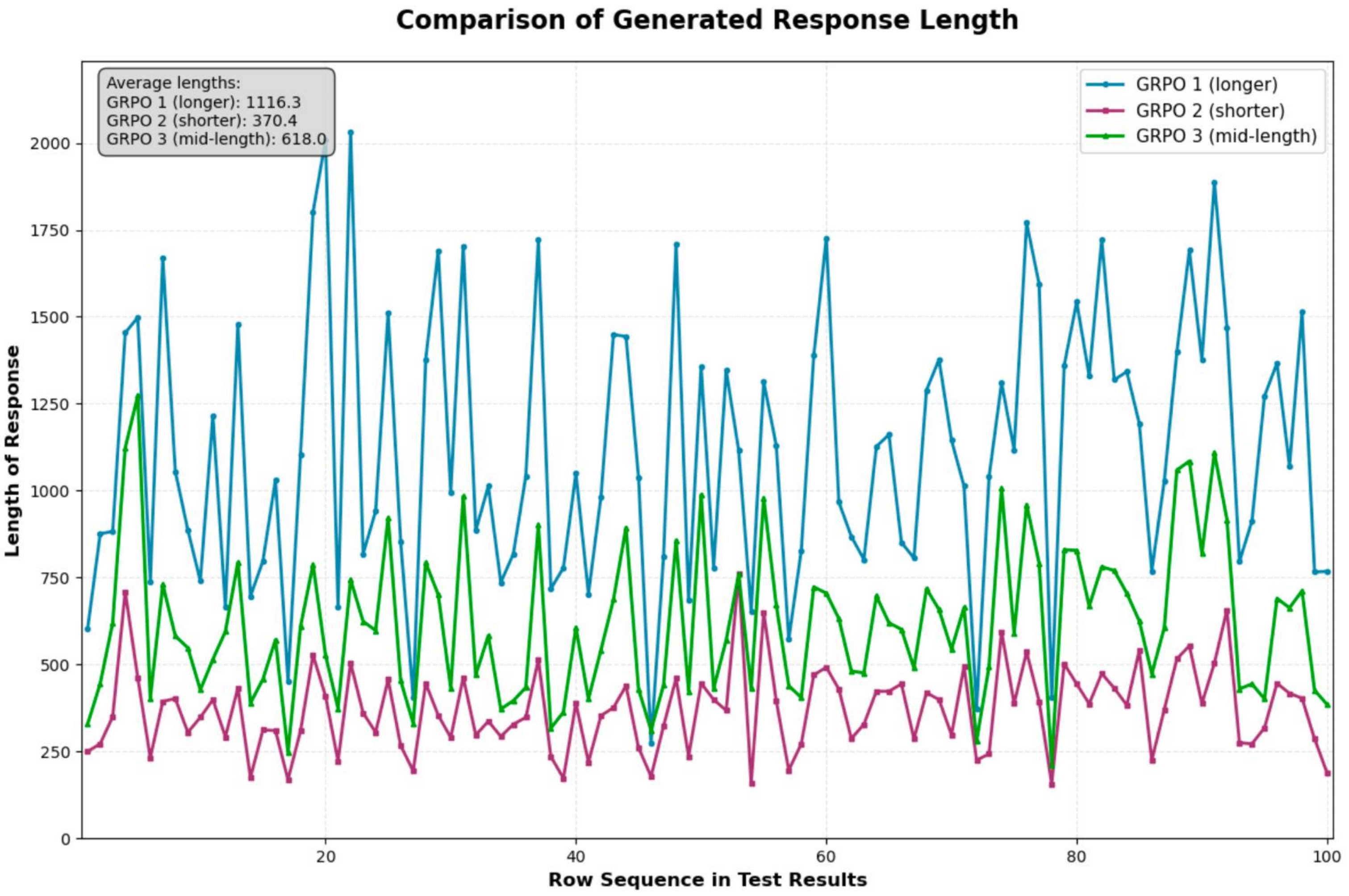


Fig. 4. GRPO comparison

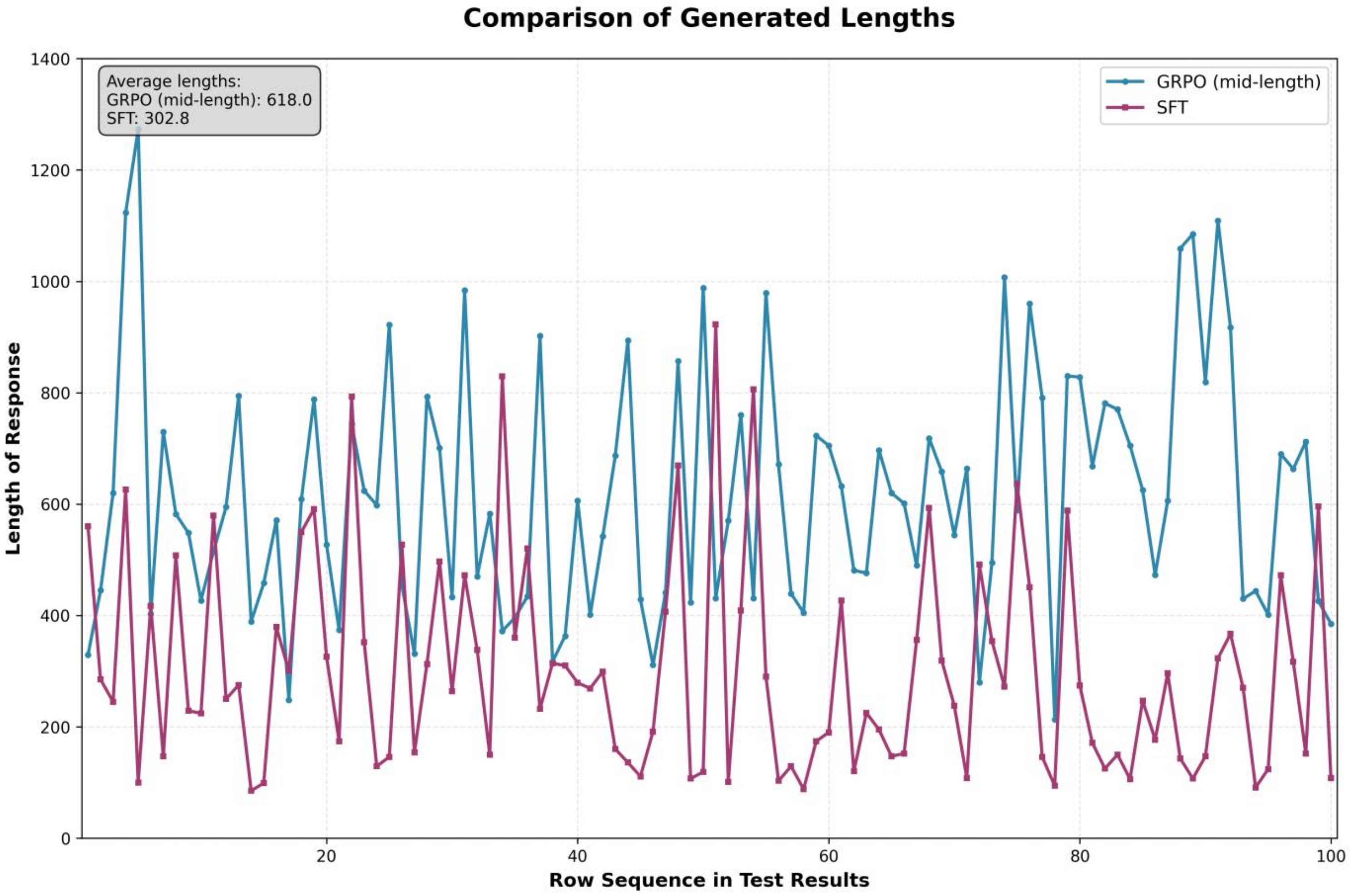


Fig. 5. SFT v. GRPO (mid-length)

- Experiment 4
  - generate more ***new entities*** in patent claims
    - an entity is considered representative of a small idea or a part of the idea
      - e.g., system, agent, contact center, remote party, processor, etc.
    - extracted using Google's gemma3:12b model
    - to approximate the inventive concept in the patent claim
  - Analogy
    - drafting a patent claim is like connecting dots
    - use RL to add more new dots

- reward function
    - check dependency to avoid reward hacking

```
if len(response) > len(claim1):
  score = -2
else:
  set1 = extract_entities(claim1)
  set2 = extract_entities(response)
  dependency = intersection(set1, set2)
  new_ideas = set2 - set1
  if len(dependency) == 0:
    score = -2
  else:
    score = len(new_ideas)
```

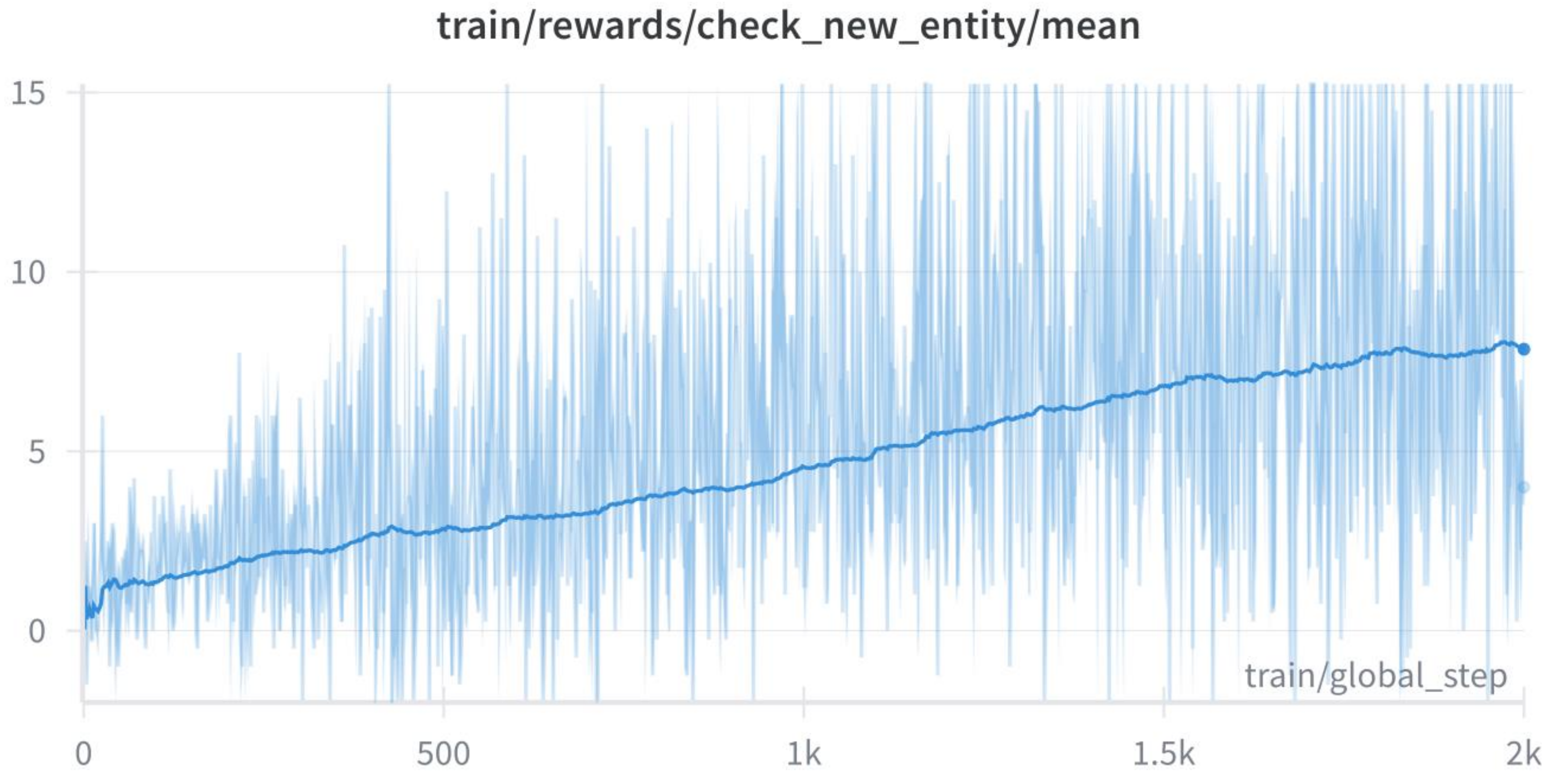
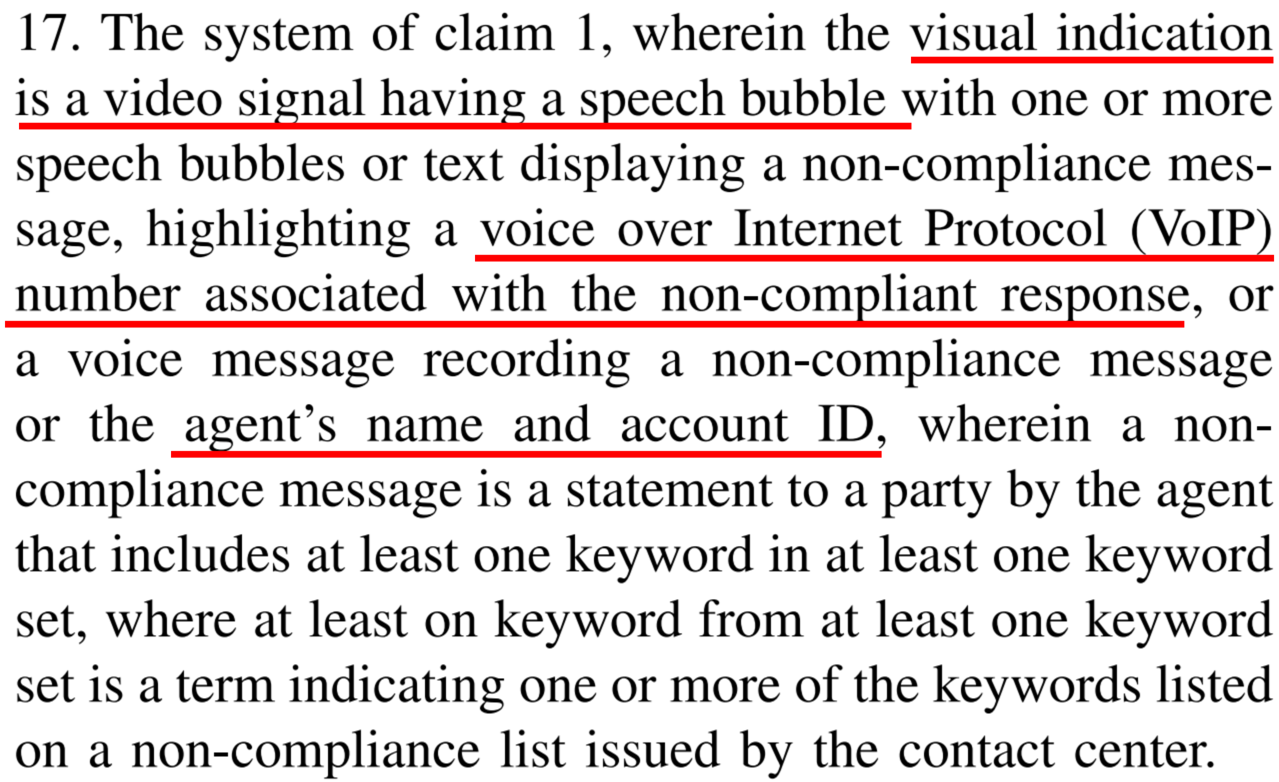


Fig. 6. Reward new entities.



- 17. The system of claim 1, wherein the visual indication is a video signal having a speech bubble with one or more speech bubbles or text displaying a non-compliance message, highlighting a voice over Internet Protocol (VoIP) number associated with the non-compliant response, or a voice message recording a non-compliance message or the agent's name and account ID, wherein a non-compliance message is a statement to a party by the agent that includes at least one keyword in at least one keyword set, where at least on keyword from at least one keyword set is a term indicating one or more of the keywords listed on a non-compliance list issued by the contact center.

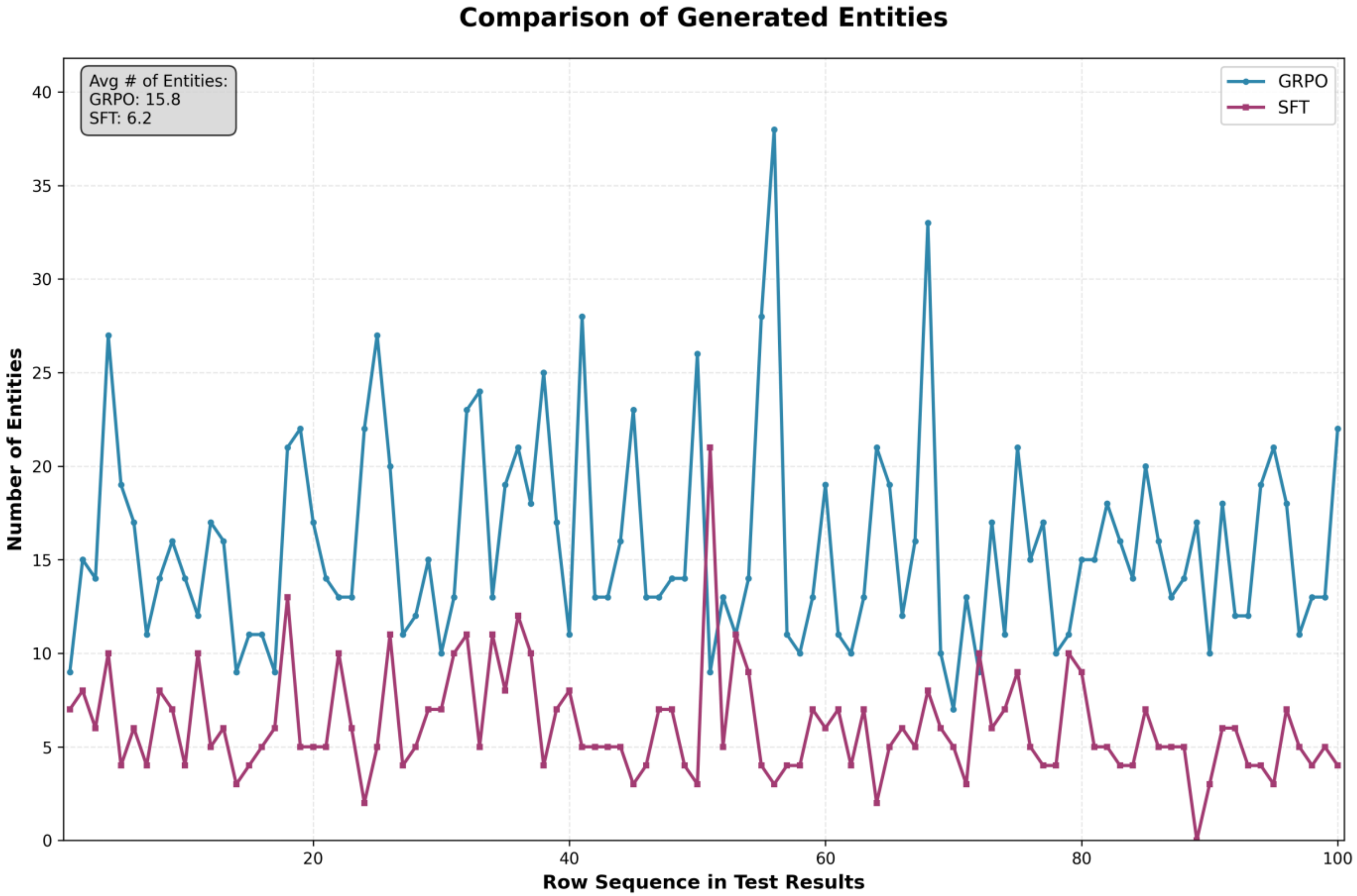
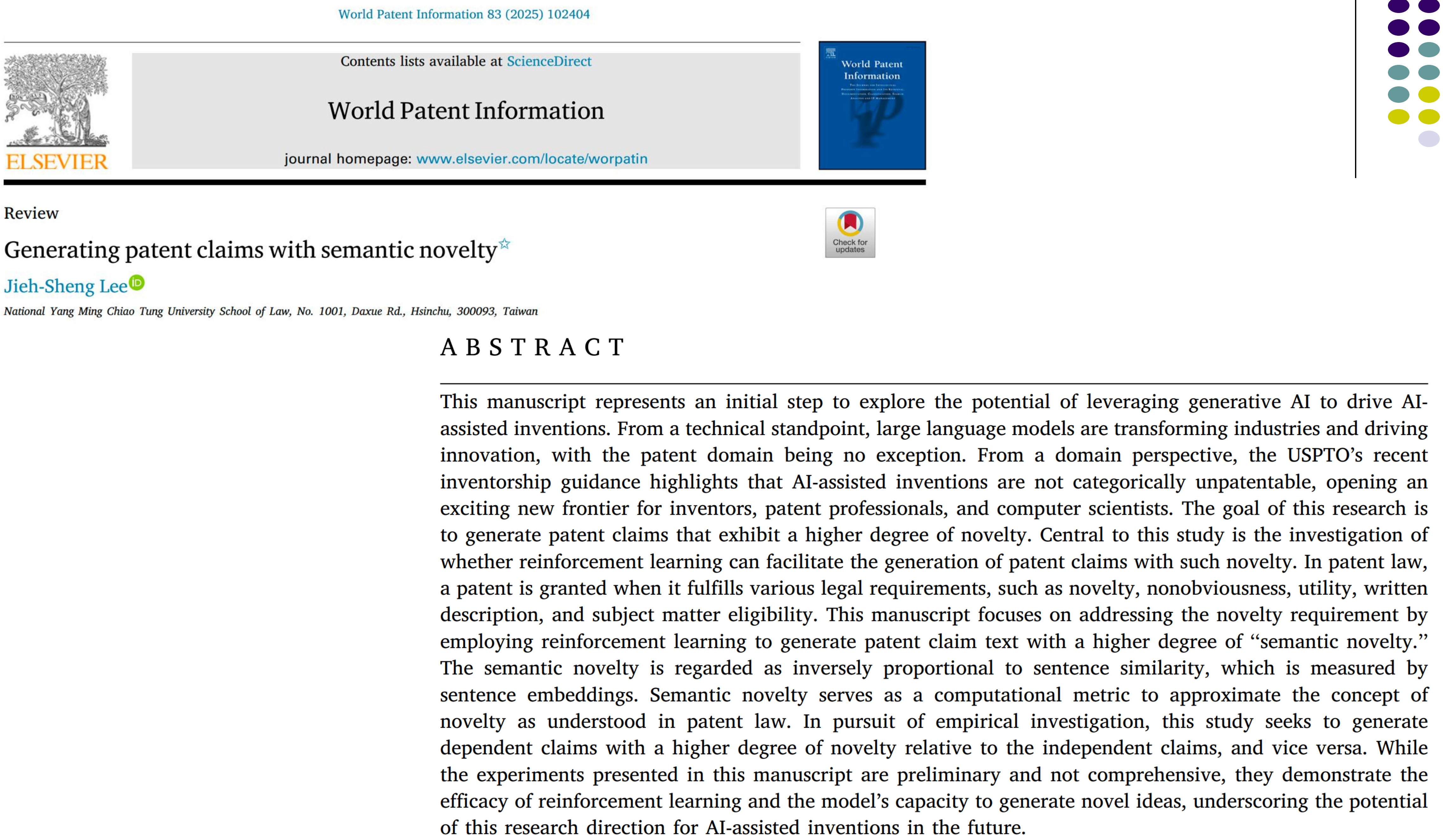


Fig. 7. SFT v. GRPO (entity count)

Conclusion

- Trained GRPO models demonstrate efficacy in
  - controlling claim length across different targets
  - generating more new entities in patent claims
    - a potential application of RL to harness new inventive ideas
- Future Work
  - Qualitative analysis
  - Convert the legal requirements in Patent Law to verifiable rewards
    - e.g., eligibility, novelty, nonobviousness, written description, etc.
  - Use AI to co-invent
    - through Reinforcement Learning with Verifiable Rewards

my related research



World Patent Information 83 (2025) 102404

Contents lists available at ScienceDirect

World Patent Information

journal homepage: www.elsevier.com/locate/worpatin

Review

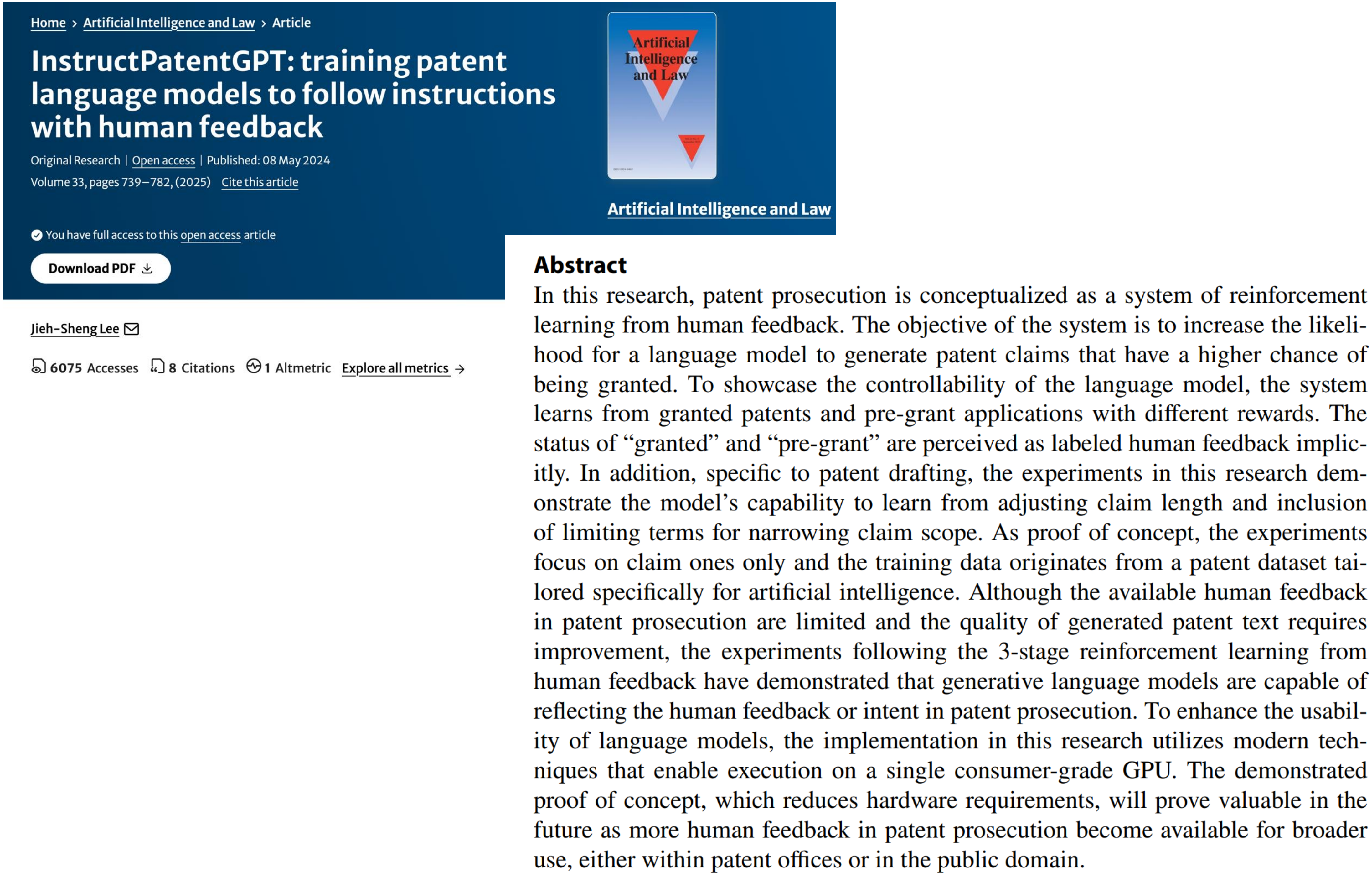
Generating patent claims with semantic novelty☆

Jieh-Sheng Lee

National Yang Ming Chiao Tung University School of Law, No. 1001, Daxue Rd., Hsinchu, 300093, Taiwan

ABSTRACT

This manuscript represents an initial step to explore the potential of leveraging generative AI to drive AI-assisted inventions. From a technical standpoint, large language models are transforming industries and driving innovation, with the patent domain being no exception. From a domain perspective, the USPTO's recent inventorship guidance highlights that AI-assisted inventions are not categorically unpatentable, opening an exciting new frontier for inventors, patent professionals, and computer scientists. The goal of this research is to generate patent claims that exhibit a higher degree of novelty. Central to this study is the investigation of whether reinforcement learning can facilitate the generation of patent claims with such novelty. In patent law, a patent is granted when it fulfills various legal requirements, such as novelty, nonobviousness, utility, written description, and subject matter eligibility. This manuscript focuses on addressing the novelty requirement by employing reinforcement learning to generate patent claim text with a higher degree of "semantic novelty." The semantic novelty is regarded as inversely proportional to sentence similarity, which is measured by sentence embeddings. Semantic novelty serves as a computational metric to approximate the concept of novelty as understood in patent law. In pursuit of empirical investigation, this study seeks to generate dependent claims with a higher degree of novelty relative to the independent claims, and vice versa. While the experiments presented in this manuscript are preliminary and not comprehensive, they demonstrate the efficacy of reinforcement learning and the model's capacity to generate novel ideas, underscoring the potential of this research direction for AI-assisted inventions in the future.



Home › Artificial Intelligence and Law › Article

InstructPatentGPT: training patent language models to follow instructions with human feedback

Original Research | Open access | Published: 08 May 2024

Volume 33, pages 739–782, (2025) Cite this article

Artificial Intelligence and Law

You have full access to this open access article

Jieh-Sheng Lee

6075 Accesses 8 Citations 1 Altmetric Explore all metrics →

Abstract

In this research, patent prosecution is conceptualized as a system of reinforcement learning from human feedback. The objective of the system is to increase the likelihood for a language model to generate patent claims that have a higher chance of being granted. To showcase the controllability of the language model, the system learns from granted patents and pre-grant applications with different rewards. The status of "granted" and "pre-grant" are perceived as labeled human feedback implicitly. In addition, specific to patent drafting, the experiments in this research demonstrate the model's capability to learn from adjusting claim length and inclusion of limiting terms for narrowing claim scope. As proof of concept, the experiments focus on claim ones only and the training data originates from a patent dataset tailored specifically for artificial intelligence. Although the available human feedback in patent prosecution are limited and the quality of generated patent text requires improvement, the experiments following the 3-stage reinforcement learning from human feedback have demonstrated that generative language models are capable of reflecting the human feedback or intent in patent prosecution. To enhance the usability of language models, the implementation in this research utilizes modern techniques that enable execution on a single consumer-grade GPU. The demonstrated proof of concept, which reduces hardware requirements, will prove valuable in the future as more human feedback in patent prosecution become available for broader use, either within patent offices or in the public domain.

Q & A

Thank You