Langrage Model:  
4\_1.

Alpha = 0.059 map->0.6382307

4\_2.

在Alpha 參數= 0.059前提下，加入K-means 將文章略為分群，跑10次的K-means :

beta 參數 = 0.0004 map->0.6400256

beta參數 = 0.01 map->0.64312387

beta參數 = 0.009 map->0.6397073

beta參數 = 0.0095 map->0.6303829

beta參數 = 0.0098 map->0.613603

在beta參數為0.01下，實驗分群次數的結果:

8次 0.01 map->0.6406543

9次 0.01 map->0.6335494

10次 0.01 map->0.64312387

11次 0.01 map->0.63503027

19次 0.01 map->0.63101757

20次 0.01 map->0.6551658 (在0.01下分群最好)

21次 0.01 map->0.63124686

22次 0.01 map->0.6284691 (由以下得知在22次分群中收斂)

25次 0.01 map->0.6284691

30次0.01 map->0.6284691

40次 0.01 map->0.6284691

50次 0.01 map->0.6284691

將原來公式調整為下列 :

(alpha\*P(w|D) + (1-alpha)\*P(w|Corpus) + beta\*P(w|Group)

P(w|Group)指的是query出現的word在某篇D文章出現，而該文章隸屬於某群，其P(w|Group)值為word出現在某群的文章中出現次數/Group是指該群所有文章的總字數。而K-means做法是分做10群，距離是以cosine-similar去算，中心點取平均虛擬資料。

而我試過將分群的次數變多或減少，同樣beta=0.01參數之下，再嘗試以20次分群為基準調整beta參數，發現比0.01高的參數，map是越來越低，還是沒有0.01高，應該是指不能大幅依賴分群的結果，僅僅微調。

20次 分群開始調動beta值

0.008 map->0.65567446

0.007 map->0.6235123

0.006 map->0.6416563

0.005 map->0.63305205

0.004 map->0.6403073

0.003 map->0.6535916

0.002 map->0.63854855

0.001 map->0.63802606

0.0009 map->0.6354963

0.0008 map->0.64547515

0.0007 map->0.6430392

0.0006 map->0.6301028

0.0005 map->0.6438254

0.0004 map->0.63936055

0.0003 map->0.63228333

0.0002 map->0.64189446

0.0001 map->0.62994343

目前alpha = 0.059 , beta = 0.008 K-means 20 次 MAP會最高

MAP->0.65567446