DESCRIPTION

Help a leading mobile brand understand the voice of the customer by analyzing the reviews of their product on **Amazon** and the topics that customers are talking about. Yo
parts of speech. You'll finally interpret the emerging topics.

Problem Statement

A popular mobile phone brand, Lenovo has launched their budget smartphone in the Indian market. The client wants to understand the VOC (voice of the customer) on the
evaluate the current product, but to also get some direction for developing the product pipeline. The client is particularly interested in the different aspects that customers ca
on a leading e-commerce site should provide a good view.

Steps to perform
:

Discover the topics in the reviews and present it to business in a consumable format. Employ techniques in syntactic processing and topic modeling.

Perform specific cleanup, POS tagging, and restricting to relevant POS tags, then, perform topic modeling using LDA. Finally, give business-friendly names to the topics an

Content
:

Dataset: 'K8 Reviews v0.2.csv'

Columns:

Sentiment: The sentiment against the review (4,5 star reviews are positive 😀 , 1,2 are negative 😒)

Reviews: The main text of the review

Tasks
:

<➤> Read the .csv file using Pandas. Take a look at the top few records.

<➤> Normalize casings for the review text and extract the text into a list for easier manipulation.

<➤> Tokenize the reviews using NLTKs word_tokenize function.

<➤> Perform parts-of-speech tagging on each sentence using the NLTK POS tagger.

<➤> For the topic model, we should want to include only nouns.

```
1. Find out all the POS tags that correspond to nouns.

2. Limit the data to only terms with these tags.
```

<➤> Lemmatize.

```
1. Different forms of the terms need to be treated as one.

2. No need to provide POS tag to lemmatizer for now.
```

<➤> Remove stopwords and punctuation (if there are any).

<➤> Create a topic model using LDA on the cleaned-up data with 12 topics.

```
1. Print out the top terms for each topic.

2. What is the coherence of the model with the c_v metric?
```

<➤> Analyze the topics through the business lens.

```
1. Determine which of the topics can be combined.
```

<➤> Create topic model using LDA with what you think is the optimal number of topics

```
1. What is the coherence of the model?
```

<➤> The business should be able to interpret the topics.

```
1. Name each of the identified topics.

2. Create a table with the topic name and the top 10 terms in each to present to the  business.
```

1.Read the .csv file using Pandas. Take a look at the top few records.

2.Normalize casings for the review text

3.extract the text into a list for easier manipulation

4.Tokenize the reviews using NLTKs word_tokenize function

5.Perform parts-of-speech tagging on each sentence using the NLTK POS tagger.

6.Find out all the POS tags that correspond to nouns.

7.Lemmatize.

8.Print out the top terms for each topic.

9.Create topic model using LDA

10.What is the coherence of the model?

11.Create a table with the topic name and the top 10 terms in each to present to the business.

12.What is the coherence of the model?

In [1]:

```
##Import the Required libraries

import pandas as pd
import numpy as np
import tqdm
import nltk
import os
import texthero as hero
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
import gensim
```

Read the .csv file using Pandas. Take a look at the top few records.

In [2]:

```
amazon=pd.read_csv("K8 Reviews v0.2.csv")
amazon.head()
```

Out[2]:

	sentiment	review
0	1	Good but need updates and improvements
1	0	Worst mobile i have bought ever, Battery is dr...
2	1	when I will get my 10% cash back.... its alrea...
3	1	Good
4	0	The worst phone everThey have changed the last...

In [3]:

```
#check the unique values for sentimment and their counts

amazon['sentiment'].value_counts()
```

Out[3]:

```
0    7712
1    6963
Name: sentiment, dtype: int64
```

Normalize casings for the review text

In [4]:

```
stopnltk=stopwords.words('english')
stop_new=['lenevo']
stopnltk=stopnltk+stop_new
```

In [5]:

```
from texthero import preprocessing#create a custom cleaning pipeline
custom_pipeline = [preprocessing.fillna
                   , preprocessing.lowercase
                   , preprocessing.remove_digits
                   , preprocessing.remove_punctuation
                   , preprocessing.remove_diacritics
                   , preprocessing.remove_stopwords
                   , preprocessing.remove_whitespace]
                   #, preprocessing.stem]#pass the custom_pipeline to the pipeline argument
amazon['review'] = hero.clean(amazon['review'], pipeline = custom_pipeline)
amazon.head()
```

Out[5]:

	sentiment	review
0	1	good need updates improvements
1	0	worst mobile bought ever battery draining like...
2	1	get cash back already january
3	1	good
4	0	worst phone everthey changed last phone proble...

extract the text into a list for easier manipulation

In [6]:

```
article0=list(amazon.review.values)
article0[0]
```

Out[6]:

```
'good need updates improvements'
```

In [7]:

```
type(article0)
```

Out[7]:

```
list
```

Tokenize the reviews using NLTKs word_tokenize function

In [8]:

```
article_tkn=[]
for term in article0:
    article_tkn.append(nltk.word_tokenize(term))

article_tkn[0]
```

Out[8]:

```
['good', 'need', 'updates', 'improvements']
```

Perform parts-of-speech tagging on each sentence using the NLTK POS tagger.

In [9]:

```
article_pos=[]

for term in article_tkn:
    article_pos.append(nltk.pos_tag(term))

article_pos[0]
```

Out[9]:

```
[('good', 'JJ'), ('need', 'NN'), ('updates', 'NNS'), ('improvements', 'NNS')]
```

Find out all the POS tags that correspond to nouns.

In [10]:

```
[term for (term,pos) in article_pos[0] if pos.startswith ('NN')]
```

Out[10]:

```
['need', 'updates', 'improvements']
```

In [11]:

```
onlynouns=[]

for term in article_pos:
    onlynouns.append([term for (term,pos) in term if pos.startswith ('N')])

onlynouns[1]
```

Out[11]:

```
['mobile',
 'hell',
 'backup',
 'hours',
 'uses',
 'lie',
 'amazon',
 'lenove',
 'battery',
 'booster',
 'charger',
 'hours',
 'regret']
```

Lemmatize.

In [12]:

```
lemma=WordNetLemmatizer()
article_lema=[]
for term in onlynouns:
    article_lema.append([lemma.lemmatize(word) for word in term])


article_lema[0]
```

Out[12]:

```
['need', 'update', 'improvement']
```

Create a topic model using LDA on the cleaned-up data with 12 topics.

In [13]:

```
dictionary=gensim.corpora.Dictionary(article_lema)
```

In [14]:

```
count=0
for k,v in dictionary.iteritems():
    print(k,v)
    count+=1
    if count>12:
        break
```

```
0 improvement
1 need
2 update
3 amazon
4 backup
5 battery
6 booster
7 charger
8 hell
9 hour
10 lenove
11 lie
12 mobile
```

Print out the top terms for each topic.

In [15]:

```
bow_corpus=[dictionary.doc2bow(doc) for doc in article_lema]
bow_corpus[0]
```

Out[15]:

```
[(0, 1), (1, 1), (2, 1)]
```

In [16]:

```
document_num=25
bow_doc_x=bow_corpus[document_num]
for i in range(len(bow_doc_x)):
    print("word {} (\"{}\") appears {} time in document# {}".format(bow_doc_x[i][0],dictionary[bow_doc_x[i][0]],bow_doc_x[i][1],
                                                                    document_num))
```

```
word 38 ("option") appears 2 time in document# 25
word 50 ("screen") appears 1 time in document# 25
word 71 ("call") appears 1 time in document# 25
word 72 ("cast") appears 1 time in document# 25
word 73 ("hotspot") appears 1 time in document# 25
```

Create topic model using LDA

In [17]:

```
lda_model=gensim.models.LdaMulticore(bow_corpus,
                                     num_topics=12,
                                     id2word=dictionary,
                                     passes=20,
                                     workers=2)
```

In [18]:

```
for idx,topic in lda_model.print_topics():
    print("Topic :{} \n Words :{} ".format(idx,topic))
    print("\n")
```

```
Topic :0
 Words :0.355*"product" + 0.035*"amazon" + 0.030*"return" + 0.016*"experience" + 0.014*"box" + 0.012*"love" + 0.011*"heat" + 0.011*"issue


Topic :1
 Words :0.074*"delivery" + 0.045*"super" + 0.040*"awesome" + 0.033*"gallery" + 0.026*"fast" + 0.024*"thanks" + 0.016*"player" + 0.015*"re
ility"


Topic :2
 Words :0.107*"network" + 0.051*"sim" + 0.043*"problem" + 0.036*"support" + 0.030*"work" + 0.030*"jio" + 0.029*"issue" + 0.029*"handset" -


Topic :3
 Words :0.102*"phone" + 0.047*"issue" + 0.039*"day" + 0.034*"time" + 0.028*"problem" + 0.026*"call" + 0.021*"service" + 0.020*"option" + 0


Topic :4
 Words :0.121*"charger" + 0.060*"hai" + 0.052*"turbo" + 0.044*"h" + 0.020*"ho" + 0.015*"hi" + 0.014*"charge" + 0.013*"bhi" + 0.012*"ka" +


Topic :5
 Words :0.085*"camera" + 0.042*"phone" + 0.037*"battery" + 0.023*"mode" + 0.014*"performance" + 0.013*"feature" + 0.012*"depth" + 0.012*"


Topic :6
 Words :0.157*"note" + 0.097*"money" + 0.088*"k8" + 0.079*"lenovo" + 0.046*"waste" + 0.036*"value" + 0.027*"device" + 0.011*"k4" + 0.010*


Topic :7
 Words :0.101*"performance" + 0.100*"phone" + 0.033*"heat" + 0.024*"issue" + 0.023*"hang" + 0.020*"lot" + 0.018*"time" + 0.015*"review" +


Topic :8
 Words :0.195*"mobile" + 0.053*"glass" + 0.044*"service" + 0.034*"superb" + 0.032*"screen" + 0.028*"gorilla" + 0.015*"item" + 0.015*"use"


Topic :9
 Words :0.206*"camera" + 0.157*"quality" + 0.048*"phone" + 0.035*"sound" + 0.025*"speaker" + 0.020*"display" + 0.016*"good" + 0.015*"head
stem"


Topic :10
 Words :0.274*"battery" + 0.104*"problem" + 0.060*"backup" + 0.051*"heating" + 0.032*"hour" + 0.030*"issue" + 0.028*"charge" + 0.028*"life


Topic :11
 Words :0.413*"phone" + 0.080*"price" + 0.051*"feature" + 0.030*"range" + 0.016*"buy" + 0.016*"budget" + 0.013*"processor" + 0.012*"ok" +
```

What is the coherence of the model?

In [19]:

```
from gensim.models import CoherenceModel
def compute_coherence_values(corpus,texts,dictionary,k,a,b):
    lda_model=gensim.models.LdaMulticore(corpus=corpus,
                                         id2word=dictionary,
                                         random_state=100,
                                         chunksize=100,
                                         passes=20,
                                         alpha=a,
                                         eta=b,
                                         per_word_topics=True)
    coherence_model_lda=CoherenceModel(model=lda_model,texts=texts,dictionary=dictionary,coherence='c_v')
    return coherence_model_lda.get_coherence()
```

In [20]:

```
grid={}
grid['validation_set']={}
#topics range
min_topics=3
max_topics=19
step_size=3
topics_range=range(min_topics,max_topics,step_size)
#Alpha Parameter
alpha=list(np.arange(0.01,1,0.3))
#Beta Parameter
beta=list(np.arange(0.01,1,0.3))
#Validation Sets
num_of_docs=len(bow_corpus)
corpus_sets=[gensim.utils.ClippedCorpus(bow_corpus,num_of_docs*0.75).corpus,bow_corpus]
corpus_sets[0][0]
```

Out[20]:

```
[(0, 1), (1, 1), (2, 1)]
```

In [22]:

```
corpus_title = ['75% Corpus', '100% Corpus']
model_results = {'Validation_Set': [],
                 'Topics': [],
                 'Alpha': [],
                 'Beta': [],
                 'Coherence': []
                }
# Can take a long time to run
if 1 == 1:
    pbar = tqdm.tqdm(total=126)

    # iterate through validation corpuses
    for i in range(len(corpus_sets)):
        # iterate through number of topics
        for k in topics_range:
            # iterate through alpha values
            for a in alpha:
                # iterare through beta values
                for b in beta:
                    # get the coherence score for the given parameters
                    cv =0
                    #print( "i is {}, k is {}, a is {}, b is {}".format(i,k,a,b) )
                    cv = compute_coherence_values(corpus_sets[i],article_lema, dictionary,
                                           k, a, b)
                    print( "i is {}, k is {}, a is {}, b is {} and final cv is {}".format(i,k,a,b,cv) )
                    # Save the model results
                    model_results['Validation_Set'].append(corpus_title[i])
                    model_results['Topics'].append(k)
                    model_results['Alpha'].append(a)
                    model_results['Beta'].append(b)
                    model_results['Coherence'].append(cv)

                    pbar.update(1)
    pd.DataFrame(model_results).to_csv('lda_tuning_results.csv', index=False)
    pbar.close()
```

  1%|▉ | 1/126 [01:43<3:36:21, 103.85s/it]
i is 0, k is 3, a is 0.01, b is 0.01 and final cv is 0.3882555592099666
  2%|▉ | 2/126 [03:22<3:31:41, 102.43s/it]
i is 0, k is 3, a is 0.01, b is 0.31 and final cv is 0.4380762586310488
  2%|▉▉ | 3/126 [04:59<3:26:20, 100.65s/it]
i is 0, k is 3, a is 0.01, b is 0.61 and final cv is 0.41497010132338863
  3%|▉▉▉ | 4/126 [06:35<3:21:32, 99.12s/it]
i is 0, k is 3, a is 0.01, b is 0.9099999999999999 and final cv is 0.4125262561837527
  4%|▉▉▉▉ | 5/126 [08:15<3:20:25, 99.38s/it]
i is 0, k is 3, a is 0.31, b is 0.01 and final cv is 0.4303766645256206
  5%|▉▉▉▉▉ | 6/126 [09:51<3:16:44, 98.37s/it]
i is 0, k is 3, a is 0.31, b is 0.31 and final cv is 0.5266822638891019
  6%|▉▉▉▉▉▉ | 7/126 [11:22<3:10:53, 96.25s/it]
i is 0, k is 3, a is 0.31, b is 0.61 and final cv is 0.5885321093408831
  6%|▉▉▉▉▉▉ | 8/126 [12:52<3:05:55, 94.54s/it]
i is 0, k is 3, a is 0.31, b is 0.9099999999999999 and final cv is 0.6699513202848246
  7%|▉▉▉▉▉▉▉ | 9/126 [14:32<3:07:10, 95.99s/it]
i is 0, k is 3, a is 0.61, b is 0.01 and final cv is 0.4130211612788922
  8%|▉▉▉▉▉▉▉▉ | 10/126 [16:04<3:03:09, 94.74s/it]
i is 0, k is 3, a is 0.61, b is 0.31 and final cv is 0.4410368684446031
  9%|▉▉▉▉▉▉▉▉▉ | 11/126 [17:37<3:00:45, 94.31s/it]
i is 0, k is 3, a is 0.61, b is 0.61 and final cv is 0.23923873579335783
 10%|▉▉▉▉▉▉▉▉▉▉ | 12/126 [19:06<2:56:00, 92.64s/it]
i is 0, k is 3, a is 0.61, b is 0.9099999999999999 and final cv is 0.2664457986806179
 10%|▉▉▉▉▉▉▉▉▉▉ | 13/126 [20:46<2:59:03, 95.08s/it]
i is 0, k is 3, a is 0.9099999999999999, b is 0.01 and final cv is 0.415492830793346
 11%|▉▉▉▉▉▉▉▉▉▉▉ | 14/126 [22:20<2:56:47, 94.71s/it]
i is 0, k is 3, a is 0.9099999999999999, b is 0.31 and final cv is 0.48420202151449165
 12%|▉▉▉▉▉▉▉▉▉▉▉▉ | 15/126 [23:51<2:52:54, 93.46s/it]
i is 0, k is 3, a is 0.9099999999999999, b is 0.61 and final cv is 0.4158645706959013
 13%|▉▉▉▉▉▉▉▉▉▉▉▉▉ | 16/126 [25:21<2:49:27, 92.43s/it]
i is 0, k is 3, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.37694018719582895
 13%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 17/126 [27:03<2:53:18, 95.40s/it]
i is 0, k is 6, a is 0.01, b is 0.01 and final cv is 0.38114302801690947
 14%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 18/126 [28:41<2:53:12, 96.23s/it]
i is 0, k is 6, a is 0.01, b is 0.31 and final cv is 0.4234271575416105
 15%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 19/126 [30:17<2:51:31, 96.19s/it]
i is 0, k is 6, a is 0.01, b is 0.61 and final cv is 0.408699510867609
 16%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 20/126 [31:52<2:49:19, 95.84s/it]
i is 0, k is 6, a is 0.01, b is 0.9099999999999999 and final cv is 0.3989156562484791
 17%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 21/126 [33:33<2:50:17, 97.31s/it]
i is 0, k is 6, a is 0.31, b is 0.01 and final cv is 0.4267673047501773
 17%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 22/126 [35:09<2:47:46, 96.79s/it]
i is 0, k is 6, a is 0.31, b is 0.31 and final cv is 0.5101012248240817
 18%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 23/126 [36:39<2:42:57, 94.93s/it]
i is 0, k is 6, a is 0.31, b is 0.61 and final cv is 0.613788029577681
 19%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 24/126 [38:09<2:38:56, 93.49s/it]
i is 0, k is 6, a is 0.31, b is 0.9099999999999999 and final cv is 0.6708726441835889
 20%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 25/126 [39:52<2:41:54, 96.18s/it]
i is 0, k is 6, a is 0.61, b is 0.01 and final cv is 0.43324865939464174
 21%|▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉▉ | 26/126 [41:29<2:40:35, 96.36s/it]

```
i is 0, k is 6, a is 0.61, b is 0.31 and final cv is 0.42638596149597846
 21%|██████████████▏                                                          | 27/126 [43:02<2:37:21, 95.37s/it]
i is 0, k is 6, a is 0.61, b is 0.61 and final cv is 0.2907266533466791
 22%|███████████████▏                                                         | 28/126 [44:33<2:33:33, 94.02s/it]
i is 0, k is 6, a is 0.61, b is 0.9099999999999999 and final cv is 0.3020718624268877
 23%|████████████████                                                         | 29/126 [46:20<2:38:42, 98.17s/it]
i is 0, k is 6, a is 0.9099999999999999, b is 0.01 and final cv is 0.41409883183590296
 24%|████████████████▉                                                        | 30/126 [47:56<2:35:58, 97.48s/it]
i is 0, k is 6, a is 0.9099999999999999, b is 0.31 and final cv is 0.4861493797995127
 25%|█████████████████▊                                                       | 31/126 [49:29<2:32:10, 96.11s/it]
i is 0, k is 6, a is 0.9099999999999999, b is 0.61 and final cv is 0.4120129558764784
 25%|██████████████████▋                                                      | 32/126 [51:03<2:29:28, 95.41s/it]
i is 0, k is 6, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.38312281320420616
 26%|███████████████████▌                                                     | 33/126 [52:46<2:31:33, 97.78s/it]
i is 0, k is 9, a is 0.01, b is 0.01 and final cv is 0.3849856686509164
 27%|████████████████████▍                                                    | 34/126 [54:29<2:32:22, 99.37s/it]
i is 0, k is 9, a is 0.01, b is 0.31 and final cv is 0.4449383511856151
 28%|█████████████████████▎                                                   | 35/126 [56:05<2:29:11, 98.37s/it]
i is 0, k is 9, a is 0.01, b is 0.61 and final cv is 0.4340651461326259
 29%|██████████████████████▏                                                  | 36/126 [57:42<2:26:34, 97.71s/it]
i is 0, k is 9, a is 0.01, b is 0.9099999999999999 and final cv is 0.42330446588774506
 29%|███████████████████████                                                  | 37/126 [59:25<2:27:17, 99.30s/it]
i is 0, k is 9, a is 0.31, b is 0.01 and final cv is 0.41681135718693274
 30%|███████████████████████▉                                                 | 38/126 [1:01:05<2:26:07, 99.63s/it]
i is 0, k is 9, a is 0.31, b is 0.31 and final cv is 0.5210348438282614
 31%|████████████████████████▊                                                | 39/126 [1:02:39<2:21:51, 97.84s/it]
i is 0, k is 9, a is 0.31, b is 0.61 and final cv is 0.6105043609134588
 32%|█████████████████████████▋                                               | 40/126 [1:04:11<2:18:01, 96.30s/it]
i is 0, k is 9, a is 0.31, b is 0.9099999999999999 and final cv is 0.5324600464864792
 33%|██████████████████████████▌                                              | 41/126 [1:05:54<2:18:53, 98.04s/it]
i is 0, k is 9, a is 0.61, b is 0.01 and final cv is 0.41107059312315464
 33%|███████████████████████████▍                                             | 42/126 [1:07:29<2:16:22, 97.41s/it]
i is 0, k is 9, a is 0.61, b is 0.31 and final cv is 0.42728863095368685
 34%|████████████████████████████▎                                            | 43/126 [1:09:01<2:12:30, 95.78s/it]
i is 0, k is 9, a is 0.61, b is 0.61 and final cv is 0.24720298110087452
 35%|█████████████████████████████▏                                           | 44/126 [1:10:35<2:09:56, 95.08s/it]
i is 0, k is 9, a is 0.61, b is 0.9099999999999999 and final cv is 0.29779974115327795
 36%|██████████████████████████████                                           | 45/126 [1:12:20<2:12:16, 97.98s/it]
i is 0, k is 9, a is 0.9099999999999999, b is 0.01 and final cv is 0.4238093810731966
 37%|██████████████████████████████▉                                          | 46/126 [1:14:01<2:12:07, 99.09s/it]
i is 0, k is 9, a is 0.9099999999999999, b is 0.31 and final cv is 0.4813948224674933
 37%|███████████████████████████████▊                                         | 47/126 [1:15:42<2:11:12, 99.65s/it]
i is 0, k is 9, a is 0.9099999999999999, b is 0.61 and final cv is 0.41894341905205257
 38%|████████████████████████████████▋                                        | 48/126 [1:17:16<2:07:06, 97.78s/it]
i is 0, k is 9, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.37238613572847684
 39%|█████████████████████████████████▌                                       | 49/126 [1:19:05<2:09:53, 101.22s/it]
i is 0, k is 12, a is 0.01, b is 0.01 and final cv is 0.3848655782123114
 40%|██████████████████████████████████▍                                      | 50/126 [1:20:53<2:10:43, 103.21s/it]
i is 0, k is 12, a is 0.01, b is 0.31 and final cv is 0.43926803203161435
 40%|███████████████████████████████████▎                                     | 51/126 [1:22:33<2:07:51, 102.29s/it]
i is 0, k is 12, a is 0.01, b is 0.61 and final cv is 0.419140600588007
```

41%|████████████████████████▍ | 52/126 [1:24:12<2:04:53, 101.26s/it]

i is 0, k is 12, a is 0.01, b is 0.9099999999999999 and final cv is 0.4277662321126458

42%|████████████████████████▊ | 53/126 [1:25:55<2:03:59, 101.91s/it]

i is 0, k is 12, a is 0.31, b is 0.01 and final cv is 0.414913691072357

43%|█████████████████████████▏ | 54/126 [1:27:33<2:00:51, 100.71s/it]

i is 0, k is 12, a is 0.31, b is 0.31 and final cv is 0.5416607537523503

44%|█████████████████████████▌ | 55/126 [1:29:08<1:57:02, 98.91s/it]

i is 0, k is 12, a is 0.31, b is 0.61 and final cv is 0.6155798793258714

44%|█████████████████████████▉ | 56/126 [1:30:42<1:53:53, 97.62s/it]

i is 0, k is 12, a is 0.31, b is 0.9099999999999999 and final cv is 0.6926102644616506

45%|██████████████████████████▎ | 57/126 [1:32:26<1:54:17, 99.38s/it]

i is 0, k is 12, a is 0.61, b is 0.01 and final cv is 0.4174542547167866

46%|██████████████████████████▋ | 58/126 [1:34:01<1:51:06, 98.04s/it]

i is 0, k is 12, a is 0.61, b is 0.31 and final cv is 0.4341000167467995

47%|███████████████████████████ | 59/126 [1:35:34<1:47:46, 96.51s/it]

i is 0, k is 12, a is 0.61, b is 0.61 and final cv is 0.2410159350773721

48%|███████████████████████████▍ | 60/126 [1:37:06<1:44:43, 95.21s/it]

i is 0, k is 12, a is 0.61, b is 0.9099999999999999 and final cv is 0.27025369528962145

48%|███████████████████████████▊ | 61/126 [1:38:53<1:47:03, 98.82s/it]

i is 0, k is 12, a is 0.9099999999999999, b is 0.01 and final cv is 0.4193570936609025

49%|████████████████████████████▏ | 62/126 [1:40:29<1:44:19, 97.81s/it]

i is 0, k is 12, a is 0.9099999999999999, b is 0.31 and final cv is 0.48202692042768613

50%|████████████████████████████▌ | 63/126 [1:42:03<1:41:44, 96.90s/it]

i is 0, k is 12, a is 0.9099999999999999, b is 0.61 and final cv is 0.4140201754896179

51%|████████████████████████████▉ | 64/126 [1:43:37<1:39:01, 95.84s/it]

i is 0, k is 12, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.38552756118110326

52%|█████████████████████████████▎ | 65/126 [1:45:20<1:39:41, 98.06s/it]

i is 0, k is 15, a is 0.01, b is 0.01 and final cv is 0.3814176280315676

52%|█████████████████████████████▋ | 66/126 [1:47:04<1:39:52, 99.87s/it]

i is 0, k is 15, a is 0.01, b is 0.31 and final cv is 0.4524330012422638

53%|██████████████████████████████ | 67/126 [1:48:40<1:37:04, 98.72s/it]

i is 0, k is 15, a is 0.01, b is 0.61 and final cv is 0.4279095568482014

54%|██████████████████████████████▍ | 68/126 [1:50:17<1:34:56, 98.22s/it]

i is 0, k is 15, a is 0.01, b is 0.9099999999999999 and final cv is 0.4145831916481902

55%|██████████████████████████████▊ | 69/126 [1:51:59<1:34:13, 99.19s/it]

i is 0, k is 15, a is 0.31, b is 0.01 and final cv is 0.42105783027581245

56%|███████████████████████████████▏ | 70/126 [1:53:34<1:31:35, 98.14s/it]

i is 0, k is 15, a is 0.31, b is 0.31 and final cv is 0.5293507652138245

56%|███████████████████████████████▌ | 71/126 [1:55:06<1:28:05, 96.10s/it]

i is 0, k is 15, a is 0.31, b is 0.61 and final cv is 0.6110058203078086

57%|███████████████████████████████▉ | 72/126 [1:56:37<1:25:10, 94.64s/it]

i is 0, k is 15, a is 0.31, b is 0.9099999999999999 and final cv is 0.6485298407901995

58%|████████████████████████████████▎ | 73/126 [1:58:17<1:25:02, 96.27s/it]

i is 0, k is 15, a is 0.61, b is 0.01 and final cv is 0.4190702837577503

59%|████████████████████████████████▋ | 74/126 [1:59:49<1:22:21, 95.04s/it]

i is 0, k is 15, a is 0.61, b is 0.31 and final cv is 0.4290261199834309

60%|█████████████████████████████████ | 75/126 [2:01:20<1:19:39, 93.71s/it]

i is 0, k is 15, a is 0.61, b is 0.61 and final cv is 0.2397179996678875

60%|█████████████████████████████████▍ | 76/126 [2:02:49<1:17:03, 92.47s/it]

i is 0, k is 15, a is 0.61, b is 0.9099999999999999 and final cv is 0.26715939021615265

61%|█████████████████████████████████▊ | 77/126 [2:04:31<1:17:42, 95.16s/it]

```
i is 0, k is 15, a is 0.9099999999999999, b is 0.01 and final cv is 0.4172831739782996
 62%|████████████████████████████████████████████████▎                             | 78/126 [2:06:04<1:15:45, 94.70s/it]
i is 0, k is 15, a is 0.9099999999999999, b is 0.31 and final cv is 0.484389439262988
 63%|████████████████████████████████████████████████▉                             | 79/126 [2:07:36<1:13:22, 93.68s/it]
i is 0, k is 15, a is 0.9099999999999999, b is 0.61 and final cv is 0.4175839421451788
 63%|█████████████████████████████████████████████████▌                            | 80/126 [2:09:07<1:11:19, 93.04s/it]
i is 0, k is 15, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.36912723375509593
 64%|██████████████████████████████████████████████████                            | 81/126 [2:10:50<1:11:58, 95.97s/it]
i is 0, k is 18, a is 0.01, b is 0.01 and final cv is 0.38067916687248854
 65%|██████████████████████████████████████████████████▊                           | 82/126 [2:12:34<1:12:10, 98.42s/it]
i is 0, k is 18, a is 0.01, b is 0.31 and final cv is 0.4494214351229595
 66%|███████████████████████████████████████████████████▍                          | 83/126 [2:14:12<1:10:17, 98.09s/it]
i is 0, k is 18, a is 0.01, b is 0.61 and final cv is 0.420976766759084
 67%|████████████████████████████████████████████████████                          | 84/126 [2:15:53<1:09:24, 99.14s/it]
i is 0, k is 18, a is 0.01, b is 0.9099999999999999 and final cv is 0.430323334132151434
 67%|████████████████████████████████████████████████████▌                        | 85/126 [2:17:38<1:08:55, 100.87s/it]
i is 0, k is 18, a is 0.31, b is 0.01 and final cv is 0.42152397518543183
 68%|█████████████████████████████████████████████████████▏                        | 86/126 [2:19:14<1:06:16, 99.42s/it]
i is 0, k is 18, a is 0.31, b is 0.31 and final cv is 0.5234800493042454
 69%|█████████████████████████████████████████████████████▊                        | 87/126 [2:20:54<1:04:48, 99.71s/it]
i is 0, k is 18, a is 0.31, b is 0.61 and final cv is 0.5799604744895173
 70%|██████████████████████████████████████████████████████▍                       | 88/126 [2:22:29<1:02:10, 98.16s/it]
i is 0, k is 18, a is 0.31, b is 0.9099999999999999 and final cv is 0.6970157115035246
 71%|███████████████████████████████████████████████████████                       | 89/126 [2:24:13<1:01:39, 99.98s/it]
i is 0, k is 18, a is 0.61, b is 0.01 and final cv is 0.4070879760740588
 71%|███████████████████████████████████████████████████████▋                        | 90/126 [2:25:53<59:56, 99.90s/it]
i is 0, k is 18, a is 0.61, b is 0.31 and final cv is 0.4329755593157821
 72%|████████████████████████████████████████████████████████▎                       | 91/126 [2:27:25<56:51, 97.48s/it]
i is 0, k is 18, a is 0.61, b is 0.61 and final cv is 0.2431743015095899
 73%|████████████████████████████████████████████████████████▉                       | 92/126 [2:28:54<53:52, 95.07s/it]
i is 0, k is 18, a is 0.61, b is 0.9099999999999999 and final cv is 0.2664457986806179
 74%|█████████████████████████████████████████████████████████▌                      | 93/126 [2:30:35<53:18, 96.92s/it]
i is 0, k is 18, a is 0.9099999999999999, b is 0.01 and final cv is 0.4116189523049355
 75%|██████████████████████████████████████████████████████████▏                     | 94/126 [2:32:09<51:13, 96.05s/it]
i is 0, k is 18, a is 0.9099999999999999, b is 0.31 and final cv is 0.4796599608330167
 75%|██████████████████████████████████████████████████████████▊                     | 95/126 [2:33:40<48:50, 94.54s/it]
i is 0, k is 18, a is 0.9099999999999999, b is 0.61 and final cv is 0.418506548216691
 76%|███████████████████████████████████████████████████████████▍                    | 96/126 [2:35:11<46:38, 93.29s/it]
i is 0, k is 18, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.37971303354285524
 77%|████████████████████████████████████████████████████████████                    | 97/126 [2:36:52<46:17, 95.76s/it]
i is 1, k is 3, a is 0.01, b is 0.01 and final cv is 0.38943977257045753
 78%|████████████████████████████████████████████████████████████▋                   | 98/126 [2:38:30<45:01, 96.47s/it]
i is 1, k is 3, a is 0.01, b is 0.31 and final cv is 0.44341867119778927

 79%|█████████████████████████████████████████████████████████████▎                  | 99/126 [2:40:07<43:23, 96.42s/it]
i is 1, k is 3, a is 0.01, b is 0.61 and final cv is 0.41500053531106446
 79%|█████████████████████████████████████████████████████████████▉                 | 100/126 [2:41:43<41:46, 96.39s/it]
i is 1, k is 3, a is 0.01, b is 0.9099999999999999 and final cv is 0.42823735136520535
 80%|██████████████████████████████████████████████████████████████▌                | 101/126 [2:43:23<40:39, 97.57s/it]
i is 1, k is 3, a is 0.31, b is 0.01 and final cv is 0.4147921039276403
 81%|███████████████████████████████████████████████████████████████▏               | 102/126 [2:44:58<38:43, 96.80s/it]
i is 1, k is 3, a is 0.31, b is 0.31 and final cv is 0.4950298084364224
```

 82%|▊| 103/126 [2:46:31<36:34, 95.39s/it]

i is 1, k is 3, a is 0.31, b is 0.61 and final cv is 0.6134847860305175

 83%|▊| 104/126 [2:48:01<34:24, 93.86s/it]

i is 1, k is 3, a is 0.31, b is 0.9099999999999999 and final cv is 0.685975192485473

 83%|▊| 105/126 [2:49:41<33:30, 95.72s/it]

i is 1, k is 3, a is 0.61, b is 0.01 and final cv is 0.41463188396765005

 84%|▊| 106/126 [2:51:16<31:49, 95.49s/it]

i is 1, k is 3, a is 0.61, b is 0.31 and final cv is 0.42664343327105014

 85%|▊| 107/126 [2:52:53<30:21, 95.87s/it]

i is 1, k is 3, a is 0.61, b is 0.61 and final cv is 0.2432989924139912

 86%|▊| 108/126 [2:54:24<28:23, 94.62s/it]

i is 1, k is 3, a is 0.61, b is 0.9099999999999999 and final cv is 0.2699973461389365

 87%|▊| 109/126 [2:56:06<27:24, 96.74s/it]

i is 1, k is 3, a is 0.9099999999999999, b is 0.01 and final cv is 0.4278101822991737

 87%|▊| 110/126 [2:57:41<25:40, 96.26s/it]

i is 1, k is 3, a is 0.9099999999999999, b is 0.31 and final cv is 0.4774077478339174

 88%|▊| 111/126 [2:59:13<23:43, 94.90s/it]

i is 1, k is 3, a is 0.9099999999999999, b is 0.61 and final cv is 0.42083241949122735

 89%|▊| 112/126 [3:00:44<21:54, 93.87s/it]

i is 1, k is 3, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.3845606919323589

 90%|▉| 113/126 [3:02:27<20:55, 96.56s/it]

i is 1, k is 6, a is 0.01, b is 0.01 and final cv is 0.38440691864770815

 90%|▉| 114/126 [3:04:05<19:24, 97.08s/it]

i is 1, k is 6, a is 0.01, b is 0.31 and final cv is 0.4414375881813523

 91%|▉| 115/126 [3:05:42<17:44, 96.81s/it]

i is 1, k is 6, a is 0.01, b is 0.61 and final cv is 0.43047506884065156

 92%|▉| 116/126 [3:07:18<16:06, 96.63s/it]

i is 1, k is 6, a is 0.01, b is 0.9099999999999999 and final cv is 0.4140081037201993

 93%|▉| 117/126 [3:08:59<14:41, 97.92s/it]

i is 1, k is 6, a is 0.31, b is 0.01 and final cv is 0.4189438179935906

 94%|▉| 118/126 [3:10:36<13:00, 97.58s/it]

i is 1, k is 6, a is 0.31, b is 0.31 and final cv is 0.5279497050200496

 94%|▉| 119/126 [3:12:08<11:12, 96.09s/it]

i is 1, k is 6, a is 0.31, b is 0.61 and final cv is 0.5948475479690934

 95%|▉| 120/126 [3:13:39<09:27, 94.57s/it]

i is 1, k is 6, a is 0.31, b is 0.9099999999999999 and final cv is 0.678840762344901

 96%|▉| 121/126 [3:15:20<08:02, 96.45s/it]

i is 1, k is 6, a is 0.61, b is 0.01 and final cv is 0.4328410603507472

 97%|▉| 122/126 [3:16:53<06:21, 95.36s/it]

i is 1, k is 6, a is 0.61, b is 0.31 and final cv is 0.4288742424203636

 98%|▉| 123/126 [3:18:24<04:42, 94.20s/it]

i is 1, k is 6, a is 0.61, b is 0.61 and final cv is 0.24167095483356413

 98%|▉| 124/126 [3:19:56<03:06, 93.46s/it]

i is 1, k is 6, a is 0.61, b is 0.9099999999999999 and final cv is 0.31152017165720886

 99%|▉| 125/126 [3:21:38<01:36, 96.02s/it]

i is 1, k is 6, a is 0.9099999999999999, b is 0.01 and final cv is 0.42714357313186574

100%|▉| 126/126 [3:23:13<00:00, 95.60s/it]

i is 1, k is 6, a is 0.9099999999999999, b is 0.31 and final cv is 0.47622940059462776

127it [3:24:44, 94.33s/it]

i is 1, k is 6, a is 0.9099999999999999, b is 0.61 and final cv is 0.417981357542426

128it [3:50:23, 527.76s/it]

i is 1, k is 6, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.3809655941186651

129it [3:52:07, 400.49s/it]

i is 1, k is 9, a is 0.01, b is 0.01 and final cv is 0.375312935850365

130it [3:53:52, 311.99s/it]

i is 1, k is 9, a is 0.01, b is 0.31 and final cv is 0.43860211387509596

131it [3:55:37, 249.94s/it]

i is 1, k is 9, a is 0.01, b is 0.61 and final cv is 0.4120859350049133

132it [3:57:19, 205.59s/it]

i is 1, k is 9, a is 0.01, b is 0.9099999999999999 and final cv is 0.42099414997193024

133it [3:59:06, 175.77s/it]

i is 1, k is 9, a is 0.31, b is 0.01 and final cv is 0.42003690510426606

134it [4:00:50, 154.47s/it]

i is 1, k is 9, a is 0.31, b is 0.31 and final cv is 0.5336740783991056

135it [4:02:31, 138.17s/it]

i is 1, k is 9, a is 0.31, b is 0.61 and final cv is 0.6139104792483356

136it [4:04:09, 126.26s/it]

i is 1, k is 9, a is 0.31, b is 0.9099999999999999 and final cv is 0.6816037619986307

137it [4:05:56, 120.37s/it]

i is 1, k is 9, a is 0.61, b is 0.01 and final cv is 0.42564055990864136

138it [4:07:34, 113.83s/it]

i is 1, k is 9, a is 0.61, b is 0.31 and final cv is 0.4094487306651065

139it [4:09:16, 110.25s/it]

i is 1, k is 9, a is 0.61, b is 0.61 and final cv is 0.25711832600086987

140it [4:10:55, 106.98s/it]

i is 1, k is 9, a is 0.61, b is 0.9099999999999999 and final cv is 0.3005331295492658

141it [4:12:44, 107.34s/it]

i is 1, k is 9, a is 0.9099999999999999, b is 0.01 and final cv is 0.41363730735032717

142it [4:14:25, 105.57s/it]

i is 1, k is 9, a is 0.9099999999999999, b is 0.31 and final cv is 0.47917319347747783

143it [4:16:04, 103.48s/it]

i is 1, k is 9, a is 0.9099999999999999, b is 0.61 and final cv is 0.41901902114740835

144it [4:17:41, 101.72s/it]

i is 1, k is 9, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.3850905146088963

145it [4:19:31, 104.10s/it]

i is 1, k is 12, a is 0.01, b is 0.01 and final cv is 0.38195675853947597

146it [4:21:18, 104.98s/it]

i is 1, k is 12, a is 0.01, b is 0.31 and final cv is 0.4372370523757942

147it [4:23:02, 104.64s/it]

i is 1, k is 12, a is 0.01, b is 0.61 and final cv is 0.42455640929029914

148it [4:24:45, 104.28s/it]

i is 1, k is 12, a is 0.01, b is 0.9099999999999999 and final cv is 0.42997036062245203

149it [4:26:34, 105.56s/it]

i is 1, k is 12, a is 0.31, b is 0.01 and final cv is 0.43327139951157634

150it [4:28:16, 104.40s/it]

i is 1, k is 12, a is 0.31, b is 0.31 and final cv is 0.5266710804088617

151it [4:31:09, 124.99s/it]

i is 1, k is 12, a is 0.31, b is 0.61 and final cv is 0.5281082728329707

152it [4:34:02, 139.60s/it]

i is 1, k is 12, a is 0.31, b is 0.9099999999999999 and final cv is 0.6527262078275798

153it [4:37:23, 158.06s/it]

i is 1, k is 12, a is 0.61, b is 0.01 and final cv is 0.4340818680017015

154it [4:40:31, 167.01s/it]

i is 1, k is 12, a is 0.61, b is 0.31 and final cv is 0.4196288389079814

155it [4:43:27, 169.49s/it]

i is 1, k is 12, a is 0.61, b is 0.61 and final cv is 0.24565778727867243

156it [4:46:21, 170.89s/it]

i is 1, k is 12, a is 0.61, b is 0.9099999999999999 and final cv is 0.26644579868061785

157it [4:49:41, 179.71s/it]

i is 1, k is 12, a is 0.9099999999999999, b is 0.01 and final cv is 0.4230339538439726

158it [4:52:44, 180.82s/it]

i is 1, k is 12, a is 0.9099999999999999, b is 0.31 and final cv is 0.481485757390164

159it [4:55:40, 179.37s/it]

i is 1, k is 12, a is 0.9099999999999999, b is 0.61 and final cv is 0.44852812175328294

160it [4:58:34, 177.66s/it]

i is 1, k is 12, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.4760037235631593

161it [5:02:02, 186.76s/it]

i is 1, k is 15, a is 0.01, b is 0.01 and final cv is 0.3846933454180325

162it [5:05:17, 189.35s/it]

i is 1, k is 15, a is 0.01, b is 0.31 and final cv is 0.43152985597173704

163it [5:08:27, 189.50s/it]

i is 1, k is 15, a is 0.01, b is 0.61 and final cv is 0.3992852288180545

164it [5:11:37, 189.56s/it]

i is 1, k is 15, a is 0.01, b is 0.9099999999999999 and final cv is 0.4068505801457858

165it [5:15:02, 194.13s/it]

i is 1, k is 15, a is 0.31, b is 0.01 and final cv is 0.4175211014302391

166it [5:18:10, 192.29s/it]

i is 1, k is 15, a is 0.31, b is 0.31 and final cv is 0.5255246806803391

167it [5:21:04, 186.94s/it]

i is 1, k is 15, a is 0.31, b is 0.61 and final cv is 0.6102520058889652

168it [5:23:58, 182.86s/it]

i is 1, k is 15, a is 0.31, b is 0.9099999999999999 and final cv is 0.6519089398295763

169it [5:27:20, 188.75s/it]

i is 1, k is 15, a is 0.61, b is 0.01 and final cv is 0.4053179078217494

170it [5:30:26, 187.93s/it]

i is 1, k is 15, a is 0.61, b is 0.31 and final cv is 0.4178413950435354

171it [5:33:22, 184.39s/it]

i is 1, k is 15, a is 0.61, b is 0.61 and final cv is 0.2422135931911789

172it [5:36:16, 181.20s/it]

i is 1, k is 15, a is 0.61, b is 0.9099999999999999 and final cv is 0.2797183747594749

173it [5:39:35, 186.58s/it]

i is 1, k is 15, a is 0.9099999999999999, b is 0.01 and final cv is 0.42010573108856597

174it [5:42:39, 185.76s/it]

i is 1, k is 15, a is 0.9099999999999999, b is 0.31 and final cv is 0.480266450120485

175it [5:45:35, 182.74s/it]

i is 1, k is 15, a is 0.9099999999999999, b is 0.61 and final cv is 0.4617982774025531

176it [5:48:31, 180.67s/it]

i is 1, k is 15, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.457894646149277

177it [5:51:54, 187.44s/it]

i is 1, k is 18, a is 0.01, b is 0.01 and final cv is 0.37831492393289656

178it [5:55:04, 188.40s/it]

i is 1, k is 18, a is 0.01, b is 0.31 and final cv is 0.43759500891279307

179it [5:58:12, 188.31s/it]

```
i is 1, k is 18, a is 0.01, b is 0.61 and final cv is 0.4149764552589741
180it [6:01:17, 187.17s/it]
i is 1, k is 18, a is 0.01, b is 0.9099999999999999 and final cv is 0.41785086834162194
181it [6:04:43, 192.74s/it]
i is 1, k is 18, a is 0.31, b is 0.01 and final cv is 0.4206994566674523
182it [6:07:50, 191.07s/it]
i is 1, k is 18, a is 0.31, b is 0.31 and final cv is 0.5143582048165386
183it [6:10:47, 186.77s/it]
i is 1, k is 18, a is 0.31, b is 0.61 and final cv is 0.5955462159892985
184it [6:13:37, 181.76s/it]
i is 1, k is 18, a is 0.31, b is 0.9099999999999999 and final cv is 0.6490604215488949
185it [6:16:58, 187.76s/it]
i is 1, k is 18, a is 0.61, b is 0.01 and final cv is 0.40541428003405094
186it [6:19:58, 185.38s/it]
i is 1, k is 18, a is 0.61, b is 0.31 and final cv is 0.4215553519441639
187it [6:22:50, 181.15s/it]
i is 1, k is 18, a is 0.61, b is 0.61 and final cv is 0.24334725017709583
188it [6:24:29, 156.74s/it]
i is 1, k is 18, a is 0.61, b is 0.9099999999999999 and final cv is 0.26644579868061785
189it [6:26:18, 142.20s/it]
i is 1, k is 18, a is 0.9099999999999999, b is 0.01 and final cv is 0.42653550166716786
190it [6:27:59, 129.91s/it]
i is 1, k is 18, a is 0.9099999999999999, b is 0.31 and final cv is 0.4798533982144269
191it [6:29:37, 120.30s/it]
i is 1, k is 18, a is 0.9099999999999999, b is 0.61 and final cv is 0.4186159837157392
192it [6:31:14, 122.27s/it]
i is 1, k is 18, a is 0.9099999999999999, b is 0.9099999999999999 and final cv is 0.39262346921900415
```

In [25]:

```
tuneResults = pd.read_csv('lda_tuning_results.csv')
```

In [26]:

```
tuneResults[tuneResults.Coherence == tuneResults.Coherence.values.ravel().max()]
```

Out[26]:

	Validation_Set	Topics	Alpha	Beta	Coherence
87	75% Corpus	18	0.31	0.91	0.697016

In [27]:

```
tuneResults[tuneResults.Coherence > 0.58]
```

Out[27]:

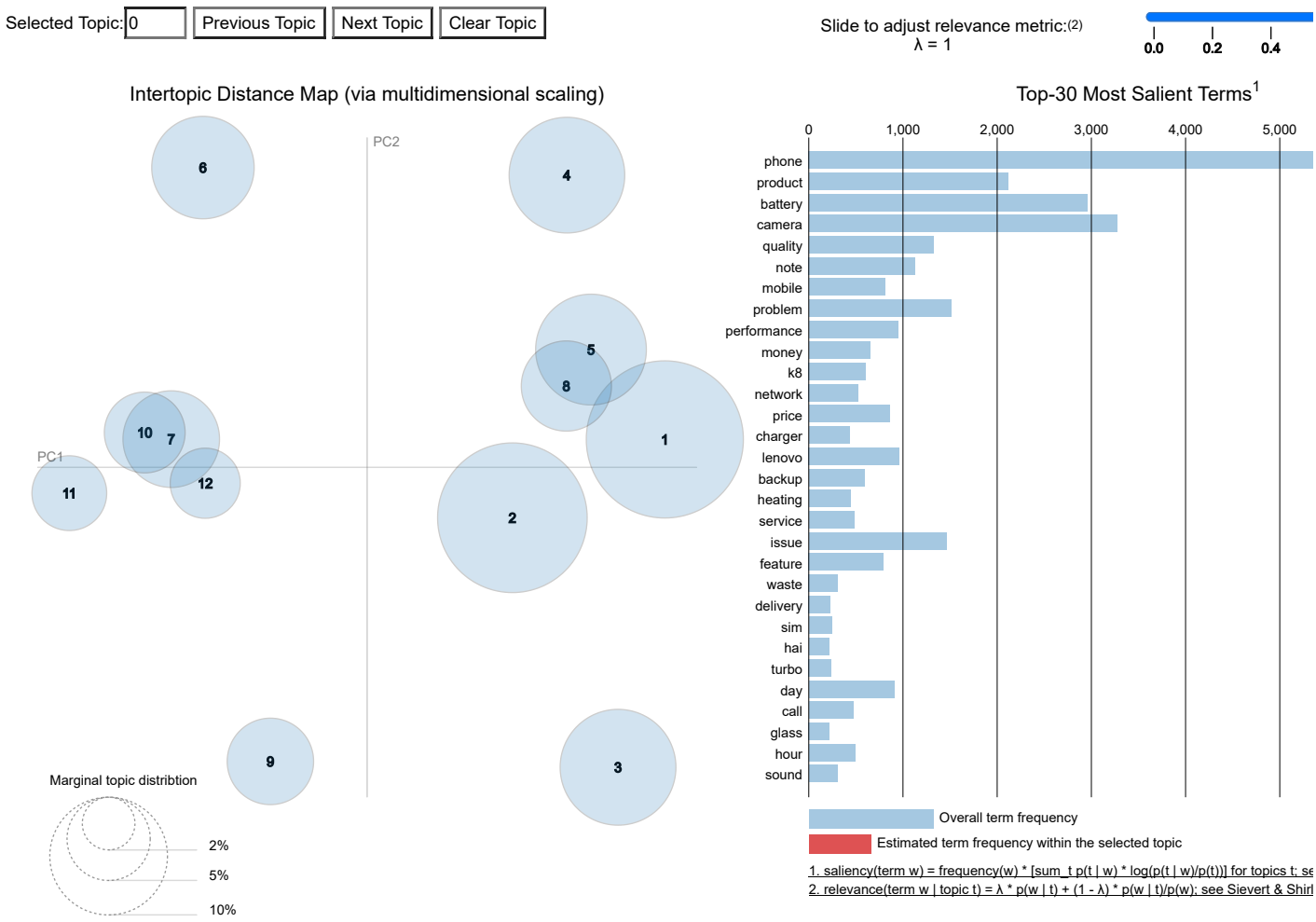
	Validation_Set	Topics	Alpha	Beta	Coherence
6	75% Corpus	3	0.31	0.61	0.588532
7	75% Corpus	3	0.31	0.91	0.669951
22	75% Corpus	6	0.31	0.61	0.613788
23	75% Corpus	6	0.31	0.91	0.670873
38	75% Corpus	9	0.31	0.61	0.610504
54	75% Corpus	12	0.31	0.61	0.615580
55	75% Corpus	12	0.31	0.91	0.692610
70	75% Corpus	15	0.31	0.61	0.611006
71	75% Corpus	15	0.31	0.91	0.648530
87	75% Corpus	18	0.31	0.91	0.697016
102	100% Corpus	3	0.31	0.61	0.613485
103	100% Corpus	3	0.31	0.91	0.685975
118	100% Corpus	6	0.31	0.61	0.594848
119	100% Corpus	6	0.31	0.91	0.678841
134	100% Corpus	9	0.31	0.61	0.613910
135	100% Corpus	9	0.31	0.91	0.681604
151	100% Corpus	12	0.31	0.91	0.652726
166	100% Corpus	15	0.31	0.61	0.610252
167	100% Corpus	15	0.31	0.91	0.651909
182	100% Corpus	18	0.31	0.61	0.595546
183	100% Corpus	18	0.31	0.91	0.649060

Create a table with the topic name and the top 10 terms in each to present to the business.

In [24]:

```
import pyLDAvis.gensim
import pickle
import pyLDAvis
# Visualize the topics
pyLDAvis.enable_notebook()
LDAvis_prepared = pyLDAvis.gensim.prepare(lda_model, corpus_sets[0], dictionary)
LDAvis_prepared
```

Out[24]:



In [ ]: