

تمرین دوم

با مسائل غیرقطعی چگونه رفتار میکنیم؟

راه حل مسائل غیر قطعی در هوش مصنوعی مرتبط با مدیریت و تصمیمگیری در شرایطی که دارای عدم قطعیت هستند میباشد. برای حل این گونه مسائل، میتوان از رویکردها و تکنیکهای زیر استفاده کرد:

. احتمالات و آمار: استفاده از مفاهیم احتمالات و آمار برای مدلسازی و پیشبینی وقوع رویدادها در شرایط عدم قطعیت.

. مدلسازی بیزی: استفاده از مدل‌های بیزی برای نمایش علاقه‌مندیها و توزیعهای احتمالی در مسائل غیر قطعی.

. تئوری تصمیمگیری: اعمال تکنیکهای تصمیمگیری چون مدل‌های مارکوف تصمیمگیری (MDP) و فرآیندهای تصمیمگیری مارکوف برای تعیین تصمیم‌های بهینه در شرایطی که دارای عدم قطعیت هستند.

. اطلاعات فازی: استفاده از اطلاعات فازی برای مدلسازی عدم قطعیت و عدم دقت در داده‌ها و تصمیم‌گیریها.

تمرین سوم

کد ۸ وزیر رو پیدا کنید و درمورد آن بررسی کنید؟

```
:def is_safe(board, row, col, n)
```

قرار داد یا خیر (row, col) چک کردن آیا می توان وزیری را در سلول #

```
چک کردن ردیف افقی (سمت جب) #
```

```
:for i in range(col)
```

```
:if board[row][i] == 1
```

```
return False
```

```
چی کردن قطر بالا به چپ #
```

```
:for i, j in zip(range(row, -1, -1), range(col, -1, -1))
```

```
:if board[i][j] == 1
```

```
return False
```

```
چی کردن قطر پایین به چپ #
```

```
:for i, j in zip(range(row, n, 1), range(col, -1, -1))
```

```
:if board[i][j] == 1
```

```
return False
```

```
return True
```

تمرین 4

به نظر شما پیچیدگی زمانی دستو پاگیر تره یا پیچیدگی حافظه؟

بستگی به نوع مسئله و الگوریتمی که برای حل آن استفاده می شود دارد که پیچیدگی حافظه یا زمانی کدامیک بیشتر است. در برخی موارد، مسائلی وجود دارند که پیچیدگی حافظه آن ها بسیار بالاست و نیاز به استفاده از منابع حافظه بالا دارند. برای مثال. الگوریتم هایی که برای پردازش تصویر و صدا استفاده می شوند، به دلیل بزرگی حجم داده های ورودی نیاز به استفاده از حافظه بالا دارند

در مقابل، در بسیاری از مسائل. پیچیدگی زمانی بیشتر از پیچیدگی حافظه است. به عنوان مثال، الگوریتم هایی که برا مرتب سازی اعداد استفاده می شوند، نیاز به حافظه کمتری دارند ولی زمان بیشتری برای اجرای آن ها لازم است

بنابراین. برای انتخاب بهترین الگوریتم برای حل یگ مسئله، باید به دو پیچیدگی حافظه و زمانی توجه کرد و الگوریتمی را انتخاب کرد که برای آن مسئله پیچیدگی کمتری داشته باشد

تحلیل WUMPUS:

Wumpus يك بازی کامپیوتری است که در آن بازیکنان باید در يك شبکه از اتاق ها و تونل ها حرکت کنند و به

دنبال هیولای وامپوس بگردند. هدف اصلی این بازی یافتن کنجینه و خروج از محل بازی است، اما بازیکنان باید

همچنین از مخاطرات مختلفی مانند چاه ها، موش های سمی و هیولای وامپوس بپرهیزند.

بازی Wumpus يك بازی استراتژيك است که نیاز به تصمیم گیری هوشمندانه برای حرکت در محیط دارد.

بازیکنان باید از شاخص های مختلف محیط استفاده کنند تا مکان های مختلف را کشف کرده و خطرات را

پیش بینی کنند. همچنین. آنها باید به طور منظم از تجهیزات مختلف مانند فانوس، شمع و نقشه استفاده

کنند تا بهترین راه حل برای پیدا کردن کنجینه و خروج از محل بازی را پیدا کنند.

با توجه به محیط پرخطر و موانع مختلف، بازی Wumpus چالش های زیادی را برای بازیکنان ارائه می دهد. این

بازی نه تنها نیاز به استراتژی و تصمیم گیری هوشمندانه دارد. بلکه همچنین مهارت های حل مسئله و تجزیه

و تحلیل را نیز تقویت می کند.

مشابه دنیای مکش. دنیای Wumpus شبکهای از مربع است که توسط دیوارهایی احاطه شده‌اند. که هر

مربع مي تواند شامل عامل ها و اشیاء باشد. وظیفه عامل یافتن طلا و بازگشتن به نقطه شروع و بالارفتن از

غار است. براي مشخص نمودن وظیفه عامل، ادراکات، عملیات و اهداف آن را باید مشخص کنیم. در دنيای

Wumpus. اینها به صورت زیر هستند:

از مربعي که شامل Wumpus است و مربع هاي مجاور (نه قطري) عامل بوي بدی را دریافت میکند.

در مربعهایی که مستقیماً مجاور با چاله ها هستند. عامل نسیمی را دریافت می کند.

در مربعي که طلا وجود دارد. عامل یک درخششي را درک میکند.