
 <p>دانشگاه صنعتی امیرکبیر</p>	<p>دوره آموزشی پردازش زبان طبیعی (NLP)</p> <p>تمرین هشتم</p> <p>مهلت تحویل: سه شنبه ۲۰ مهر ۱۴۰۰</p>	 <p>آکادمی همراه</p>
---	---	---

Dialog State Tracking Challenge 2

مقدمه:

در قسمت پایانی به یکی از چالش‌های معروف چت‌بات‌ها خواهیم پرداخت. نسخه کامل‌تر این داده در [اینجا](#) قرار گرفته است. مقالات متعددی روش خود را بر روی این داده بررسی کرده‌اند که در سایت [paperswithcode](#) قابل مشاهده هستند. همچنین پروژه‌های [DeepPavlov](#) و [RASA](#) کتابخانه‌هایی را برای چت‌بات‌ها نوشته‌اند و این داده را به عنوان نمونه بررسی کرده‌اند. در این تمرین نیز همانند تمرین‌های قبل، پروژه‌ای که تحویل می‌دهید را خودتان بنویسید و هدف درک دقیق شما از **slot filling**، **intent detection** و **state tracking** است. فایل‌های دیتای تمرین به صورت زیر قابل دریافت هستند:

```
!gdown --id 104TU-gdBAtalfRyaauuKaOobhGbB52Zf
```

فایل ner_intent.json

شامل همه چرخش‌های مربوط به کاربر است و برای هر چرخش توکن‌ها و برچسب‌های لازم برای **slot filling** و اسلات و مقدار آن مشخص شده است. همچنین **intent** کاربر نیز مشخص شده است. این فایل کاملاً دقیق نیست و برخی خطاها در آن وجود دارد. نمونه ای از داده‌های موجود در این فایل را می‌بینید.

```
"im looking for a cheap restaurant in the south part of town": {
  "TEXT": [ "im", "looking", "for", "a", "cheap", "restaurant",
            "in", "the", "south", "part", "of", "town"],
  "NERTAGS": [ "O", "O", "O", "O", "B-pricerange", "O", "O", "O",
               "B-area", "O", "O", "O" ],
  "NERVLAS": { "pricerange": "cheap", "area": "south" },
  "INTENTS": [ "inform_pricerange", "inform_area" ]
}
```

فایل‌های simple-dstc2-tst.json simple-dstc2-val.json simple-dstc2-trn.json

فایل‌ها به فرمت json هستند. هر فایل شامل چندین مکالمه است که در آن کاربر با **speaker: 1** و بات با **speaker: 2** مشخص شده است. برای هر چرخش کاربر مشخص شده چه اسلات‌هایی موجود است و مقدار آن چیست.

فایل: slot_vals.json

در این فایل برای هر نوع اسلات مقادیر معمول آن و معادل‌های آن نوشته شده است.

فایل simple-dstc2-templates.txt

یک اکشن تمپلیت است در حقیقت تمام اکشن‌های بات از روی این تمپلیت تولید شده‌اند. در هر خط نام یک اکشن و تمپلیت آن نوشته شده و با تب از هم جدا شده‌اند.

بخش‌های پروژه:

پیشنهاد می‌شود یکبار تمام سؤالات را مطالعه کنید و سپس براساس سؤالات داده‌ها را بررسی کنید تا بیشتر با ساختار آن آشنا شوید.

نکته مهم: به علت محدودیت‌های GPU کولب دقت کنید که برای تحویل تمرین نوت بوک را در حالت سی پی یو قرار دهید. مرحله ترین مدل را برای یک ایپاک انجام دهید و برای قسمت‌های بعدی از لود وزن‌ها استفاده نمایید.

سؤال ۱: براساس داده‌های موجود یک ماژول slot filling پیاده‌سازی کنید و دقت آن را به‌صورت مجزا گزارش کنید. دقت را با macro f1 برای داده آموزش ضمن تابع fit و برای داده تست محاسبه نمایید.

نکته ۱: برای این ماژول می‌توانید همانند کد ویدئو از مدل biGRU استفاده کنید و یا با برت پیاده‌سازی را انجام دهید. توجه کنید که اگر پیاده‌سازی را با برت انجام دهید امتیاز اضافه دریافت خواهید کرد.

سؤال ۱-۱: فایلی که در زیر آمده در هر خط شامل یک جمله است. در تمرین یک سلول قرار دهید که این فایل را باز کند و برای هر خط مشخص کند چه اسلات‌هایی را با چه مقادیری تشخیص داده است. محتوای فایل در هنگام تصحیح بروز خواهد شد.

```
!gdown --id 1-ay5mBPEFuzHGZpDnBJIA5e_ONAIcvjW
```

سؤال ۲: براساس داده‌های موجود یک ماژول intent detection را براساس مدل برت پیاده‌سازی کنید و دقت آن را به‌صورت مجزا گزارش کنید. توجه داشته باشید که در این داده هر turn می‌تواند بیش از یک intent داشته باشد. دقت را با micro f1 برای داده آموزش ضمن تابع fit و برای داده تست محاسبه نمایید.

سؤال ۲-۱: فایلی که در زیر آمده در هر خط شامل یک جمله است. در تمرین یک سلول قرار دهید که این فایل را باز کند و ضمن چاپ هر خط مشخص کند چه intent‌هایی را تشخیص داده است. محتوای فایل در هنگام تصحیح بروز خواهد شد.

```
!gdown --id 101uOTEhpm7t3Y4iIPpA_isGUHGKXafsa
```

سؤال ۳: از روی داده‌های موجود اطلاعات زیر را استخراج کنید:

الف) لیست اسلات‌هایی که باید بات از آن‌ها مطلع شود تا بتواند رستوران مناسب را پیشنهاد کند. مثلاً اسلات pricerange که مشخص می‌کند رستوران گرانی هست یا ارزان.

ب) برای هر رستوران یک دیکشنری به دست بیاورید که دقیقاً ترتیب آیتم‌های آن مانند نمونه زیر باشد. سپس لیستی از این دیکشنری‌ها بسازید که به ترتیب حروف الفبای نام رستوران‌ها مرتب شده باشند. در یک سلول مجزا و به فرمتی خوانا اول طول لیست و سپس کل لیست را چاپ نمایید.

```
{
  "name": "the river bar steakhouse and grill"
  "food": "modern european",
  "area": "centre",
  "pricerange": "cheap",
  "phone": "01223 307030",
  "addr": "quayside off bridge street",
  "postcode": "c.b 5, 8 a.q",
},
```

سؤال ۳-۱: فایلی که در زیر آمده در هر خط شامل یک جمله است. در تمرین یک سلول قرار دهید که این فایل را باز کند و ضمن چاپ هر خط مشخص کند چه تعداد و چه رستوران‌هایی را برای این کوئری یافت می‌شود. در خط اول کوئری، در خط بعدی تعداد و سپس لیست نام رستوران‌ها را بنویسید (در هر خط یک رستوران و به ترتیب حروف الفبا) و بعد از آن ۵ خط خالی چاپ کنید. محتوای فایل در هنگام تصحیح بروز خواهد شد.

```
!gdown --id 1-snND1ySP8qslXLVScy6z7lSPZTc4WI4
```

نکته ۵: پیشنهاد می‌شود خروجی سؤال ۳ به صورت فایل json نگهداری شود.

سؤال ۴: به کمک دو ماژولی که در سؤال ۱ و ۲ تهیه کردید و اطلاعاتی که در سؤال ۳ استخراج کردید، یک بات ساده طراحی کنید که درخواست متنی کاربر را دریافت کرده رستوران متناسب با علاقه‌مندی به وی پیشنهاد دهد. این بات در هر `turn` برخی از نیازمندی‌های کاربر را دریافت می‌کند و باید با ماژول `intent detection` قصد کاربر را تشخیص دهد و با ماژول `slot filling` اسلات‌هایی که در پیام کاربر ذکر شده است را تشخیص دهد. سپس در صورتی که اطلاعات مورد نیاز کافی نیست سوالات بیش‌تری مطرح کند (نمونه این مساله در داده‌ها موجود است). در نهایت با اطلاعات کسب شده از روی لیست رستوران‌ها یک یا چند رستوران مناسب پیشنهاد کند.

نکته ۶: در حالت ساده می‌توانید چند `if else` قرار دهید. در صورت علاقمندی برای حالت پیچیده‌تر از الگوی حالت در مهندسی نرم‌افزار و یا مدل‌های عصبی استفاده کنید.

نکته ۷: در آخرین سلول تمرین اجازه دهد با بات شما تعامل کنیم به این معنی که یک جمله بنویسیم و جواب بات را دریافت کنیم. این سلول سلول آخر باشد تا امکان اجرای چند باره آن وجود داشته باشد.

نمونه تست:

```
user: hi
bot: Hello, welcome to the Cambridge restaurant system. You can ask for res-
taurants by area, price range or food type. How may I help you?

user: a cheap restaurant plz
bot: What kind of food would you like?

user: chinese
bot: charlie chan serves chinese food in the cheap price range.

user: address
bot: Sure, charlie chan is on regent street city centre.

user: how about west is there any there
```

bot: Sorry there is no cheap restaurant in the west of town serving chinese food.

user: oh thanks

bot: You are welcome!

سوال اضافه برای افراد علاقمند:

سؤال ۵ - یک مدل چندوظیفه‌ای طراحی کنید که هر دو وظیفه **intent detection** و **slot filling** را با هم انجام دهد. یعنی با دریافت یک متن به‌طور همزمان مشخص کند که قصد این جمله چیست و چه اسلات‌هایی را پر می‌کند.

نکته ۴: تابع **compile** در کراس می‌تواند بجای یک خروجی تاپلی از خروجی‌ها را دریافت کند. در این شرایط، اگر توابع خطای خروجی‌ها یکسان نباشد، لیستی از توابع خطا را می‌توان تعریف کرد. برای اطلاعات بیشتر پارامترهای این تابع را بررسی کنید و به دنبال عبارت **multi-task** باشید.

سؤال ۶ - نام و مشخصات ۵ رستوران ایرانی را به دیتابیس اضافه کنید و در مدل قابلیت پیشنهاد این رستوران‌ها را نیز قرار دهید.

نکته ۵: توجه کنید که ادرس این رستوران‌ها هم با جهت‌های جغرافیایی قابل جست‌وجو باشد هم با نام خیابان یا محله. همچنین این رستوران‌ها غذایی سرو کنند که در دیتاست برچسب نخورده است. مثلاً در پاسخ به جمله زیر بتوان رستورانی را پیشنهاد داد.

"I have a craving for ghorme sabzi."