دانشکده مهندسی کامپیوتر

دکتر بهروز مینایی

بهار ۱۴۰۱

تمرین سری اول

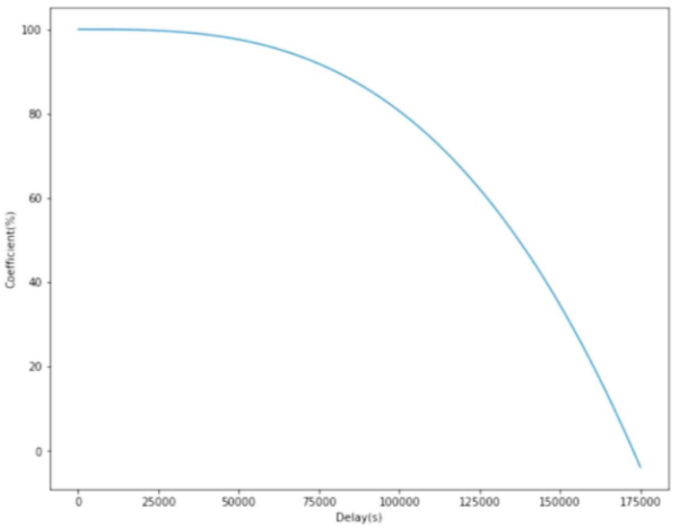
پردازش زبان و گفتار

رضا قهرمانی

تاریخ تحویل: یکشنبه ۱۴ فروردین ۱۴۰۱ساعت ۲۳:۵۹:۵۹

قوانین:

- سوالات این تمرین از مبحث «تحلیل مقدماتی متن» می‌باشد و برای پاسخ به سوالات آن نیاز به دانش نسبی درمورد این مبحث دارید.
- این تمرین شامل ۹ سوال می‌باشد. ۳ سوال تئوری و نوشتاری هستند و ۶ سوال عملی و شامل پیاده سازی هستند.
- درصورت وجود هرگونه سوال، در کلاس درس و یا در گروه تلگرامی درس بپرسید. (لطفا پی‌وی پیام ندهید.)
- هرگونه ایده گرفتن از تمرین دیگران و کدهای موجود در اینترنت که موجب تشابه غیرعادی و بالای کد شما با دیگری شود، تقلب محسوب می‌شود. درصورت مشاهده‌ی تقلب، نمره‌ی تمرین برای هر دو دانشجوی متخلف **صفر** منظور خواهد شد.
- لطفا برای انجام تمرین، زمان مناسب اختصاص دهید و انجام آن را به روزهای پایانی موکول نکنید. دقت کنید تمرین به هیچ عنوان تمدید نخواهد شد.
- پاسخ ارسالی شما باید علاوه بر کدهای مربوط به هر سوال، شامل یک گزارش در قالب یک فایل PDF باشد که محتوای گزارش مربوطه توضیحات تکمیلی شما درخصوص هر سوال و اسکرین‌شات از نتیجه اجرای کدهای شما باشد.
- تمامی فایل‌های موردنیاز برای تمرین را به صورت یک فایل ZIP با فرمت شماره دانشجویی_نام و نام خانوادگی_HW1 نام‌گذاری کرده و ارسال کنید. (برای مثال HW1_NameFamily_98000000)
- تاخیر در ارسال تمرین ها بر اساس نمودار زیر محاسبه خواهد شد. محور افقی نمودار، مقدار تاخیر به ثانیه و محور عمودی، ضریب اعمالی در نمره تمرین است.



سوالات تئوری:

۱. [ابزارهای پردازش زبان طبیعی] برای تسهیل فرآیند پردازش زبان طبیعی، ابزارهای و کتابخانه‌های زیادی وجود دارند. از شما میخواهیم به سه مورد از این ابزارها و کتابخانه اشاره کرده و توضیحی مختصر در مورد هر یک ارائه دهید.

۲. [عبارات منظم] عبارات منظم هر یک از موارد خواسته شده زیر را بنویسید:

آ. شماره تلفن که با 09، 00989، +989، ... که حداکثر ۱۶ رقم داشته باشد، را بپذیرد.

ب. تاریخ‌ها به فرمت dd-mm-yyyy را بپذیرد.

ج. URLها با پسوندهای ir و org را بپذیرد. (http:// میتواند بخشی از URL باشد.)

د. پلاک ماشین‌ها مثلا بصورت 54M235IR44 را بپذیرد.

۳. [Maximum Matching] الگوریتم Maximum Matching یکی از معروف‌ترین روش‌های word segmentation در پردازش متن است. در مورد روش این الگوریتم تحقیق کرده و توضیحی از آن ارائه دهید.

سوالات عملی:

۱. [عبارات منظم] با استفاده از regex برنامه ای بنویسید که تمامی کلمات زیر را پوشش دهد. برای تست عملکرد برنامه خود میتوانید این عبارات را در یک فایل تکست قرار داده و خروجی را با true/false نمایش دهید.

- William R. Breakey M.D.
- Pamela J. Fischer M.D.
- Leighton E. Cluff M.D.
- James S. Thompson, M.D.
- C.M. Franklin, M.D.
- Atul Gawande, M.D.
- Dr. Talcott
- Dr. J. Gordon Melton
- Dr. Etienne-Emile Baulieu

- Dr. Karl Thomae
- Dr. Alan D. Lourie
- Dr. Xiaotong Fei
- Doctor Dre
- Doctor Dolittle
- Doctor William Archibald Spooner

۲. [آشنایی با NLTK] NLTK یکی از معروف‌ترین ابزارهای پردازش متن است. برای نصب آن کافی است دستور pip install nltk را اجرا کنید. به تقسیم بندی متون به قسمت‌های کوچکتر با معنی مثل کلمه، جمله و توکن، segmentation می‌گویند. حال به کمک این ابزار، خروجی متدهای word_tokenize و sent_tokenize بر روی یک فایل متنی دلخواه را نمایش دهید.

۳. [نرمالسازی] قبل از پردازش متن بهتر است کلمات و کاراکترهای اضافی و به دردنخور حذف شوند تا از پردازش اطلاعات بیشتر جلوگیری شود. به این عمل پاک سازی می‌گویند. همچنین باید کلمات را به فرم نرمال آنها تبدیل کرد تا پردازش متن راحت تر شود که به آن اصطلاحا نرمال سازی می‌گویند. به عنوان مثال اگر در متنی عبارت love با تعداد o های بیشتری نوشته شده بود(یعنی مثلا looooove)، فرم نرمال آن باید به صورت love نوشته شود. حال از شما میخواهیم به کمک ابزار NLTK و عبارات منظم، برنامه ای بنویسید که کلمه ای را به عنوان ورودی بگیرد و فرم نرمال آن را در خروجی نمایش دهد. مثلا اگر کلمه corrrect به عنوان ورودی داده شود، فرم نرمال آن یعنی correct باید در خروجی نشان داده شود. راهنمایی:

Loooove => (loo)(o)o(ve) => (lo)(o)o(ve) => (l)(o)ove => love

۴. [Word Tokenization] مراحل زیر را برای این تمرین انجام دهید:

آ. در این تمرین شما با فایل‌های متنی و داده‌های متنی نمونه زیر کار می‌کنید. برای نوشته‌های نمونه، آنها را بدون هیچگونه تغییر استفاده کنید. ابتدا فایل‌های متنی را باز کرده و متغیرهای مربوط به متن نمونه را تعریف کنید.

متن نمونه انگلیسی	AlbertEinstein.txt
متن نمونه فارسی	Shahnameh.txt
متن کوتاه انگلیسی	ShortSampleEnglish.txt
متن کوتاه فارسی	ShortSamplePersian.txt

ب. در ادامه میخواهیم متون نمونه را به کلمات آنها تجزیه کنیم و تعداد Tokenها و تعداد Typeهای آنها را بدست آوریم. روش‌های زیادی برای تجزیه متون به کلمات در NLTK وجود دارد. ما در این تمرین میخواهیم TreebankWordTokenizer، RegexpTokenizer، WhitespaceTokenizer و WordPunctTokenizer را مورد بررسی قرار دهیم.
پ. با استفاده از TreebankWordTokenizer متن کوتاه انگلیسی، متن کوتاه فارسی، متن نمونه فارسی و متن نمونه انگلیسی را به کلمات تجزیه کنید و تعداد Tokenها و Typeهای آنها را بدست آورید.
ت. با استفاده از RegexpTokenizer، کلمات مربوط به متن کوتاه انگلیسی و متن کوتاه فارسی و اعداد مربوط به متن نمونه انگلیسی را استخراج کنید.
ث. با استفاده از WhitespaceTokenizer کلمات مربوط به متن کوتاه انگلیسی را تجزیه کنید. نحوه عملکرد این روش چگونه است؟ اگر بخواهیم با استفاده از RegexpTokenizer به نتیجه مشابه برسیم پیشنهاد شما چیست؟
ج. نحوه عملکرد WordPunctTokenizer را با استفاده از متن کوتاه انگلیسی بررسی کنید.

۵. [Stemming] در ادامه میخواهیم Stemming را بر روی برخی کلمات اعمال کنیم.
آ. نحوه استفاده از PorterStemmer و LancasterStemmer را در NLTK بررسی کنید.
ب. از لیست کلماتی که از متن نمونه انگلیسی با استفاده از TreebankWordTokenizer استخراج کرده‌اید، اندیس های ۲و۱۰و۱۸و۱۹و۲۱و۲۲و۴۲ را یک بار با اعمال PorterStemmer و یکبار با اعمال LancasterStemmer نمایش دهید و با یکدیگر مقایسه کنید.
پ. یکی دیگر از اعمالی که میتوان روی کلمات انجام داد Lemmatization است. با استفاده از WordNetLemmatizer کلمات زیر را به حالت نگارشی اولیه آنها برگردانید.
Waves, fishing, rocks, was, corpora, better, ate, broken
ت. آیا با استفاده از متد lemmatize با ورودی های پیشفرض، برای همه این کلمات پاسخ درست را برمیگرداند؟ اگر پاسخ خیر است، پیشنهاد شما برای اینکه با این روش بتوان برای همه این کلمات نتایج صحیح گرفت چیست؟

۶. [پیش پردازش داده‌ها] در این بخش قصد داریم تا با اصول پیش پردازش توییت‌های استخراج شده از توییتر آشنا شویم. داده های موردنظر در فایل tweets.csv موجود میباشد. مراحل پیش پردازش در زیر آورده شده است.

مرحله ۱ – حذف فضاهای خالی (White Space) اضافه در میان کلمات یک توییت و قرار دادن یک فضای خالی در بین هر دو کلمه.

مرحله ۲ – تبدیل حروف کلیه کلمات متن به حروف کوچک.

مرحله ۳ – حذف کلیه Handleها از داخل متن(Handleها همان username@ ها میباشد.).

مرحله ۴ – حذف علائم نگارشی، اعداد و کاراکترهای خاص از داخل متن توییت‌ها.

مرحله ۵ - با استفاده از یک روش Tokenization در NLTK، عملیات Tokenization بر روی هر توییت را انجام دهید. توضیح مختصری در مورد روش مورد استفاده ارائه دهید.

مرحله ۶ – ابتدا توضیح مختصری در ارتباط با مفهوم و دلیل استفاده از StopWordsها در پیش پردازش متون ارائه نمایید و سپس آنها را از درون متون حذف نمایید.

مرحله ۷ – حذف کلمات با طول کمتر از ۳ از داخل متن. در مورد علت انجام این کار توضیحاتی ارائه دهید.

مرحله ۸ – پس از یافتن دو روش Stemming در NLTK و ارائه مقایسه کوتاهی از آنها، عملیات Stemming را بر روی توکن‌های بدست آمده از هر توییت، با استفاده از PorterStemmer انجام دهید و سپس مجدد هر توییت را با استفاده از توکن های آن بازسازی کنید.

پس از اعمال مراحل پیش پردازش بر روی داده ها، پاسخ سوالات زیر را همراه خروجی هر مرحله از پیش پردازش در فایل توضیحات خود بیاورید.

آ. پرتکرارترین کلمات در این مجموعه داده چیست؟ آنها را در قالب یک Wordcloud نمایش دهید.

ب. ترندهای موجود در این مجموعه داده چیست؟ با استفاده از نمودار ده ترند اول را نمایش دهید. (راهنمایی: برای یافتن ترندها شما نیاز به شناسایی هشتگ‌های موجود در توییت ها دارید.)

جهت تحویل این بخش، در فایل توضیحات علاوه بر توضیحات کد هر مرحله بایستی ۳ نمونه از هر توییت پیش و پس از اعمال هر مرحله پیش پردازش آورده شود و در مورد تاثیر هر مرحله توضیحاتی ارائه دهید. در نهایت بایستی در یک فایل csv کلیه توییت ها قبل و پس از پیش پردازش در کنار هم قرار داده شود و به عنوان نتیجه نهایی در فایل ارسالی آورده شوند.

موفق باشید