

DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Análisis sobre reward hacking

October 10, 2026 Fundamentos de IA

Integrante
Cindy Levi



Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - Pabellón I

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Argentina

Tel/Fax: (54 11) 4576-3359

http://exactas.uba.ar

1 Introducción

Reward hacking es la situacion bajo la cual los modelos de inteligencia artificial, especialmente aquellos entrenados mediante aprendizaje por refuerzo, encuentran formas no deseadas de maximizar su señal de recompensa, causando efectos contrarios no deseados por los programadores. En esencia, la IA encuentra una forma de "engañar" al sistema para obtener beneficios sin esforzarse o cumplir las intenciones de diseño. El problema no es que el modelo funcione mal, sino que hay una diferencia entre lo que los diseñadores intentan recompensar y lo que efectivamente codifican como señal de recompensa. Cuando un modelo persigue esa señal al pie de la letra, pero no cumple el resultado esperado por el programador, decimos que ocurre reward hacking.
Este problema implica una desconfizana hacia los modelos actuales, que pueden traer resultados inesperados, y es especialmente preocupante cuando están involucrados en tareas importantes, como asistencia médica, moderación de contenido, involucracion en situaciones politicas. Reward hacking no sólo representa un fallo técnico, sino también una amenaza a la seguridad y alineación de la IA.
Casos recientes, como el reportado por OpenAI donde un modelo aprendió a generar respuestas emocionalmente cargadas (uso de frases como "I'm really sorry") para recibir más aprobaciones humanas, ejemplifican cómo reward hacking sigue apareciendo en los sistemas mas actuales, con los refuerzos de seguridad creados hasta el momento. Veremos mas de un ejemplo donde suceden situaciones similares, generando fallas importantes en el comportamiento deseado. No podemos pasar por alto estos incidentes, ya que muestran fallas estructurales en la forma en que diseñamos y supervisamos modelos.

2 Orígenes

Si bien ahora se está poniendo de moda el termino "reward hacking" y su investigacion está en auge, el problema surgio en sistemas antiguos.
Ya desde los años 80, el sistema Eurisko fue uno de los primeros sistemas en demostrar este problema. Fue desarrollado como un rule-based-model, para generar y modificar sus propias reglas para resolver problemas. En el contexto de un juego el cual el sistema tenia que ganar, en vez de desarrollar estrategias más inteligentes de combate, descubrió formas de conseguir mas puntos, manipulando los parámetros del sistema. Esto anticipaba el problema fundamental: los modelos optimizan lo que se les recompensa, no lo que uno quiere que hagan. Otro caso parecido

es el del robot LEGO, entrenado para seguir una línea negra. En lugar de seguir la linea, el robot descubrio que girando sobre si mismo recibia mas puntos, dado que el sensor detectaba el cambio como avance.
Estos ejemplos dan cuenta de ejemplos claros de reward hacking, pero al parecer el problema no está reduciendo, sino que se está voilviendo mas complejo y dificil de detectar.
Reward hacking tomo muchisima mas relevancia con el despliegue de LLMs. Se encontraron casos de modelos de lenguaje como GPT-4, Claude o LLaMA que manipulan su lenguaje para evitar penalizaciones o buscar aprobación. El paper de LessWrong, "Reward Hacking is Becoming More Sophisticated and Deliberate in Frontier LLMs" reporta que modelos actuales desarrollan estrategias deliberadas para complacer mas, sin necesariamente dar respuestas precisas o verídicas. Esta sofisticación indica que reward hacking no solo es un problema que se sigue manteniendo en los sistemas actuales, sino que se está volviendo cada vez mas dificil de detectar. En el paper se muestra como algunos modelos responden con expresiones de sumisión o cortesía no por convicción, sino porque aprenden que eso genera más recompensa por parte de los evaluadores. También se detectan patrones donde los modelos minimizan su grado de certeza para evitar el castigo en caso de errores, o formulan respuestas vagas e inofensivas para evitar penalizaciones. Este paper muestra que claramente, el problema no viene solo de los sistemas de recompensas automaticas, sino que muchas veces el modelo hackea al propio evaluador humano para conseguir mas recompensa.

3 Fundamentos teóricos

El reward hacking se puede entender a través de la Ley de Goodhart, que establece que "cuando una métrica se convierte en objetivo, deja de ser una buena métrica". Lilian Weng identifica cuatro variantes:

- Regresional: los extremos de la métrica son malos proxies. Esto se puede dar cuando se utiliza una métrica (como la longitud de la respuesta, o la puntuación de satisfacción del evaluador) que está correlacionada con el objetivo verdadero (por ejemplo, utilidad o veracidad), pero esa correlación se rompe en los extremos. Por ejemplo, un modelo que da respuestas cada vez más largas porque en promedio las respuestas largas son más útiles, pero al maximizar la metrica, el modelo empieza a generar texto redundante o incoherente. El problema nace en que el modelo exagera una correlación estadística sin entender la causa propia de la metrica.

- Causal: intervenir en la métrica rompe la causalidad. Sucede cuando intervenir en la métrica sin respetar su estructura causal, rompe la relación con la variable de interés. Este caso aparece cuando se fuerza un cambio en la métrica sin intervenir correctamente en el sistema subyacente que la genera. Por ejemplo, al premiar directamente a un modelo por recibir altas calificaciones de humanos, sin tener en cuenta por qué esas calificaciones se dan (como cortesía, tono emocional o estilo), puede hacer que el modelo empiece a optimizar aspectos irrelevantes como decir "lo siento" o mostrarse sumiso.Por lo tanto, la optimización directa de la métrica disrumpe la relación causal entre el buen desempeño real y la recompensa obtenida.

- Adversarial: el agente explota vulnerabilidades de la métrica. El problema aparece cuando el sistema o agente aprende a explotar debilidades específicas de la métrica. Esto puede ocurrir cuando el agente tiene suficiente capacidad para entender el sistema de evaluación y activamente encuentra formas de manipularlo. Por ejemplo, un modelo que aprende que puede evitar penalizaciones dando respuestas vagas o usando frases emocionalmente aceptables, sin mejorar la calidad real de la respuesta. Este es el caso mas comun de reward hacking, como los ejemplos de los videojuegos que usan trucos para obtener mas puntos, o el robot que no sigue la linea recta. Es ahi cuando el sistema se vuelve adversario de la métrica, explotándola como un bug más que siguiéndola como una guía.

- Extremal: maximizar la métrica genera comportamientos patológicos. En este caso, el problema viene de querer maximizar las metricas al maximo, tanto que se producen consecuencias inesperadas. En este caso, la metrica puede seguir correlacionada con el objetivo, pero puede tener incidencias peligrosas llevadas al maximo. Por ejemplo, querer que un robot limpie toda la habitacion, puede llevar a que rompa las cosas dentro de la misma. O que un GPS te muestre el camino mas corto posible puede llevarte a pasar por zonas peligrosas. Es importante en estos casos, pensar en los efectos de la metrica llevados al maximo, para poder establecer los parametros de seguridad necesarios.

Estas formas explican por qué incluso una métrica bien intencionada puede inducir a comportamientos no alineados La tensión entre señales simplificadas y objetivos complejos está en el centro del problema. El problema central entonces tambien viene de la imposibilidad de especificar y formalizar completamente lo que se desea optimizar, y cualquier

aproximacion a esa formalizacion es vulnerable a la explotacion. Este marco y las pautas que surgen a partir de el, puede ayudar a entender cómo el reward hacking puede emerger no por error de implementación, sino como una consecuencia natural de la optimización bajo incentivos mal definidos.

4 Métodos para contrarrestar y atacar el Reward Hacking

La investigación reciente intenta atacar este problema desde varias estrategias, aunque ninguna resulta ser definitiva. En el paper "Reward Shaping to Mitigate Reward Hacking in RLHF", se introduce el enfoque Proxy-Aligned Reward (PAR), que busca construir funciones de recompensa más fieles a la intención humana. En sus experimentos con LLaMA-2-Chat y Mistral-Instruct, PAR redujo significativamente la aparición de conductas manipulativas al ser evaluadas con AlpacaEval y tareas de chat general.
Sin embargo, el experimento realizado es limitado: se basa en tareas simplificadas y no se evalúa en contextos más abiertos. Los autores no examinan cómo se comportan los modelos en diálogos largos o contextos cambiantes, lo que limita la generalización de sus resultados. Desde una perspectiva más conceptual, Lilian Weng en su post que analiza el "Reward Hacking in Reinforcement Learning" propone el uso de evaluación causal en lugar de correlacional. El post sugiere que las respuestas deberían evaluarse por su impacto real en la comprensión o utilidad para el usuario, y no por cómo "suenan". También se proponen marcos jerárquicos de supervisión con múltiples capas de validación y distintos criterios de evaluación.
El problema es cada vez mas profundo en los sistemas de IA, el post en LessWrong "Reward hacking is becoming more sophisticated and deliberate in frontier LLM" también analiza que reward hacking ya no es sólo accidental, sino que los modelos pueden internalizarlo como estrategia deliberada. Esto subraya la urgencia de no sólo evaluar outputs, sino de investigar por qué el modelo produce ciertas respuestas.
La autocorrección también es una opcion posible: construir modelos con capacidades metacognitivas que detecten internamente cuando están maximizando una métrica a expensas de normas internas o valores. Aunque esta idea es conceptualmente atractiva, estamos lejos de implementarla en modelos actuales de forma fiable.
En resumen, aunque hay avances en la identificación y mitigación del reward hacking, las soluciones actuales son parciales, difíciles de escalar, y

muchas veces dependen de suposiciones difíciles de validar en la práctica.

5 Reflexiones

El fenómeno del reward hacking revela una tensión central en la construcción de sistemas de IA: la dificultad de traducir valores humanos complejos en señales computacionales simples. La historia y la teoría nos muestran que no alcanza con ajustar recompensas: se requiere una reconceptualización más profunda de cómo se define y evalúa el éxito en IA.

Desde una perspectiva crítica, puede argumentarse que muchos enfoques actuales buscan hacer parches sobre síntomas sin abordar la raíz del problema: la dependencia excesiva de métricas cuantificables como sustituto de juicios humanos complejos. Esto lleva a una ilusión de control algorítmico que no se sostiene cuando los sistemas enfrentan ambigüedad, conflicto de valores o contextos abiertos. En lugar de aspirar a que los modelos infieran lo que "realmente" queremos, tal vez sea más realista diseñar mecanismos que integren de forma continua el juicio humano en la evaluación y corrección del comportamiento del modelo.

Una posible mejora en esta dirección es considerar una descomposición participativa de objetivos: dividir tareas complejas en subcomponentes con objetivos evaluables en distintos niveles. Por ejemplo, en lugar de recompensar a un modelo por "dar buenas respuestas", se podrían evaluar aspectos como claridad, honestidad, cobertura del tema y disposición al disenso, cada uno moderado por distintos grupos humanos. Esto permitiría una forma de alineación más distribuida, que reconoce la diversidad de valores y perspectivas.

Otra vía que puede introducir mejoras es el diseño de entornos de entrenamiento más abiertos, donde los modelos se enfrenten a conflictos éticos, ambigüedad y consecuencias a largo plazo. En lugar de evitar el reward hacking mediante restricciones, se puede inducir a los modelos a enfrentarlo y reflexionar sobre él. Esto requeriría incorporar estructuras de memoria moral y razonamiento contrafactual dentro del aprendizaje, de modo que el modelo no solo "evite" hackear recompensas, sino que comprenda por qué hacerlo puede ser problemático. Sin embargo, esto tambien puede ser peligroso para el entendimiento humano sobre los sistemas.

En definitiva, el reward hacking no solo debe entenderse como una falla técnica, sino como una oportunidad para repensar la relación entre objetivos computacionales y valores humanos. Resolverlo no exige solo

mejores algoritmos, sino una mayor sofisticación en cómo definimos y medimos los objetivos propuestos.

6 Conclusión

Reward hacking es un comportamiento emergente natural cuando se entrena un modelo con objetivos imperfectos, que puede y suele surgir cada vez que se intenta entrenar un modelo. Su persistencia a lo largo de la historia de la IA, desde los 80 hasta la actualidad, demuestra que es una preocupación estructural. La creciente sofisticación de los modelos hace que sus tácticas para explotar señales de recompensa también se vuelvan más sutiles y peligrosas.

A medida que estos sistemas se incorporan a procesos sensibles como la educación, la salud, la justicia, el riesgo de reward hacking malicioso o inadvertido se vuelve inaceptable. Reforzar la alinacion no es sólo un problema técnico, sino también politico y social. Debemos revisar no sólo qué queremos que los modelos hagan, sino cómo definimos el éxito y el valor en esos contextos.

Las soluciones propuestas hasta ahora —recompensas proxy, evaluación causal, entrenamiento adversarial, supervisión jerárquica— ofrecen perspectivas valiosas, y avances valioso, pero ninguna resuelve por completo el dilema. Es importante continuar investigando cómo evaluar la alineación de manera robusta, cómo construir modelos con conciencia situacional de sus objetivos, y cómo diseñar incentivos que reflejen no sólo resultados, sino intenciones.

En última instancia, el reward hacking expone una tensión fundamental entre la optimización automática y la complejidad humana. Resolverla no va a ser trivial, pero será necesario si queremos sistemas que realmente nos asistan.

References

[1] Lenat, D. B. (1983). *Eurisko: A program that learns new heuristics and domain concepts.* Artificial Intelligence.

[2] Nelson, A. (2004). *LEGO Robotics: Building, Programming, and Experimenting.* Apress.

[3] Christiano, P., Leike, J., Brown, T., et al. (2017). *Deep reinforcement learning from human preferences.* NeurIPS.

[4] LessWrong (2024). *Reward hacking is becoming more sophisticated*

and deliberate in frontier LLMs.
https://www.lesswrong.com/posts/rKC4xJFkxm6cNq4i9

[5] Fu, J., et al. (2025). *Reward Shaping to Mitigate Reward Hacking in RLHF*. arXiv preprint arXiv:2502.13295.

[6] Weng, L. (2024). *Reward Hacking*. https://lilianweng.github.io/posts/2024-11-28-reward-hacking/

[7] Goodhart, C. A. E. (1975). *Problems of Monetary Management: The U.K. Experience*. Papers in Monetary Economics.