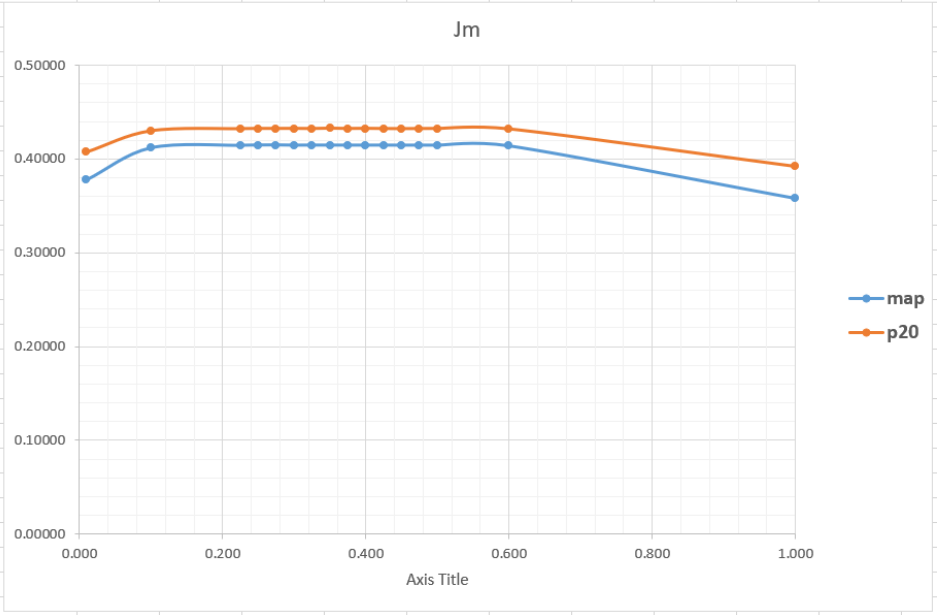
سوال 1)

:jm

Jm	map	p20
0.010	0.37753	0.40738
0.100	0.41155	0.42975
0.225	0.41429	0.43201
0.250	0.41463	0.43226
0.275	0.41473	0.43226
0.300	0.41446	0.43231
0.325	0.41454	0.43216
0.350	0.41450	0.43251
0.375	0.41449	0.43221
0.400	0.41448	0.43236
0.425	0.41451	0.43216
0.450	0.41445	0.43206
0.475	0.41448	0.43216
0.500	0.41440	0.43221
0.600	0.41372	0.43181
1.000	0.35825	0.39252

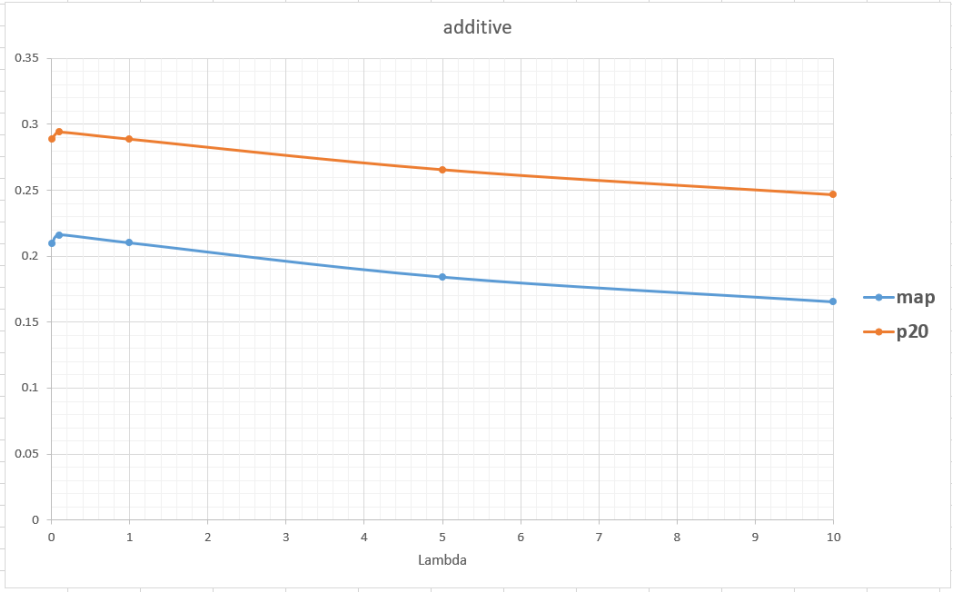


بهترین مقدار برای حدود 0.275 یا 0.35 است.

در این روش به کمک پارامتر لاندا ترکیبی از تاثیر دو عامل maximum likelihood و مدل مرجع را داریم. در این مجموعه لاندا کمی کمتر از 0.5 خوب جواب داده که انگار کلمات زیادی هستند که در سند نیستند و باید ارزش گذاری شوند و اتفاقا می توانند مهم نیز باشند! مشاهده می شود که با لاندا 1 (حذف تاثیر مدل مرجع) نتایج بدتر از زمانی که حذف تقریبی maximum likelihood را داریم شده است.

:additive smoothing

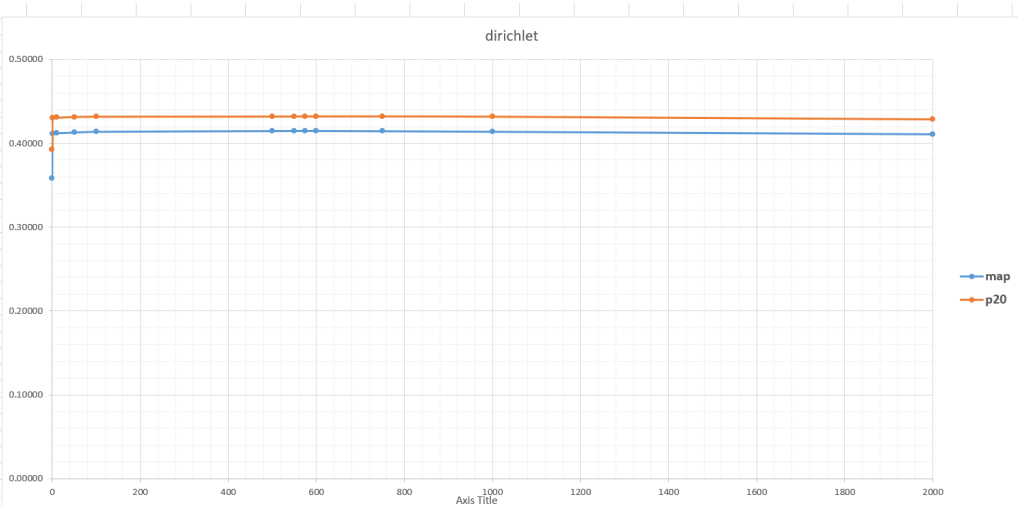
additive	map	p20
0.01	0.20942	0.28841
0.1	0.21598	0.29412
1	0.20989	0.28876
5	0.18425	0.26554
10	0.16568	0.24652



در این مدل با 0.1 بهترین نتیجه را تقریبا داریم. در این روش به تعداد کلمات مقداری افزوده می شود. گویا بیشتر کلمات به اندازه کافی در متن هستند و اضافه کردن حتی یک کلمه باعث می شود نسبت تعداد کلمات به طول سند بیش از حد زیاد شود که مطلوب نیست.

:dirichlet

dirichlet	map	p20
0	0.35832	0.39252
1	0.41165	0.43055
10	0.41192	0.43070
50	0.41281	0.43146
100	0.41370	0.43186
500	0.41456	0.43206
550	0.41470	0.43216
575	0.41473	0.43221
600	0.41469	0.43226
750	0.41436	0.43226
1000	0.41376	0.43191
2000	0.41075	0.42870

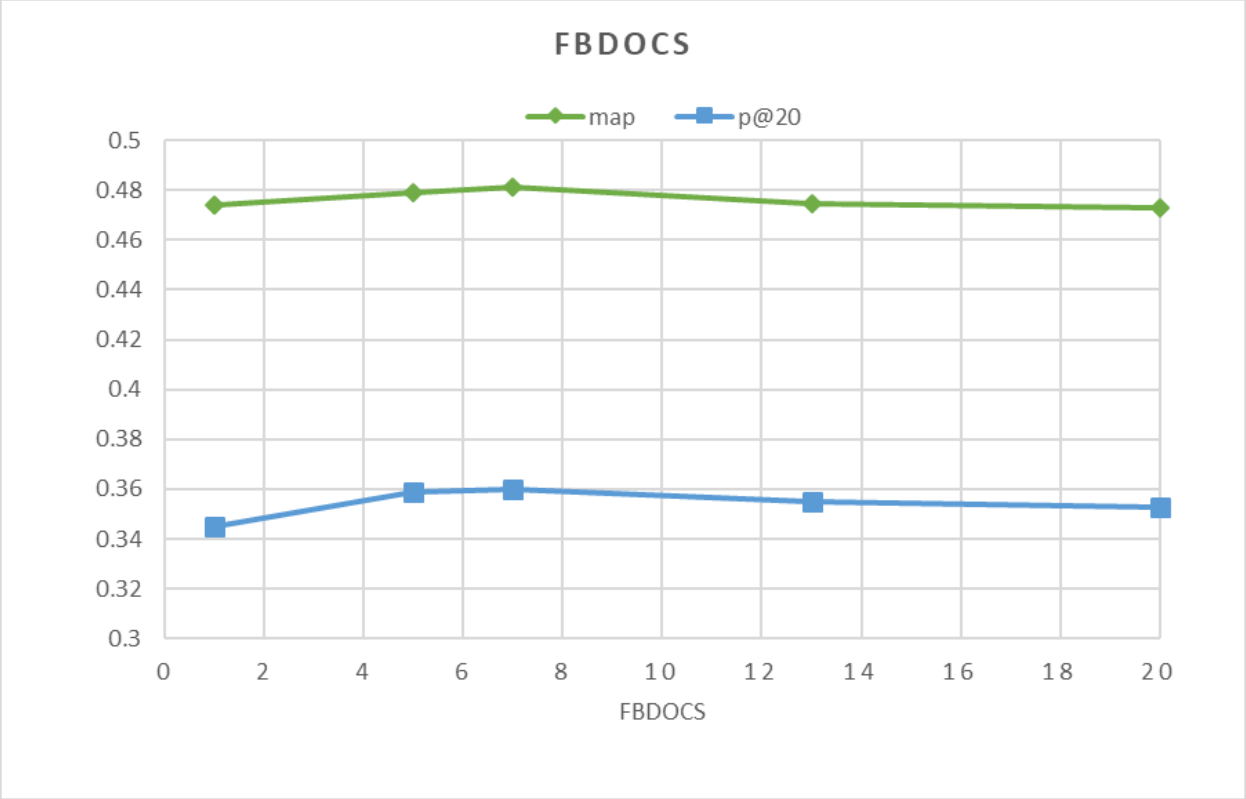


همانطور که دیده می شود در تغییر از 0 به 1 برای مو پیشرفت داریم اما در مقادیر بیشتر پیش چشمی دیده نمی شود. این روش smoothing زمانی که تعداد کلمات دیده نشده یا کم دیده شده در سند زیاد باشد به درد میخورد و هرچه طول سند کمتر باشد تاثیر آن بیشتر است. در مورد ما احتمالا تعداد خوبی از کلمات پرس و جو در سند ها دیده می شوند و یا با تاثیر مقدار کمی مو نیز قابل تمییز از یکدیگر می شوند و smoothing از یه جا به بعد خیلی تاثیر نمی گذارد.

best = 575 - 600

سوال 2)

الف) fbDocs

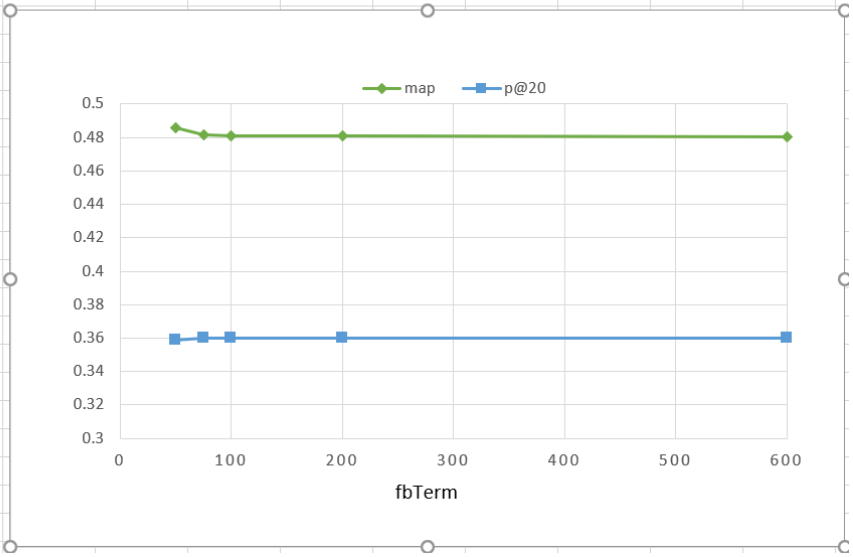


برای fbDocs = 7 مقادیر بهینه اند.

اگر تعداد اسناد بازخورد خیلی کم باشد عملا از فیدبک در آپدیت مدل کوئری استفاده خاصی نـ
شاید تاثیر مدل مرجع بیشتر دیده می شود. اگر هم خیلی زیاد باشد ممکن است تعداد خوبی از کلمات موجود در فیدبک ها که خیلی هم اهمیت ندارند وزن زیادی بگیرند و در مدل teta f وزن بسیاری بگیرند.

ب) fbTerms

fbTerm	map	p@20
50	0.48572	0.359
75	0.48165	0.36
100	0.48106	0.36
200	0.48078	0.36
600	0.48047	0.36



برای 50 در موارد ازمایشی ما نتیجه کمی بهتر است. نباید خیلی به همه کلمات اسناد بازخورد اکتفا کرد. لزوما مدل کوئری را بهتر نمی کنند. اینکه نتیجه از مقداری ب بعد تغییر نکرده شاید ب این دلیل است که تعداد کلمات اسناد بازخورد تمام شده و هرچه بیشتر کنیم تاثیری ندارد. (در تست fbTerm از مقدار بهینه قسمت قبل یعنی fbDoc = 7 استفاده کردیم)

سوال 3)

– B

الف)

$$q = \text{" بازیابی اطلاعات اسناد "} \qquad |d_1| = 8 \qquad |d_2| = 8$$

در نگاه اول ، به نظر می رسد سندی که کلیات کلیدی " بازیابی " ، " اطلاعات " و " اسناد " و شاید حتی " هوشمند " را بیشتر دارد ، برای کاربر مورد انتظار و مورد پسند است :

کلمه " بازیابی " در سند دوم 2 بار بیشتر از سند اول ، کلمه " هوشمند " در سند دوم یک بار تکرار شده اما در سند اول نیامده ، کلمه " اطلاعات " در سند دوم 2 بار و در سند اول 3 بار آمده ، کلمه " اسناد " در هر دو سند به تعداد برابر آمده .

انتظار می رود سند دوم به دلیل تاکید بیشتر بر روی واژه " بازیابی " و تعداد تکرار تقریبا یکسان در دیگر کلمات ، نسبت به سند اول بهتر باشد .

1) d_2

2) d_1

ب)

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

$$p(w|d_1) = p(w = \text{"بازیابی"} \mid d_1) \times p(w = \text{" اطلاعات "} \mid d_1) \times p(w = \text{" اسناد"} \mid d_1) =$$

$$\frac{c(\text{بازیابی}, d_1)}{|d_1|} \times \frac{c(\text{اطلاعات}, d_1)}{|d_1|} \times \frac{c(\text{اسناد}, d_1)}{|d_1|} =$$

$$\frac{2}{9} \times \frac{3}{9} \times \frac{1}{9} \cong 0.0082304$$

$$p(w|d_2) = p(w = \text{"بازیابی"} \mid d_2) \times p(w = \text{" اطلاعات "} \mid d_2) \times p(w = \text{" اسناد"} \mid d_2) =$$

$$\frac{c(\text{بازیابی}, d_2)}{|d_2|} \times \frac{c(\text{اطلاعات}, d_2)}{|d_2|} \times \frac{c(\text{اسناد}, d_2)}{|d_2|} =$$

$$\frac{4}{8} \times \frac{2}{8} \times \frac{1}{8} \cong 0.0156 \rightarrow d_2 \gg d_1$$

ج)

$$d = 0.5 \qquad p(w|d) = \frac{c(w,d)+0.5}{|d|+0.5 \times |v|} \qquad v: \text{vocab size}$$

$$p(w|d_1) = p(w = \text{"بازیابی"} \mid d_1) \times p(w = \text{" اطلاعات "} \mid d_1) \times p(w = \text{"اسناد"} \mid d_1) =$$

$$\frac{2+0.5}{9+0.5\times10} \times \frac{3+0.5}{9+0.5\times10} \times \frac{1+0.5}{9+0.5\times10} = \frac{2.5}{14} \times \frac{3.5}{14} \times \frac{1.5}{14} \cong 0.0047$$

$$p(w|d_2) = \frac{4+0.5}{8+0.5\times10} \times \frac{2+0.5}{8+0.5\times10} \times \frac{1+0.5}{8+0.5\times10} = \frac{4.5}{13} \times \frac{2.5}{13} \times \frac{1.5}{13}$$

$$\cong 0.0076 \rightarrow d_2 \gg d_1$$

د)

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

$$|d|_u : \text{number of unique words in } d$$

$$d = 0.5$$

$$p(w|d_1) = p(w = \text{"بازیابی"} \mid d_1) \times p(w = \text{" اطلاعات "} \mid d_1) \times p(w = \text{"اسناد"} \mid d_1) =$$

$$\frac{1.5+0.5\times5\times0.6}{9} \times \frac{2.5+0.5\times5\times0.1}{9} \times \frac{0.5+0.5\times5\times0.1}{9} =$$

$$\frac{3}{9} \times \frac{2.75}{9} \times \frac{0.75}{9} = \frac{6.1875}{729} \cong 0.0084$$

$$p(w|d_2) = p(w = \text{"بازیابی"} \mid d_2) \times p(w = \text{" اطلاعات "} \mid d_2) \times p(w = \text{" اسناد"} \mid d_2) =$$

$$\frac{3.5+0.5\times4\times0.6}{8} \times \frac{1.5+0.5\times4\times0.1}{8} \times \frac{0.5+0.5\times4\times0.1}{8}$$

$$\frac{4.7}{8} \times \frac{1.7}{8} \times \frac{0.7}{9} = \frac{5.593}{512} \cong 0.0109 \quad \rightarrow \quad d_2 \gg d_1$$

ه) بله ممکن است . ممکن است کلمه ای را داشته باشیم که اصلا در سند نیست و $c(w,d)$ ان برابر 0 است اما $p(w,c)$ آن بزرگ است و عام است .

در این حالت

$c(w,d) - S + s|d|_u p(w,c)$ برای این کلمه می تواند به اندازه ای باشد که از $s|d|_u p(w,c)$ کلمه دیگری که در سند وجود دارد بیشتر شود .

$$w_1 = \text{زیبا} \qquad c(w_1,d) = 0 \quad p(w_1|c) = 0.9$$

$$w_2 = \text{علی} \quad c(w_2,d) = 1 \quad p(w_2|c) = 0.1 \quad s(lambda) = 0.5 \quad |d| = 10 \quad |d|u = 5$$

$$p(w1|d) = \frac{0.5\times5\times9}{10} > \frac{0.5+0.5\times5\times0.1}{10} = p(w_2|d)$$

-A

۱_غلط است.

در این مدل به ازای یک μ ثابت، هرچه طول سند بیشتر باشد smoothing کمتر انجام می شود.

اما μ برآوردی از میانگین طول سند نیست.

۲_درست است.

این الگوریتم در هر مرحله likelihood را افزایش می دهد تا به یک max local optima برسد. پس برای نقاط مختلف از یک مقدار شروع کرده و به یک max محلی می رساند و minimum optimisation انجام نمی دهد.

۳_درست است.

Smoothing دو هدف دارد. یکی نسبت دادن وزن برای کلماتی که در سند نیامده اند. و دیگری نزدیک کردن وزن کلمات عام مانند stopword ها در دو داکیومنت.

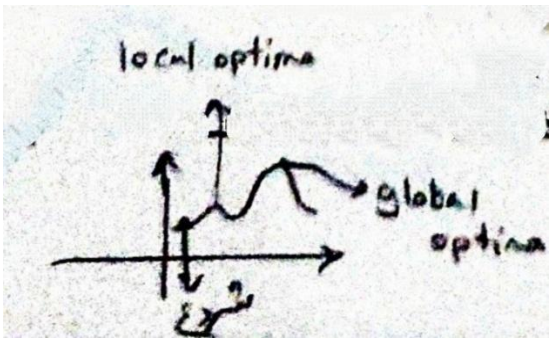
۴_غلط است.

این الگوریتم در هر مرحله مقدار هدف خود را بیشتر می کند و likelihood را در هر مرحله افزایش می دهد تا اهداف به local optima خود برسد.

۵_غلط است.

این الگوریتم برای وزن هر کلمه به طور اتفاقی از یک جا شروع می شود آنقدر iteration ها را ادامه می دهد تا به یک local optima برسد و پس از آن دیگر ادامه نمی دهد.

ممکن است آن max محلی، بیشترین مقدار ممکن نباشد!



سوال 4)

طبق توضیحی که در باره maxRank داریم .

(maxRank(n)_ maxRank(n+1) = تعداد کلماتی که n بار تکرار میشوند

طبق قانون زیپ داریم :

r*freq() = A * N

maxRank کلمات با n بار تکرار زیرا در فرمول فوق جایگزاری می کنیم :

n Rank(n) × n = A * N ⟶ maxRank =

maxRank(n) _ maxRank (n+1) = بارتکرارمیشوند n تعداد کلماتی که

$\frac{A*N}{n} - \frac{A*N}{n+1}$

$= A * N \left(\frac{1}{n} - \frac{1}{n+1}\right) = A * N \left(\frac{1}{n(n+1)}\right) = \frac{A*N}{n(n+1)}$

حال باید A * N را حساب کنیم.

فرض میکنیم کمترین تعداد تکرار یک کلمه 1 بار می باشد(کلمات unique)

کلمه با کمترین تکرار (1بار) در رتبه آخر رتبه بندی کلمات می باشد.

اگر فرض کنیم اندازه کلمات مجزا 7 باشد.

طبق قانون زیپ داریم :

$$r * freq(r) = A * N \; -> V \times 1 = A * N \}$$

فرمول تعداد کلمات با n بار تکرار صفحه قبل را با مقدار یافت شده برای A * N بازنویسی می کنیم :

تعداد کلمات با n بار تکرار = $\frac{A*N}{n(n+1)} - \frac{V}{n(n+1)}$

الف) چه نسبتی از عبارات فقط یک بار ظاهر می شود؟

$\frac{\text{تعدادکلمات،با1بارتکرار}}{\text{کلمات تعداد مجزا}} = \frac{V \times \left(\frac{1}{1(1+1)}\right)}{V} = \frac{1}{2}$

ب) چه نسبتی 10بار ظاهر می شود ؟

$$\frac{\text{تعدادکلمات،با}}{\text{کلمات،مجزا تعداد}} = \frac{V\left(\frac{1}{10(11)}\right)}{V} = \frac{1}{110}$$

ج)

$$\frac{\text{تعدادکلمات،با}k\text{بارتکرار}}{\text{تعدادکلمات،مجزا}} = \frac{7\left(\frac{1}{k(k+1)}\right)}{7} = \frac{1}{k(k+1)}$$