Step 0 - install and import dependencies

```
In [1]:  !pip install pythainlp
         !pip install tensorflow_text
         !pip install umap-learn
```

```
Collecting pythainlp
  Downloading pythainlp-2.3.2-py3-none-any.whl (11.0 MB)
     |████████████████████████████████| 11.0 MB 6.6 MB/s
Collecting python-crfsuite>=0.9.6
  Downloading python_crfsuite-0.9.7-cp37-cp37m-manylinux1_x86_64.whl (743 kB)
     |████████████████████████████████| 743 kB 46.6 MB/s
Requirement already satisfied: requests>=2.22.0 in /usr/local/lib/python3.7/dist-packages (from
pythainlp) (2.23.0)
Collecting tinydb>=3.0
  Downloading tinydb-4.5.2-py3-none-any.whl (23 kB)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (fr
om requests>=2.22.0->pythainlp) (2021.10.8)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (fro
m requests>=2.22.0->pythainlp) (3.0.4)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from req
uests>=2.22.0->pythainlp) (2.10)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python
3.7/dist-packages (from requests>=2.22.0->pythainlp) (1.24.3)
Requirement already satisfied: typing-extensions<4.0.0,>=3.10.0 in /usr/local/lib/python3.7/dis
t-packages (from tinydb>=3.0->pythainlp) (3.10.0.2)
Installing collected packages: tinydb, python-crfsuite, pythainlp
Successfully installed pythainlp-2.3.2 python-crfsuite-0.9.7 tinydb-4.5.2
Collecting tensorflow_text
  Downloading tensorflow_text-2.7.0-cp37-cp37m-manylinux2010_x86_64.whl (4.9 MB)
     |████████████████████████████████| 4.9 MB 7.4 MB/s
Requirement already satisfied: tensorflow-hub>=0.8.0 in /usr/local/lib/python3.7/dist-packages
(from tensorflow_text) (0.12.0)
Requirement already satisfied: tensorflow<2.8,>=2.7.0 in /usr/local/lib/python3.7/dist-packages
(from tensorflow_text) (2.7.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.7/dist-packages (fro
m tensorflow<2.8,>=2.7.0->tensorflow_text) (1.6.3)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.7/dist-packages (fro
m tensorflow<2.8,>=2.7.0->tensorflow_text) (3.3.0)
Requirement already satisfied: wheel<1.0,>=0.32.0 in /usr/local/lib/python3.7/dist-packages (fr
om tensorflow<2.8,>=2.7.0->tensorflow_text) (0.37.0)
Requirement already satisfied: libclang>=9.0.1 in /usr/local/lib/python3.7/dist-packages (from
tensorflow<2.8,>=2.7.0->tensorflow_text) (12.0.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.7/dist-packages (from
tensorflow<2.8,>=2.7.0->tensorflow_text) (1.1.0)
Requirement already satisfied: protobuf>=3.9.2 in /usr/local/lib/python3.7/dist-packages (from
tensorflow<2.8,>=2.7.0->tensorflow_text) (3.17.3)
Requirement already satisfied: keras<2.8,>=2.7.0rc0 in /usr/local/lib/python3.7/dist-packages
(from tensorflow<2.8,>=2.7.0->tensorflow_text) (2.7.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.7/dist-packag
es (from tensorflow<2.8,>=2.7.0->tensorflow_text) (3.10.0.2)
Requirement already satisfied: absl-py>=0.4.0 in /usr/local/lib/python3.7/dist-packages (from t
ensorflow<2.8,>=2.7.0->tensorflow_text) (0.12.0)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.7/dist-packages (f
rom tensorflow<2.8,>=2.7.0->tensorflow_text) (0.2.0)
Requirement already satisfied: gast<0.5.0,>=0.2.1 in /usr/local/lib/python3.7/dist-packages (fr
om tensorflow<2.8,>=2.7.0->tensorflow_text) (0.4.0)
Requirement already satisfied: tensorflow-estimator<2.8,~=2.7.0rc0 in /usr/local/lib/python3.7/
dist-packages (from tensorflow<2.8,>=2.7.0->tensorflow_text) (2.7.0)
Requirement already satisfied: flatbuffers<3.0,>=1.12 in /usr/local/lib/python3.7/dist-packages
(from tensorflow<2.8,>=2.7.0->tensorflow_text) (2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.7/dist-packages (from tens
orflow<2.8,>=2.7.0->tensorflow_text) (3.1.0)
Requirement already satisfied: keras-preprocessing>=1.1.1 in /usr/local/lib/python3.7/dist-pack
ages (from tensorflow<2.8,>=2.7.0->tensorflow_text) (1.1.2)
Requirement already satisfied: tensorboard~=2.6 in /usr/local/lib/python3.7/dist-packages (from
tensorflow<2.8,>=2.7.0->tensorflow_text) (2.7.0)
Requirement already satisfied: wrapt>=1.11.0 in /usr/local/lib/python3.7/dist-packages (from te
nsorflow<2.8,>=2.7.0->tensorflow_text) (1.13.3)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.7/dist-packages (from tens
orflow<2.8,>=2.7.0->tensorflow_text) (1.15.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.7/dist-packages (f
rom tensorflow<2.8,>=2.7.0->tensorflow_text) (1.41.1)
```

```
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-packages (from te
nsorflow<2.8,>=2.7.0->tensorflow_text) (1.19.5)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.21.0 in /usr/local/lib/python3.
7/dist-packages (from tensorflow<2.8,>=2.7.0->tensorflow_text) (0.21.0)
Requirement already satisfied: cached-property in /usr/local/lib/python3.7/dist-packages (from 
h5py>=2.9.0->tensorflow<2.8,>=2.7.0->tensorflow_text) (1.5.2)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.7/dist-packages (f
rom tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (2.23.0)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /usr/local/lib/python3.7/dis
t-packages (from tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (0.4.6)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.7/dist-packages (from 
tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (3.3.4)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.7/dist-packages 
(from tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (1.35.0)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /usr/local/lib/python3.7/dist-p
ackages (from tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (1.8.0)
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.7/dist-packages (fro
m tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (1.0.1)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in /usr/local/lib/python3.
7/dist-packages (from tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (0.6.1)
Requirement already satisfied: setuptools>=41.0.0 in /usr/local/lib/python3.7/dist-packages (fr
om tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (57.4.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.7/dist-packages (from go
ogle-auth<3,>=1.6.3->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (4.7.2)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.7/dist-packages 
(from google-auth<3,>=1.6.3->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (0.2.8)
Requirement already satisfied: cachetools<5.0,>=2.0.0 in /usr/local/lib/python3.7/dist-packages 
(from google-auth<3,>=1.6.3->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (4.2.4)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.7/dist-packag
es (from google-auth-oauthlib<0.5,>=0.4.1->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow
_text) (1.3.0)
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.7/dist-packages (fr
om markdown>=2.6.8->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (4.8.2)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /usr/local/lib/python3.7/dist-packages 
(from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->t
ensorflow_text) (0.4.8)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python
3.7/dist-packages (from requests<3,>=2.21.0->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorfl
ow_text) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (fr
om requests<3,>=2.21.0->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (2021.10.8)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from req
uests<3,>=2.21.0->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (fro
m requests<3,>=2.21.0->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (3.0.4)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.7/dist-packages (from 
requests-oauthlib>=0.7.0->google-auth-oauthlib<0.5,>=0.4.1->tensorboard~=2.6->tensorflow<2.8,>=
2.7.0->tensorflow_text) (3.1.1)
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (from import
lib-metadata->markdown>=2.6.8->tensorboard~=2.6->tensorflow<2.8,>=2.7.0->tensorflow_text) (3.6.
0)
Installing collected packages: tensorflow-text
Successfully installed tensorflow-text-2.7.0
Collecting umap-learn
  Downloading umap-learn-0.5.2.tar.gz (86 kB)
     |████████████████████████████████| 86 kB 4.0 MB/s 
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.7/dist-packages (from umap
-learn) (1.19.5)
Requirement already satisfied: scikit-learn>=0.22 in /usr/local/lib/python3.7/dist-packages (fr
om umap-learn) (0.22.2.post1)
Requirement already satisfied: scipy>=1.0 in /usr/local/lib/python3.7/dist-packages (from umap-
learn) (1.4.1)
Requirement already satisfied: numba>=0.49 in /usr/local/lib/python3.7/dist-packages (from umap
-learn) (0.51.2)
Collecting pynndescent>=0.5
  Downloading pynndescent-0.5.5.tar.gz (1.1 MB)
     |████████████████████████████████| 1.1 MB 18.0 MB/s 
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from umap-learn)
(4.62.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.7/dist-packages (from numba
>=0.49->umap-learn) (57.4.0)
Requirement already satisfied: llvmlite<0.35,>=0.34.0.dev0 in /usr/local/lib/python3.7/dist-pac
kages (from numba>=0.49->umap-learn) (0.34.0)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages (from pyn
ndescent>=0.5->umap-learn) (1.1.0)
Building wheels for collected packages: umap-learn, pynndescent
  Building wheel for umap-learn (setup.py) ... done
  Created wheel for umap-learn: filename=umap_learn-0.5.2-py3-none-any.whl size=82709 sha256=9b
adb4390e1ee540154a482578b98719e041d06eea0c2bbd000b97df279f103f
  Stored in directory: /root/.cache/pip/wheels/84/1b/c6/aaf68a748122632967cef4dffef68224eb16798
b6793257d82
  Building wheel for pynndescent (setup.py) ... done
  Created wheel for pynndescent: filename=pynndescent-0.5.5-py3-none-any.whl size=52603 sha256=
b3c9daf79707cd7713f09f535c9934d827c01b93e5eeaded222e6504c7a1076d
  Stored in directory: /root/.cache/pip/wheels/af/e9/33/04db1436df0757c42fda8ea6796d7a8586e23c8
5fac355f476
Successfully built umap-learn pynndescent
Installing collected packages: pynndescent, umap-learn
Successfully installed pynndescent-0.5.5 umap-learn-0.5.2
```

In [2]:

```
!pip install --upgrade tensorflow_hub
```

```
Requirement already satisfied: tensorflow_hub in /usr/local/lib/python3.7/dist-packages (0.12.
0)
Requirement already satisfied: numpy>=1.12.0 in /usr/local/lib/python3.7/dist-packages (from te
nsorflow_hub) (1.19.5)
Requirement already satisfied: protobuf>=3.8.0 in /usr/local/lib/python3.7/dist-packages (from
tensorflow_hub) (3.17.3)
Requirement already satisfied: six>=1.9 in /usr/local/lib/python3.7/dist-packages (from protobu
f>=3.8.0->tensorflow_hub) (1.15.0)
```

In [3]:

```
import numpy as np
import pandas as pd
import re

import tensorflow as tf
import tensorflow_hub as hub
import tensorflow_text
import umap

from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

from sklearn.cluster import AgglomerativeClustering
from sklearn.neighbors import kneighbors_graph

import pythainlp
from pythainlp.corpus.common import thai_words
from pythainlp.util import Trie
import collections
```

In [4]:

```
module_url = 'https://tfhub.dev/google/universal-sentence-encoder-multilingual/3' #'https://tfh

model = hub.load(module_url)
```

In [5]:

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

In [6]:

```
df.head()
```

Out[6]:

	Review ID	Review
0	1	เป็นคนที่ชอบทาน Macchiato เป็นประจำ มีวันนึงเด...
1	2	Art of Coffee Kasetsart เป็นร้านกาแฟรสชาติเยี่...
2	3	กวงทะเลเผา อาหารทะเลเค้าสดจริงๆเนื้อปูหวานไม่ค...
3	4	วันนี้มีโอกาสตื่นเช้าครับเลยถึงโอกาสออกมาหาอะไ...
4	5	ชอบมาทานร้านนี้ถ้าอยากกินอาหารเวียดนามใกล้บ้าน...

Step 1 - document embedding and dimension reduction

In [7]:

```
#embed sentences using Universal Sentence Encoder (USE)
```

```
embed_comments_array = model(df['Review'].values).numpy()
embed_comments_array
```

Out[7]:

```
array([[ 0.08993827,  0.01941084,  0.03787038, ..., -0.03488849,
          0.06299512,  0.04635989],
       [ 0.00634244,  0.00814594,  0.03071941, ..., -0.01478723,
         -0.03080936, -0.03316405],
       [ 0.0633687 , -0.02027139, -0.05077003, ..., -0.06530775,
         -0.00952999, -0.03439987],
       ...,
       [ 0.08775924,  0.03609736,  0.01263062, ..., -0.03102781,
         -0.03361677,  0.01928871],
       [ 0.05691195,  0.05381691, -0.0399575 , ..., -0.06598807,
         -0.05390478, -0.01037725],
       [ 0.0777048 ,  0.05080631,  0.02680681, ..., -0.0061413 ,
         -0.01313567,  0.02236264]], dtype=float32)
```

In [8]:

```
#reduce array dimensions using umap (you can chagne n_components)

reducer = umap.UMAP(random_state=42,n_components=50)
umap_embed_comments_array = reducer.fit_transform(embed_comments_array)
```

```
/usr/local/lib/python3.7/dist-packages/numba/np/ufunc/parallel.py:363: NumbaWarning: The TBB th
reading layer requires TBB version 2019.5 or later i.e., TBB_INTERFACE_VERSION >= 11005. Found
TBB_INTERFACE_VERSION = 9107. The TBB threading layer is disabled.
  warnings.warn(problem)
```

Step 2 - document clustering using KMeans

In [9]:

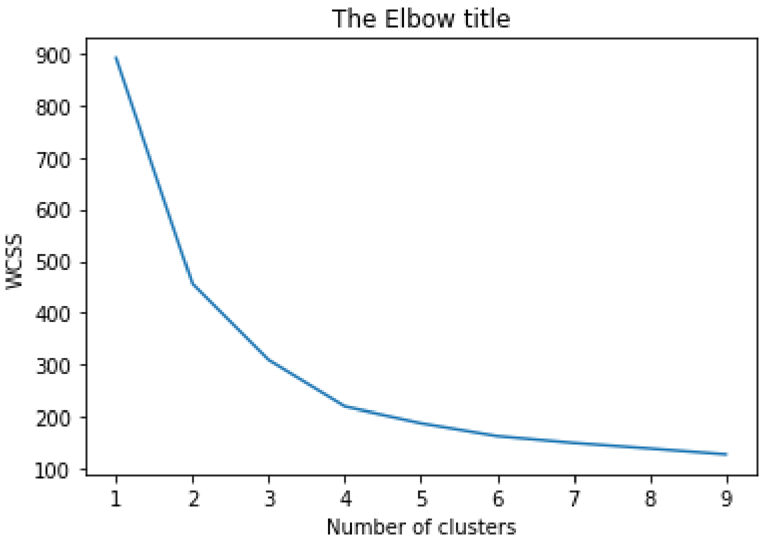
```
#run kmeans with various number of k. evaluate no. of k based on the elbow plot

wcss=[]
max_k = 10
for i in range(1, max_k):
  kmeans = KMeans(i)
  kmeans.fit(umap_embed_comments_array)
  wcss_iter = kmeans.inertia_
  wcss.append(wcss_iter)

number_clusters = range(1, max_k)
plt.plot(number_clusters,wcss)
plt.title('The Elbow title')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
```

Out[9]:

```
Text(0, 0.5, 'WCSS')
```



In [58]:

```
#run kmeans with no. of clusters you see fit the most

k = 3
```

```
kmeans = KMeans(n_clusters = k)
kmeans.fit(umap_embed_comments_array)

df['KMeans ID'] = kmeans.labels_
```

In [59]:

```
#merge all reviews of each cluster into one big sentence

df_kmeans = pd.DataFrame(columns=["KMeans ID", "texts"])


for i in range(0, k):
  row = []
  row.append(i)
  row.append(df['Review'][df['KMeans ID'] == i].to_string())
  df_kmeans.loc[len(df_kmeans)] = row
```

In [60]:

```
df_kmeans
```

Out[60]:

	KMeans ID	texts
0	0	13 เคยเป็นไหมกันไหมคะ หลังอาหารมื้อใหญ่ ต่...
1	1	0 เป็นคนที่ชอบทาน Macchiato เป็นประจำ มีว...
2	2	2 กวงทะเลเผา อาหารทะเลเค้าสดจริงๆเนื้อปูห...

In [61]:

```
#create regex compiler for removal of a character you don't want

special_characters = "/[!@#$%^&*']/g"

specialchar_pattern = re.compile(special_characters)
```

In [62]:

```
#create regex compiler for removal of any emoji

emoji_pattern = re.compile("["
        u"\U0001F600-\U0001F64F"  # emoticons
        u"\U0001F300-\U0001F5FF"  # symbols & pictographs
        u"\U0001F680-\U0001F6FF"  # transport & map symbols
        u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                           "]+", flags=re.UNICODE)
```

In [63]:

```
#create regex compiler for removal of digit

number_pattern = re.compile("[0-9]")
```

In [64]:

```
#create regex compiler for removal of white space

space_pattern = re.compile("\s+")
```

In [65]:

```
#create regex compiler for removal of .

dot_pattern = re.compile(r"\.+")
```

In [66]:

```
#create regex compiler for removal of \

backslash_pattern = re.compile(r"\\+")
```

In [67]:

```
#define a function to tokenize a sentence into words - you can define words you want to remove

stopwords = list(pythainlp.corpus.thai_stopwords())
removed_words = ['u', 'b', 'n', 'nn', 'nn-', '\n', 'ร้าน', ':','@','--------','สรุป','ๆ','ฮ่า','ค่ะ
screening_words = stopwords + removed_words

new_words = {"สตารบัก"}
```

```
words = new_words.union(thai_words())

custom_dictionary_trie = Trie(words)

def tokenize_to_list(sentence):
  merged = []
  words = pythainlp.word_tokenize(str(sentence), engine='newmm', custom_dict=custom_dictionary_
  for word in words:
    if word not in screening_words:
      merged.append(word)
  return merged
```

In [68]:

```
#clean and tokenize sentences. count the occurences of each word

df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: emoji_pattern.sub(r'', x))
df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: specialchar_pattern.sub(r'', x))
df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: number_pattern.sub(r'', x))
df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: space_pattern.sub(r'', x))
df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: dot_pattern.sub(r'', x))
df_kmeans['texts'] = df_kmeans['texts'].apply(lambda x: backslash_pattern.sub(r'', x))
df_kmeans['texts_tokenized'] = df_kmeans['texts'].apply(lambda x: tokenize_to_list(x))
df_kmeans['texts_count'] = df_kmeans['texts_tokenized'].apply(lambda x: collections.Counter(x).
```

In [69]:

```
#results of tokenization

df_kmeans
```

Out[69]:

	KMeans ID	texts	texts_tokenized	texts_count
0	0	เคยเป็นไหมกันไหมคะหลังอาหารมื้อใหญ่ต่อให้อิ่เช...	[ไหม, ไหม, หลังอาหาร, มื้อ, ต่อให้, อิ่, เช้า,...	[(ชา, 18), (นม, 14), (ไข่มุก, 14), (เครื่องดื่...
1	1	เป็นคนที่ชอบทานMacchiatoเป็นประจำมีวันนึงเดArt...	[คน, Macchiato, เป็นประจำ, นึง, เด, ArtofCoffe...	[(ร้านกาแฟ, 25), (กาแฟ, 22), (คาเฟ่, 6), (ดี, ...
2	2	กวงทะเลเผาอาหารทะเลเค้าสดจริงๆเนื้อปูหวานไม่คว...	[กวง, ทะเล, เผา, อาหารทะเล, สด, เนื้อ, ปู, หวา...	[(อร่อย, 11), (บ้าน, 6), (ส้มตำ, 6), (ซอย, 6),...

In [70]:

```
#show top keywords of each cluster

top_N_words = 15

for i in range(0, len(df_kmeans)):
  print(f"Cluster ID : {i}\n")
  print(f"Most common words include : {list(df_kmeans['texts_count'][i])[:top_N_words]}\n")

#tune a model by remove unwanted characters and words and add more words to a custom dictionary
```

```
Cluster ID : 0

Most common words include : [('ชา', 18), ('นม', 14), ('ไข่มุก', 14), ('เครื่องดื่ม', 4), ('ร้า', 3),
('น้ำ', 3), ('ตั้งอยู่', 3), ('ลอง', 3), ('เดิน', 3), ('ปั่น', 3), ('ไต้หวัน', 3), ('ไหม', 2), ('เติม',
2), ('นขา', 2), ('ชาเขียว', 2)]

Cluster ID : 1

Most common words include : [('ร้านกาแฟ', 25), ('กาแฟ', 22), ('คาเฟ่', 6), ('ดี', 6), ('อร่อย', 5),
('กา', 5), ('น่ารัก', 5), ('สวัสดี', 5), ('เจอ', 5), ('หา', 5), ('คน', 4), ('นึง', 4), ('อ', 4), ('รี
', 4), ('เบเกอรี่', 4)]

Cluster ID : 2

Most common words include : [('อร่อย', 11), ('บ้าน', 6), ('ส้มตำ', 6), ('ซอย', 6), ('สาขา', 6),
('กาแฟ', 6), ('เพื่อน', 5), ('ไทย', 5), ('เมนู', 5), ('สวัสดี', 4), ('ถนน', 4), ('แซ่บ', 4), ('คน',
4), ('รอบ', 4), ('บอ', 4)]
```

Step 3 - document clustering using Agglomorative Clustering with cosine similarity

```
In [71]: #clustering using agglomorative clustering

         knn_graph = kneighbors_graph(embed_comments_array, 5, include_self=False)
         model = AgglomerativeClustering(linkage="average", connectivity=knn_graph, n_clusters=10, affir
         model.fit(embed_comments_array)
         df['Agglomerative ID'] = model.labels_
```

```
In [72]: #merge all reviews of each cluster into one big sentence

         df_Agglomerative = pd.DataFrame(columns=["Agglomerative ID", "texts"])


         for i in range(0, k):
           row = []
           row.append(i)
           row.append(str(df['Review'][df['Agglomerative ID'] == i].tolist()))
           df_Agglomerative.loc[len(df_Agglomerative)] = row
```

```
In [73]: #clean and tokenize sentences. count the occurences of each word

         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: emoji_pattern.sub(r'', x)
         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: specialchar_pattern.sub(r
         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: number_pattern.sub(r'', x
         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: space_pattern.sub(r'', x)
         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: dot_pattern.sub(r'', x))
         df_Agglomerative['texts'] = df_Agglomerative['texts'].apply(lambda x: backslash_pattern.sub(r''
         df_Agglomerative['texts_tokenized'] = df_Agglomerative['texts'].apply(lambda x: tokenize_to_lis
         df_Agglomerative['texts_count'] = df_Agglomerative['texts_tokenized'].apply(lambda x: collectic
```

```
In [74]: #show top keywords of each cluster

         top_N_words = 10

         for i in range(0, len(df_Agglomerative)):
           print(f"Cluster ID : {i}\n")
           print(f"Most common words include : {list(df_Agglomerative['texts_count'][i])[:top_N_words]}\
```

```
Cluster ID : 0

Most common words include : [('อร่อย', 508), ('รสชาติ', 407), ('ดี', 347), ('กาแฟ', 311), ('เมนู',
309), ('สั่ง', 301), ('(', 270), ('ชา', 262), (')', 250), ('บาท', 242)]

Cluster ID : 1

Most common words include : [('แตงโม', 22), ('น้ำ', 8), ('ปั่น', 6), ('เนื้อ', 6), ('เลือก', 4), ('ซื้
อ', 4), ('ดื่ม', 4), ('พันธุ์', 3), ('รับประทาน', 3), ('แก้', 3)]

Cluster ID : 2

Most common words include : [('ดิชั้น', 4), ('แย่มาก', 3), ('โต๊ะ', 2), ('รอง', 2), ('แก้ว', 2), ("
['", 1), ('ดิ', 1), ('ชั้น', 1), ('ทบ', 1), ('เวลา', 1)]
```

Step 4 - result discussion

From the comparison of Kmean and Cosine Similarity which used the Wongnai Reviews, it found out that the Kmean can do better clustering on word segmentation than the Cosine Similarity in order to grouping the reviews.

From KMean clustering which classified into 3 groups of reviewers, which the result shows that the most of customer reviews the Coffee Cafe, the 2nd one is restaurants and Taiwan Tea cafe/shops.

From Cosine Similarity, it could be classified mainly on the satisfaction of the customer to review, which 98% of customers satisfied on restaurant and cafe with positive reviews, at least 2% of customers disatisfied on the restaurant and provided the negative feedback in Wongnai.