Aprendizaje en MAS

Pablo Bernabeu Pérez
Liam James Glennie England

Introducción

Introducción

Agente
Sistema informático que es capaz de actuar de forma autónoma en representación de su usuario o propietario.

Sistemas Multiagente
Sistema compuesto por múltiples agentes inteligentes que interactúan entre ellos mediante cooperación y negociación.

Aprendizaje en Sistemas Multiagente
Aplicación de Machine Learning en un sistema multiagente. Conjunto de técnicas y algoritmos usados para entrenar a los agentes para la toma de decisiones. Más difícil que entrenar un único agente ya que el entorno no es estático.

Primeros enfoques al MAL

Propiedades importantes

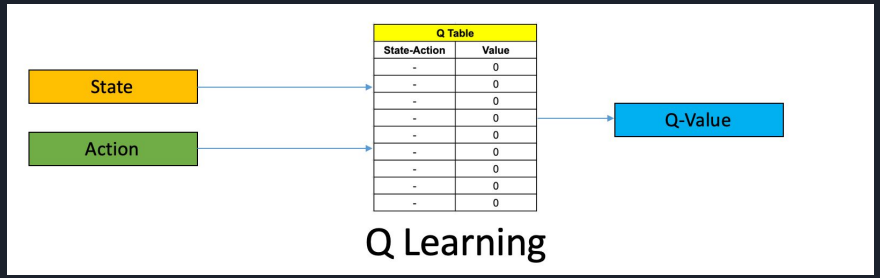
Convergencia
Un agente llegará a converger en una política estacionaria, es decir, una política que decide la misma acción para cada estado independientemente del tiempo.

Racionalidad
Si las políticas de otros agentes convergen en políticas estacionarias, el algoritmo de aprendizaje convergerá en una política de mejor respuesta.

No arrepentimiento
Asegura que la política en uso tiene un rendimiento mayor que la mejor política estática.

Q-Learning

- Inicialmente diseñado para agentes únicos para encontrar políticas óptimas en MDPs
- Es **racional** si los otros agente emplean una estrategia estacionaria
- No es **convergente** porque no utiliza políticas estocásticas
- Base para muchos algoritmos diseñados para entornos multiagente:
  - Minimax-Q
  - Nash-Q
  - Friend-or-Foe Q-Learning

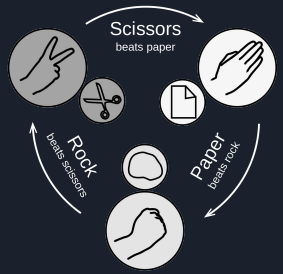


Minimax-Q

- Extensión de Q-Learning
- Utiliza el concepto de equilibrio para estimar el valor de un estado
- Actualiza el valor de los estados con cada transición para aprender el equilibrio Nash del agente
- Es **convergente** porque es un algoritmo independiente de los otros jugadores para aprender la solución de equilibrio
- No es **racional** porque no siempre produce la mejor respuesta

Opponent Modeling

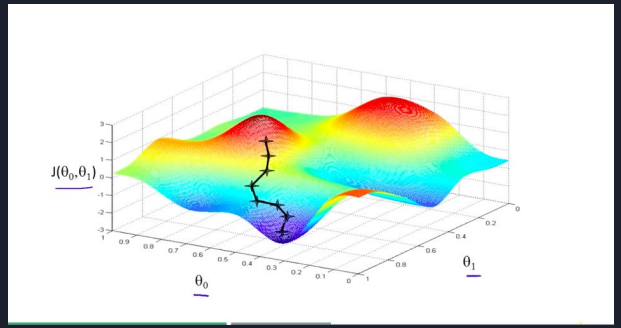
- Basado en aprenderse los modelos explícitos de los otros agentes si estos aplican una política estacionaria
- Es **racional** porque las estimaciones sobre la política del oponente convergen en un política verdadera de mejor respuesta
- No es **convergente** porque solo aplica políticas puras y no puede converger en juegos de equilibrio mixto

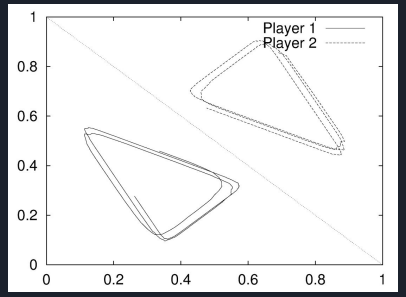


Principio “Win or Learn Fast” (WoLF)

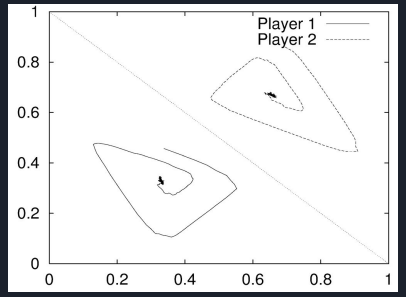
Principio WoLF

Ascenso del Gradiente
Algoritmo iterativo de optimización para encontrar los máximos en una función.





Ritmo de Aprendizaje Constante



Ritmo de Aprendizaje Variable

Ritmo de Aprendizaje Variable
La magnitud de la corrección de los parámetros del modelo depende de si el agente gana o pierde con la estrategia actual.

Principio WoLF

WoLF-PHC

Algoritmo de Q-Learning modificado que implementa el principio WoLF en combinación con la política de escalada.

Es capaz de converger en una política óptima sin sacrificar la propiedad de racionalidad introducida por el algoritmo de Q-Learning.

Noción de victoria en comparación con la política media.

GIGA-WoLF

Algoritmo basado en WoLF-PHC que introduce la propiedad de “no arrepentimiento”

Calcula dos estrategias modificadas en cada iteración mediante el algoritmo GIGA (Generalized Infinitesimal Gradient Ascent)

Reduce los requerimientos estrictos de WoLF-PHC al no necesitar conocer la matriz de recompensas y la distribución de las acciones del oponente

Deep Learning

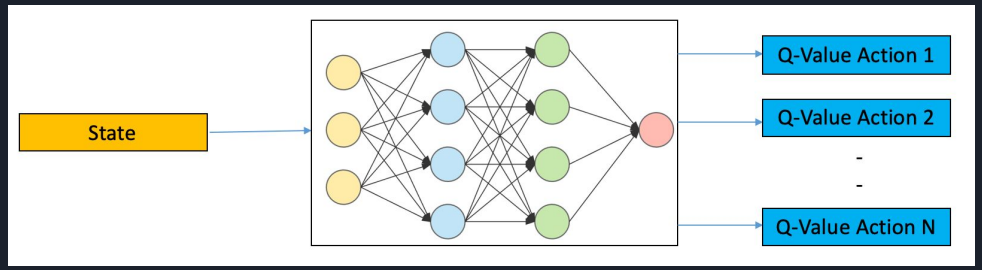
Deep Reinforcement Learning

- Nace como una solución a las limitaciones del aprendizaje reforzado:
  - Se generalizan los métodos entre estados
  - Elimina la necesidad de representar los estados manualmente

- Tiene sus propios problemas rompiendo las condiciones de:
  - **Independencia** porque en RL los datos de entrenamiento tiene una alta correlación entre agente y entorno
  - **Datos idénticamente distribuidos** porque el agente explora el espacio de estados aprendiendo de forma activa y creando una distribución de datos no estacionaria

Deep Q-Networks

- Combinación de Deep Learning y RL enfocada para agentes únicos con el fin de enfrentarse a la maldición de la dimensionalidad
- Produce los Q-values de todas las acciones tomadas desde el estado de entrada
- Red de políticas con entrenamiento continuo para aproximar a una política óptima
- En su introducción en 2015, un agente empleando DQN consiguió jugar competentemente a 49 juegos de Atari



Variantes de DQN

- DDQN (Double DQN)
  - Reduce la sobrestimación de los Q-values
  - Introdujo la repetición prioritaria para romper correlaciones entre muestras
  - Obtuvo un rendimiento 5 veces mayor que DQN en 57 juegos de Atari
- DRQN (Deep Recurrent Q-Networks)
  - Resuelve el problema de la memoria de 4 frames como entrada en la red de políticas
  - Sustituye la capa después de la última capa de convolución de la red política por una memoria recurrente a corto plazo
  - Supera a DQN en 700% de rendimiento en los juegos Double Dunk y Frostbite
  - Vence al jugador medio de Doom
- DARQN (Deep Attention Recurrent Q-Networks)
  - Añade un mecanismo de atención para que la red se concentre en la regiones importantes del juego
  - Alcanza una puntuación 5 veces mayor que DRQN en el juego Seaquest

Federated deep Reinforcement Learning (FedRL)

- Combate las dificultades del aprendizaje en MAS
  - Limitación de datos de entrenamiento
  - Datos privados que no pueden ser compartidos entre agentes
- Aprende un Q-network privado para cada agente compartiendo información limitada
- La información compartida es cifrada
- Existen agentes que no pueden construir buenas políticas de decisión sin la información compartida por otros agentes.
- Todos los agentes se benefician al unirse a la federación para construir las políticas de decisión
- Permite el uso de MAS en entornos donde los datos de cada agente deben ser privados. Por ejemplo, en el mundo de la sanidad los agentes pueden compartir información útil de tratamientos sin pérdida de privacidad de los pacientes

Juegos Estáticos

AlphaZero

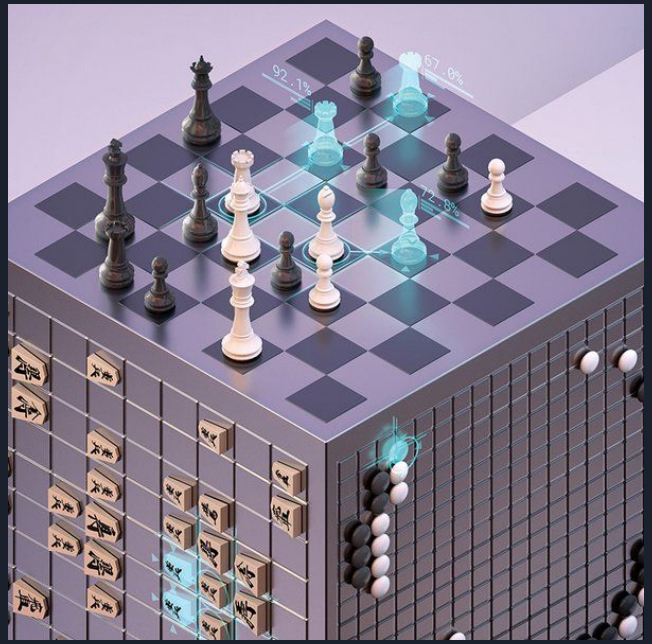
Los juegos de información perfecta forman parte del ámbito más estudiado de la inteligencia artificial. Se han diseñado programas que juegan a un nivel sobrehumano pero estos están muy ajustados a su dominio.

AlphaZero

- Un sistema que mediante selfplay llega a dominar juegos como ajedrez, Go y shogi
- Utiliza una deep neural network y algoritmos con un conocimiento muy básico de las reglas del juego
- Entrenado jugando millones de partidas contr sí mismo ajustando los parámetros de la red neuronal con cada partida

Éxito de AlphaZero

- Ajedrez (Stockfish)
  - Se jugaron 1000 partidas
  - AphaZero ganó 155 y perdió solo 5
- Shogi (Elmo)
  - Venció en 91.2% de la partidas
- Go (AlphaGo Zero)
  - Ganó en el 61% de las partidas



Juegos Dinámicos

DOTA 2

Fundamentos del juego

- 2 equipos de 5 jugadores
- Cada jugador controla un héroe que tiene habilidades únicas y cumple una función para su equipo
- Mismo mapa en todas las partidas formado por 3 líneas
- El objetivo de la partidas es destruir la base enemiga

Dota 2 vs Juegos Estáticos

	Ajedrez	Go	Dota 2
Movimientos por partida	40	150	80000
Acciones válidas por movimiento	35	250	1000
Tipo de información	Completa	Completa	Parcial
Representación del espacio de observación	70	400	20000

Human View

AI View

```
 3.006  -1.386  -0.4695   0.883  1 0.84
-0.3154 -0.5425    -0.5   0.866  0 0.82
  3.11   -1.36  -0.9336  0.3584  1 0.78
-2.324   2.863   0.9746   0.225  0 0.86
 3.037  -1.361  -0.7773  0.6294  1 0.82
-1.387   2.951   0.988   0.1565  0 0.74
 3.023  -0.9395 0.05234 -0.9985  0 0.66
 2.951  -0.5747 0.01746       1  0 0.72
 2.963  -1.303   0.3906  0.9204  0 0.68
 2.834  -3.164 0.01746      -1  0 0.68
 3.127  -1.368  0.6562   0.755  1 0.55
 3.088  -1.366  0.4695   0.883  0 0.55
 2.984  -1.398  -0.225   0.9746 1 0.55
 3.037  -1.391   0.788   0.6157 0 0.55
 3.076  -1.438   0.883   0.4695 0 0.55
-2.412   2.846   0.996 0.08716 1  0.3
```

OpenAI Five

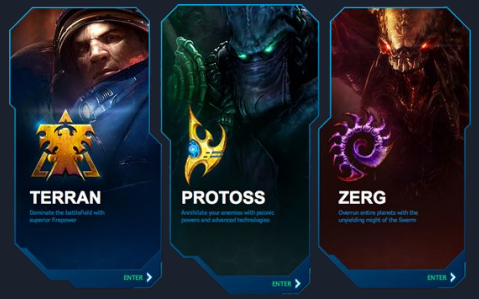
OpenAI

Entrenamiento

- Autoaprendizaje
- Algoritmos de Reinforced Learning (Proximal Policy Optimization)
- 180 años de entrenamiento diario por héroe (900 años diarios en total)

Evolución

- 2017: Agente 1vs1 capaz de derrotar a los mejores jugadores del mundo
- 2018: Primer modelo de MAS entrenado con una versión del videojuego con restricciones
- 2019: OpenAI Five derrota al campeón del mundo de Dota 2



- Cuenta con una jugabilidad rica de múltiples capas diseñada para desafiar el intelecto humano
- El formato más común es el torneo 1v1 donde cada jugador
  - Elige una entre las tres razas alienígenas
  - Empieza con trabajadores que recogen recursos básicos para construir estructuras y crear tecnologías que ayudarán al jugador a vencer el oponente
  - Debe equilibrar la gestión a gran escala de su economía (macros) y el control a bajo nivel de sus unidades individuales (micro)
- Existen retos relacionados con la inteligencia artificial
  - Teoría de juegos
  - Información imperfecta
  - Planificación a largo plazo
  - Tiempo real
  - Gran espacio de acciones

AlphaStar

AlphaStar es una inteligencia artificial que mediante deep neural networks, aprendizaje supervisado y multiagente reforzado consiguió derrotar a un jugador profesional en Starcraft II.

- Su comportamiento es generado por una deep neural network que recibe como entrada la interfaz del juego
- Aprendió las estrategias básicas por medio de la imitación de partidas humanas. Con este entrenamiento era capaz de derrotar al 95% de sus oponentes de nivel oro
- Se creó una liga de agentes donde se jugaban entre ellos donde cada agente aprende y descubre nuevas estrategias
- Tras 14 días de liga, AlphaStar se enfrentó a dos jugadores profesionales y ganó las dos partidas 5 a 0
- Descubrió estrategias nuevas que una enorme comunidad no había descubierto en 20 años

Aplicaciones en la práctica

Control de Semáforos

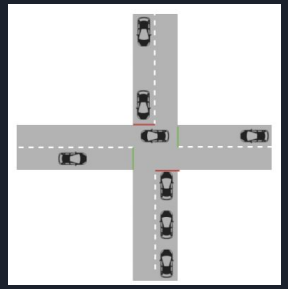
Buenos resultados a pequeña escala pero problemas de escalabilidad, solucionables con Deep Reinforcement Learning

Modelo de agentes heterogéneos con Dueling Double Deep Q-Networks que representa el estado de un cruce con dos matrices de 64x64:

- Matriz de posiciones
- Matriz de velocidades

Función de recompensas: Inversa del tiempo de espera de los coches

Estado del cruce



Matriz de posiciones

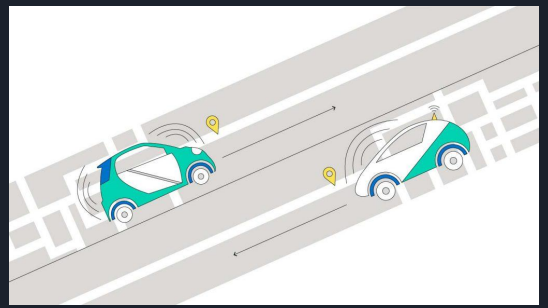
$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.4 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

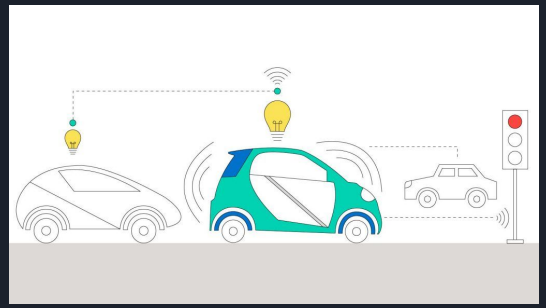
Matriz de velocidades

Conducción Autónoma

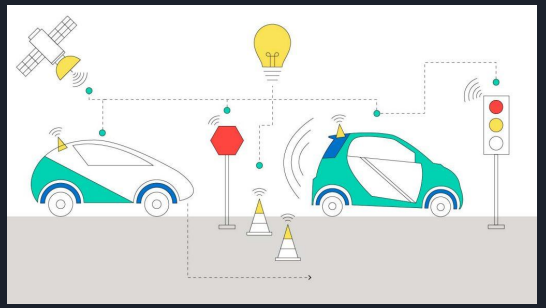
Tesla
Conocimiento en tiempo real mediante análisis y reconocimiento de los alrededores



Volkswagen
Carreteras inteligentes y comunicación entre vehículos (V2V)



Lidar
Mapas 3D de alta resolución captados previamente. Usado por Mercedes Benz, General Motors y Ford



ZeroEnergy Communites

- Los nZECs son conjunto de edificios cuya energía neta es casi 0
- Modelado como un entorno multiagente donde cada edificio representa un agente.
- Cada agente aprende una política de actuación óptima y realiza transacciones de energía
- Existen dos componentes principales:
  - El agente DRL
  - CMS (Community Monitoring Service)
- Mejora de 40 kWh en energía neta en invierno y una mejora de 60 kWh en verano

Conclusión

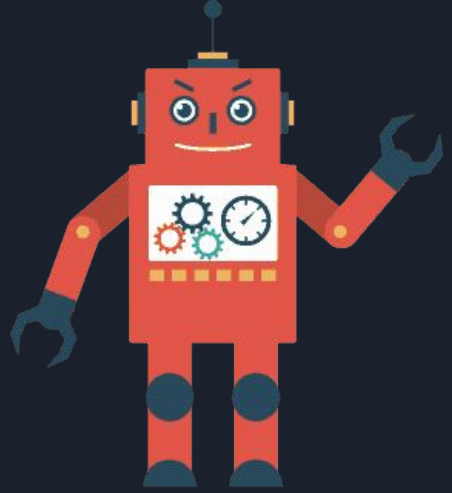
Conclusión

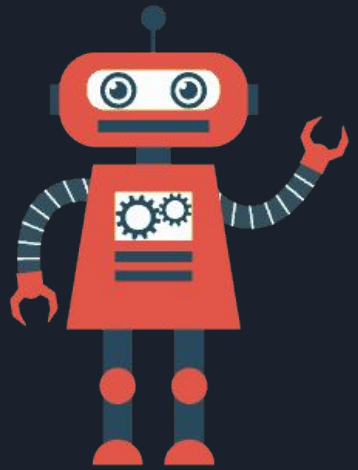
El aprendizaje en sistemas multiagente es un campo de investigación muy útil y interesante. Ofrece herramientas únicas y potentes para la representación y resolución de problemas.

La implementación del Deep Reinforcement Learning es representa el mayor avance e innovación dentro del campo. Abre puertas para afrontar nuevos retos que antes eran imposibles dada la cantidad de datos e información requerida para entrenar a los agentes.

Aplicaciones en multitud de áreas:

- Transporte
- Logística y gestión de recursos
- Comunicación
- Simulación





FIN