بسم الله الرحمن الرحیم

دانشکده مهندسی برق و کامپیوتر

بازیابی هوشمند اطلاعات - تمرین سوم

سید مهدی رضوی

استاد : خانم دکتر شاکری

آذر ماه ۱۴۰۲

فهرست مطالب

فهرست تصاویر

۱ تمرین اول

۱.۱ مقدمه تمرین اول عملی

ما برای این تمرین با ابزار پایتون به بررسی دقیق روابط بین کلمات پرداخته‌ایم.
همینطور قبل از آن به PreProcessing داده‌ها با ابزار nltk پرداخته‌ایم.
توضیح کافی به همراه متن مختصر توضیحات در کد آمده است.
استفاده از جدول co-occurrence و همچنین کمک گرفتن از Entropy ،
و بردار شبه داکیومنت همه و همه در دفترچه پایتون آمده است.
در این قسمت فقط به نتایج و بررسی استنباط می‌پردازیم.

۲.۱ بهترین روابط جایگزینی با fix

('fix', 'back') 1.0623060171738952
('fix', 'app') 1.016075313874879
('fix', 'bug') 1.0364397442720932
('fix', 'get') 1.0622791875418032
('fix', 'sinc') 1.0692632769285833
('fix', 'last') 1.067330317401901
('fix', 'year') 1.0550606309553927
('fix', 'paus') 1.0595872457288265
('fix', 'premium') 1.0687727217677196
('fix', 'it') 1.0429723858531248
('fix', 'pleas') 0.9185091206082435
('fix', 'play') 0.9895712070877878
('fix', 'song') 1.0565295551010694
('fix', 'tri') 1.0401968428307167
('fix', 'updat') 0.9771419385081199
('fix', 'time') 1.0468334593375355
('fix', 'even') 1.0442387540510312
('fix', 'work') 1.0233635631035687
('fix', 'the') 1.0609905349178204
('fix', 'randomli') 1.06743523012607
('fix', 'stop') 1.02208694231523
('fix', 'spotifi') 1.047006920901704
('fix', 'pay') 1.0626880494243536
('fix', 'recent') 1.0507026978987422
('fix', 'problem') 1.0105734818283583
('fix', 'still') 1.0613495899137018
('fix', 'phone') 1.0261614147374967
('fix', 'go') 1.0687379149010132
('fix', 'issu') 0.9770285020813666
('fix', 'i') 1.0124065337042083
('fix', 'use') 1.0473040592247491
('fix', 'ca') 1.0174383469304147

۳.۱ بهترین روابط جایگزینی با like

('like', 'ad') 1.0317677258237679
('like', 'app') 0.9229778814133005
('like', 'get') 1.0040567367101727
('like', 'premium') 1.0439347699260568
('like', 'good') 1.030460778318152
('like', 'it') 0.9829904601490487
('like', 'thing') 1.0605016478739646
('like', 'play') 0.9796620236910933
('like', 'song') 0.8858461570469413
('like', 'also') 1.022906102263539
('like', 'listen') 0.96205054161234
('like', 'one') 1.0362880459064914
('like', 'would') 1.0122838282218183
('like', 'time') 1.043814620724446
('like', 'even') 1.0170738008587068
('like', 'spotifi') 0.95817325098131
('like', 'give') 1.0695569394312143
('like', 'music') 0.9430891003170647
('like', 'realli') 1.0331453647389297
('like', 'make') 1.0509801813192157
('like', 'go') 1.0618861871469552
('like', 'want') 1.0117116458507387
('like', 'love') 1.0495384622024877
('like', 'i') 0.852055750591527
('like', 'use') 0.9948504527393771
('like', 'playlist') 0.9981797037927899
('like', 'ca') 1.0538455086251424

۴.۱ استنباط ۱

با توجه به نتایج آزمایش‌ها مشخص است که کلماتی بهتر جایگزین می‌توانند بشوند که با این کلمه در یک مجموعه از نقش قرار بگیرند. Part Of Speech(POS) به نظر می‌رسد که علت اصلی این که علاوه بر افعال ، اسم‌ها نیز برای این دو کلمه برگردانده‌شده است ، این است که این کلمات در زبان انگلیسی می‌تواند هم نقش فعل و هم نقش اسم به خود بگیرند. مقدار مشاهده شده در بالا مقادیر زاویه بین شبه‌بردارهای بین این کلمات است. Psedu document این مقادیر زاویه برحسب رادیان است.

۵.۱ بهترین روابط همنشینی با fix

('ux', 'fix') 1.8366988247838922
('superb', 'fix') 1.1673441367723636
('format', 'fix') 1.982919565634936
('school', 'fix') 1.2692237507915771
('covid', 'fix') 1.3169204928228733
('fix', 'ux') 1.8366988247838922
('fix', 'superb') 1.1673441367723636
('fix', 'format') 1.982919565634936
('fix', 'school') 1.2692237507915771
('fix', 'covid') 1.3169204928228733
('fix', 'outstand') 1.982919565634936
('fix', 'promot') 1.2117382561308172
('fix', 'kpop') 1.982919565634936
('fix', 'speech') 1.7118923690476116
('fix', 'kannada') 1.417322389780711
('fix', 'misinform') 0.9590728236805686
('fix', 'freedom') 1.1673441367723636
('fix', 'beauti') 0.8023473199931154
('fix', 'polit') 1.2344583326309007
('fix', 'tast') -0.021881420628885068
('fix', 'censor') 1.6118254139436896
('outstand', 'fix') 1.982919565634936
('promot', 'fix') 1.2117382561308172
('kpop', 'fix') 1.982919565634936
('speech', 'fix') 1.7118923690476116
('kannada', 'fix') 1.417322389780711
('misinform', 'fix') 0.9590728236805686
('freedom', 'fix') 1.1673441367723636
('beauti', 'fix') 0.8023473199931154
('polit', 'fix') 1.2344583326309007

۶.۱ بهترین روابط همنشینی با like

('recognis', 'like') 1.867219266737927
('thoroughli', 'like') 2.296062565541801 ('fabul', 'like') 2.2685818291196944
('cough', 'like') 2.411539782961737
('v', 'like') 2.411539782961737
('citi', 'like') 1.95210816432444
('unexpectedli', 'like') 2.352646093908169
('kannada', 'like') 2.2020864173327874
('abruptli', 'like') 2.1891473616252894
('june', 'like') 1.7485747702393077
('fold', 'like') 2.1385212885553213
('disast', 'like') 2.241614781519425
('creation', 'like') 2.4729403276258806
('like', 'recognis') 1.867219266737927
('like', 'fabul') 2.2685818291196944
('like', 'thoroughli') 2.296062565541801
('like', 'cough') 2.411539782961737
('like', 'v') 2.411539782961737
('like', 'citi') 1.95210816432444
('like', 'unexpectedli') 2.352646093908169
('like', 'kannada') 2.2020864173327874
('like', 'abruptli') 2.1891473616252894
('like', 'june') 1.7485747702393077
('like', 'fold') 2.1385212885553213
('like', 'disast') 2.241614781519425
('like', 'creation') 2.4729403276258806
('like', 'ongo') 2.241614781519425
('like', 'drama') 2.411539782961737 ('like', 'protect') 2.41153978296173
('like', 'sincer') 2.352646093908169
('like', 'score') 2.1891473616252894
('like', 'iheartradio') 2.352646093908169 ('like', 'rington') 2.241614781519425
('ongo', 'like') 2.241614781519425
('protect', 'like') 2.41153978296173
('drama', 'like') 2.411539782961737
('sincer', 'like') 2.352646093908169
('score', 'like') 2.1891473616252894
('rington', 'like') 2.241614781519425
('iheartradio', 'like') 2.352646093908169

۷.۱ استنباط ۲

همانطور که کاملا از نتایج مشهود است ، کلمات غالبا در یک موضوع محتوایی با کلمات مدنظر سوال را دارند ، نتیجه‌گیری ما این است که الگوریتم EOWC تا حد بسیار خوبی توانسته است که کلمات در یک حوزه معنایی را به کمک تشکیل بردار شبه سند با کمک Mutual Information به دست بیاورد.

به ازای هر زوج کلمه در بالا ، میزان Mutual Information بین این زوج کلمات را مشاهده خواهیدکرد. این کلمات روابط هم‌نشینی خوبی را تشکیل می‌دهند.

در واقع تفاوت این دو نوع رابطه در آن است که در یکی الگوریتم به دنبال یافتن کلماتی است که در یک حوزه معنایی زیاد تکرار شده‌اند.

اما در رابطه دیگر رویکرد الگوریتم به دنبال آن است که کلماتی را که در یک حوزه جایگاهی از لحاظ نحوی هستند را بیابد تا بتوانند به جای یکدیگر قرار بگیرند.

۲ تمرین دوم

۱.۲ ۲ـالف

$$\gamma(x) = concat(Unary(Binary(x) - 1), Binary(x))$$

$$\delta(x) = concat(\gamma(Binary(x) - 1), Binary(x))$$

$$9 = (1001)_{Binary}$$

$$9 = (1110, 001)_{\gamma}$$

$$9 = (10, 1, 001)_{\delta}$$

$$100 = (1100100)_{Binary}$$

$$100 = (1111110, 100100)_{\gamma}$$

$$9 = (110, 10, 100100)_{\delta}$$

۲.۲ ۲ـب

$$|\gamma(x)| = |\delta(x)|$$

$$|Unary(Binary(x) - 1)| + |Binary(x)| = |\gamma(x)| + |Binary(x)|$$

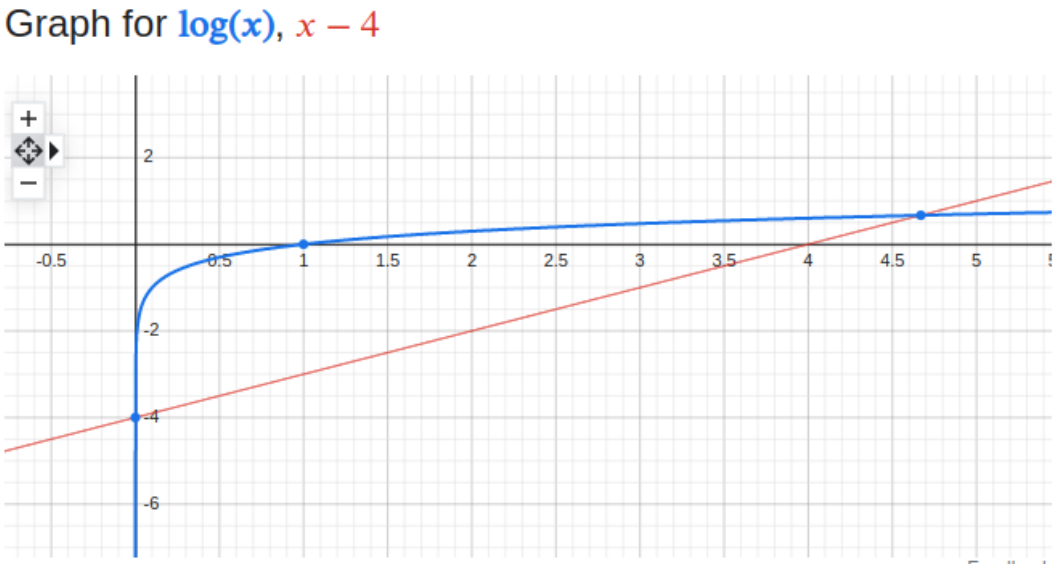
$$|Unary(Binary(x) - 1)| = |\gamma(x)|$$

$$2\log(\log(x)) + 1 = \log(x) + 1$$

$$\log(x) = x - 4$$

$$x = 4.699$$

۳.۲ ۲ـج



شکل ۱: رسم نمودار طول کدهای گاما و دلتا

با توجه به نمودار بالا کاملا به صورت شهودی قضیه مدنظر سوال اثبات می‌شود که از نقطه حدودا ۶.۴ به بعد طول کد رشته دلتای هر عدد از طول رشته گاما کوچکتر است.

۳ تمرین سوم

۱.۳ ۳ـالف

$$S(Q \cup q, D1) = S(Q, D1) + S(q, D1)$$

$$S(Q \cup q, D2) = S(Q, D2) + S(q, D2)$$

$$S(q, D1) = 0$$

$$S(Q, D2) = S(Q, D1)$$

$$S(Q \cup q, D1) < S(Q \cup q, D2) \implies S(q, D2) > 0$$

تنها ترمی که باید از صفر بزرگ‌تر باشد را در زیر می‌نویسیم :

$$\ln \frac{N - df(q) + 0.5}{df(q) + 0.5} * \frac{(k_1 + 1) * c(q, D2)}{k_1(1 - b + b\frac{D}{avdl})} \frac{(k_3 + 1)c(q, Q)}{k_3 + c(q, Q)} > 0$$

$$\ln \frac{N - df(q) + 0.5}{df(q) + 0.5} > 0$$

$$\ln \frac{N - df(q) + 0.5}{df(q) + 0.5} > \ln(1)$$

$$\frac{N - df(q) + 0.5}{df(q) + 0.5} > 1$$

$$\frac{N}{2} > df(q)$$

شرط برقرار بودن محدودیت Lower Bounding Constraint 1 برای روش Okapi آن است که تعداد وقوع ترم اضافه‌شده در اسناد ، کمتر از نصف کل اسناد باشد.

۲.۳ ۳ـب

$$S(Q, D \cup q) > S(Q, D)$$

$$S(Q, D) = \sum_{t \in q \cap D} \frac{1 + \ln(1 + \ln(c(t, D)))}{1 - s + s\frac{D}{avdl}} * c(t, Q) * \ln \frac{N + 1}{df(t)}$$

$$S(Q, D \cup q) = \sum_{t \in q \cap D \cup q} \frac{1 + \ln(1 + \ln(c(t, D)))}{1 - s + s\frac{D+1}{avdl}} * c(t, Q) * \ln \frac{N + 1}{df(t)}$$

$$S(Q, D \cup q) > S(Q, D)$$

برای ارضای شرط فوق می‌بایستی چند حالت را در نظر بگیریم :

۱. عبارت سمت راستی یک ترم بیشتر دارد که تنها شرط مثبت بودن آن این است که کوئری ترم q حداقل یک‌بار در سند ظاهر شده باشد. (اثبات در پایین)

۲. بزرگی عبارت سمت چپ به ازای ترم‌های مشترک دو عبارت وابسته به مقدار s آن‌هاست. چون که طول سند عبارت سمت راست یک واحد بیشتر از طول سند عبارت سمت چپ می‌باشد ، باید مقدار s به درستی تنظیم شود.

اثبات (۱)

$$1 + \ln(1 + \ln(c(q, D))) > 0$$

$$1 + \ln(c(q, D)) > \frac{1}{e}$$

$$\ln(c(q, D)) > \frac{1}{e} - 1$$

$$c(q, D) > e^{(\frac{1 - e}{e})}$$

$$c(q, D) > 0.531463$$

$$c(q, D) > 0$$

۳.۳ ۳-ج

محدودیت Lower Bound 2 یک محدودیت در مدل‌های بازیابی است که بیان می‌کند که امتیاز مربوط بودن یک سند نمی‌تواند منفی باشد.
این بدان معناست که اگر سندی حاوی هیچ‌یک از اصطلاحات پرس‌وجو هم نباشد ، همچنان باید دارای امتیاز غیرمنفی باشد.
این محدودیت تضمین می کند که اسناد نامربوط بالاتر از اسناد مربوطه رتبه بندی نمی شوند، حتی اگر حاوی هیچ یک از اصطلاحات پرس و جو نباشند.