

Proyecto de Simulación e IA

Integrantes:

Hivan Cañizares Diaz
Franco Hernández Piloto
Carlos Mauricio Reyes Escudero

Universidad de La Habana
Curso 2024

Abstract

Este proyecto desarrolla una simulación de combates Pokémon que integra inteligencia artificial (IA) para crear entidades Pokémon cada vez más poderosas. El objetivo principal es explorar y verificar estrategias de combate efectivas, con un enfoque particular en la hipótesis de que seleccionar un único ataque altamente poderoso (overpowered) es una estrategia común y eficaz.

Para verificar esta hipótesis, la simulación implementa un sistema de evolución basado en algoritmos genéticos, donde los Pokémon aprenden y adaptan sus estrategias a lo largo de múltiples generaciones de combates. Se analizará la frecuencia de uso de movimientos específicos, la tasa de victorias, y la evolución de las estrategias de combate a lo largo del tiempo.

Este enfoque no solo busca confirmar la efectividad de la estrategia de "ataque único", sino también identificar posibles contramedidas y estrategias alternativas que puedan surgir en respuesta a esta táctica dominante.

1 Introducción

Este proyecto desarrolla una simulación de combates Pokémon utilizando Python, aprovechando bibliotecas como NumPy para cálculos numéricos, Pandas para manejo de datos, y PUREPLES[2] para implementar algoritmos genéticos. La simulación busca obtener ejemplares poderosos de Pokémon y verificar la hipótesis de que una estrategia muy común y efectiva es seleccionar un solo ataque overpowered.

Para evaluar esta hipótesis, implementamos un proceso en dos fases:

1.1 Fase 1: Evolución mediante Algoritmos Genéticos

Utilizamos algoritmos genéticos para evolucionar a los diferentes Pokémon. Este proceso genera múltiples generaciones de Pokémon, seleccionando y reproduciendo los más exitosos.

1.2 Fase 2: Evaluación de Estrategias

Los Pokémon más fuertes obtenidos al final del proceso evolutivo se enfrentan en múltiples combates. Durante estos enfrentamientos, registramos la cantidad y variedad de ataques utilizados por cada Pokémon.

1.3 Análisis Estadístico

Para analizar los resultados y verificar nuestra hipótesis, llevaremos a cabo experimentos y analizaremos los resultados.

Este enfoque nos permitirá cuantificar la tendencia de los Pokémon evolucionados a utilizar un único ataque poderoso en comparación con una estrategia más variada.

2 Detalles de Implementación

2.1 Evolución de Especies Pokémon

Los Pokémon se representan como entidades que evolucionan a lo largo de generaciones. Este proceso implica:

- **Reproducción:** Se utiliza el entrecruzamiento uniforme (uniform crossover) para combinar los atributos de dos Pokémon progenitores.
- **Mutación:** Los movimientos, habilidades y naturalezas de los Pokémon pueden mutar. La tasa de mutación está controlada por un índice de temperatura que disminuye a medida que

aumenta el fitness del Pokémon, siguiendo la fórmula:

$$\text{temperatura} = (1 - \text{fitness})^{\text{COOLER}}$$

donde COOLER es una constante que controla la velocidad de enfriamiento.

2.2 Inteligencia Artificial y ES-HyperNEAT

La IA que guía a cada Pokémon se implementa como una red neuronal evolucionada usando ES-HyperNEAT (Evolvable-Substrate HyperNEAT):

- **Entrada:** Estado del terreno y características de los Pokémon en combate.
- **Salida:** Movimiento a efectuar en el turno actual.
- **Evolución:** La topología de la red evoluciona a lo largo de las generaciones, adaptándose para mejorar la toma de decisiones en combate.

2.3 Simulación de Combates

Para la simulación de combates, nos apoyamos en una implementación existente de mecánicas de batalla Pokémon [1], que proporciona una base sólida y precisa para las interacciones entre Pokémon durante el combate.

2.4 Manejo de Experiencia y Niveles

El sistema de experiencia y niveles se basa en datos actualizados de fuentes especializadas en Pokémon[3][4]. La fórmula para el cálculo de experiencia es:

$$\Delta EXP = \left(\frac{b \times L}{5} \times \frac{1}{s} \times \left(\frac{2L+10}{L+L_p+10} \right)^{2.5} + 1 \right) \times t \times e \times v \times f \times p$$

Figure 1: Fórmula de cálculo de experiencia en Pokémon

2.5 Cálculo de Fitness

El fitness de cada espécimen se calcula considerando múltiples factores:

$$\text{fitness} = p \cdot \left(\frac{\text{moves_used}}{4} \right)^{2.9} + (1 - p) \cdot x$$

donde:

- $p = \frac{3}{\text{age}+2.7}$ es un factor que disminuye con la edad del Pokémon.

- $x = 0.4 \cdot \frac{\text{level}}{100} + 0.6 \cdot \frac{\text{won_battles}}{\text{total_battles}}$
- moves_used es el número de movimientos diferentes utilizados por el Pokémon.

Esta fórmula está diseñada para:

- Fomentar inicialmente la variedad de ataques usados.
- Reducir gradualmente la importancia de la variedad a medida que el Pokémon "envejece".
- Balancear la importancia entre el nivel alcanzado y la tasa de victorias.

2.6 Almacenamiento de Datos

Los Pokémon más poderosos se serializan y guardan usando la biblioteca pickle de Python, permitiendo un análisis estadístico posterior de las estrategias evolutivas más exitosas.

2.7 Test de Hipótesis

Para verificar la hipótesis de que los Pokémon tienden a especializarse en un único ataque poderoso, se implementa el siguiente proceso:

1. Recopilación de datos de uso de ataques para cada Pokémon evolucionado.
2. Cálculo de la distribución acumulativa empírica de uso de ataques.
3. Definición de una distribución teórica de uso equitativo de ataques.
4. Aplicación de la prueba de Kolmogorov-Smirnov de una muestra para comparar las distribuciones empírica y teórica.
5. Análisis de los resultados para determinar si hay una desviación significativa hacia el uso predominante de un solo ataque.

Este enfoque permite cuantificar objetivamente la tendencia de los Pokémon evolucionados a especializarse en estrategias de ataque único o diversificadas.

2.8 Diseño Experimental

Como se había mencionado anteriormente respecto a la varianza en el uso de movimientos, se decidió realizar dos experimentos distintos para evaluar la hipótesis principal:

1. **Experimento sin énfasis en la variedad de movimientos:** En este primer experimento, la función de fitness no considera la variedad de movimientos utilizados por el

Pokémon. La fórmula de fitness se simplifica a:

$$\text{fitness} = 0.4 \cdot \frac{\text{level}}{100} + 0.6 \cdot \frac{\text{won.battles}}{\text{total.battles}}$$

Este enfoque permite que la evolución se centre únicamente en la efectividad de los Pokémon en términos de nivel alcanzado y tasa de victorias.

2. **Experimento con énfasis decreciente en la variedad de movimientos:** En este segundo experimento, se implementa la función de fitness completa descrita anteriormente:

$$\text{fitness} = p \cdot \left(\frac{\text{moves_used}}{4} \right)^{2.9} + (1 - p) \cdot x$$

donde $p = \frac{3}{\text{age}+2.7}$ disminuye con el tiempo, reduciendo gradualmente la importancia de la variedad de movimientos a medida que el Pokémon "envejece" y evoluciona.

Estos dos experimentos nos permitirán comparar cómo la consideración de la variedad de movimientos afecta la evolución de las estrategias de los Pokémon. El primer experimento sirve como línea base, mientras que el segundo nos ayudará a entender si la presión inicial para usar una variedad de movimientos conduce eventualmente a estrategias más especializadas o si mantiene un conjunto más diverso de tácticas.

Los resultados de ambos experimentos se analizarán utilizando la prueba de Kolmogorov-Smirnov descrita anteriormente, lo que nos permitirá cuantificar y comparar la tendencia hacia la especialización en un único ataque poderoso en ambos escenarios.

3 Resultados

stat	exp. 1	exp. 2
dks	0.8200	0.8467
p-val	0.0004	0.0002
entropy	0.3251	0.8412
mean	1.10	1.33
variance	0.09	0.69

Table 1: Resultados de los Experimentos de Uso de Movimientos

Los resultados obtenidos de ambos experimentos proporcionan información valiosa sobre el comportamiento de los Pokémon en combate. A continuación, se detalla el significado de cada estadística y su relevancia en el contexto del estudio:

- D de Kolmogorov-Smirnov: Los valores obtenidos (0.8200 para el Experimento 1 y 0.8467 para el Experimento 2) indican una diferencia significativa entre la distribución observada de uso de movimientos y una distribución uniforme. Esto sugiere que los Pokémon no utilizan sus movimientos de manera equitativa en ambos experimentos.
- p-valor: Los p-valores muy bajos (0.0004 y 0.0002) proporcionan evidencia estadística fuerte para rechazar la hipótesis nula de que los movimientos se utilizan de manera uniforme. Esto indica que los Pokémon tienden a especializarse en ciertos movimientos.
- Entropía de Shannon: La entropía de Shannon aumentó de 0.3251 en el Experimento 1 a 0.8412 en el Experimento 2, lo que sugiere una mayor diversidad en el uso de movimientos en el segundo experimento. Sin embargo, el valor de entropía sigue siendo relativamente bajo, indicando que, en general, los Pokémon aún utilizan un número limitado de movimientos.
- Promedio de movimientos usados: El promedio de movimientos utilizados aumentó de 1.10 a 1.33, lo que sugiere que al introducir un énfasis decreciente en la variedad de movimientos, los Pokémon comenzaron a usar un poco más de movimientos diferentes, aunque todavía se especializan en un solo ataque.
- Varianza de movimientos usados: La varianza también aumentó de 0.09 a 0.69, lo que indica una mayor dispersión en el uso de movimientos. Esto sugiere que algunos Pokémon están utilizando más movimientos que otros, lo que podría ser un resultado de la estrategia de combate más flexible implementada en el segundo experimento.

Estos hallazgos refuerzan la idea de que los Pokémon tienden a especializarse en ciertos movimientos, pero también indican que al permitir una mayor variedad de movimientos, existe la posibilidad de diversificación en sus estrategias de combate.

4 Integración con la API de Google y Modelo Gemini 1.5 Flash

En este proyecto, se utiliza la API de Google para acceder a su modelo de inteligencia artificial Gemini 1.5 Flash. Este modelo es capaz de generar texto de manera coherente y contextual, lo que lo convierte en una herramienta ideal para enriquecer la

experiencia de combate en nuestra simulación de Pokémon.

4.1 Narración de Batallas

La API de Google se utiliza para narrar las batallas en tiempo real, proporcionando comentarios y descripciones detalladas de las acciones de los Pokémon durante el combate. Esto incluye:

- Descripción de movimientos: La API genera descripciones sobre cómo cada Pokémon utiliza sus movimientos, lo que añade un nivel de inmersión a la experiencia del usuario.
- Análisis de estrategias: Al narrar las batallas, el modelo también puede ofrecer análisis sobre las decisiones tomadas por los Pokémon, lo que ayuda a los jugadores a comprender mejor las estrategias de combate.
- Eventos clave: La narración resalta eventos importantes, como cuando un Pokémon lanza un ataque crítico o cuando uno de ellos queda incapacitado, mejorando la dinámica del combate.
- Consejos basados en el estado de la batalla: El modelo proporciona consejos adaptados al estado actual de la batalla, sugiriendo movimientos u estrategias óptimas en función de la salud y las condiciones de los Pokémon involucrados.

4.2 Resumen de Batallas Finalizadas

Al finalizar una batalla entre dos IA, la API de Google también se utiliza para generar un resumen conciso de la batalla. Este resumen incluye:

- Resultados de la batalla: Se presenta quién fue el ganador y quién el perdedor, junto con estadísticas relevantes como el número de movimientos utilizados y la duración de la batalla.
- Estrategias empleadas: El resumen puede incluir un análisis de las estrategias utilizadas por cada Pokémon, destacando los movimientos más efectivos y las decisiones críticas que llevaron a la victoria.
- Comentarios finales: Se generan comentarios finales que pueden ofrecer recomendaciones sobre cómo mejorar las estrategias en futuras batallas, basándose en el desempeño observado.

4.3 Implementación Técnica

Para implementar esta funcionalidad, se realiza una llamada a la API de Google desde el código del proyecto, enviando la información relevante sobre la batalla en curso o finalizada. El modelo Gemini 1.5 Flash procesa esta información y devuelve el texto generado, que luego se presenta al usuario en la interfaz de la simulación.

Este uso de la API de Google no solo mejora la experiencia del usuario al proporcionar una narración dinámica y atractiva, sino que también permite una comprensión más profunda de las mecánicas de combate y las estrategias utilizadas por los Pokémon en la simulación.

5 Fuentes y recursos usados

- The **pokemon-python** battle simulator, which provided a solid foundation for the battle mechanics and interactions between Pokémon from its (GitHub Repository)[1].
- The **PUREPLES** library, which provided the ES-HyperNEAT (Evolvable-Substrate HyperNEAT) implementation used for evolving the neural network topologies (GitHub Repository)[2].
- **Bulbapedia**, the community-driven Pokémon encyclopedia (see website)[3].
- The **Pokémon Database** data (see website)[4].
- The **Google API for devs** (see website)[5].