HW5

1.

a. With λ = 0, the second model is equivalent to the first one since the probability of choosing the background is 0 and we can ignore the background term.

b. $p(w \mid D) = \frac{\sum_{d \in D} c(w,d)}{\sum_{d \in D} |d|}$ is the maximum likelihood estimate for this LM.

c. Hypothesis: A large λ will make the topics indistinguishable from each other, which means that every topic has similar word probabilities. A small λ will make every topic have high probabilities for common words.
To test the hypothesis, we need to use a fixed collection of documents, a fixed background topic model and a fixed set of initial values as the starting point of EM algorithm. Then we can generate several large, small and normal values of λ and use them to run EM algorithm with other parameters stay the same. In the end, we can compare the results of topics generated by different values of λ.

2. Similar to the derivation in Chase's note section 4.
E step:

$$n_{d,k} = \sum_w c(w,d) \frac{(1-\lambda)\mathrm{p}(z_{d,w}=k|\pi_d)p(w|\theta_k)}{\lambda p(w|D) + (1-\lambda)\sum_{k'}^{K}\mathrm{p}(z_{d,w}=k'|\pi_d)p(w|\theta_{k'})}$$

$$n_{w,k} = \sum_d c(w,d) \frac{(1-\lambda)\mathrm{p}(z_{d,w}=k|\pi_d)p(w|\theta_k)}{\lambda p(w|D) + (1-\lambda)\sum_{k'}^{K}\mathrm{p}(z_{d,w}=k'|\pi_d)p(w|\theta_{k'})}$$

Here d_j denotes the jth word token in document d.

M step:

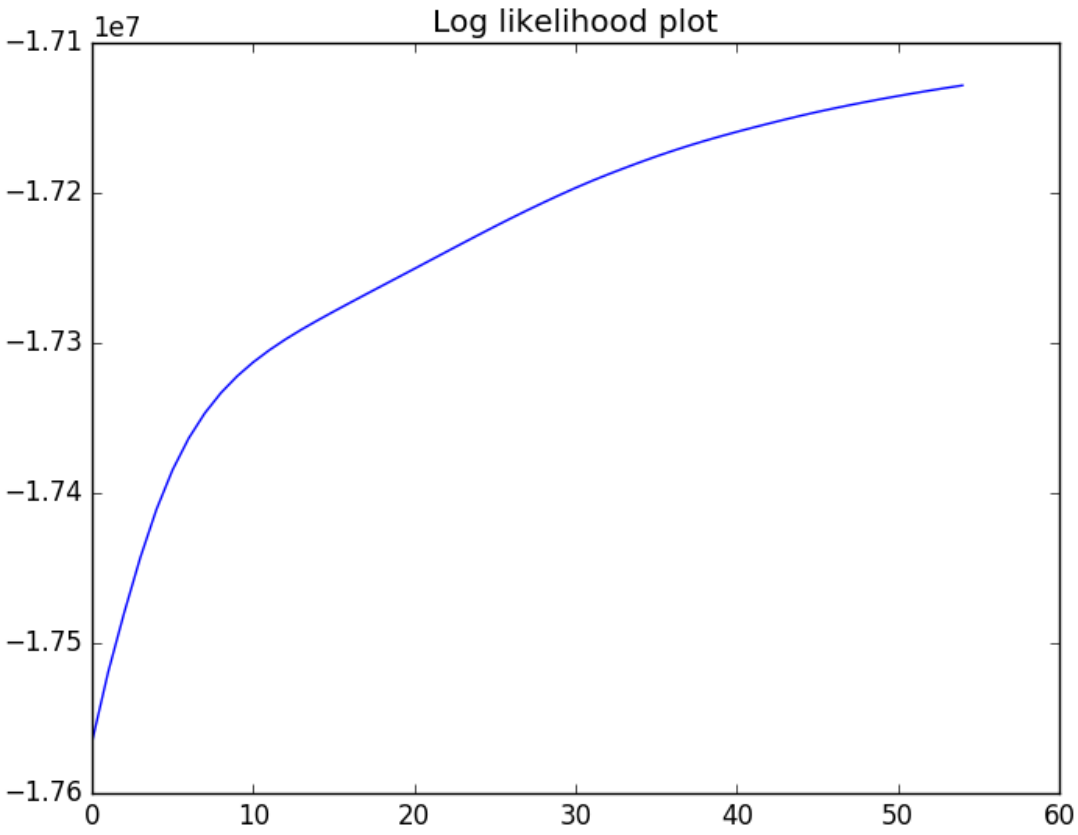
$$p(z=k|\pi_d) = \frac{n_{d,k}}{\sum_{k'}^{K} n_{d,k'}}$$

$$p(w|\theta_k) = \frac{n_{w,k}}{\sum_{w' \in V} n_{w',k}}$$

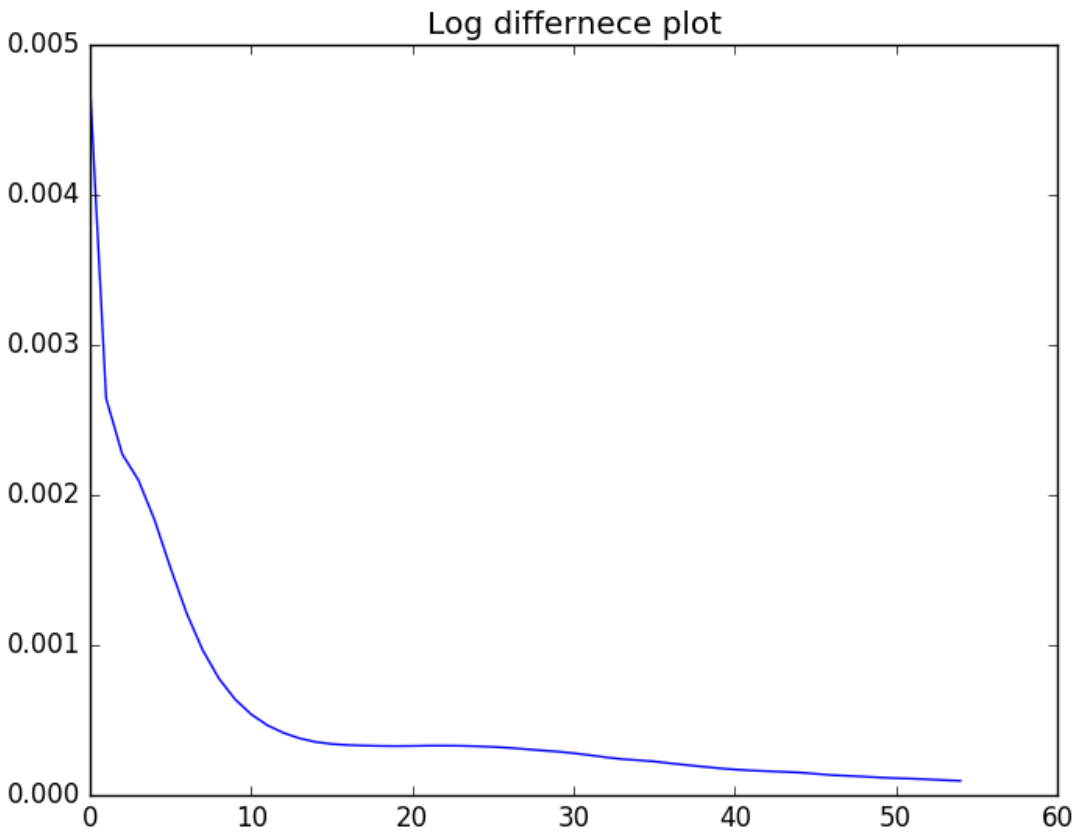
3. b. The plot would look like a line goes up very quickly in the first few iterations, then gradually reach a stable value and the curve goes smoothly.
c. The plot would look like a line goes down drastically in the first few iterations, then goes smoothly in the next iterations.
d.

Log likelihood:

The graph goes up because in the M step we maximized the log likelihood and it supposed to converge after few iterations, so it grows quickly first, then it went smoothly.



Log Difference:
The log difference should go down quickly then, went smoothly. Because after each iteration, it converges gradually, so it goes smoothly in the end.



e.
Lambda = 0.3 :
[('the', 0.63842111230692744), ('of', 0.053615273689081429), ('loop', 0.020976284983202545), ('and', 0.014883730666404511), ('web', 0.009650546351953471), ('collector', 0.0047384269581109252), ('testing', 0.0046888841725004323), ('placement', 0.0039701699390995221), ('push', 0.0034911998898949381), ('routing', 0.0034884882676994118)]

[('the', 0.2798643833922187), ('of', 0.12205183582027265), ('to', 0.016125872628803769), ('and', 0.014140199527020508), ('page', 0.011416409405013634), ('schema', 0.011169766660668105), ('soft', 0.010877272222636843), ('malware', 0.010604553628537353), ('coverage', 0.010194348855423557), ('xml', 0.0098399193337421788)]

[('the', 0.50201220255111689), ('of', 0.1391438404025071), ('and', 0.031700339414969705), ('grid', 0.020527633825643783), ('rules', 0.01643498805822349), ('workflow', 0.013782309781977854), ('association', 0.010287947822245914), ('disk', 0.009821646795896052), ('web', 0.0090956777210780548), ('sampling', 0.0088920973938169737)]

[('of', 0.21549679338056268), ('the', 0.13043058466457638), ('and', 0.034380563865328041), ('to', 0.028244644290128917), ('multicast', 0.022968896493773414), ('overlay', 0.013781002291645451), ('in', 0.012722851208147257), ('memory', 0.012471189770500988), ('xml', 0.0094937592543362318), ('disks', 0.0070499137088275016)]

[('the', 0.54202587998917107), ('of', 0.14036073699244192), ('and', 0.020256705055857305), ('to', 0.013355400021026703), ('itemsets', 0.009739527092757246), ('frequent', 0.0095170573336142619), ('color', 0.0068577369708738697), ('in', 0.0058976907389378967), ('scp', 0.005585045185315011), ('dns', 0.0054152247180090105)]

[('the', 0.59411302387405651), ('of', 0.067291313429775859), ('branch', 0.013201069957099614), ('news', 0.013068261493145678), ('similarity', 0.0074122185168114937), ('sensor', 0.0061060103614062958), ('and', 0.0058718402088139875), ('to', 0.0055758505049872328), ('intrusion', 0.0048886254509352281), ('cube', 0.0045218597054107972)]

[('the', 0.78189987801860894), ('xml', 0.016913770772209191), ('of', 0.016586331529028622), ('type', 0.0050951902616571346), ('transform', 0.0050661786935006911), ('opinion', 0.0045707544340150372), ('view', 0.0043886847055290869), ('in', 0.0040237686932056357), ('controller', 0.003856047535597068), ('data', 0.0038507548770832091)]

[('the', 0.67381630421218008), ('and', 0.017078392001698244), ('of', 0.016689789639467319), ('to', 0.012002004461669767), ('vm', 0.0069185499009318556), ('prediction', 0.0068578877089178186), ('epsiv', 0.0068258492931781824), ('ownership', 0.0058231244346155734), ('branches', 0.0055124112326990892), ('corba', 0.0053038926040695177)]

[('the', 0.32649583762319612), ('of', 0.19851551438844595), ('test', 0.069197298632353993), ('and', 0.040501756369726999), ('prefetching', 0.015566515906002521), ('ranking', 0.013352498839023671), ('adaptation', 0.0082768381769729414), ('social', 0.0079291941285533872), ('chain', 0.005713020220199181), ('job', 0.0057119272475829571)]

[('the', 0.75853768639830976), ('and', 0.017000038687378819), ('svm', 0.01136596960445573), ('of', 0.0091806142298482968), ('skyline', 0.0076656899730841305), ('skin', 0.0054123571053606226), ('in', 0.0049803541438927113), ('mappings', 0.0047532534868382197), ('schema', 0.0047453893087231287), ('to', 0.0041074733891890572)]

[('the', 0.56205252240701598), ('of', 0.092798677436030749), ('to', 0.0153410200963167), ('qos', 0.014028515644592676), ('web',

0.011074562701208235), ('pages', 0.0087571282620506081), ('search', 0.0072785204725492377), ('tm', 0.0054327844590847637), ('graphs', 0.0053031951392958129), ('schemas', 0.0042734081906206254)]

[('the', 0.664841166135971), ('and', 0.068512639951782348), ('fuzzy', 0.01818857393818258), ('cache', 0.0092583955415291262), ('vague', 0.0068223145595125859), ('to', 0.0055520798902249889), ('sensor', 0.0047988232228445577), ('hypermedia', 0.0036135120255472216), ('operators', 0.0033598945156355776), ('probabilistic', 0.0030998440664674501)]

[('the', 0.55053800190996416), ('and', 0.079970886532979041), ('cache', 0.052184245186051227), ('to', 0.034387196641021266), ('defect', 0.0058785235819818447), ('energy', 0.0042865340150624572), ('tiles', 0.0039157611364222591), ('filter', 0.0039012774682327151), ('ratings', 0.0036658486446176967), ('geospatial', 0.0035794334383375342)]

[('the', 0.55524114261681179), ('of', 0.074401056927866735), ('mobility', 0.013079632142942131), ('slicing', 0.010322716324105561), ('energy', 0.0095091329883551338), ('to', 0.0070789085123012831), ('hair', 0.0062916146378733289), ('rdf', 0.0056297345486938039), ('and', 0.0052355180905063548), ('ads', 0.0047283709826825273)]

[('the', 0.41428536417686002), ('of', 0.10773559450254602), ('and', 0.095032079241963607), ('peer', 0.025693582961492909), ('tcp', 0.011918477586551065), ('to', 0.011029887307159829), ('spatial', 0.0083717964153862453), ('peers', 0.0076708410512714657), ('clustering', 0.0072180194332132942), ('recognition', 0.0058196956115491843)]

[('the', 0.47685787057949702), ('to', 0.043100037582617864), ('and', 0.026179932882816104), ('of', 0.015890186832728956), ('project', 0.0085626139367555396), ('web', 0.0065326473886269717), ('routing', 0.0064765435526340821), ('suffix', 0.0062683963470456863), ('is', 0.0059292541530948867), ('type', 0.0058905534666913832)]

[('the', 0.75768766037692681), ('of', 0.013673497204555929), ('clustering', 0.011029988476512388), ('in', 0.0060772938634874075), ('web', 0.0058381775192037919), ('cloud', 0.0040613405318449423), ('exact', 0.0031908757310985937), ('and', 0.0031549338873991125), ('migration', 0.003120801213237402), ('graphs', 0.0030671887087031073)]

[('the', 0.59493221005157948), ('query', 0.02989678857243009), ('and', 0.019729683825909 54), ('trust', 0.018694266961355337), ('image', 0.017076910834478983), ('join', 0.013379434282972389), ('direction',

0.0068171490171036403), ('gui', 0.0061426539026234834), ('of',
0.0056420742606706209), ('dl', 0.0046386841343897013)]

[('the', 0.5613480727346718), ('of', 0.1623111678054622), ('to',
0.022110398829003102), ('and', 0.012806551085900258), ('cal',
0.0073433821549505531), ('topic', 0.0072300935865950728), ('adaptation',
0.0067504303519173095), ('domain', 0.0052984426415186988), ('views',
0.0052370787259981523), ('xml', 0.004083054941042991)]

[('the', 0.476254531509192), ('of', 0.11080583283843817), ('and',
0.06821732404078773), ('usability', 0.015403528222565932), ('routing',
0.011578954458417404), ('data', 0.0080613546978549333), ('policy',
0.0080582962223333058), ('wavelet', 0.0058327080564210361), ('replay',
0.0056835931307423742), ('uml', 0.0055545937668293856)]

Lambda = 0.9 :
[('query', 0.066691659500111383), ('queries', 0.055347804109732925), ('xml',
0.038021030697058741), ('processing', 0.01865810826937106), ('index',
0.018464096538045066), ('database', 0.016576752222187761), ('join',
0.016348854099209512), ('tree', 0.014771043283475277), ('relational',
0.013515117770683809), ('graph', 0.012571992797254235)]

[('access', 0.044136109249608341), ('control', 0.038916129453191679), ('security',
0.029977821965596473), ('policies', 0.02793329241296218), ('translation',
0.022814025246927105), ('policy', 0.022301069060288635), ('role',
0.018133510845233947), ('workflow', 0.01776105881871453), ('loop',
0.016476881353364926), ('skyline', 0.0092274643057246915)]

[('user', 0.05446303886063468), ('users', 0.036633800617292989), ('collaborative',
0.016368670687795885), ('filtering', 0.01492002482015131 5), ('recommender',
0.011774771776388004), ('recommendation', 0.011411120134540877), ('preferences',
0.010542844073806091), ('recommendations', 0.010438872843950699), ('information',
0.0099218838596616798), ('adaptation', 0.0094710426234635627)]

[('network', 0.077651771606208078), ('networks', 0.051872701293298323), ('sensor',
0.045015155839571802), ('nodes', 0.040276461472136572), ('routing',
0.039912794906878171), ('wireless', 0.025069905339153147), ('node',
0.022132639327997541), ('traffic', 0.019837210657268508), ('protocol',
0.015336567428471582), ('communication', 0.014501912952917419)]

[('power', 0.051515841424353913), ('energy', 0.040429617904424481), ('design',
0.031951545718362062), ('consumption', 0.021228321838744173), ('architecture',
0.019644154840645779), ('performance', 0.018867870017800003), ('processor',

0.016513689122817594), ('chip', 0.014205763505000283), ('instruction', 0.014148423747296326), ('embedded', 0.013061398088470009)]

[('software', 0.066876419121524436), ('test', 0.044701607552319743), ('testing', 0.030034230627364916), ('development', 0.020861328882980666), ('requirements', 0.019297148150953895), ('tool', 0.01093331142908203), ('components', 0.009972718112580842), ('engineering', 0.0095853011840517383), ('suite', 0.0085548210472387726), ('usability', 0.0076168805124106906)]

[('programs', 0.035192369309759809), ('type', 0.034269380991378821), ('code', 0.032778323494144036), ('program', 0.029270285325069059), ('language', 0.028094106205166784), ('programming', 0.023088868888530394), ('java', 0.018658781193363796), ('languages', 0.017379835937094607), ('compiler', 0.016895326073953248), ('implementation', 0.010364469877775525)]

[('cache', 0.059418792222892806), ('memory', 0.043949911422687865), ('performance', 0.041299880316932311), ('scheduling', 0.019976155586705832), ('processor', 0.013596093844326527), ('processors', 0.012293827993042912), ('disk', 0.011667425477813092), ('hardware', 0.011361598554134567), ('caches', 0.0097517073175411705), ('prefetching', 0.0096345463724305715)]

[('image', 0.087798131145895769), ('images', 0.045895906129789082), ('texture', 0.023721296486937517), ('retrieval', 0.020843428770939088), ('method', 0.01810497140664544), ('semantic', 0.017813714198794076), ('ontology', 0.015305973936719213), ('shape', 0.013191308945456637), ('features', 0.012571901192546022), ('local', 0.012206216643856778)]

[('fault', 0.023954061867109456), ('execution', 0.017528975830797353), ('failure', 0.015196650447680321), ('failures', 0.013524827695089808), ('program', 0.013303298644988708), ('bugs', 0.011415010092572914), ('faults', 0.011388695638678921), ('software', 0.011092737711091682), ('debugging', 0.010846749708788491), ('recovery', 0.010299672365317038)]

[('ranking', 0.038386166401609634), ('human', 0.021748720366010246), ('game', 0.017546177497590118), ('retrieval', 0.01634999615652458), ('facial', 0.013270085427834442), ('social', 0.012485434675152601), ('user', 0.011544775231346586), ('rank', 0.01104860770277827), ('games', 0.010763292846935191), ('question', 0.010115193076656363)]

[('surface', 0.021518415893776462), ('rendering', 0.020715505238603941), ('camera', 0.014019132788939088), ('objects', 0.01206108399440908), ('motion', 0.011630848345313237), ('mesh', 0.011480660975251187), ('visualization', 0.010724877835489385), ('scene', 0.010699183060804422), ('video', 0.010633095485048559), ('interactive', 0.010224953865611241)]

[('logic', 0.025712470436846126), ('schema', 0.025149162184338918), ('database', 0.019840735115588384), ('knowledge', 0.018327830495483154), ('integration', 0.015089953191670721), ('rules', 0.014205785580986922), ('reasoning', 0.013034212231223455), ('constraints', 0.012659411227291155), ('semantics', 0.011357123627257951), ('sources', 0.011232313921219454)]

[('service', 0.041541691765848848), ('services', 0.022297736278338937), ('security', 0.019445902199299384), ('client', 0.016915024644465496), ('protocol', 0.016141270882747066), ('protocols', 0.014581722429205668), ('broadcast', 0.014147572182854952), ('analysis', 0.012629572789233828), ('server', 0.011438493006172475), ('slicing', 0.01065183490458761)]
[('mobile', 0.03360561008759428), ('peer', 0.02750730189627618), ('services', 0.020531029465874043), ('distributed', 0.017747667589930027), ('service', 0.015747456040259629), ('pp', 0.015307701114375138), ('sharing', 0.013042734936522894), ('location', 0.012894037896291802), ('resource', 0.012813309095903839), ('management', 0.01190207285041155)]

[('learning', 0.06256093893265767), ('classification', 0.042068789074943568), ('feature', 0.023514382036446676), ('training', 0.022273245814992921), ('methods', 0.018097941578695011), ('classifier', 0.016483945555555282), ('classifiers', 0.015855856789739494), ('accuracy', 0.014322616043546052), ('selection', 0.013296959211384402), ('svm', 0.012365409160654125)]

[('mining', 0.053579669814611493), ('algorithm', 0.034503986016046449), ('data', 0.033340183836846761), ('patterns', 0.029141562407127914), ('clustering', 0.025793995358818763), ('frequent', 0.024973985745088106), ('algorithms', 0.017201945252255779), ('association', 0.015673712293182811), ('pattern', 0.014994417643764062), ('rules', 0.01355896622926926)]

[('grid', 0.031723186115267837), ('interaction', 0.027193917735845535), ('motion', 0.016553333995235141), ('interface', 0.014404285448855636), ('deformation', 0.012714465689970391), ('input', 0.012609747494116268), ('animation', 0.011628093361869513), ('jobs', 0.011140338741177789), ('shape', 0.009528867280827654 4), ('job', 0.0093648695007153074)]

[('web', 0.095646702199219003), ('search', 0.055166108537673238), ('documents', 0.02989072892927256), ('information', 0.028996986150944971), ('document', 0.025741946045067434), ('pages', 0.023152006202179384), ('text', 0.022013637240811262), ('content', 0.018588888460794734), ('query', 0.017771558601035832), ('page', 0.016637602576578769)]

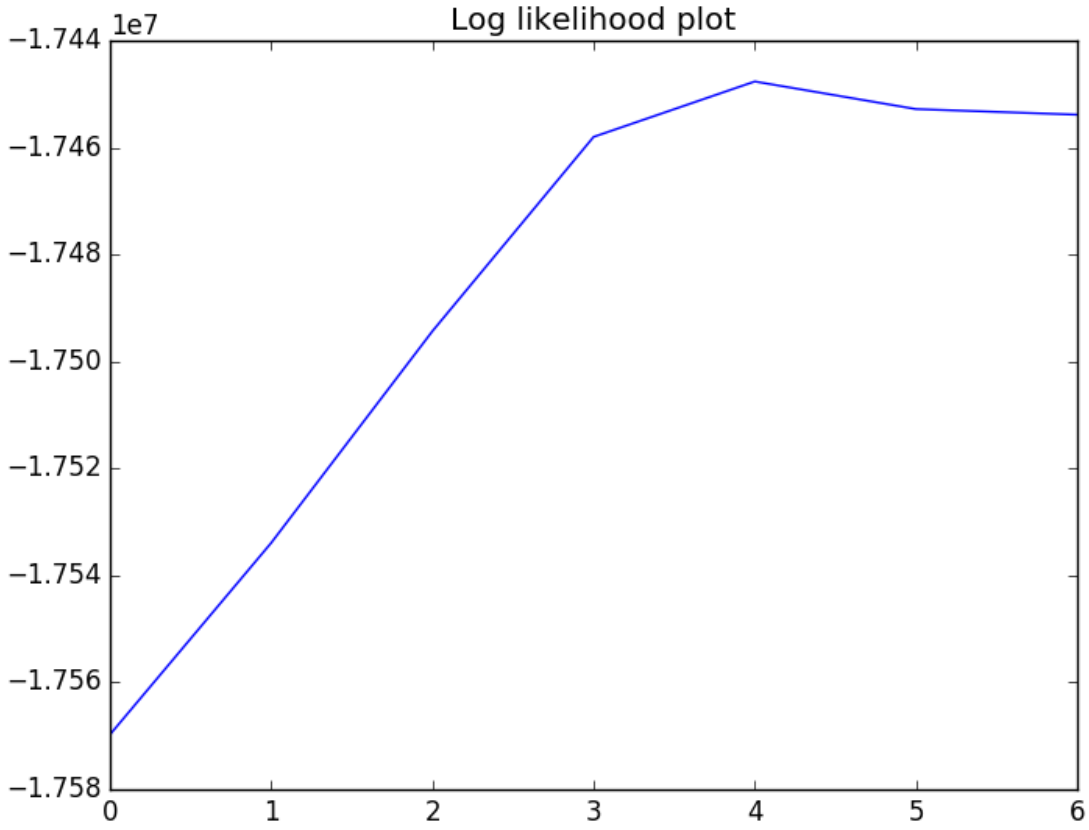
[('trust', 0.024380489083400684), ('parallel', 0.01377079280812967), ('agent', 0.013453935533733547), ('replication', 0.011400032613412697), ('parallelization',

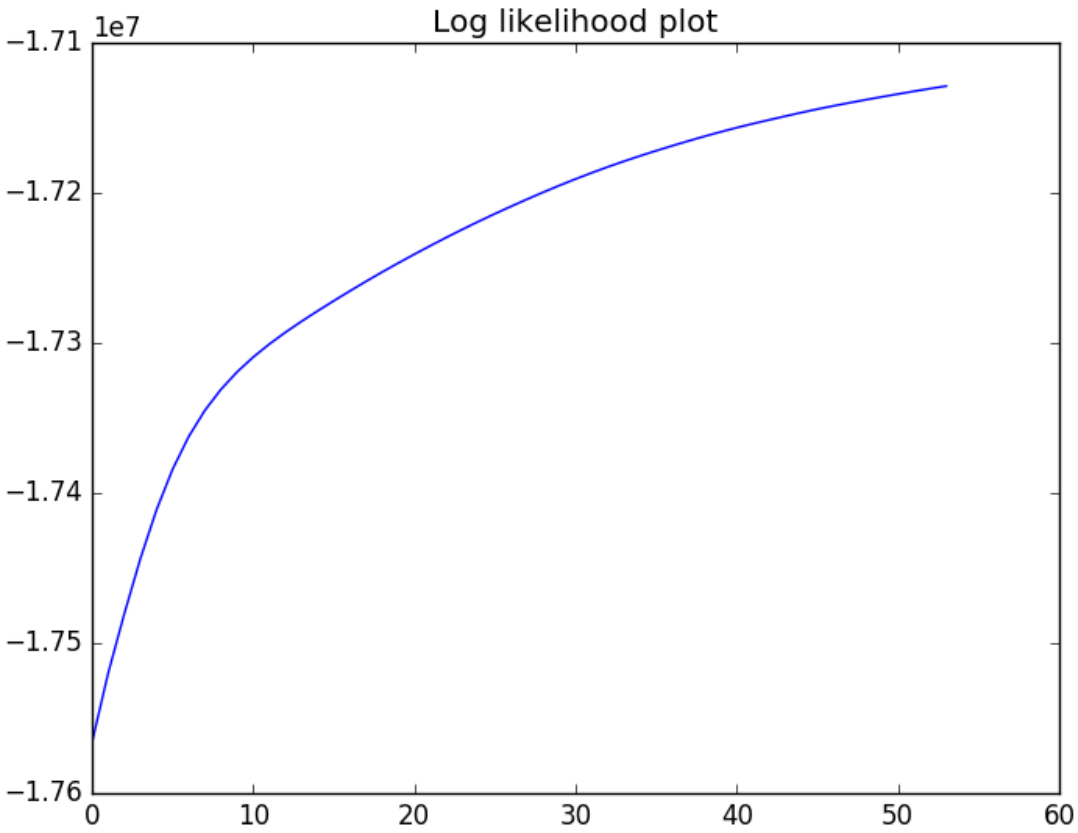
0.010758657531276861), ('matrix', 0.0091492269087972283), ('reputation', 0.0084740313892509404), ('operator', 0.0080692874801193398), ('optimal', 0.0078774621200437527), ('replica', 0.0078348907826617242)]

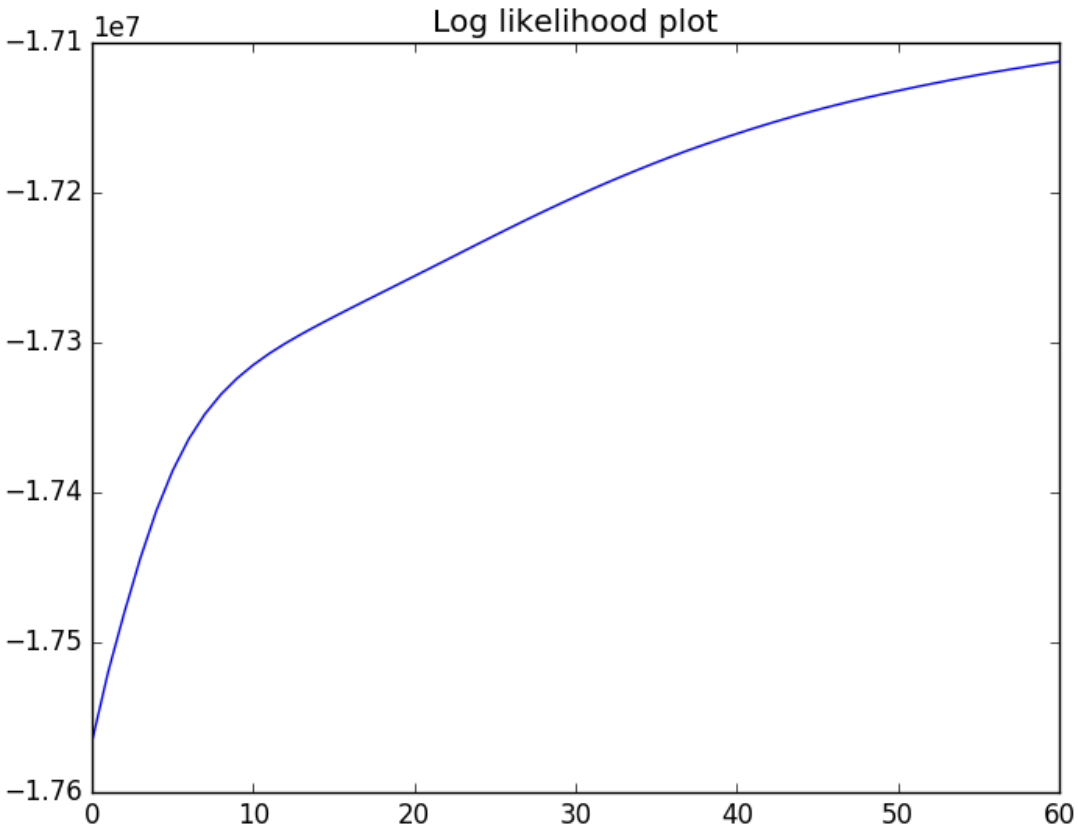
When lambda = 0.9, each topic have is very distinguishable, because with large lambda, the background words and stop words was filtered to the background, on the contrary, with small lambda, the background word mixed up with each topics.
This is what we expected.
If we want high descriptive topics, we should set lambda big.

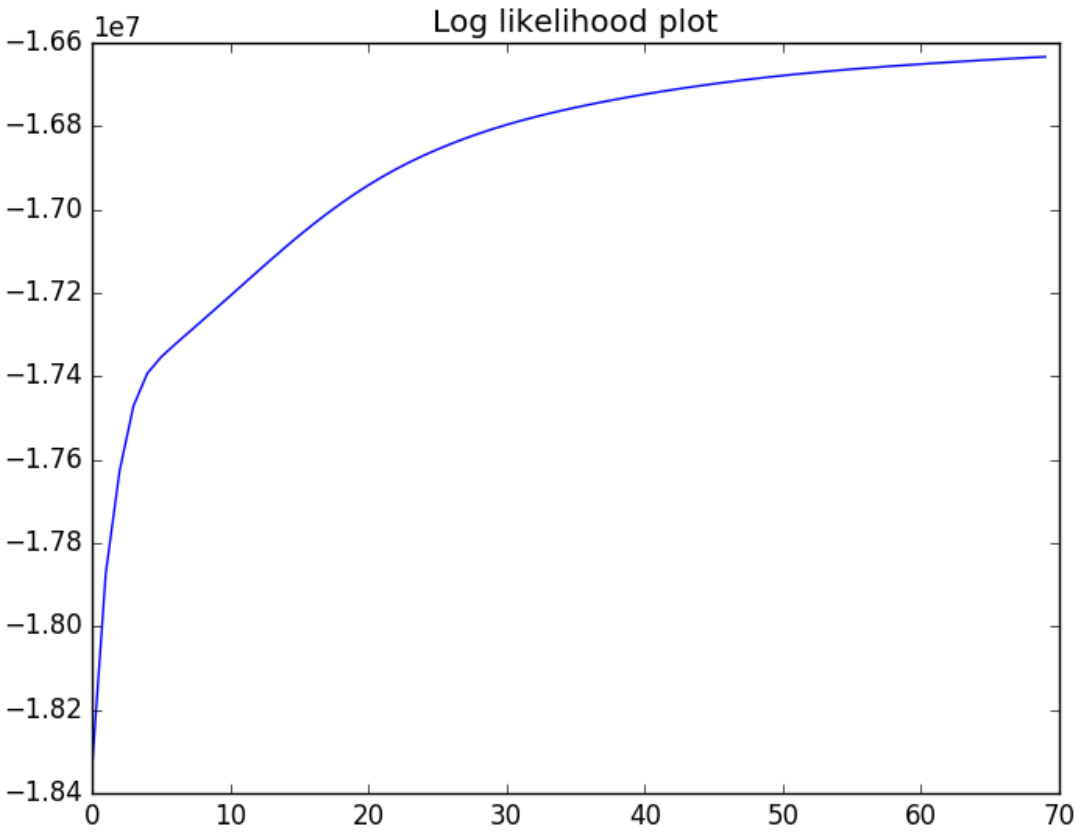
f. we can add more topics, if we didn't set background lambda appropriately, we can add more topics, and in this case, the stop words may be filtered into the new topics we brought in.

g.
Four log likelihood:




Log likelihood plot
1e7
−1.71
−1.72
−1.73
−1.74
−1.75
−1.76
0
10
20
30
40
50
60


Log likelihood plot
1e7
−1.71
−1.72
−1.73
−1.74
−1.75
−1.76
0
10
20
30
40
50
60



We don’t have exactly the same four log likelihood values, this is general true, because we don’t have the same start points, and each would end in different local maximum. If we want to have the highest log likelihood, we would suggest them to run multiple times with different start random points to see which one reaches the highest.