



مدرس: دکتر فدایی و دکتر یعقوبزاده

طراحان: سهیل شیروانی، طاها شعبانی

مهلت تحویل: جمعه ۲۰ فروردین ۱۴۰۰، ساعت ۲۳:۵۵

مقدمه

در این پروژه، با روش‌هایی که برگرفته از طبیعت و انتخاب طبیعی هستند، آشنا می‌شویم. در این روش‌ها که مجموعاً الگوریتم‌های ژنتیک نامیده می‌شوند، ایده‌هایی برای مدل‌سازی جفت‌گیری، جهش و انتخاب طبیعی به کار می‌گیریم. همچنین یاد خواهیم گرفت که بعضی اوقات با انتخاب معیارهای ساده انتخاب طبیعی، این الگوریتم‌ها ضعیف عمل می‌کنند و باید معیار انتخابی در نظر بگیریم که علاوه بر عملکرد فردی، به گوناگونی جمعیت نیز اهمیت دهد. الگوریتم‌های ژنتیک عموماً در مسئله‌هایی با فضای حالت بزرگ کاربرد دارند؛ در این الگوریتم‌ها، با نمونه گرفتن از جمعیت و ترکیب و تغییر افراد و ارزیابی آنان سعی می‌شود نسل به نسل جواب‌ها بهبود یابد تا به جواب مورد نظر برسد.

در این مسئله با برنامه‌ریزی^۱ که یکی از مسائل پرکاربرد در حوزه ژنتیک است آشنا می‌شویم.

^۱ Scheduling

شرح مسئله

در روزهای اخیر و مخصوصاً بعد از شیوع ویروس کرونا، نرخ مراجعه به بیمارستان‌ها بسیار افزایش پیدا کرده است. این امر سبب شده است که بیمارستان‌ها به فکر تعیین دقیق شیفت‌های کاری پزشکان خود باشند تا بتوانند جواب‌گوی بیماران مراجعه کننده باشند. مدیر یک بیمارستان از یک مهندس هوش مصنوعی خواسته است تا به او کمک کند تا یکی از بهینه‌ترین حالات ممکن برای شیفت پزشکان را با توجه به محدودیت‌ها بیابد.

این مدیر اطلاعاتی را در اختیار مهندس هوش مصنوعی قرار داده است و از او خواسته با توجه به این اطلاعات یکی از برنامه‌ریزی‌های بهینه برای پزشکان بیمارستان را پیدا کند. علاوه بر آن، به دلیل اینکه مدت خوبی از شیوع کرونا گذشته است، بیمارستان تخمین خوبی از تعداد بیماران مراجعه کنند بدست آورده است که با توجه به آن می‌تواند تخمین بزند در هر روز به طور حدودی چه تعداد پزشک برای بیمارستان نیاز دارد.

برای برنامه ریزی باید محدودیت‌های زیر اعمال شود:

۱. پزشکان در این بیمارستان دارای سه شیفت کاری **صبح، عصر و شب** هستند و برای هر روز تخمینی از **کمترین و بیشترین تعداد پزشک مورد نیاز** برای هر شیفت در بیمارستان وجود دارد که تعداد پزشکان تخصیص داده شده به آن شیفت باید در این بازه باشد.

۲. به دلیل اهمیت سلامتی پزشکان و همین‌طور به دلیل اینکه پزشکان باید بر کار خود تمرکز کافی داشته باشند و با بازده بالا بیماران را معالجه کنند، پزشکی که در شیفت شب کار می‌کند نمی‌تواند فردای آن روز در شیفت های صبح و عصر فعالیت داشته باشد. همچنین هیچ پزشکی نمی‌تواند سه شیفت شب پیاپی در بیمارستان کار کند.

۳. پزشکان نمی‌توانند بیشتر از تعدادی مشخصی، شیفت داشته باشند (ظرفیت برای تمام پزشکان یکسان است).

علاوه بر اطلاعات بالا، مهندس هوش مصنوعی اطلاعاتی از جمله تعداد کل پزشکان بیمارستان و تعداد روزهایی که نیاز است این برنامه ریزی انجام شود نیاز دارد. با توجه به اینکه مدیر این برنامه را برای بعدها نیز ممکن است استفاده کند از مهندس خواسته تا در صورت تغییر هر کدام از اطلاعات (استخدام پزشک جدید، تغییر تغییر تعداد بیشینه و کمینه پزشکان هر شیفت و ...) بتواند برنامه ریزی جدیدی برای پزشکان بدست آورد. طبق نظر مدیر، تنها یک برنامه که تمام شرایط را برقرار سازد برای او کافی است.

پیاده سازی مسئله

در این هندز آن از شما خواسته شده تا راه حلی را با استفاده از الگوریتم ژنتیک برای مسئله ی بالا پیدا کنید. برای پیاده سازی آن، شما باید مراحل زیر را همانطور که در درس نیز آموخته اید پیاده سازی کنید و سپس با تجمیع تمام این مراحل یک الگوریتم کلی با توجه به ورودی و خروجی های خواسته شده برای حل مسئله پیاده سازی کنید.

۱. مشخص کردن مفهوم ژن و کروموزوم

در الگوریتم ژنتیک ابتدا باید یک تعریف برای ژن پیدا کنید و سپس با استفاده از آن یک کروموزوم را بسازید. توجه داشته باشید که در ژنتیک اکثر کارها را باید به صورت رندوم انجام دهید چرا که اگر فضای حالت بزرگ باشد پیدا کردن شرطی که همه ی محدودیت ها را برقرار سازد بسیار دشوار است.

۲. تولید جمعیت اولیه

پس از پیاده سازی روند ساخت یک کروموزوم، باید یک جمعیت اولیه از کروموزوم ها بسازید. تعداد این جمعیت می تواند به عنوان یک پارامتر حل مسئله باشد و به راه حل شما (انتخاب های شما) بستگی دارد. دقت داشته باشید که در ساخت جمعیت اولیه باید کاملاً به صورت رندوم عمل کنید و سعی در پیاده سازی محدودیت های صورت مساله در جمعیت اولیه نکنید.

۳. مشخص کردن تابع معیار سازگاری²

در ادامه می‌بایست یک تابع برای معیار سازگاری کروموزوم‌های خود با مسئله پیدا کنید. این تابع باید برای هر کروموزوم یک معیار ارزیابی (نسبت به مسئله) پیدا کند. این معیار باید به گونه ای باشد تا تشخیص دهد کروموزوم چه مقدار از محدودیت‌های مسئله را پوشش می‌دهد و کدام کروموزوم‌ها بهتر و کدام یک ضعیف تر هستند.

۴. پیاده‌سازی crossover، mutation، ... و تولید نسل بعدی

شما باید یک تابع به عنوان crossover و یک تابع به عنوان mutation داشته باشید. تابع crossover بر روی دو کروموزوم اعمال می‌شود و آن‌ها را با یک روش (از روش‌های گفته شده در کلاس می‌توانید استفاده کنید) ترکیب می‌کند تا به کروموزوم‌های بهتر برسد. این ترکیب و نرخ ایجاد آن می‌تواند به عنوان پارامترهای مسئله باشد که با توجه به تعریف شما از تابع می‌تواند بهبود یابد. تابع mutation بر روی یک کروموزوم اعمال می‌شود و با استفاده از روشی آن را جهش داده و تغییر می‌دهد، به امید آن که بتواند به کروموزوم بهتری دست یابد. این کار باعث می‌شود تا کروموزوم‌های جدیدی در جمعیت به وجود آیند. همچنین می‌توانید از درصد معقولی از ژن‌های برتر برای انتقال مستقیم به نسل‌های آینده نیز استفاده کنید.

۵. ایجاد الگوریتم ژنتیک

در آخر باید این توابع را در یک الگوریتم استفاده کنید. توجه کنید که می‌توانید پارامترهایی برای راه خود داشته باشید که با تغییر آن به جواب بهتری برسید.

فرمت ورودی

فرمت ورودی که به شما داده می‌شود یک فایل شامل خط‌های زیر است که مدیر آن را آماده کرده است.

در خط اول به ترتیب ابتدا تعداد روزها (d) و تعداد پزشکان (p).

² Fitness Function

در خط دوم حداکثر ظرفیت پزشک‌ها (c).

در خط بعدی و تا d خط مقادیر کمینه و بیشینه پزشکان لازم برای شیفت‌های روز به صورت:

min-morning,max-morning min-evening,max-evening min-night,max-night

برای مثال:

4 5

9

2,2 1,4 2,2

1,2 1,2 3,5

2,2 2,3 1,4

1,2 1,2 1,3

فرمت خروجی

خروجی برنامه شما باید یک برنامه ریزی برای پزشکان باشد که تمام شرایطی که مسئله خواسته است را رعایت کرده باشد. توجه کنید که چون خروجی شما با برنامه چک می شود شما باید حتما خروجی خود را در یک فایل به شکل زیر گزارش کنید.

Id هر پزشک را به صورت [0, 1, ..., p-1] در نظر بگیرید و در یک فایل به تعداد d خط (تعداد روزهای برنامه‌ریزی) برای هر شیفت، پزشکان تخصیص یافته را بنویسید. هر شیفت را با یک 'space' و هر پزشک را با یک '،' جدا کنید. برای مثال:

5,1 1,2,4 2,3,0

1,4,5 1 0,3,4

...

5,1 1,2,4 2,3,0

نکته: همچنین شما باید اجرای تست‌ها را در زمان معقولی به اتمام برسانید و این زمان اجرا را در انتهای فایل **notebook** خود قرار دهید. (هر تست را دوبار انجام داده و میانگین این دو زمان را گزارش کنید):

تست ۱: کمتر از ۱۵ ثانیه

تست ۲: کمتر از ۶۰ ثانیه

ملاحظات

- موعده تحویل غیرحضوری تا پایان روز 20 فروردین می‌باشد.
- تمامی نتایج باید در یک فایل فشرده با عنوان `AI-HandsOn1-<#STID>.zip` تحویل داده شود.
- توجه داشته باشید که علاوه بر ارسال فایل `jupyter`، فایل‌های خروجی با نام‌های `output1.txt` و `output2.txt` را نیز ارسال کنید.
- شباهت غیر معقول در انجام این پروژه بین افراد مختلف پذیرفته نمی‌شود. در صورت کشف هرگونه تقلب برای همه افراد متقلب نمره ۰- در نظر گرفته می‌شود.

- استفاده از مراجع با ارجاع به آنها بلامانع است. اما در صورتی که گزارش شما ترجمه عینی از آن‌ها باشد، یا از گزارش افراد دیگر استفاده کرده باشید کار شما تقلب محسوب می‌شود.
- در صورتی که سوالی در مورد پروژه داشتید بهتر است در فروم درس مطرح کنید تا بقیه از آن استفاده کنند، در غیر این صورت به طراحان پروژه ایمیل بزنید و از یکی از آن‌ها بپرسید.

taha.shabani.m@gmail.com

shirvani.soheil@ut.ac.ir