



گزارش تمرین پایانی

درس مبانی و کاربردهای هوش مصنوعی

دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

استاد درس : دکتر کارشناس

تهیه کننده :

محمد حسین دهقانی اشکذری

تیر ۱۴۰۲

مقدمه :

در این پروژه قصد داشتیم تا سیستمی پیاده سازی شود که بر اساس علائم دریافتی از بیمار و با استفاده از پایگاه دانش تعبیه شده و استنتاج منطقی، بیماری که بیشترین میزان تطابق با علائم وارد شده دارد را مشخص کنیم. در اجرای این پروژه از زبان برنامه نویسی پایتون و پرولوگ برای نوشتن برنامه و تعبیه و استفاده از پایگاه دانش استفاده کردیم.

نحوه ساخت پایگاه دانش:

در این پروژه برای ساخت پایگاه دانش از کتابخانه pyswip استفاده کردیم. به این نحو که پس از خواندن اطلاعات بیماری ها، که شامل نام بیماری و علائم آن بود، و پردازش متون خوانده شده؛ هر بیماری را متناظر با علائمش و با استفاده از دستور assertz به پایگاه دانش وارد می کردیم تا در ادامه کار از آنها برای استنتاج منطقی و در نهایت تعیین نام آن بیماری که با علائم بیشترین هم خوانی را دارد استفاده کنیم. سه تابع زیر اعمال خواندن از فایل، پردازش متن خوانده شده از فایل و تشکیل پایگاه دانش را انجام می دهند.

```
# STEP1: Define the knowledge base of illnesses and their symptoms
```

```
1 usage - Mohammad Hossein Dehghani
```

```
def read_from_file(filename):...
```

```
1 usage - Mohammad Hossein Dehghani
```

```
def parse_illnesses(illnesses):...
```

```
1 usage - Mohammad Hossein Dehghani
```

```
def create_knowledge_base(statements):...
```

```
#####
```

نحوه تعیین بیماری هایی که بیشترین همخوانی را با علائم وارد شده دارند :

تابع diagnose با دریافت لیست علائم وارد شده، لیستی از بیماری هایی که بیشترین همخوانی را با آن علائم دارند را خروجی می دهد. به این صورت که ابتدا تمام بیماری هایی که با علائم دریافت شده همخوانی دارند را از طریق کوئری زدن به پایگاه دانش پیدا می کند و سپس بیماری هایی که بیشترین تطابق را دارند از میان تمامی

بیماری‌ها انتخاب کرده و آن‌ها را خروجی می‌دهد. اگر تعداد بیماری‌های خروجی داده شده بیشتر از یک باشد برنامه وارد فاز پرسش و پاسخ می‌شود.

نحوه پرسش و پاسخ:

برای پرسش و پاسخ دو تابع `ask_question` و `on_question_answer` که در تمپلیت هم وجود داشت به علاوه تابع `find_diff_symptoms` وجود دارد. در ابتدا تابع `ask_question` صدا زده می‌شود. در این تابع سوالاتی پرسیده می‌شود و با فراخوانی تابع `on_question_answer` و تحلیل پاسخ‌ها تصمیم مناسبی برای حذف یا نگه داشتن بیماری‌ها گرفته می‌شود تا در نهایت به یک بیماری برسیم که پاسخ ماست.

در این بخش تابع `find_diff_symptoms` اضافه شد تا فرایند پرسش و پاسخ را بهینه‌تر و دقیق‌تر کند. این تابع باعث می‌شود از پرسیده شدن سوالات تکراری و سوالاتی که پاسخ آن‌ها مشخص است صرف نظر شود و صرفاً سوالاتی کلیدی پرسیده شود تا بهینگی برنامه و الگوریتم افزایش یابد.