Hierarchical Reinforcement Learning: Maze with Tasks

Rubén Cid Costa¹, Aimar Nicuera Usandizaga², Daniel Obreo Sanz³

^{1, 2, 3}Universidad Carlos III de Madrid ¹100538592@alumnos.uc3m.es, ²100538592@alumnos.uc3m.es, ³100538592@alumnos.uc3m.es

Abstract

La realización de objetivos secuenciales en un entorno es una tarea compleja de modelar y aprender. Para poder representar estos escenarios, una de las técnicas más usadas es el Aprendizaje por Refuerzo Jerárquico (*Hierarchical Reinforcement Learning (HIL)*). En este contexto, este trabajo se enfoca en Feudal Learning, una variante de IRL que organiza las tareas en una estructura jerárquica de niveles de abstracción. Este documento detallará las bases teóricas y su aplicación sobre un ambiente de desarrollo con tareas de navegación y obtención de subobjetivos.

Introducción

En muchos dominios, como la robótica o sistemas autónomos, las tareas implican la realización de objetivos secuenciales en entornos complejos. La capacidad de modelar y aprender estos escenarios es un desafío crucial para la inteligencia artificial.

El aprendizaje por refuerzo jerárquico (HIL) se presenta como un enfoque para la resolución de estos problemas. Mientras que otras técnicas previas enfrentan dificultades para escalar con el número de tareas y su complejidad, el Aprendizaje por Refuerzo Jerárquico (HIL) organiza el proceso en niveles de abstracción. Dentro de este marco, el Aprendizaje Feudal se presenta como un enfoque que permite modelar las tareas mediante un jerarquía de abstracción.

Este trabajo se centra en el estudio de Feudal Learning, una variante de HIL. Se presentan las bases teóricas de esta técnica como diferentes algoritmos o métodos de aprendizaje que se han desarrollado en este campo. También, se mostrará su aplicación sobre un entorno simulado diseñado para tareas de navegación en laberintos y obtención de subobjetivos.

Marco Teórico y Estado del Arte

De manera general, el aprendizaje jerárquico

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Sistemas de Control Feudal FeUdal Networks: FuN

Las redes feudales se basan en la arquitectura de aprendiaje por refuerzo feudal, una arquitectura del aprendizaje por refuerzo jerárquico. Esta arquitectura emplea un sistema de control, conocido como "manager", que asigna tareas a un subsistema conocido como "worker" que debe aprender a ejecutarlas de manera óptima.

La arquitectura del sistema se muestra en la imagen siguiente (Science 2024).

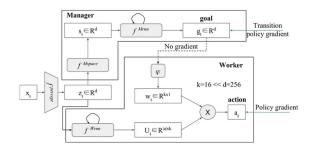


Figure 1: Arquitectura de una red feudal.

La entrada de esta red es procesada por una capa de percepción, que emplea capas convolucionales para extraer características de la imagen de entrada. A continuación, estas características son procesadas tanto por el worker como por el manager, cada uno de manera distinta; el manager extrae objetivos y el worker aprende a alcanzar esos objetivos.

El objetivo principal del manager es generar metas que el worker debe cumplir. Recibe la percepción del entorno, proporcionada por el módulo de percepción, ese estado es procesado por una red recurrente LSTM para mantener un estado interno y poder capturar información relevante en horizontes temporales largos. El manager emplea esta información para predecir un objetivo direccional en el espacio latente, este objetivo es un vector unitario, lo que asegura que el worker se enfoque en la dirección y no en la posición absoluta.

Para entrenar el manager, se emplea la recompensa obtenida, y emplea la similitud coseno entre la dirección en la que se movió el worker y la compara con el objetivo establecido, empleando la similitud coseno como función de pérdida. Esta pérdida incentiva al Manager a emitir objetivos que maximicen el progreso hacia estados ventajosos.

El vector de objetivos se envía al worker sin propagar gradientes, esto garantiza que los objetivos mantengan un significado semántico independiente, en lugar de ser simples variables latentes optimizadas de manera conjunta.

En el caso del worker, también se emplea una red LSTM para mantener un estado interno y poder capturar información relevante, pero en este caso, el worker recibe tanto la percepción del entorno como el objetivo del manager. El worker emplea esta información para predecir la acción que debe realizar para alcanzar el objetivo. La acción se predice en el espacio de acciones, y se emplea

Definición Estado del arte

Evaluación práctica

 $Q_L earning v2$

Debido a la disipación de las paredes que se experimentaba al realizar las abstracciones del mapa de estados a medida que se disminuía el nivel del gerente, el aprendizaje resultaba altamente perjudicado llegando trasmitirse ordenes imposibles de realizar debida a la naturaleza cerrada del mapa del problema.

Para enfrentar dicha adversidad se probó a disminuir el número de gerentes/subgerentes, limitándose al uso del gerente de menor nivel de abstracción; aquel que veía el mapa de estados en la escala original. Por lo tanto, se realizó una disminución de agentes para quedarse con únicamente 2: un gerente encargado de determinar la siguiente zona destino y otro agente encargado en navegar hasta dicha posición de interés. Dicha estructura contaba con las mismas limitaciones y políticas de recompensa definidas para el proceso anterior.

Para facilitar el arranque y el uso del mánager desde el principio, se facilitaba al mánager una pista de la localización de las monedas que debían ser recogidas para que guiara al trabajador (o worker) hacía dichas zonas sin necesidad de un proceso inicial donde las directrices del worker podían ser "ignoradas" debida a su aleatoriedad.

El trabajador a su vez realizaba un proceso de exploración que se veía recompensado cuando alcanzaba el punto objetivo determinado por el mánager. Mediante estas exploraciones, iba creando sus caminos o matrices Q que iba actualizando y afinando con el fin de encontrar el camino óptimo a partir de las recompensas obtenidas por llegar a los destinos asignados por el manager.

En cuanto al proceso de aprendizaje, se incluyeron diferentes implementaciones para ver la influencia que estas variaciones o perturbaciones tenían en el proceso de aprendizaje del super-agente. Las perturbaciones más comunes consistían en limitar el conocimiento que tenía el teacher (dándole solo el conocimiento de la solución empezando desde las coordenadas (1,1) o el conocimiento completo) o variando la posición de comienzo del agente.

En cuento a los resultados se observó como sin ayudas externas el proceso de aprendizaje era más lento al tener que explorar muchos estados y no contar con una ayuda externa en caso de caer en bucles debido a los estrechos caminos que

componían el mapa. Cuando se le introducía la ayuda externa, teacher, dicho proceso de aprendizaje se veía acortado siendo capaz de definir la ruta optima con mayor rapidez. Sin embargo, la influencia de un teacher con conocimiento de un solo camino ante el que siempre tenía la solución optima suponía una gran diferencia también en los momentos en los que se partía de zonas inexploradas. Esto se debía a que el agente tenía que lograr llegar hasta le punto más cercano de conocimiento del teacher para poder obtener su ayuda, lo que lo ponía en similar situación que cuando no contaba con ayuda externa.

En conclusión, esta nueva estructura feudal, de un noble (mánager) y un subordinado (worker), supuso una mejora en comparación con el nivel anterior debida a la simplicidad y mayor énfasis en las limitaciones del mapa de estados que permitía definir. Mediante los experimentos que se realizaron se observó no solo una mejora en el proceso de aprendizaje si no que una capacidad de llegar a soluciones en mapas de 20x20 tras 9000 iteraciones en caso de no incluir la ayuda del teacher o 1000 iteraciones cuando se le facilitaba un teacher para poder preguntar (limitando el número de consultas a realizar en 3xP; siendo P los pasos del camino más corto para solucionar el problema).

Conclusiones

References

Science, T. D. 2024. Hierarchical Reinforcement Learning: Feudal Networks.