

به نام خالق یکتا

سیستم تشخیص بیماری بر اساس علائم کاربر

درس مبانی و کاربردهای هوش مصنوعی

ترم دوم سال تحصیلی 1401-1402

در این تمرین، هدف پیاده سازی یک سیستم تشخیص بیماری ساده بر اساس علائم دریافتی از بیمار است. سیستم در گام‌های مختلف، علائم یک بیمار را دریافت می‌کند و براساس دانش ذخیره شده در پایگاه دانش و استنتاج منطقی، بیماری با بیشترین تطابق با علائم را بر می‌گرداند. برای پیاده سازی این سیستم از زبان‌های برنامه‌نویسی پایتون و پرولوگ استفاده می‌کنیم. در ادامه گام‌های مختلف تمرین توضیح داده شده‌اند.

1. ساخت پایگاه دانش

به عنوان اولین گام، باید پایگاه دانشی از بیماری‌ها و علائم مختلف آن‌ها ساخته شود. به شما فایلی به نام `illnesses.txt` داده شده است. این فایل شامل جملات توصیف کننده‌ی بیماری‌ها به شکل زیر است:

`[illness_name] symptoms are [symptom_1], [symptom_2], ..., and [last symptom].`

در ابتدا به جای `[illness_name]`، نام یک بیماری می‌آید و سپس تعدادی از علائم به شکل `[symptom_i]` در ادامه جمله خواهند آمد که همگی با `"`، از یکدیگر جدا شده‌اند. آخرین علامت با `and` بعد از `"`، شروع می‌شود. به عنوان مثال، برای توصیف بیماری آنفولانزا (`flu`)، جمله توصیف کننده می‌تواند به شکل زیر باشد:

`flu symptoms are fever, body_aches, and fatigue.`

یا بیماری سرماخوردگی (`cold`) می‌تواند به شکل زیر توصیف شده باشد:

`cold symptoms are cough, and sore_throat.`

شما باید هر جمله را از فایل `illnesses.txt` بخوانید، اطلاعات لازم را استخراج کنید و در پایگاه دانش پرولوگ ذخیره کنید.

مثالی از وارد کردن بیماری آنفولانزا (`flu`) در پایگاه داده را می‌توانید در شکل 1 ببینید:

```

prolog.assertz("symptom(flu, fever)")
prolog.assertz("symptom(flu, body_aches)")
prolog.assertz("symptom(flu, fatigue)")
prolog.assertz("illness(flu)")

prolog.assertz("symptom(cold, sore_throat)")
prolog.assertz("symptom(cold, cough)")
prolog.assertz("illness(cold)")

```

شکل 1- مثالی وارد کردن علائم بیماری آنفولانزا و سرماخوردگی در پایگاه دانش

2. دریافت علائم و تشخیص بیماری:

در این گام باید در تعامل با کاربر (که در اینجا محیط مسأله را مشخص می‌کند) علائم بیماری را از او دریافت کنید. با توجه به تعداد بیماری‌هایی که بیشترین تطابق را با علائم دارند، دو حالت قابل در نظر گرفتن است:

1. یک بیماری با بیشترین تطابق وجود دارد: در این صورت این بیماری باید به عنوان بیماری نهایی کاربر برگردانده شود. برای مثال پایگاه دانش شکل 2 را در نظر بگیرید. اگر کاربری علائم **chills, cough** و **fatigue** را وارد کند، با توجه به اینکه علائم بیشتری با بیماری **flu** مطابقت دارند، بیماری نهایی **flu** می‌باشد.
2. اگر بیش از یک بیماری با بیشترین تعداد تطابق با علائم وارد شده وجود داشته باشند، سیستم وارد وضعیت پرسیدن سوال های بله/خیر درباره علائم می‌شود تا زمانی که یک بیماری را به عنوان بیماری نهایی اعلام کند. در این فاز باید علائم غیرمشترک بیماری‌هایی که در گام اول، همگی بیشترین تطابق را با علائم وارد شده داشته‌اند مورد پرسش قرار گیرند و پس از هر پاسخ کاربر، لیست بیماری‌های ممکن بروزرسانی شود تا زمانی که یک بیماری با بیشترین علائم وارد شده یافت شود. در این هنگام آن بیماری به عنوان پاسخ بازگردانده می‌شود.

```

prolog.assertz("symptom(fever, high_temperature)")
prolog.assertz("symptom(fever, chills)")

prolog.assertz("symptom(flu, chills)")
prolog.assertz("symptom(flu, body_aches)")
prolog.assertz("symptom(flu, fatigue)")

```

شکل 2- مثالی از یک پایگاه دانش

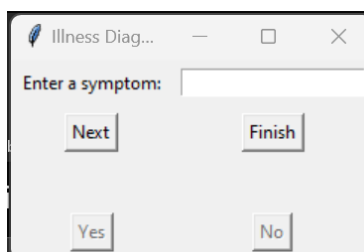
نحوه پیاده سازی

برای پیاده سازی این تمرین، از زبان برنامه‌نویسی پایتون و پرولوگ استفاده خواهید کرد. برای اینکه بتوانید با استفاده از زبان پایتون، با پایگاه دانش پرولوگ ارتباط برقرار کنید، باید از رابط **PySwip** استفاده کنید. این رابط از طریق راهنمایی‌های [این لینک](#) قابل دریافت است.

همینطور باید خود زبان پرولوگ را نیز در سیستم عامل خود نصب کنید. دقت کنید که برای اینکه زبان و رابط PySwip، بتوانند با هم ارتباط برقرار کنند، بهتر است نسخه 8.4.3 از این زبان را نصب کنید که از [این لینک](#) قابل دریافت است و همینطور در پوشه مربوط به فایل های پروژه نیز قرار داده شده است.

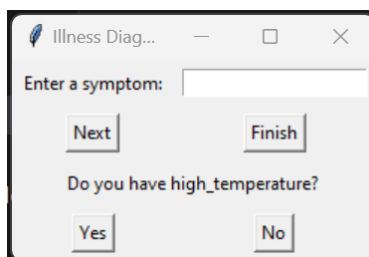
برای انجام تمرین، شما باید فایل `template.py` که در اختیار شما قرار داده شده است را تکمیل کنید (قسمت های مشخص شده با **TODO** را کامل کنید).

اجرای این فایل، محیط گرافیکی ساده ای به شکل زیر را در اختیار شما قرار می دهد:



شکل 3- پنجره ورود علائم اولیه بیماری

نحوه کار این محیط به این شکل است که کاربر در هر گام، یکی از علائم بیماری را در کادر مشخص شده وارد می کند، و سپس دکمه **Next** را انتخاب می کند. پس از اینکه همه علائم وارد شدند، کاربر دکمه **Finish** را انتخاب می کند. پس از انتخاب این دکمه، چنانچه یک بیماری شناسایی شده بود، آن بیماری در فایل خروجی به نام `diagnosed_illness.txt` ذخیره شده و تعامل با کاربر به پایان می رسد. در غیر این صورت، برنامه وارد بخش پرسش و پاسخ می شود که در آن سؤالاتی درباره علائم غیر مشترک بیماری ها از کاربر پرسیده می شود تا بیماری منحصر به فردی مشخص شود و به عنوان بیماری نهایی در فایل گفته شده ذخیره شده و برنامه به پایان برسد. مثالی از این موضوع در شکل 4 قابل مشاهده است. در اینجا ابتدا علائم **chills** مربوط به بیماری **fever** و **cough** مربوط به بیماری **cold** وارد شده اند و سپس دکمه **finish** زده شده است. به دلیل اینکه دو علامت مختلف از دو بیماری وارد شده است، سیستم قادر به تصمیم گیری درباره بیماری نهایی نیست و وارد فاز پرسش و پاسخ می شود که و در آن درباره علائم نا مشترک بیماری ها سؤالاتی می پرسد. مثلاً در شکل 4 درباره علامت **high_temperature** بیماری **fever** سوال پرسیده شده و از کاربر خواسته شده که بگوید آیا این نشانه را دارد یا خیر.



شکل 4- مثالی از پرسش یک سوال بله/خیر

آنچه شما باید انجام دهید:

در فایل `template.py`، چهار بخش `TODO` مشخص شده اند. در اولین آن‌ها باید خواندن فایل بیماری‌ها و ساخت پایگاه دانش را انجام دهید.

در دومین `TODO`، باید تابعی را کامل کنید که لیستی از همه علائم وارد شده از ابتدای برنامه تا انتخاب دکمه `finish` را از ورودی می‌گیرد (symptoms)، و بیماری یا بیماری‌هایی که بیشترین تطابق را با لیست دارند، در غالب یک لیست برمی‌گرداند. اگر هیچ بیماری مطابقی وجود نداشته باشد، باید `"Unknown illness"` را به عنوان یک لیست برگردانید.

در سومین و چهارمین `TODO` ها، باید فاز پرسش برنامه را کامل کنید.

در تابع `ask_question`، باید مانند مثال آورده شده در کامنت، درباره علائم مورد سوال قرار نگرفته، سوال بپرسید و پاسخ کاربر را به تابع بعدی یا `TODO` چهارم بفرستید تا تغییرات لازم بر اساس پاسخ کاربر را اعمال کنید. می‌توانید لیست ورودی‌های توابع را با توجه به نیاز خودتان تغییر دهید. دقت کنید که تابع چهارم مشخص می‌کند بر اساس پاسخ کاربر، چه اتفاقی باید رخ دهد. اگر بیماری یکتایی برای مشخص کردن وجود نداشت، باید `"Unknown illness"` را به عنوان بیماری برگردانده شود تا در فایل ذخیره شود.

دقت کنید که نیازی به تغییر در هیچ بخش دیگری از برنامه نیست.

مواردی که باید تحویل داده شوند:

در نهایت یک فایل زیپ شامل موارد زیر باید تحویل داده شود:

1. فایل تکمیل شده‌ی `template.py` با تغییر نام به `[Your_lastname_StudentNumber].py`

2. فایل گزارش مختصر از نحوه ساخت پایگاه دانش و توضیح قوانین تعریف شده و توضیحی مختصر درباره

بخش‌های کامل شده به نام `[Your_lastname_StudentNumber].doc`.

نام گذاری فایل زیپ باید به شکل `Your_lastname_StudentNumber.zip` باشد.

مهلت انجام این تمرین تا پایان روز جمعه 1402/03/26 بوده و تمرین باید به صورت انفرادی انجام شود و در موعد مقرر در کوئرا بارگزاری شود.

موفق باشید