Load libraries and data

```
!pip install --upgrade pythainlp
!pip install pyLDAvis
```

```
Requirement already satisfied: pythainlp in /usr/local/lib/python3.7/dist-packages (2.3.2)
Requirement already satisfied: requests>=2.22.0 in /usr/local/lib/python3.7/dist-packages (from py
Requirement already satisfied: python-crfsuite>=0.9.6 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: tinydb>=3.0 in /usr/local/lib/python3.7/dist-packages (from pythair
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.7
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from reques
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from r
Requirement already satisfied: typing-extensions<4.0.0,>=3.10.0 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: pyLDAvis in /usr/local/lib/python3.7/dist-packages (3.3.1)
Requirement already satisfied: pandas>=1.2.0 in /usr/local/lib/python3.7/dist-packages (from pyLDA
Requirement already satisfied: jinja2 in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (2
Requirement already satisfied: funcy in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (1.
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (1.
Requirement already satisfied: sklearn in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (
Requirement already satisfied: gensim in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (3
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (1
Requirement already satisfied: numpy>=1.20.0 in /usr/local/lib/python3.7/dist-packages (from pyLDA
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-packages (from pyLDAv
Requirement already satisfied: future in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (0
Requirement already satisfied: setuptools in /usr/local/lib/python3.7/dist-packages (from pyLDAvis
Requirement already satisfied: numexpr in /usr/local/lib/python3.7/dist-packages (from pyLDAvis) (
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dat
Requirement already satisfied: smart-open>=1.2.1 in /usr/local/lib/python3.7/dist-packages (from g
Requirement already satisfied: MarkupSafe>=0.23 in /usr/local/lib/python3.7/dist-packages (from ji
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-packages (fro
```

```
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore", category=FutureWarning)

import pandas as pd
import numpy as np
import re
import string
from pprint import pprint
import matplotlib.pyplot as plt
import seaborn as sns
import networkx as nx


import pythainlp
from pythainlp.tokenize import sent_tokenize, word_tokenize
import gensim

# import tensorflow_hub as hub
# import tensorflow_text
```

```
# import tensorflow as tf

import pyLDAvis.gensim_models
pyLDAvis.enable_notebook()
```

```
/usr/local/lib/python3.7/dist-packages/past/types/oldstr.py:5: DeprecationWarning: Using or import
  from collections import Iterable
```

```
import matplotlib as mpl
mpl.font_manager.fontManager.addfont('THSarabunNew.ttf')
mpl.rc('font', family='TH Sarabun New', size=20, weight=400)


df = pd.read_csv('Wongnai Reviews - Small.csv')


df.tail()
```

	Review ID	Review
295	296	ค่ำนี้คุณเพื่อนอยากส้มตำ หมูเฮาเลยพากันลงมากิน...
296	297	ร้านสะอาดดี ตกแต่งสวยงาม มีที่จอดรถ ราคาเมนูต่...
297	298	เช้าๆ รีบๆ วิ่งมาเข่าห้องเรียนแทบไม่ทันแต่ต้อง...
298	299	ร้านนี้เป็นร้านกาแฟเล็กๆ ข้างๆ ร้านๆ Happy Man...
299	300	ทรูคอฟฟี่สาขาซีคอนอยู่ในศูนย์บริการของทรู ชั้น...

Tokenize Words

```
stopwords = list(pythainlp.corpus.thai_stopwords())
removed_words = [' ', '  ', '\n', 'ร้าน', '(', ')','ๆ','-','–','.',',','{','}',':',';','/','>','<','"']
screening_words = stopwords + removed_words

def tokenize_sentence(sentence):
  merged = ''
  words = pythainlp.word_tokenize(str(sentence), engine='newmm')
  for word in words:
    if word not in screening_words:
      if merged == '':
        merged = word
      else:
        merged = merged + ',' + word
  return merged


df['Review_tokenized'] = df['Review'].apply(lambda x: tokenize_sentence(x))


df.tail()
```

	Review ID	Review	Review_tokenized
295	296	ค่ำนี้คุณเพื่อนอยากส้มตำ หมูเฮาเลยพากันลงมากิน...	ค่ำ,เพื่อน,ส้มตำ,หมู,เฮา,ลงมา,กิน,ส้มตำ,ออฟฟิศ...
296	297	ร้านสะอาดดี ตกแต่งสวยงาม มีที่จอดรถ ราคาเมนูต่...	สะอาด,ดี,ตกแต่ง,สวยงาม,ที่จอดรถ,ราคา,เมนู,เทีย...
297	298	เช้าๆ รีบๆ วิ่งมาเข่าห้องเรียนแทบไม่ทันแต่ต้อง...	เช้า,รีบ,วิ่ง,เข่า,ห้องเรียน,แทบ,ต้องหา,ของกิน...
298	299	ร้านนี้เป็นร้านกาแฟเล็กๆ ข้างๆ ร้านๆ Happy Man...	ร้านกาแฟ,Happy,Mango,อาทิตย์,นัด,เพื่อน,นั่ง,ค...

Create Dictionary

```
documents = df['Review_tokenized'].to_list()
texts = [[text for text in doc.split(',')] for doc in documents]
dictionary = gensim.corpora.Dictionary(texts)
```

```
print(dictionary.token2id.keys())
```

```
dict_keys(['20', 'Macchiato', 'กาแฟ', 'กาแฟร้อน', 'กิน', 'คน', 'ครึ่ง', 'ความคิด', 'ชอบ', 'ดื่ม', 'ตอน'
```

```
gensim_corpus = [dictionary.doc2bow(text, allow_update=True) for text in texts]
word_frequencies = [[(dictionary[id], frequence) for id, frequence in couple] for couple in gensim_corp
```

```
print(word_frequencies)
```

```
[[('20', 1), ('Macchiato', 2), ('กาแฟ', 1), ('กาแฟร้อน', 1), ('กิน', 1), ('คน', 1), ('ครึ่ง', 1), ('ค
```

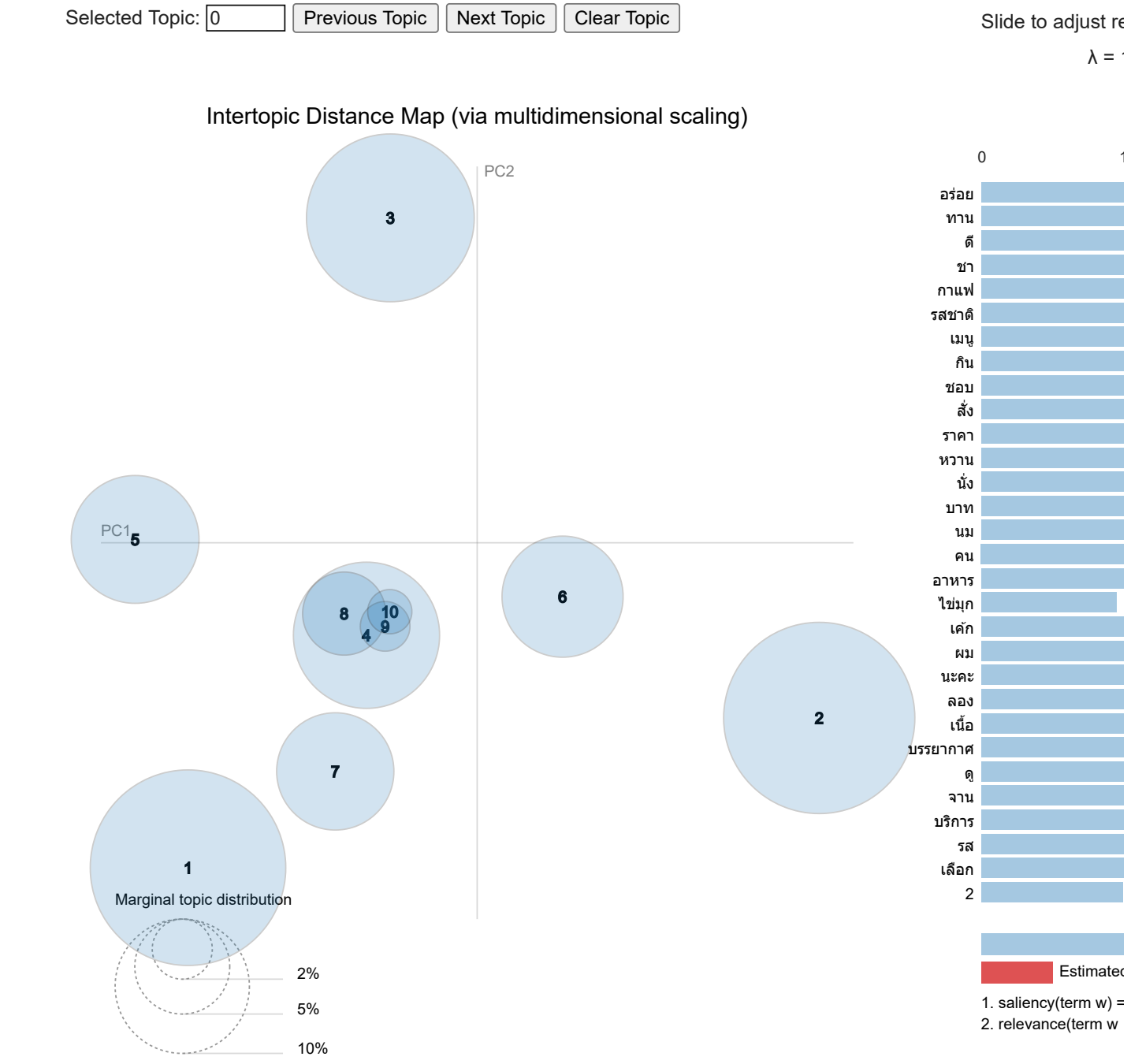
Topic Modeling

```
num_topics = 10
chunksize = 4000 # size of the doc looked at every pass
passes = 20 # number of passes through documents
iterations = 50
eval_every = 1  # Don't evaluate model perplexity, takes too much time.

# Make a index to word dictionary.
temp = dictionary[0]  # This is only to "load" the dictionary.
id2word = dictionary.id2token

%time model = gensim.models.LdaModel(corpus=gensim_corpus, id2word=id2word, chunksize=chunksize, \
                       alpha='auto', eta='auto', \
                       iterations=iterations, num_topics=num_topics, \
                       passes=passes, eval_every=eval_every)
```

```
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
```

```
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call

  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
/usr/local/lib/python3.7/dist-packages/gensim/models/ldamodel.py:1077: DeprecationWarning: Call
  score += np.sum(cnt * logsumexp(Elogthetad + Elogbeta[:, int(id)]) for id, cnt in doc)
CPU times: user 11.5 s, sys: 1.39 s, total: 12.9 s
Wall time: 11.5 s
```

```
pyLDAvis.gensim_models.prepare(model, gensim_corpus, dictionary)
```




```
model.show_topic(3)
```

```
[('ทาน', 0.013288857),
 ('อร่อย', 0.012208763),
 ('รสชาติ', 0.012171247),
 ('อาหาร', 0.010088017),
 ('ดี', 0.00995331),
 ('กาแฟ', 0.009764029),
 ('บาท', 0.009613954),
 ('ราคา', 0.0091974735),
 ('สั่ง', 0.008951393),
 ('เมนู', 0.008182236)]
```

```
result = []
topn = 15
```

```
for n in range(num_topics):
    temp_df = pd.DataFrame(model.get_topic_terms(n, topn=topn), columns=['word_id','prob'])
    temp_df['topic'] = n
    result.append(temp_df)

topic_terms_df = pd.concat(result)
topic_terms_df['word'] = topic_terms_df['word_id'].apply(lambda x: dictionary.get(x))
topic_terms_df.head()
```

	word_id	prob	topic	word
0	11	0.011818	0	ทาน
1	156	0.011723	0	อร่อย
2	38	0.011249	0	ดี
3	45	0.010717	0	รสชาติ
4	2	0.008100	0	กาแฟ

```
topic_terms_df['word'].unique()
```

```
array(['ทาน', 'อร่อย', 'ดี', 'รสชาติ', 'กาแฟ', 'สั่ง', 'เมนู', 'ราคา',
       'ชา', 'กิน', 'อาหาร', 'ชอบ', 'คน', 'บาท', 'นั่ง', 'นม', 'ไข่มุก',
       'เค้ก', 'หวาน', 'ลอง', 'จาน', 'ผม', 'นะคะ', 'ยำ', 'เนื้อ', 'รส',
       'กก', 'พาย'], dtype=object)
```

```
topic_terms_df.to_pickle('topic_term.pkl')
```

```
df['topics'] = df['Review_tokenized'].apply(lambda x: model.get_document_topics(dictionary.doc2bow(x.sp
df['score'] = df['Review_tokenized'].apply(lambda x: model.get_document_topics(dictionary.doc2bow(x.spl
```

```
df[df['topics'] == 3]
```

	Review		
1	เป็นคนที่ชอบทาน Macchiato	คน,ชอบ,ทาน,Macchiato,เป็น	0.994915

```
df.to_pickle('result.pkl')
```

Result

```
df = pd.read_pickle('result.pkl')
topic_terms_df = pd.read_pickle('topic_term.pkl')
```

2	3	\nร้านเบอเกอรีร้านนี้อยู่ในร้านล้าง			

```
topic_terms_df.word.unique()
```

```
array(['ทาน', 'อร่อย', 'ดี', 'รสชาติ', 'กาแฟ', 'สั่ง', 'เมนู', 'ราคา',
       'ชา', 'กิน', 'อาหาร', 'ชอบ', 'คน', 'บาท', 'นั่ง', 'นม', 'ไข่มุก',
       'เค้ก', 'หวาน', 'ลอง', 'จาน', 'ผม', 'นะคะ', 'ยำ', 'เนื้อ', 'รส',
       'กก', 'พาย'], dtype=object)
```

275	276		รีวิว,ลอง,ทาน,บ้าน,หม่บ้าน,รัชดา,นิเวศน์,แยก,...	3	0.998691

```
ttdf = topic_terms_df.sort_values(['topic','prob'],ascending=[True, False]).groupby('topic').head(10)
ttdf['word'].unique()
```

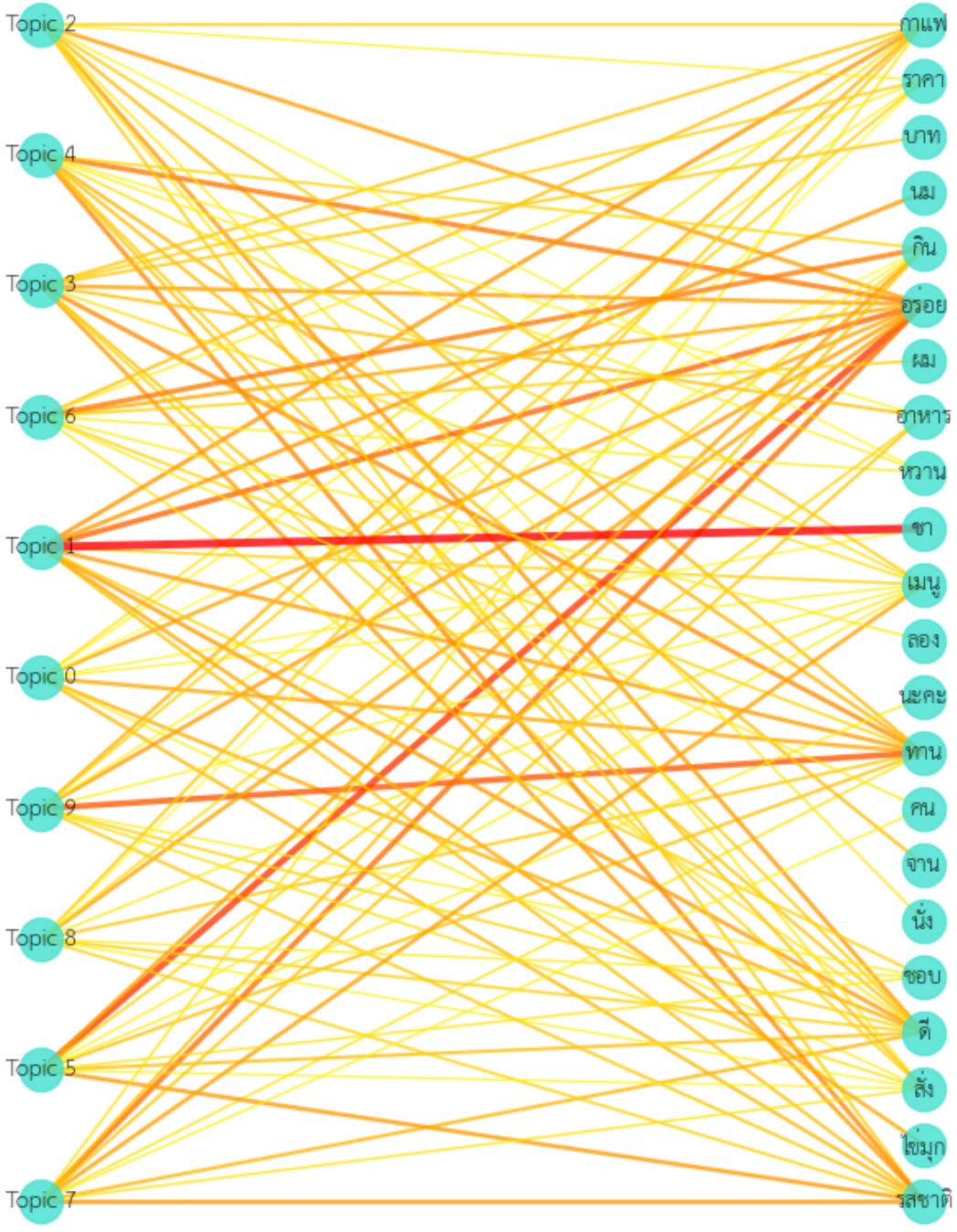
```
array(['ทาน', 'อร่อย', 'ดี', 'รสชาติ', 'กาแฟ', 'สั่ง', 'เมนู', 'ราคา',
       'ชา', 'กิน', 'นม', 'ไข่มุก', 'ชอบ', 'นั่ง', 'หวาน', 'อาหาร', 'บาท',
       'จาน', 'นะคะ', 'ผม', 'คน', 'ลอง'], dtype=object)
```

```
import networkx as nx
G = nx.Graph()

G.add_weighted_edges_from([(f"Topic {r['topic']}", r['word'], round(r['prob'],4)) for i,r in ttdf.iterr

print(nx.info(G))
```

```
Graph with 32 nodes and 100 edges
```

```
# nodelist, node_size = zip(*[(n,d['support']) for n,d in G.nodes(data=True)])
# node_size = 150 + ((np.array(node_size) - min(node_size)) / (max(node_size) - min(node_size)))*1200
topic_nodes = [ f"Topic {i}" for i in range(num_topics)]
edgelist, weights = zip(*[((u,v), d['weight']) for u,v,d in G.edges(data=True)])


width = 1 + ((np.array(weights) - min(weights)) / (max(weights) - min(weights)))*4

labels_params = {'font_family': 'TH Sarabun New', 'alpha':.8, 'font_size':20}

plt.figure(figsize=(9,12))

# pos = nx.spring_layout(G, k=5, weight='lift', iterations=120, seed=120, scale=2.5)
# pos = nx.spring_layout(G, k=5, weight='weight', iterations=80, seed=90, scale=2.5)
pos = nx.bipartite_layout(G, topic_nodes)
nx.draw(G, pos, with_labels=True,
        edgelist=edgelist, width=width, edge_color=width, edge_cmap=plt.cm.autumn_r, #edge_color='.75',
        node_color='turquoise', node_size=700,
        **labels_params)
# edge_labels = nx.get_edge_attributes(G, 'weight')
# nx.draw_networkx_edge_labels(G, pos, edge_labels=edge_labels, font_color='tomato', **labels_params)
```

```
# plt.title('Association Rules of 1-itemsets')
# plt.savefig('plots/association-rules-1.jpg', dpi=120)
plt.show()
```



✓ 0s completed at 17:51