به نام خدا

دانشگاه تهران

دانشکدگان فنی

دانشکده مهندسی برق و کامپیوتر

درس بازیابی هوشمند اطلاعات

پاسخ بخش تئوری تمرین ۱

نام و نام خانودگی: سهیل حاجیان منش

شماره دانشجویی: 810100119

مهر ماه ۱۴۰۳

فهرست

اظهارنامه

تأیید می کنم که از LLM ها مطابق با دستورالعمل های بارگذاری شده در سامانه Elearn درس به طورمسئولانه استفاده کرده ام. تمام اجزای کار خود را درک میکنم و آماده بحث شفاهی درباره آنها هستم.

پاسخ سوال اول

پاسخ بخش اول

W = "یادگیری"

- Jelinek-Mercer Smoothing (λ = 0.6)

$$p(w|d) = (1-\lambda)\frac{c(w,d)}{|d|} + \lambda\, p(w|REF)$$

$$p(W|D_3) = (0.4) * \frac{1}{13} + 0.6 * 0.003 \approx 0.032$$

- Dirichlet Prior Smoothing (μ=2000)

$$p(w|d) = \frac{|d|}{|d|+\mu}\frac{c(w,d)}{|d|} + \frac{\mu}{|d|+\mu}p(w|REF)$$

$$p(W|D_3) = \frac{13}{2013} * \frac{1}{13} + \frac{2000}{2013} * 0.003 \approx 0.0034$$

- Absolute Discounting Smoothing (δ=0.5)

$$p(w|d) = \frac{\max(c(w,d)-\delta, 0) + \delta|d|_u\, p(w|REF)}{|d|}$$

$$p(W|D_3) = \frac{\max(1-0.5, 0) + 0.5 * 12 * 0.003}{13} \approx 0.5$$

- Additive (Laplace) Smoothing (α=1)

$$p(w|d) = \frac{c(w,d)+\alpha}{|d|+|V|}$$

$$p(W|D_3) = \frac{1+1}{13+10000} \approx 0.0001$$

روش Absolute Discounting Smoothing احتمال بیشتری می دهد چون p(W|D) در صورت دیده شدن در سند احتمال زیادی دارد : ۰.۰۷ = ۱/۱۳ = c(w,d) / ۱۳ بنابر این روش هایی که باعث شوند وزن بیشتری به این احتمال داده شود نسبت به P(W|REF) = ۰.۰۰۳ احتمال بالاتری می دهند. روش اول و دوم احتمال بیشتری به احتمال مدل زبانی مرجع دادند. روش چهارم هم به دلیل مقدار زیادی |V| احتمال حضور کلمات به شدت کاهش می یابد و تاحدودی شبیه هم می شوند (احتمال کلمه ای که حضور دارد با کلمه ای که حضور ندارد تفاوت چندانی نمی کند.)

پاسخ بخش دوم

$$\log p(q|d) = \sum_i \log p(w_i|d) \qquad where, \quad q = w_1 w_2 \dots w_n$$

W1 = "یادگیری"

W2 = "بازیابی"

μ=2000

$$\log p(w_1|D_3) = \log \frac{13}{2013} * \frac{1}{13} + \frac{2000}{2013} * 0.003 \approx \log\ 0.0034 \approx -2.525$$

$$\log p(w_2|D_3) = \log \frac{13}{2013} * \frac{1}{13} + \frac{2000}{2013} * 0.002\ \approx \log\ 0.0024 \approx -2.701$$

$$\log(q|D_3) = \log(w_1|D_3) + \log(w_2|D_3) \approx -2.525 - 2.701 = -5.226$$

$$\log p(w_1|D_1) = \log \frac{8}{2008} * \frac{1}{8} + \frac{2000}{2008} * 0.003 \approx \log\ 0.0034 \approx -2.457$$

$$\log p(w_2|D_1) = \log \frac{8}{2008} * \frac{1}{8} + \frac{2000}{2008} * 0.002\ \approx \log\ 0.0024 \approx -2.7$$

$$\log(q|D_1) = \log(w_1|D_1) + \log(w_2|D_1) \approx -2.457 - 2.7 = -5.157$$

چون در فرمول جز $|d|$ در $\frac{c(w,d)}{|d|}$ نقش Document Length Normalization را ایفا می کند. بعنوان مثال برای کلمه w1 در سند D1 این مقدار برابر 0.125 و در سند D3 برابر 0.07 می باشد. با وجود اینکه این کلمه در هر دو سند یک بار تکرار شده است.

پاسخ بخش سوم (امتیازی)

- **از انجا که تعداد کلمات منحصر بفرد تنها در روش AD تاثیرگذار است (فرمول روش را در بخش اول سوال نوشته ام) پس با پارامتر γ می توان آن را کنترل کرد. با توجه به فرمول با افزایش $|d|_u$ مقدار $P_{AD}(w|d)$ زیاد می شود، و با پارامتر γ می توان این تاثیرگذاری را کنترل کرد.**
- **در مدل JM وزن دهی بین مدل سند و مدل زبانی مرجع یکسان است. پس هر چه طول سند بیشتر باشد، چون فراوانی کلمات در سند قابل اعتماد ترند (c(w,d)) و بخش ۱-λ فرمول بهتر عمل می کند پس اهمیت روش JM و پارامتر α بیشتر است.**

- در مدل Dir مقدار μ کنترل می کند که چقدر احتمال تحت تاثیر مدل زبانی مرجع قرار بگیرد، هر چه طول سند کوتاه تر باشد، تاثیر REF بیشتر می شود پس احتمال ها معنادار تر می شود. پس در اینجا اهمیت روش Dir و پارامتر β بیشتر است.
- می توان با استفاده از روش K-Fold Cross Validation وزن های بهینه را برای این چهار پارامتر پیدا کرد.
- هم چنین می توانیم وزن ها به صورت تابعی از ویژگی های سند تعیین کنیم:

بعنوان مثال :

  - هرچه سند بلندتر بود وزن Jm بیشتر باشد.
  - هر چه سند کوتاه تر بود، وزن Dir بیشتر باشد.
  - هر چه تنوع واژگان بیشتر بود، وزن AD بیشتر باشد.

پاسخ سوال دوم

پاسخ بخش اول

$$P(w_i|\theta_F)^0 = \frac{1}{3}$$

E-Step :

$$P(z_i = 1|w_i) = \frac{\lambda P(w_i|C)}{\lambda P(w_i|C) + (1-\lambda)P(w_i|\theta_F)}$$

$$P(z_i = 1|\text{الگوریتم}) = \frac{0.3 * 0.001}{0.3 * 0.001 + 0.7 * P(w_i|\theta_F)} = \frac{0.0003}{0.0003 + 0.23333} \approx 0.00128$$

$$P(z_i = 1|\text{یادگیری}) = \frac{0.3 * 0.002}{0.3 * 0.002 + 0.7 * P(w_i|\theta_F)} = \frac{0.0006}{0.0006 + 0.23333} \approx 0.00257$$

$$P(z_i = 1|\text{عمیق}) = \frac{0.3 * 0.0015}{0.3 * 0.0015 + 0.7 * P(w_i|\theta_F)} = \frac{0.00045}{0.00045 + 0.23333} \approx 0.00193$$

M-Step:

- C(الگوریتم,F) = 2 (D_1 و D_3)
- C(یادگیری,F)= 3 (D_1 ،D_2 ،D_4)
- C(عمیق,F)= 4 (D_1 ،D_2 ،D_4 ،D_5)

$$P^{new}(w_i|\theta_F) = \frac{c(w_i, F)(1 - p^{(n)}(z_i = 1|w_i))}{\sum_{w_j \in vocabulary} c(w_j, F)\,(1 - p^{(n)}(z_j = 1|w_j))}$$

$$P^1(\text{الگوریتم}|\theta_F) = \frac{2 * (1 - 0.00128)}{1.99744 + 2.99229 + 3.99228} \approx \frac{1.99744}{8.98201} \approx 0.22$$

$$P^1(\text{یادگیری}|\theta_F) = \frac{3 * (1 - 0.00257)}{.99744 + 2.99229 + 3.99228} \approx \frac{2.99229}{8.98201} \approx 0.33$$

$$P^1(\text{عمیق}|\theta_F) = \frac{4 * (1 - 0.00193)}{.99744 + 2.99229 + 3.99228} \approx \frac{3.99228}{8.98201} \approx 0.44$$

پاسخ بخش دوم

- **کدام کلمات وزن بیشتری میگیرند و چرا ؟**

طبق نتایج به ترتیب : 1- عمیق، 2- یادگیری و 3- الگوریتم بیشترین وزن را دارند.

چون p(z=1|w_i) تقریبا به هم نزدیک و خیلی کوچک است. پس نقش P(z) در تمایز بین کلمات کم است و اسناد فیدبک اهمیت بیشتری دارند. یعنی تفاوت اصلی رو c(w,F) می آید:

عمیق در بیشترین تعداد سند ظاهر شده، بعد یادگیری و بعد الگوریتم.

در نتیجه سهم بیشتری از وزن واژه ها در مدل بازخوردی به واژه‌هایی داده می‌شود که در اسناد فیدبک بیشتر تکرار شده‌اند.

- **اگر λ افزایش یابد، تأثیر آن چیست؟**

با افزایش:λ

- با توجه به فرمول E-step وزن مدل پس‌زمینه بیشتر می‌شود.
- در نتیجه P(z=1|w) بزرگ‌تر می‌شود و P(z=0|w) کوچک‌تر.
- یعنی در M-step وزن کمتری به θ_F می‌رسد.

- اثرش این است که مدل بازخورد کمتر از اسناد فیدبک می‌گیرد و توزیع θ_F به توزیع کلی مجموعه (C) نزدیک‌تر می‌شود.

- **آیا D_3 باعث نویز می شود؟ چگونه می توان آن را شناسایی کرد ؟**

- تنها واژهٔ مشترک این سند با پرس‌وجو الگوریتم است که نسبت به بقیه واژه های جست و جو عمومی تر می باشد.
- از نظر معنایی مشخص است که سند D3 دربارهٔ الگوریتم‌های ژنتیک» صحبت می کند و ارتباطی به الگوریتم های یادگیری عمیق ندارد.

پس سند D3 تا نویزی محسوب می شود و اگر در Query Expansion از این سند ترم جدید گرفته شود، مطلوب نیست.

چطور می‌شود آن را تشخیص داد؟

امتیاز اولیه D3 از باقی سند ها پایین تر است.می توان یک Threshold بر روی امتیاز اسناد گذاشت و یا اسناد را به ترتیب امتیاز مرتب کرد و k سند اول را بعنوان اسناد فیدبک انتخاب کرد.

سوال سوم

پاسخ بخش اول

توضیح شباهت توزیعی :

در مدل های مبتنی بر شباهت توزیعی، هر سند به عنوان یک توزیع احتمال بر روی واژگان مدل سازی می شود.

- در روش Query-Likelihood احتمال وجود واژه های پرس و جو در توزیع احتمالی هر سند حساب می شود تا مشخص شود پرس و جو به مدل زبانی کدام سند نزدیک تر است.
- در روش KL-Divergence توزیع احتمالی بر روی واژگان پرس و جو هم حساب می شود و فاصله آماری (شباهت توزیعی) بین توزیع سند و توزیع پرس و جو محاسبه می شود و اسناد با توجه به میزان شباهت توزیعی شان با پرس و جو مرتب می شوند. (D(p || q))

دلیل اینکه گاهی اوقات مدل های شباهت توزیعی برای اسناد بلندتر و کلی تر بهتر عمل می کنند :

- اسناد طولانی واژه‌های بیشتری دارند، بنابراین احتمال واژه‌های پرس و جو در توزیع مدل زبانی آن‌ها بالاتر است و در مدل Query Likelihood امتیاز بیشتری می‌گیرند.
- اسناد کلی و طولانی توزیع زبانی یکنواخت‌تری دارند و ممکن است خیلی از کلمات پرس و جو با احتمالی بزرگتر از صفر در مدل زبانی سند باشند؛ همین باعث کاهش KL-Divergence و افزایش احتمال P(q|d) می‌شود، حتی بدون اینکه لزوما ارتباط معنایی دقیقی وجود داشته باشد.

بطور کلی این مدل‌ها فقط شباهت آماری واژه‌ها را می‌سنجند، نه رابطهٔ معنایی. در نتیجه اسنادی که چند واژهٔ پرسجو را داشته باشند حتی اگر بی‌ربط باشند امتیاز بهتری از اسناد کوتاه اما کاملاً مرتبط می‌گیرن که همانطور که در سوال هم گفته شده ممکن است این رفتار با شهود انسانی یکسان نباشد.

پاسخ بخش دوم

نقش هموارسازی در این مدل ها

در مدل های زبانی احتمال واژه ها در سند محاسبه می شود : $p(w|d)$

تا در انتها از ضرب احتمالات $p(q|d)$ محاسبه شود.

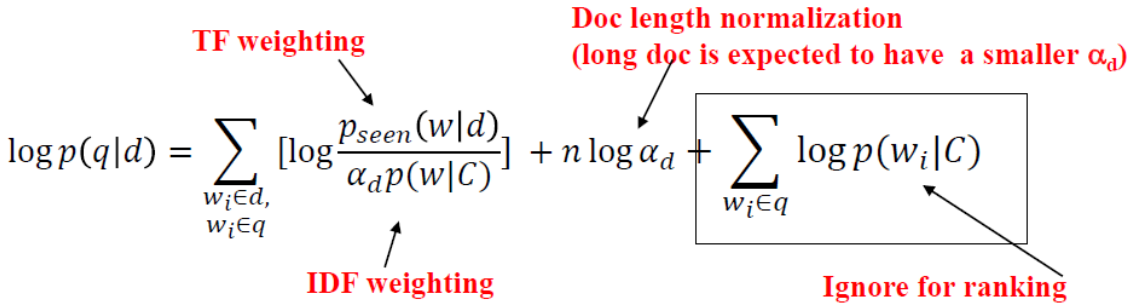
از آنجا که ممکن است بعضی واژه ها در سند اصلا ظاهر نشده باشند : $p(w|d) = 0$

در نتیجه $p(q|d)$ برای سند 0 می شود یعنی ممکن است سند خیلی مرتبط با پرس و جو باشد اما چون یکی از کلمات پرس و جو در آن نیست احتمال 0 بگیرد.

تمام توابع هموارسازی تلاش میکنند :

- از احتمال کلماتی که در سند ظاهر شده اند مقداری کاهش دهند.
- مقداری کاهش داده شده و اضافی را به واژه های دیده نشده اختصاص دهند، تا مقدار غیر صفر داشته باشند.

طبق فرمول query likelihood هموارسازی بطور غیرمستقیم نقش IDF weighting را هم ایفا می کنند :



پس می توان گفت هموارسازی دو نقش ایفا می کند :

1. تخمین مدل زبانی را وقتی طول سند کوتاه است بهتر کند.
2. بین واژه های جست و جو تمایز قائل می شود و واژه های پرتکرار و عمومی جریمه می شوند.

وقتی یک سند کوتاه یا خیلی تخصصی باشد، چند واژهٔ کلیدی دارد که هویت اصلی آن را می‌سازند. اما در مدل‌های زبانی، این سند با مدل زبانی مرجع (که بسیار عمومی‌تر است) ترکیب می‌شود. اگر وزن مدل مرجع زیاد باشد، احتمال واژه‌های تخصصی سند به اندازهٔ قبل بالا نمی‌ماند.
بعنوان مثال به حالت زیر توجه کنید که به کمک ChatGPT نوشته ام :

فرض کن پرس و جو «عوارض جانبی داروی Pembrolizumab در درمان سرطان ملانوما» باشد.

یک سند تخصصی کوتاه پیدا می‌شود که دقیقاً همین را توضیح می‌دهد و واژه‌های **Pembrolizumab** و **melanoma** چند بار در آن آمده‌اند.

از آن طرف، یک مقاله عمومی ۲۰ صفحه‌ای دربارهٔ سرطان هم وجود دارد که فقط یک بار همین دو واژه را ذکر کرده است.

حالا اگر مدل زبانی سند تخصصی را زیاد هموارسازی کند، دیگر احتمال این واژه‌ها در سند تخصصی زیاد نمی‌ماند. سند تخصصی از نظر مدل شبیه یک سند معمولی می‌شود. چون احتمال این واژه ها در مدل زبانی مرجع خیلی پایین است.

پاسخ بخش سوم

Query Drift

در مدل‌های مبتنی بر شباهت توزیعی مانند،همانطور که در قسمت های قبل گفته شد، مدل زبانی جست و جو نقش مهمی در تعیین شباهت اسناد دارد.Query Drift زمانی رخ می‌دهد که توزیع زبانی پرس‌وجو به دلیل اضافه‌شدن واژه‌های جدید، مخصوصاً واژه‌های عمومی و پرتکرار از معنای اصلی خود فاصله بگیرد. در نتیجه، مدل دیگر تمرکزش روی واژه‌های اصلی و کلیدی پرس‌وجو نیست و وزن بیشتری روی واژه‌های عمومی قرار می‌دهد، و همین موضوع باعث تغییر ناخواسته در رتبه‌بندی اسناد می‌شود.

چرا اضافه‌شدن واژه‌های عمومی باعث تغییر توزیع پرس‌وجو می‌شود؟

در مدل‌های زبانی، توزیع پرس‌وجو از روی فراوانی واژه‌ها (در خود پرس و جو + مدل زبانی مرجع) ساخته می‌شود. وقتی کلمات عمومی مثل "data" ,"use", "system", "information" به پرس‌وجو اضافه شوند:

احتمال این کلمات در مدل زبانی مرجع بسیار بالا است.

از طرف در فرمول های شباهت توزیعی ، احتمال واژه ها در مدل زبانی جست و جو، مدل زبانی سند و مدل زبانی مرجع اهمیت دارد و احتمال وجود این کلمات پرتکرار در مدل زبانی مرجع و سند ها بالاست.

بنابراین اسنادی که این کلمات عمومی را زیاد دارند، حتی اگر از نظر مفهومی بی‌ربط باشند، امتیاز بیشتری کسب می‌کنند و اسناد کوتاه ولی تخصصی که دقیقاً به موضوع مرتبط هستند، به دلیل اینکه آن واژه‌های عمومی را ندارند، در رتبه پایین‌تری قرار می‌گیرند.

روش های کاهش Query Drift

1. وزن‌دهی کمتر به واژه‌های عمومی

   واژه هایی که در مدل زبانی مرجع احتمال بالایی دارند را وزن پایینی بدهیم. همچنین واژه هایی که در اسناد زیادی ظاهر شده اند را هم وزن کمتری نسبت به واژه های کم تکرارتر در اسناد بدهیم یعنی همان IDF را اینجا هم انجام بدهیم.

2. گسترش پرس‌وجو برای جلوگیری از ورود واژه‌های عمومی

   مدل تلاش می‌کند پرس‌وجو را با اضافه‌کردن واژه‌های مرتبط و تخصصی که از اسناد اولیه استخراج شده‌اند تقویت کند. این کار باعث می‌شود توزیع پرس‌وجو حول مفاهیم اصلی پایدار بماند و واژه‌های عمومی یا غیرمرتبط که باعث Drift می‌شوند وارد مدل پرس‌وجو نشوند.

پاسخ بخش چهارم

پدیدهٔ Distribution Shift زمانی رخ می‌دهد که الگوی آماری داده‌هایی که مدل براساس آن‌ها آموزش دیده با الگوی داده‌هایی که در زمان استفادهٔ واقعی به سیستم داده می‌شود متفاوت شود.
در حوزهٔ بازیابی اطلاعات، این اتفاق کاملاً رایج است؛ زیرا رفتار زبانی کاربران همواره در حال تغییر است، در حالی که مدل‌های زبانی (خصوصاً مدل زبانی پس‌زمینه) معمولاً یک بار ساخته می‌شوند و ثابت باقی می‌مانند.
بنابراین اگر کاربران ناگهان از اصطلاحات جدید، حوزه‌های تخصصی تازه یا واژه‌هایی استفاده کنند که قبلاً در مجموعهٔ آموزش وجود نداشت، توزیع واقعی پرس‌وجوها از توزیع اولیه اسناد فاصله می‌گیرد و سیستم دچار کاهش دقت می‌شود.

هستهٔ مشکل اینجاست که مدل‌های زبانی بر پایهٔ احتمال‌سازی عمل می‌کنند. اگر واژه‌ای در اسناد آموزشی وجود نداشته باشد، مقدار اولیهٔ $P(w|D)$ برای آن صفر است و حتی روش‌های هموارسازی نیز معمولاً نمی‌توانند مقدار معناداری برای آن واژه فراهم کنند، زیرا آن واژه در مدل پس‌زمینه نیز حضور ندارد.
در نهایت، مدل واژه‌های جدید را نه به‌عنوان مفاهیم مرتبط، بلکه به‌عنوان نویز تلقی می‌کند؛ و همین موضوع باعث می‌شود ارتباط معنایی میان پرس‌وجوی جدید و اسناد مناسب تشخیص داده نشود و دقت سیستم کاهش یابد.

راه ۱

برای کاهش حساسیت مدل نسبت به واژه‌های دیده‌نشده، می‌توان به جای تکیه بر واژه‌های کامل، از روش‌هایی مانند Byte-Pair Encoding (BPE) یا WordPiece استفاده کرد.
این روش‌ها واژه‌ها را به واحدهای کوچک‌تر و پرتکرار می‌شکنند.
به‌عنوان مثال، اگر واژهٔ تازه‌ای مانند BIOHACKING وارد سیستم شود، مدل می‌تواند آن را به اجزای آشنا مانند:
Bio + hack + ing
تقسیم کند و از معنای این اجزاء برای تخمین احتمال و ارتباط استفاده کند.
در نتیجه، ورود واژه‌های جدید باعث شکست مدل نمی‌شود.

راه ۲

مشکل Distribution Shift معمولاً زمانی رشد می‌کند که مدل‌ها ثابت باقی بمانند.
می‌توان با طراحی یک سیستم یادگیری مستمر، مدل زبانی پس‌زمینه را به شکل دوره‌ای و منظم با استفاده از پرس‌وجوهای جدید، اسناد تازه و تغییرات زبانی واقعی کاربران بروزرسانی کرد.
بروزرسانی مداوم توزیع واژگان باعث می‌شود مدل بمرور با تغییرات زبانی سازگار شود و فاصلهٔ میان داده‌های آموزشی و داده‌های واقعی کاهش یابد؛ و این موضوع پایداری سیستم بازیابی را در برابر تغییرات زبانی افزایش می‌دهد.