



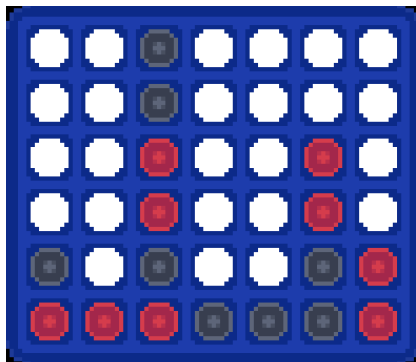
فاز سوم پروژه مبانی و کاربردهای هوش مصنوعی

۱- مقدمه

رقابت یکی از ویژگی‌های مهم خیلی از محیط‌های چندعاملی است. در این مرحله از پروژه هدف اصلی آشنایی و بررسی محیط‌های رقابتی بوده و باید عامل هوشمندی توسعه داده شود که بتواند در محیط یک بازی دونفره، عامل مقابل (حریف) را شکست دهد.

برای این مرحله از پروژه کتابخانه PettingZoo انتخاب شده است. این کتابخانه مخصوص محیط‌های چندعاملی بوده و پیشنهاد می‌گردد برای آشنایی با آن یکبار به توضیحات قرار داده شده در این کتابخانه رجوع کنید. در این فاز نیز همانند فاز گذشته باید قسمت، TODO کد قرار داده شده را تکمیل نمایید.

۲- توضیح پروژه



محیط استفاده شده در این فاز، محیط بازی [Connected four](#) است. در این محیط، صفحه بازی به صورت یک صفحه 6×7 بوده و هر بازیکنی که بتواند چهار مهره خود را به صورت پشت سرهم در حالت عمودی، افقی یا قطری بگذارد برنده می‌شود. شما می‌تواند به صورت آنلاین در [این جا](#) این بازی را تمرین کرده و با شرایط آن بیشتر آشنا شوید.

موجودیت‌های محیط:

این محیط احتمالا ساده‌تر از چیزی است که فکر می‌کنید. در این محیط شما صرفا با دو موجودیت خانه خالی و خانه پر روبه‌رو هستید و باید مهره‌های خود را در یکی از ستون‌های بازی بیاندازید و دقیقا مهره شما در اولین خانه خالی آن ستون قرار می‌گیرد.

فضای کنش در محیط:

کنش‌های ممکن در این محیط با یک عدد بین صفر تا شش تعیین می‌شود که هر کدام از اعداد بیانگر شماره ستون برای انداختن مهره در آن است. **همه کنش‌ها در این محیط قطعی هستند.**

در این محیط ما یک دیکشنری از مشاهدات خود داریم که شامل دو کلید `action mask` و `observation` می‌باشد. ما در این بخش بیشتر با قسمت `observation` سر و کار داریم. در این حالت ما یک آرایه سه بعدی به ابعاد (۶,۷,۲) داریم به این معنی که انگار دو صفحه بازی وجود دارد که هر صفحه بازی برای یکی از عامل‌ها است. مقادیر موجود در درایه‌های این آرایه می‌توانند صفر یا یک باشند. در صورتی که مهره عامل در یک درایه باشد، مقدار آن یک و در غیر این صورت صفر خواهد بود.

کلید بعدی که `Action mask` است نشان‌دهنده ستون‌هایی است که می‌توانیم مهره‌های خود را در آن‌ها بیاندازیم. این کلید یک آرایه هفت‌تایی بوده و هر کدام از اندیس‌ها نمایان‌گر یک ستون از صفحه بازی است و در صورتی که برابر یک باشد، عامل حق قرار دادن مهره خود را در آن ستون دارد.

شرایط اتمام بازی در محیط:

بازی در دو حالت ممکن است به اتمام برسد:

- ۱- در صورتی که عامل بتواند به هدف نهایی خود برسد و چهار مهره خود را به صورت افقی، عمودی یا قطری در کنار هم قرار دهد برنده می‌شود.
- ۲- در صورتی که هیچ عاملی نتواند در بیشینه کنش‌های ممکن بازی (در این محیط ۲۰۰ حرکت) برنده شود.

امتیازدهی در محیط:

امتیازدهی در این محیط به صورت زیر است:

(۱) ۱- به ازای پیروزی عامل حریف

(۲) ۰ (صفر) به ازای تساوی

(۳) ۱+ به ازای پیروزی عامل شما

۳- نکات پیاده‌سازی

در پیاده‌سازی چند تابع مهم وجود دارد که بهتر است با آن‌ها آشنا شوید. با توجه به اینکه یکی از اهداف مراحل پروژه آشنایی با محیط‌های پیاده‌سازی مخلف می‌باشد، در این محیط برخی از توابع با محیط‌های قبلی متفاوت است.

(۱) در این فاز برای تعامل با محیط همچنان از تابع `step` استفاده می‌شود اما با این تفاوت که دیگر این تابع خروجی‌ای به ما بر نمی‌گرداند برای مشاهده وضعیت جدید عامل، پاداش محیط و همین‌طور بررسی شرایط اتمام بازی باید از تابع دیگری به نام `last` استفاده کنیم. این تابع همان‌طور که در کد هم مشخص است، مربوط به برگرداندن پاسخ‌های محیط می‌باشد و به نوعی وظیفه بازگرداندن وضعیت محیط، پاداش و سایر مواردی که در فازهای گذشته دیدیم را دارد.

(۲) نکته دیگری که شاید عجیب بنظر برسد این است که ما ابتدا تابع `last` را استفاده کرده و پس از آن از تابع `step` استفاده می‌نماییم. به دلیل دو عاملی بودن محیط ما نمی‌توانیم همانند فازهای قبلی در صورتی که عامل نتواند حرکت مجازی انجام دهد از بازی خارج شویم زیرا عامل دیگر شاید بتواند حرکت صحیحی داشته باشد. برای همین ما ابتدا از تابع `last` استفاده کرده و این تابع به ما کمک می‌کند تا آخرین کنش عامل را بر روی محیط اعمال و از تاثیراتش بر روی محیط آگاه شویم.

(۳) کتابخانه استفاده شده در این فاز از پروژه کتابخانه `PettingZoo` می‌باشد و در صورتی که آن را نصب ندارید می‌توانید از دستور زیر استفاده نمایید.

```
Pip install 'pettingzoo[classic']
```

۴- نکات تحویل پروژه

(۱) با توجه به اهمیت یادگیری مفاهیم درس حتما باید برنامه خود را از ابتدا پیاده‌سازی نمایید. هر تشابه موجود با کدهای موجود در اینترنت موجب گرفتن نمره صفر خواهد شد. در صورتی که از مقاله، سایت یا کد گیت‌هایی الهام گرفته‌اید حتما باید در بخش منابع مستندات خود آن‌ها را وارد نمایید.

(۲) در صورتی که به مشکلی در پیاده‌سازی برنامه برخوردید حتما با دستیار پروژه درس درمیان گذاشته و یا در سایت کوئرا مشکل خود را اعلام نمایید. پروژه خود را حتما تا زمان مقرر در سایت کوئرا اپلود کنید.

(۳) گروه‌بندی دانشجویان در انجام این مرحله از پروژه همانند مرحله قبلی است و امکان تغییر گروه‌ها وجود ندارد.

۴) فایل ارسالی شما باید شامل فایل پیاده‌سازی و مستندات پروژه باشد. در مستندات پروژه باید راجع به استراتژی‌های پیاده‌سازی و همینطور الگوریتم‌ها و روش‌هایی که اشاره شده، صحبت شده باشد. **در مستند ارسالی خود حتما به الگوریتم انتخابی خود و دلیل آن اشاره نمایید.** در مسیر انجام پروژه هر ابتکار یا فعالیتی که منجر به عملکرد بهتر عامل شده است را توضیح دهید. در نهایت منابعی که برای پیاده‌سازی پروژه استفاده کرده‌اید را ذکر نمایید.

۵) راه‌های ارتباط با دستیار پروژه به صورت زیر می‌باشد:

- Telegram ID: @arshia_hemmat

- Email: arshiahemmat93@gmail.com

در صورتی که در باز کردن از هر کدام از هایپرلینک‌ها مشکل دارید، می‌توانید از این قسمت لینک‌ها را باز کنید.

[/https://pettingzoo.farama.org/environments/classic/connect_four](https://pettingzoo.farama.org/environments/classic/connect_four)

[/https://pypi.org/project/pettingzoo](https://pypi.org/project/pettingzoo)

موفق باشید