بسم الله الرحمن الرحیم

دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

پردازش زبان طبیعی

نیم‌سال دوم ۰۳-۰۲

مدرس: احسان‌الدین عسگری

تمرین سوم | **طبقه‌بندی سند - طبقه‌بندی کلمه** | مهلت ارسال: ۳ تیر

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص شده است.
- در تمرین‌هایی که چند چالش دارند، فقط یک نفر از هر گروه در گوگل فرم باید چالش مورد نظر گروه را انتخاب کند. امکان تغییر چالش تا قبل از زمان ددلاین انتخاب چالش وجود دارد. البته ذکر این نکته ضروری است که هر چالش محدودیتی برای تعداد افرادی که آن را انتخاب می‌کنند، دارد. بنابرین در اسرع وقت برای انتخاب چالش اقدام کنید.
- در طول ترم امکان ارسال با تاخیر تمرین‌ها بدون کسر نمره تا سقف ۱۲ روز وجود دارد. محل بارگذاری جواب تمرین‌ها بعد از ۵ روز ، بسته خواهد شد و پس از گذشت این مدت، پاسخ‌های ارسال‌شده پذیرفته نخواهند شد. همچنین، به ازای هر روز تأخیر غیر مجاز ۱۵ درصد از نمره تمرین کسر خواهد شد. لازم به ذکر است که به دلیل تداخل زمان مجاز تاخیرها بین اعضای گروه در تمارین گروهی تمرین اول شامل تاخیر مجاز **نمی‌باشد**.
- توجه داشته‌باشید که نوت‌بوکهای شما باید قابلیت بازاجرای ۱۰۰ درصد داشته باشند و در صورت نیاز به نصب یک کتابخانه یا دسترسی به یک فایل، مراحل نصب و دانلود (از یک محل عمومی) در نوت‌بوک وجود داشته باشد.
- تمامی فایل‌های مرتبط به پروژه که حجم کمی دارند باید به شکل فایل زیپ در سامانه CW اپلود شوند. اگر حجم یک فایل زیاد بود (مانند فایل ذخیره شده یک مدل در صورتیکه بیش از ۲۰۰ مگابایت باشد)، تنها همان فایل را در یک محل عمومی، مثل گوگل درایو آپلود بفرمایید و لینک دانلود را در نوت‌بوک و مستندات قرار دهید.
- در پروژه‌های گروهی کافی است که فقط یکی از اعضای گروه پروژه را آپلود کند. اما حتما در گزارش کار نام همه اعضای گروه همراه با شماره دانشجویی آن‌ها آورده شود.
- بخشی از نمره شما به گزارش کار شما اختصاص دارد. در گزارش کار لازم نیست خط به خط کاری را که کرده‌اید توضیح دهید. بلکه باید به شکل کلی ایده‌تان برای حل مساله را شرح دهید. لازم است چند نمونه از خروجی‌های مساله را در گزارش بیاورید و براساس آن رفتار برنامه‌تان را تحلیل کنید. همچنین اگر پارامتری در صورت مساله خواسته شده (مانند دقت، صحت و مواردی از این دست) که در گزارش آورده شود شما باید آن را حساب کنید و در گزارش خود بیاورید.
- دقت داشته باشید، موارد امتیازی که در این تمرین آمده است، صرفا بر روی امتیاز همین تمرین اثر دارد و بر روی نمرات تمارین و یا بخش‌های دیگر درس، تاثیر ندارد.
- در صورت وجود هرگونه ابهام یا مشکل، در کوئرای درس آن مشکل را بیان کنید و از پیغام دادن مستقیم به تیم تدریس خودداری کنید.

ساز و کار تمرین (ابتدا این بخش را به صورت کامل مطالعه نمایید.)

در این تمرین هر گروه یکی از موضوع‌های پیشنهادی را انتخاب خواهد کرد. هر موضوع شامل دو بخش طبقه‌بندی سند[1] و طبقه‌بندی کلمه[2] است. درصورتی که در هرکدام از موضوعات تمایل دارید تا روی مجموعه دادگان دیگری کار کنید، توضیحات و آدرس مجموعه دادگان مدنظر را برای تیم تدریس در کوئرا ارسال کنید تا پس از بررسی و تایید تیم تدریس، بتوانید بر روی آن تمرین خود را انجام دهید.

[1] Document Classification
[2] Token Classification

برای موضوعاتی که در ترم‌های قبل نیز توسط دانشجویان مورد بررسی قرار گرفته است، بهترین کد آن‌ها در اختیار شما قرار داده خواهد شد تا از دوباره‌کاری جلوگیری شود. به تبع انتظار می‌رود تلاش شما روی آن موضوع باعث بهبود عملکرد کدهای قبلی شود. یکی از ایده‌های اولیه برای ارتقای عملکرد مدل‌های یادگیری ماشین، تحلیل خطا است. به این معنی که برخی از نمونه‌هایی که مدل آن‌ها را اشتباه پیش‌بینی می‌کند را بررسی کرده و درصورت مشاهده ساختارهای پرتکرار در آن‌ها که احتمالا دلیل بروز خطا در مدل هستند، تلاش کرد که مواردی که این ساختار را دارند اصلاح کرد یا مدل را نسبت به این موارد مقاوم نمود.

در بخش **طبقه‌بندی سند** نیاز است تا برای موضوع پیشنهادی، دو مدل زیر پیاده‌سازی شود:

۱. **مدل پایه:** اجرای مدل Logistic Regression یا Linear-SVM یا Nïve Bayes بر روی بردار ویژگی tf-idf
۲. **مدل اصلی:** استفاده از طبقه‌بندهای بر پایه ترنسفورمر (به عنوان مثال تنظیم کردن[3] پارامترهای مدل BERT)

در بخش **طبقه‌بندی کلمه** نیاز است تا برای موضوع پیشنهادی، دو مدل زیر پیاده‌سازی شود:

۱. **مدل پایه:** استفاده از مدل LSTM/CRF و یا HMM
۲. **مدل اصلی:** استفاده از مدل‌های بر پایه ترنسفورمر

به منظور استفاده از دادگان در هر بخش، می‌بایست دادگان خود را به سه بخش ۸۰، ۱۰ و ۱۰ درصد تقسیم کنید که به ترتیب دادگان آموزش، اعتبارسنجی و تست خواهد بود (بعضی از مجموعه دادگان به صورت پیش‌فرض برای این منظور تقسیم‌بندی شده‌اند، در این موارد نیازی به تقسیم بندی مجدد نیست و از همان تقسیم بندی پیش‌فرض خود مجموعه استفاده کنید تا مدل نهایی شما با دیگران قابل مقایسه باشد). در نهایت پس از بررسی کامل مدل و انتخاب تمامی هایپرپارامترها، عملکرد هرکدام از مدل‌ها را بر روی دادگان تست گزارش کنید. توجه داشته باشید که حتما ابتدا دادگان را تقسیم بندی کرده ذخیره کنید و سپس مدل‌های مختلف را بر روی آن‌ها تست نمایید، تا به این ترتیب مقایسه مدل‌های مختلف با هم عادلانه‌تر باشد. (توجه داشته باشید که در بخش استفاده از **مدل پایه ۱ بخش طبقه‌بندی سند** نیازی به ۱۰ درصد اعتبارسنجی نیست (دادگان اعتبارسنجی را با آموزش ترکیب کنید) و به این ترتیب دادگان به نسبت ۹۰ به ۱۰ تقسیم شده و به صورت Cross Validation با دادگان آموزش، معیارهای ارزیابی که در ادامه گفته شده است را محاسبه و میانگین و انحراف معیار آن‌ها را گزارش کنید).

برای هر دو بخش **طبقه‌بندی سند** و **طبقه‌بندی کلمه** نیاز است تا معیارهای F1(macro/micro)، Accuracy، Precision، Recall و ماتریس درهم‌ریختگی[4] محاسبه شود.

به منظور استفاده بهتر از مدل نهایی که شما توسعه داده اید و بررسی آن در ترم‌های آینده نیاز است تا بهترین مدل در هر بخش را در فضای Huggingface درس (این لینک) بارگذاری نمایید.

[3] Fine-tune
[4] Confusion matrix

تشخیص ژانر فیلم

طبقه‌بندی متن

در این تمرین می‌خواهیم مدلی را آموزش دهیم که با گرفتن خلاصه فیلم بتواند ژانر فیلم را تشخیص دهد. در این تسک از دیتاست (لینک دیتاست) استفاده می‌کنیم، این دیتاست خلاصه و ژانر فیلم‌ها و اطلاعات دیگر مثل سال تولید را برای فیلم‌های ایرانی به دو زبان انگلیسی و فارسی دارد.

با استفاده از این دادگان مدل‌های خود را بر روی هر دو **زبان فارسی و انگلیسی** آموزش دهید. برای مدل ترنسفورمر می‌توانید از Bert استفاده کنید.

برای مثال:

خلاصه فیلم: جلال، دانشجوی سابق رشته فلسفه، متوجه می‌شود خواهرش که به اختلال روانی دوقطبی مبتلاست، با مردی ثروتمند به نام شاهرخ ازدواج کرده که به شرط‌بندی بازی فوتبال اعتیاد دارد. جلال با عصبانیت از خانه بیرون می‌زند تا به خانه دوستش، بهمن، برود که یک آهنگ‌ساز زیرزمینی است. در مسیر جلال با یک راننده تاکسی به نام ناصر آشنا می‌شود و این دو شب عجیبی را در کنار هم سپری می‌کنند.

ژانر: درام

طبقه‌بندی کلمه

در این قسمت باید طبقه‌بندی را بر حسب کلمه انجام دهید. برای اینکار می توانید از دیتاست آرمان برای آموزش فارسی و CONLL ۲۰۰۳ برای انگلیسی استفاده کنید و در نهایت بعد از آموزش مدل بر روی ۱۰ اسم فیلم به زبان فارسی و ۱۰ اسم فیلم به زبان انگلیسی به صورت دستی لیبل بزنید و با مدل‌های خود نتیجه را گزارش دهید.

برای مثال:

اسم فیلم: پریناز

طبقه: B-pers

توجه

۱. دیتاست داده شده را **ضروری است** که حتما پیش‌پردازش کنید و مطمئن شوید که دیتاستی که برای آموزش مدل استفاده می‌کنید، داده‌های بالانس و مناسبی برای هر ژانر دارد. در این آنالیز اولیه دیتاست می‌توانید به دلخواه خودتان ۴ یا ۵ ژانر منتخب را انتخاب کنید و ژانرهای مشابه را در آن ژانر اصلی پوشش دهید. قسمت پیش‌پردازش و علت کارهایی که برای آنالیز داده انجام داده‌اید را در گزارش کار ذکر کنید.

۲. برای تست عملکرد حتما دیتاست را به ۳ بخش train،validation،test تقسیم کنید و بعد از آموزش عملکرد مدل را گزارش کنید. همچنین از ۵۰ خلاصه فیلم غیر ایرانی نیز برای تست کردن عملکرد تشخیص ژانر مدل انگلیسی استفاده کنید.

۳. در نهایت باید گزارش شما شامل ۴ بخش برای ۲ زبان انگلیسی و فارسی و عملکردهای هر کدام باشد .

۴. در گزارش تفاوت عملکرد برای زبان فارسی و انگلیسی را هم بر روی داده های test و هم بر روی دادگان OOD را ذکر کنید.

دستیار هوشمند

طبقه‌بندی متن

در این تمرین می‌خواهیم یک دستیار هوشمند پیاده‌سازی کنیم به صورتی که می‌تواند گفتار ما را به ۴ دستور پیش‌فرض تبدیل کند که این دستورات شامل گرفتن تاکسی، پخش موسیقی، نمایش آب و هوا و سفارش غذا می‌باشد.
فرض ما این است که بخش تبدیل سیگنال گفتار به متن از قبل انجام شده است (و درگیر جزئیات این بخش نیستیم) و متن خروجی به همراه اطلاعات آن را در اختیار داریم. (لینک دیتاست)
هدف ما در این قسمت طبقه‌بندی متن درخواست به یکی از ۴ فرمان گفته شده است،
به عنوان مثال:
متن درخواست: یه دونه موسیقی از استاد شجریان پخش کن
برچسب جمله: پخش موسیقی

طبقه‌بندی کلمه

در این قسمت باید طبقه‌بندی در سطح کلمه را پیاده‌سازی کنید. دیتاست مورد اشاره در قسمت اول علاوه بر برچسب فرمان، برچسب‌گذاری کلمات را نیز انجام داده است که در این قسمت باید برچسب کلمات داخل متن درخواست را طبقه بندی کنیم.
به عنوان مثال:
متن درخواست: یه دونه موسیقی از استاد شجریان پخش کن
برچسب گذاری کلمات(از راست به چپ) :
O O I-Artist B-Artist O O O O

توجه

۱. دیتاست مورد اشاره شامل ۵۰۰ نمونه آموزشی و ۱۲۲ نمونه آزمایشی است که در صورت استفاده از این دیتاست، همین ترتیب را برای آموزش و ارزیابی مدل‌های پایه و عمیق در نظر بگیرید و معیارهای ارزیابی مناسب برای هر قسمت را گزارش کنید.

۲. به جای این دیتاست می‌توانید از هر دیتاست مشابه دیگری (به زبان فارسی) که حجم مناسبی داشته باشد، استفاده کنید.

تشخیص زبان کد

همان‌طور که مستحضر هستید، یکی از کاربردهای بسیار مهم مدل‌های زبانی در حوزه برنامه‌نویسی است که به برنامه‌نویسان در توسعه برنامه‌ها یاری می‌رسانند. برای این موضوع مدل‌های زبانی تخصصی زیادی توسعه یافته‌اند. در این تمرین قصد داریم تا با استفاده از مدل‌های زبانی مختص به کد، دو مساله، یکی در سطح متن و یکی در سطح توکن را در این حوزه حل کنیم.

طبقه‌بندی متن

همان‌طور که مستحضر هستید، زبان‌های برنامه‌نویسی بسیار متنوعی در کاربردهای مختلف مورد استفاده قرار می‌گیرند. در این تمرین قصد داریم تا با ورودی گرفتن یک قطعه کد به یک زبان دلخواه، زبان مورد استفاده در نگارش آن کد را تشخیص دهیم. از خروجی این مدل می‌توان برای نمایش مناسب کد با استفاده از ظاهر مناسب در صفحات وب استفاده کرد. دقت بفرمایید که در این کار دو چالش اصلی وجود دارد. اول آنکه قطعه کد ورودی لزوما کامل نیست و می‌تواند تنها شامل یک خط کد باشد. دوم آنکه بسیاری از زبان‌های برنامه‌نویسی دستور نحوی مشابهی دارند به طوری که به صورت قانون‌ـ‌محور نمی‌توان گفت که کد نوشته شده به چه زبانی است. به عنوان مثال قطعه کد زیر را در نظر بگیرید. این قطعه کد از نظر نحوی با بسیاری از زبان‌ها از جمله جاوا، سی و پایتون سازگار است لیکن قرائن معنایی ما را به این جمع‌بندی می‌رساند که قطعه کد زیر در زبان پایتون نوشته شده است.

```
model = torch.sequential(
torch.Linear(10, 2),
torch.Sigmoid()
)

```

همچنین ممکن است که این قرائن معنایی در نحوه نام‌گذاری متغیرها نهفته باشد. به عنوان مثال در صورت استفاده از CamelCase کد مورد نظر احتمالا به زبان جاوا یا سی است ولی در صورت استفاده از snake_case زبان مورد نظر احتمالا پایتون است.

طبقه‌بندی توکن

پیرو مسئله مطروحه برای نمایش مناسب کد در صفحات وب، نیاز است تا نقش دستوری توکن‌های مختلف در قطعه کد مشخص شود. به عنوان مثال لازم است مشخص شود که آیا یک توکن اسم متغیر است یا اسم تابع و یا ادات تعریف. برای این منظور نیاز به یک مدل چند زبانه است که مستقل از زبان بتواند این کار را انجام دهد. حتی ممکن است که از مدل بخش قبلی به عنوان جزئی از این خط لوله استفاده شود.

برای انجام این تمرین می‌توانید از مجموعه دادگان و مدل‌های موجود در اینترنت استفاده کنید.

تشخیص موجودیت های پزشکی

طبقه‌بندی سند

برای این بخش لازم است از قسمت اول داده‌ها استفاده نمایید. شما باید مدلی را پیاده سازی کنید که در ورودی شرح حال بیمار را دریافت کند و در خروجی مشخص کند وضعیت برچسب مربوط به بیماری‌های زیر چیست. هر برچسب میتواند وضعیت met یا not met داشته باشد.

ABDOMINAL

CREATININE

MAJOR-DIABETES

طبقه‌بندی کلمه

برای این بخش لازم است از قسمت دوم داده‌ها استفاده نمایید. شما باید مدلی را پیاده سازی کنید که در ورودی مشخصات بیمار و نسخه‌ی ورودی را دریافت کند و سپس موجودیت‌های زیر را در خروجی تشخیص دهد.

Drug

Strength

Form

Dosage

Duration

Frequency

Route

ADE

Reason

دسترسی به داده‌ها

مجموعه داده‌های مورد استفاده برای این بخش، داده‌های n۲c۲ هستند که از طریق لینک زیر قابل دریافت است. توجه کنید این داده‌ها عمومی نیستند و تنها برای استفاده از این تکلیف از آن‌ها استفاده نمایید.

لینک داده‌ها

رمز داده‌ها: nlp۱۴۰۲

تشخیص احساس نظرات طاقچه

طبقه‌بندی متن

دیتاست داده‌شده شامل نظرات کاربران نرم افزار کتابخوانی طاقچه است که کاربران علاوه بر متن نظر خود، امتیازی از ۰ تا ۵ نیز ثبت کرده‌اند. هدف این تمرین آموزش دسته‌بندهایی برای انجام وظیفه تشخیص احساس نظر کاربران است. برای این منظور می‌توانید با افراز امتیازات کاربران به سه بخش مثبت، خنثی و منفی کار را آغاز کنید.
توجه کنید که انتخاب مرز امتیازات برای تفکیک آن‌ها به این سه بخش جزو ابرپارامترهای مسئله است که باید تنظیم شوند. همچنین انجام پیش پردازشات مرسوم روی دیتاست از جمله متعادل کردن آن از لحاظ تعداد نمونه‌های هر دسته ضروری است.

طبقه‌بندی کلمه

در این قسمت می‌خواهیم به کمک همان دیتاست قسمت قبل، کلمات متن هر نظر را به برچسب‌های مقابل تقسیم کنیم: نام نویسنده، نام مترجم، نام کتاب، نام انتشارات. به عنوان مثال:
متن نظر: من پارسال موقع عید کتاب جز از کل رو از انتشارات چشمه خوندم باید بگم که موقع خوندنش واقعا محو قلم استیو تولتز شده بودم .
برچسب‌گذاری کلمات(از راست به چپ):
I-Author B-Author O O O O O O O O O O B-Pub O O O I-Book I-Book B-Book O O O O O O O
توجه کنید که دیتاست داده شده شامل این برچسب‌های گفته‌شده نمی‌باشد. بنابراین می‌توانید رویکردهای متفاوتی برای حل این چالش پیدا کنید. به عنوان مثال می‌توان به کمک اطلاعات طاقچه، مجموعه داده‌ای از نام نویسندگان، مترجمان، کتاب‌ها و انتشارات جمع آوری کرده و سپس کلمات متن نظرات دیتاست داده‌شده را برچسب بزنید.

آموزش با داده کم

یکی از چالش‌های متداول در طراحی سیستم‌های پردازش متن کمبود داده است. فرض کنید برای تسک طبقه‌بندی احساس در متن داده کمی در اختیار داریم. مدل های مبتنی بر مبدل برای این کار عملکرد عالی دارند اما متاسفانه finetune کردن یک مدل مبدل بر روی داده کم منجر به overfitting می‌شود. در این ترک دو راه حل برای این موضوع را بررسی می‌کنیم.

۱. Cross-lingual Transfer Learning

یکی از قابلیت‌های پرکاربرد مدل‌های مبدل چند زبانه، finetune بر روی یک زبان و آزمون بر روی داده دیگر است. به عنوان مثال اگر از مدلی مثل XLM- RoBERTa برای آموزش روی داده طبقه‌بندی انگلیسی استفاده کنیم، مدل قابلیت دسته‌بندی بر روی سایر زبان‌ها را نیز تا حدی به دست می‌آورد.

۲. استفاده از Masked Language Modeling

مدل‌هایی مثل BERT و RoBERTa بر روی تسک MLM پیش‌آموزش داده می‌شوند. این مساله به ما اجازه می‌دهد که از دانش نهفته در مدل به طور مستقیم بتوان استفاده کرد و برای این کار تنها کافیست که یک قالب مناسب تسک مورد نظر طراحی شود.

```
input_sentence = "از غذا لذت بردم."
template = "{input_sentence} [MASK]"

```

در مثال بالا با قرار دادن جمله ورودی در این قالب ساده و دادن آن به مدل به یک توزیع احتمال برای توکن‌ها میرسیم که می‌توانند جای کلمه Mask شده قرار بگیرند. از این توزیع می‌توان به طور مستقیم برای طبقه‌بندی استفاده کرد. اگر احتمال دو کلمه مثل "مثبت" و "منفی" را در نظر بگیریم (که جزوی از vocabulary مدل هستند) و احتمال‌ها را مقایسه کنیم می‌توانیم هر کدام که بزرگتر بود را به عنوان خروجی در نظر بگیریم. طبیعتا در این روش طراحی قالب مناسب و انتخاب کلمه‌های هر کلاس اهمیت بالایی دارد.

در این ترک شما به تعداد کمی داده طبقه‌بندی احساس در زبان فارسی دسترسی دارید که برچسب آن‌ها باینری است.
برای بخش (الف) لازم است یک مجموعه‌داده انگلیسی برای طبقه‌بندی احساس پیدا کنید که مشابه دادگان این مساله باشد و مدل XLM- RoBERTa را بر روی آن آموزش دهید.
برای بخش (ب) با بهره‌گیری از مدل برت فارسی FaBert قالب مناسبی را طراحی کرده و از آن برای توسعه یک طبقه‌بندی استفاده کنید.
در زمان تحویل عملکرد این دو طبقه‌بندی بر روی داده‌ای مجزا بررسی خواهد شد.
برای دسترسی به داده می‌توانید از این لینک استفاده کنید.