پیاده سازی مدل احتمالاتی PLSA

مجتبی روحانیان

برنامه حاضر یک پیاده سازی از مدل زبانی آماری Probabilistic Latent Semantic Analysis (PLSA) در زبان پایتون می باشد.

در قسمت documents، 15 عدد فایل متنی فارسی وجود دارد. این فایل ها از متونی برداشت شده اند که از لحاظ موضوعی در یکی از سه دسته "تشیع"، "رژیم لاغری 15 روزه کانادایی" و یا "بازی شطرنج" هستند. بر این اساس، به منظور خوشه بندی[1] پیکره، مقدار عددی K برابر با 3 منظور شده است. برای یافتن خودکار و مستقل مقدار عددی k، می توان از روش هایی مانند cross-validation استفاده نمود که در اینجا به دلیل ذیق وقت از آن صرف نظر شده است و برای نسخه های بعدی و کامل تر نرم افزار مد نظر قرار خواهد گرفت.

روند کار به این شکل است که ابتدا سندها خوانده می شوند و به یک نمایش برداری تبدیل می گردند. پیش از آن متن هر سند از علائم نگارشی پیراسته می شود و سپس ایست واژه ها[2] نیز از متن حذف می شوند. این به این منظور است که واژه های پرکاربردی که بار معنایی به سند اضافه نمی کنند در نظر گرفته نشوند. لیست واژه های استفاده شده، از مجموعه تقوا و دیگران(2003) برداشت شده است.

بعد از حذف ایست واژه ها، با استفاده از نرمالساز متنی، واژه ها از لحاظ نگارشی یکسان می شوند و بعد با استفاده از قطعه بند[3] جداسازی شده و در یک لیست قرار می گیرند. برای کار قطعه بندی و نرمالسازی، از کتابخانه هضم برای زبان فارسی استفاده شده است[4]. لازم به ذکر است که نخست از ریشه یاب[5] این کتابخانه نیز استفاده شد، اما نتایج کار مطلوب نبود و نتایج را بهبهود نبخشید. با این حال به نظر می رسد استفاده از یک ریشه یاب قوی برای نسخه های بعدی این نرم افزار ضروری باشد.

پس ازآنکه نمایش برداری از سندها به دست آمد، هر کدام از آنها در یک لیست مجزا ذخیره شده، مجموعه همه این نمایش های برداری در یک متغیر به نام documents به صورت لیست ذخیره می گردد.

---

[1] clustering
[2] stop words
[3] tokenizer
[4] https://pypi.python.org/pypi/hazm/0.4
[5] stemmer

از روی این لیست، مجموعه همه واژه های موجود در اسناد استخراج می گردد و در متغیر words قرار می گیرد. در مرحله بعد، با داشتن این داده ها، ماتریس bag of words تشکیل می گردد که سطرهای آن نشانگر سندها و ستون های آن نمایشگر واژه ها است.

پس از آن به مرحله پردازش PLSA می رسیم. تابع plsa به گونه ای پیاده سازی شده است که چهار متغیر به عنوان ورودی قبول می کند. اولی ماتریس bag of words، بعدی مقدار عددی K که در اینجا 3 فرض شده است، و دو مقدار دیگر مربوط به تعداد دفعاتی است که می خواهیم فرایند EM اتفاق بیفتد و همینظور مقدار ϵ که اگر فاصله مقادیر متوالی از آن کمتر شود نشان می دهد که متغیرهای مدل به یک مقدار واحد میل کرده اند و در آن صورت پردازش پایان می یابد.

خروجی برنامه نشاندهنده محتمل ترین واژه ها در هر دسته است. یعنی مقادیر بالای $p(w|z)$ در هر دسته. به عبارت دیگر، پس از آنکه پردازش PLSA به اتمام رسید، در هر دسته، 20 کلمه ای که بیش از همه نماینده آن دسته محسوب می شوند در خروجی به نمایش در می آیند. بنا بر این انتظار اینست که پس از خوشه بندی، کلمه های محتمل در هر دسته، از لحاظ موضوعی مرتبط با هم و متفاوت با دسته های دیگر باشند.

نتایج اجرای برنامه در فایل متنی test_results.txt همراه با کدمتن برنامه گنجانده شده است. همانطور که مشاهده می شود، نتایج با انتظارات همخوان است.

در حال حاضر میزان دفعات پردازش برابر 1000 و مقدار $\epsilon = 0.0001$ فرض شده است که طبعا تغییر و کم و زیاد کردن این اعداد روی خروجی تاثیر می گذارد.

هدف من اینست که در ادامه کار، مدل زبانی برنامه را به bigram plsa تغییر دهم و از آن برای ساخت یک غلطیاب املائی برای فارسی استفاده کنم. سپس با مقایسه نتایج آن با برنامه غلطیاب املائی که سال گذشته و با روش های unigram و bigram نوشته بودم، خروجی کار را در قالب یک مقاله ارزیابی نمایم. برای این کار البته باید از داده های واقعی و پیکره های بزرگ استفاده شود.

منابع

Kazem Taghva, Russell Beckley, Mohammad Sadeh(2003) A List of Farsi Stop words, ISRI Technical Report No. 2003-01 Information Science Research Institute University of Nevada, Las Vegas

Alireza Nourian, Mojtaba Khallash, Mohsen Imani *hazm 0.4 Python library for digesting Persian text* open source project available at: https://pypi.python.org/pypi/hazm/0.4