تمرین دوم درس پردازش زبان‌های طبیعی

استاد درس: دکتر حمیدرضا برادران کاشانی

دستیاران آموزشی: آیین کوپایی – یاسین فخار

تاریخ بارگذاری تمرین: ۱۴۰۲/۰۸/۲۳

تاریخ تحویل تمرین: ۱۴۰۲/۰۹/۰۸

دانشگاه اصفهان

تکمیل خودکار

هدف از تمرین حاضر ساخت یک سیستم تکمیل خودکار است. سیستم‌های تکمیل خودکار به طور معمول در موتورهای جستجو، ویرایشگرهای متن، برنامه‌های پیام‌رسان و ... برای بهبود تجربه کاربری استفاده می‌شوند. این سیستم‌ها معمولا با تجزیه و تحلیل متن ورودی، لیستی از کلمات احتمالی بعدی را پیشنهاد می‌دهند. در این تمرین از مجموعه‌ای از نظرات مرتبط با محصولات مختلف در سایت دیجی کالا برای مدل سازی زبانی و ساخت سیستم تکمیل خودکار استفاده می‌شود.

۱- پیش پردازش

مجموعه داده مورد نظر در فایل digikala_comment.csv موجود است. در این تمرین تنها با ستون comment از این مجموعه داده کار می‌کنیم. با توجه به اینکه مجموعه داده مورد استفاده به زبان فارسی است، می‌توانید از کتابخانه hazm استفاده کنید. مراحل پیش پردازش زیر را برای این مجموعه داده انجام دهید.

۱-۱- هر کامنت را به جملات آن تجزیه کنید.

۱-۲- فضاهای خالی اضافه را حذف کنید.

۱-۳- متن را توکن بندی کنید و stopword ها، علائم نگارشی، تگ‌های html و ایموجی‌ها را حذف کنید. سپس متن را normalize کنید.

۱-۴- اعداد را با توکن <NUM> و URL ها را با توکن <URL> جایگزین کنید.

۲- ساخت مدل زبانی

برای مجموعه داده پیش پردازش شده مراحل زیر را انجام دهید:

۲-۱- **ساخت n-gram ها:** تابعی بنویسید که لیست unigram، bigram و trigram ها را از درون مجموعه داده استخراج کند و تعداد تکرار هر n-gram را محاسبه کند. این لیست ها را نمایش دهید و سپس ۱۰ تا از پر تکرار ترین unigram وbigram و trigram ها را گزارش دهید.

۲-۲- تابعی برای محاسبه احتمال n-gram ها بنویسید. درمحاسبه احتمالات از Laplace smoothing برای unigram ها و از Good-Turing smoothing برای دیگر n-gram ها استفاده کنید. می‌توانید برای محاسبه Laplace smoothing از nltk.LaplaceProbDist و برای محاسبه Good-Turing smoothing از nltk.SimpleGoodTuringProbDist استفاده کنید. توضیح دهید چرا روش Laplace smoothing روش خوبی برای n-gram ها نیست.

۲-۳- تابعی برای محاسبه perplexity هریک مدل‌های ایجاد شده (unigram, bigram, trigram) بنویسید. perplexity مدل‌ها را برای جملات زیر محاسبه کرده و گزارش دهید.

- بوی تند ولی خوشبو داره
- بلوتوثش کار نمی کنه حالا تا بدستم رسیده باید برشگردونم
- بلند گو هاش بیس بالا و صدای زیادی بمی داره که بعد از مدتی باعث خسته شدن مغز آدم میشه
- لطفا کالای مورد نظر رو در پیشنهاد ویژه قرار بدید

۲-۴- **پیش‌بینی کلمات**

در این مرحله می‌خواهیم با استفاده از مدل‌های ایجاد شده، کلمات جدید را با استفاده از دنباله‌ای از کلمات ورودی پیش‌بینی کنیم. برای این کار تابعی طراحی کنید که مدل و دنباله ای از کلمات را به عنوان ورودی دریافت کند و کلمات بعدی را به عنوان خروجی برگرداند و جمله را تا رسیدن به طول ۱۵ تکمیل کند. در واقع جمله خروجی این تابع باید دارای طولی به اندازه ۱۵ باشد که شامل کلمات ورودی و کلمات پیش بینی شده است. با استفاده از این تابع، عبارات زیر را تکمیل کنید. (در نهایت شما باید ۱۵ جمله داشته باشید، به ازای هر مدل ۵ جمله.)

- صرفه جویی در پودر ماشین
- یکی از چراغهای وضعیت
- گوشی سامسونگ
- رنگ قرمز کفش
- یک تن ماهی خوب

۲-۵- perplexity مدل ها را برای جملات ساخته شده در مرحله قبل بدست آورده و گزارش دهید.

۳- برچسب گذاری کلمات

۳-۱- با استفاده از hazm عمل POS Tagging را روی مجموعه داده پیش پردازش شده اعمال کنید و تگ هر توکن را در خروجی نمایش دهید.

۳-۲- تعداد رخدادهای هر تگ POS را در کل مجموعه داده به دست آورید و گزارش دهید.

۳-۳- اسم‌ها را از جملات دارای تگ POS استخراج کنید و ۱۵ اسم پر تکرار اول را همراه با تعداد تکرار آن‌ها گزارش دهید.

۴- مقاله خوانی

مقاله‌ی "Efficient Neural Query Auto Completion [1]" را مورد مطالعه قرار دهید و به پرسش‌های زیر پاسخ دهید:

۴-۱- به طور کلی تکمیل خودکار کوئری[2] یا QAC شامل چه مراحلی است؟

۴-۲- چالش‌های اصلی در سیستم‌های QAC چیست؟

۴-۳- فواید مدل‌سازی زبان عصبی[3] چیست؟

۴-۴- رویکرد MCG [4] که در این مقاله ارائه شده است را توضیح دهید.

۴-۵- اجزای تشکیل دهنده‌ی سیستم رتبه‌بندی کاندیداها[5] که در این مقاله ارائه شده است را نام ببرید.

نکات تحویل

۱- پاسخ خود را در پوشه ای به اسم NLP_NAME_FAMILY_HW2 و در قالب zip بارگذاری نمایید.

۲- این پوشه باید حاوی موارد زیر باشد:

- کد نوشته شده در قالب یک فایل jupyter notebook یا py.
- فایل گزارش در قالب یک فایل PDF

۳- لازم به ذکر است که رعایت قوانین نگارشی حائز اهمیت است.

۴- پاسخ خود را به ایمیل aein.koopaei@gmail.com ارسال کنید.

---

[1] Wang, S., Guo, W., Gao, H., & Long, B. (2020, October). Efficient neural query auto completion. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management (pp. 2797-2804).
[2] Query Auto Completion
[3] Neural Language Modeling
[4] Maximum Context Generation
[5] Candidate Ranking