

October 8, 2020

## 1 Clustering Assignment

There will be some functions that start with the word “grader” ex: `grader_actors()`, `grader_movies()`, `grader_cost1()` etc, you should not change those function definition. Every Grader function has to return `True`.

Please check [clustering assignment helper functions](#) notebook before attempting this assignment.

- Read graph from the given `movie_actor_network.csv` (note that the graph is bipartite graph.)
- Using `stellergaph` and `gensim` packages, get the dense representation (128 dimensional vector) of every node in the graph. [Refer `Clustering_Assignment_Reference.ipynb`]
- Split the dense representation into actor nodes, movies nodes. (Write your code in `def data_split()`)

## 2 Task 1 : Apply clustering algorithm to group similar actors

1. For this task consider only the actor nodes
2. Apply any clustering algorithm of your choice Refer : <https://scikit-learn.org/stable/modules/clustering.html>
3. Choose the number of clusters for which you have maximum score of  $Cost1 * Cost2$
4.  $Cost1 = \frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{number of nodes in the largest connected component in the graph with the actor nodes and its movie neighbours in cluster } i)}{(\text{total number of nodes in that cluster } i)}$   
where  $N$  = number of clusters (Write your code in `def cost1()`)
5.  $Cost2 = \frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{sum of degrees of actor nodes in the graph with the actor nodes and its movie neighbours in cluster } i)}{(\text{number of unique movie nodes in the graph with the actor nodes and its movie neighbours in cluster } i)}$   
where  $N$  = number of clusters (Write your code in `def cost2()`)
6. Fit the clustering algorithm with the optimal number of clusters and get the cluster number for each node
7. Convert the d-dimensional dense vectors of nodes into 2-dimensional using dimensionality reduction techniques (preferably TSNE)
8. Plot the 2d scatter plot, with the node vectors after step e and give colors to nodes such that same cluster nodes will have same color

## 3 Task 2 : Apply clustering algorithm to group similar movies

1. For this task consider only the movie nodes

2. Apply any clustering algorithm of your choice 3. Choose the number of clusters for which you have maximum score of  $Cost1 * Cost2$

$$Cost1 = \frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{number of nodes in the largest connected component in the graph with the movie nodes and its actor neighbours in cluster } i)}{(\text{total number of nodes in that cluster } i)}$$

where N= number of clusters (Write your code in def cost1())

3.  $Cost2 = \frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{sum of degrees of movie nodes in the graph with the movie nodes and its actor neighbours in cluster } i)}{(\text{number of unique actor nodes in the graph with the movie nodes and its actor neighbours in cluster } i)}$   
where N= number of clusters (Write your code in def cost2())

### Algorithm for actor nodes

```
[59]: import networkx as nx
from networkx.algorithms import bipartite
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import numpy as np
import warnings
warnings.filterwarnings("ignore")
import pandas as pd
# you need to have tensorflow
from stellargraph.data import UniformRandomMetaPathWalk
from stellargraph import StellarGraph

from sklearn.manifold import TSNE

[3]: data=pd.read_csv('movie_actor_network.csv', index_col=False,
    ↳names=['movie', 'actor'])

[4]: edges = [tuple(x) for x in data.values.tolist()]

[5]: B = nx.Graph()
B.add_nodes_from(data['movie'].unique(), bipartite=0, label='movie')
B.add_nodes_from(data['actor'].unique(), bipartite=1, label='actor')
B.add_edges_from(edges, label='acted')

[6]: # A = list(nx.connected_component_subgraphs(B))[0]
A = [B.subgraph(element) for element in nx.connected_components(B)][0]

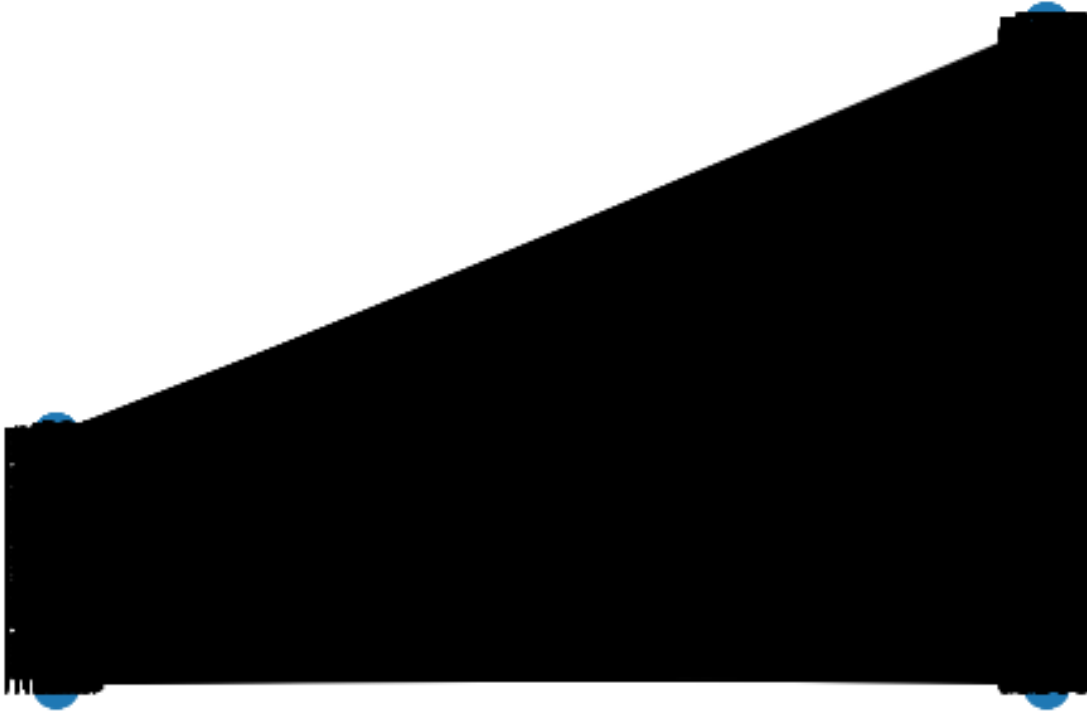
[7]: print("number of nodes", A.number_of_nodes())
print("number of edges", A.number_of_edges())

number of nodes 4703
number of edges 9650

[8]: l, r = nx.bipartite.sets(A)
pos = {}

pos.update((node, (1, index)) for index, node in enumerate(l))
pos.update((node, (2, index)) for index, node in enumerate(r))
```

```
nx.draw(A, pos=pos, with_labels=True)
plt.show()
```



```
[9]: movies = []
actors = []
for i in A.nodes():
    if 'm' in i:
        movies.append(i)
    if 'a' in i:
        actors.append(i)
print('number of movies ', len(movies))
print('number of actors ', len(actors))
```

```
number of movies 1292
number of actors 3411
```

```
[10]: # Create the random walker
rw = UniformRandomMetaPathWalk(StellarGraph(A))

# specify the metapath schemas as a list of lists of node types.
metapaths = [
    ["movie", "actor", "movie"],
```

```

    ["actor", "movie", "actor"]
]

walks = rw.run(nodes=list(A.nodes()), # root nodes
               length=100, # maximum length of a random walk
               n=1, # number of random walks per root node
               metapaths=metapaths
               )

print("Number of random walks: {}".format(len(walks)))

```

Number of random walks: 4703

```

[11]: from gensim.models import Word2Vec
      model = Word2Vec(walks, size=128, window=5)

```

```

[12]: model.wv.vectors.shape # 128-dimensional vector for each node in the graph

```

```

[12]: (4703, 128)

```

```

[13]: # Retrieve node embeddings and corresponding subjects
      node_ids = model.wv.index2word # list of node IDs
      node_embeddings = model.wv.vectors # numpy.ndarray of size number of nodes,
      ↪ times embeddings dimensionality
      node_targets = [ A.nodes[node_id]['label'] for node_id in node_ids]

```

```

[22]: def data_split(node_ids,node_targets,node_embeddings):
      '''In this function, we will split the node embeddings into
      ↪ actor_embeddings , movie_embeddings '''
      actor_nodes,movie_nodes=[],[]
      actor_embeddings,movie_embeddings=[],[]
      # split the node_embeddings into actor_embeddings,movie_embeddings based on
      ↪ node_ids
      # By using node_embedding and node_targets, we can extract actor_embedding
      ↪ and movie embedding
      # By using node_ids and node_targets, we can extract actor_nodes and movie
      ↪ nodes

      for x,y,z in zip(node_ids, node_targets, node_embeddings):
          if y == 'actor':
              actor_nodes.append(x)
              actor_embeddings.append(z)
          else:
              movie_nodes.append(x)

```

```

        movie_embeddings.append(z)
    return actor_nodes, movie_nodes, actor_embeddings, movie_embeddings

```

```

[23]: actor_nodes, movie_nodes, actor_embeddings, movie_embeddings = \
    ↳ data_split(node_ids, node_targets, node_embeddings)

```

Grader function - 1

```

[24]: def grader_actors(data):
        assert(len(data)==3411)
        return True
    grader_actors(actor_nodes)

```

```

[24]: True

```

Grader function - 2

```

[25]: def grader_movies(data):
        assert(len(data)==1292)
        return True
    grader_movies(movie_nodes)

```

```

[25]: True

```

Calculating cost1

$$\text{Cost1} = \frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{number of nodes in the largest connected component in the graph with the actor nodes and its movie neighbours})}{(\text{total number of nodes in that cluster } i)}$$
 where N= number of clusters

```

[36]: def cost1(graph, number_of_clusters):
        '''In this function, we will calculate cost1'''
        max_graph_len = len(max(nx.connected_component_subgraphs(graph), key=len))
        cost1= (1/number_of_clusters) * max_graph_len / len(graph)

        return cost1

```

```

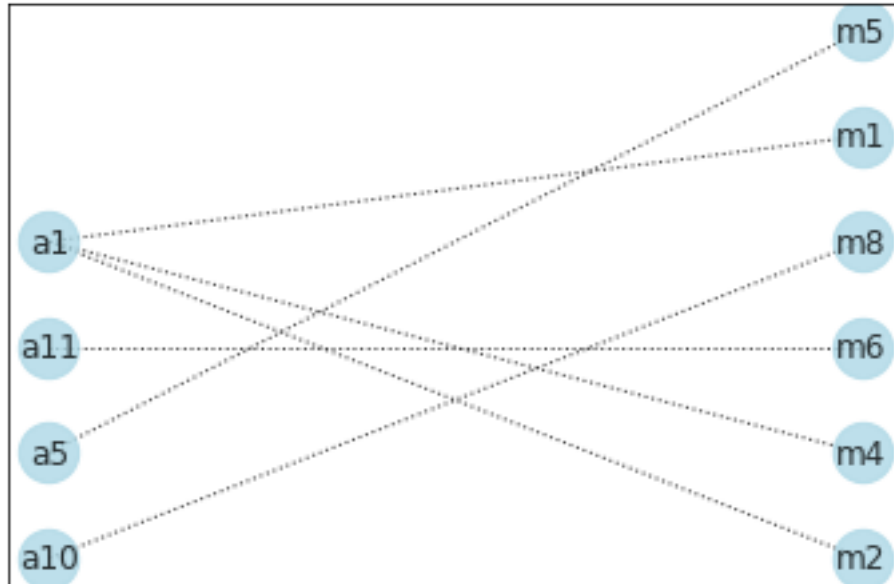
[38]: import networkx as nx
        from networkx.algorithms import bipartite
        graded_graph= nx.Graph()
        graded_graph.add_nodes_from(['a1','a5','a10','a11'], bipartite=0) # Add the \
        ↳ node attribute "bipartite"
        graded_graph.add_nodes_from(['m1','m2','m4','m6','m5','m8'], bipartite=1)
        graded_graph.
        ↳ add_edges_from([('a1','m1'),('a1','m2'),('a1','m4'),('a11','m6'),('a5','m5'),('a10','m8')])
        l={'a1','a5','a10','a11'};r={'m1','m2','m4','m6','m5','m8'}
        pos = {}
        pos.update((node, (1, index)) for index, node in enumerate(l))
        pos.update((node, (2, index)) for index, node in enumerate(r))

```

```

nx.draw_networkx(graded_graph, pos=pos,
    ↳with_labels=True,node_color='lightblue',alpha=0.
    ↳8,style='dotted',node_size=500)

```



Grader function - 3

```

[39]: graded_cost1=cost1(graded_graph,3)
def grader_cost1(data):
    assert(data==((1/3)*(4/10))) # 1/3 is number of clusters
    return True
grader_cost1(graded_cost1)

```

[39]: True

Calculating cost2

Cost2 =  $\frac{1}{N} \sum_{\text{each cluster } i} \frac{(\text{sum of degrees of actor nodes in the graph with the actor nodes and its movie neighbours in cluster } i)}{(\text{number of unique movie nodes in the graph with the actor nodes and its movie neighbours in cluster } i)}$   
 where N= number of clusters

```

[52]: def cost2(graph,number_of_clusters):
    '''In this function, we will calculate cost1'''
    actor_degree = 0
    distinct_movies = 0
    for key, val in graded_graph.degree:

        if key[0] == 'a':
            actor_degree += val
        else:

```

```

        distinct_movies += 1
    cost2= (1/number_of_clusters) * (actor_degree/distinct_movies)

    return cost2

```

Grader function - 4

```

[53]: graded_cost2=cost2(graded_graph,3)
def grader_cost2(data):
    assert(data==((1/3)*(6/6))) # 1/3 is number of clusters
    return True
grader_cost2(graded_cost2)

```

[53]: True

Grouping similar actors

```

[84]: clusters = list(range(2,101))
final_metrics = []
for k in clusters:
    number_of_clusters = k
    print('Clustering data into {} clusters'.format(k))
    kms = KMeans(n_clusters=number_of_clusters, random_state=42)
    kms.fit(actor_embeddings)
    pred = kms.predict(actor_embeddings)
    # https://stackoverflow.com/questions/60935289/ego-graph-in-networkx
    cost1_metric = 0
    cost2_metric = 0
    for ele in np.random.choice(actor_nodes, number_of_cluster):
        temp_sub_graph = A.subgraph(ele)
        cost1_metric += cost1(temp_sub_graph, number_of_clusters)
        cost2_metric += cost2(temp_sub_graph, number_of_clusters)
    final_cost = cost1_metric * cost2_metric
    final_metrics.append(final_cost)
    print('Cost1 Metric Value',cost1_metric)
    print('Cost2 Metric Value',cost2_metric)
    print('Final Metric Value',final_cost)

```

Clustering data into 1 clusters

Cost1 Metric Value 4.0

Cost2 Metric Value 4.0

Final Metric Value 16.0

Clustering data into 2 clusters

Cost1 Metric Value 2.0

Cost2 Metric Value 2.0

Final Metric Value 4.0

Clustering data into 3 clusters

Cost1 Metric Value 1.3333333333333333

Cost2 Metric Value 1.3333333333333333  
 Final Metric Value 1.7777777777777777  
 Clustering data into 4 clusters  
 Cost1 Metric Value 1.0  
 Cost2 Metric Value 1.0  
 Final Metric Value 1.0  
 Clustering data into 5 clusters  
 Cost1 Metric Value 0.8  
 Cost2 Metric Value 0.8  
 Final Metric Value 0.6400000000000001  
 Clustering data into 6 clusters  
 Cost1 Metric Value 0.6666666666666666  
 Cost2 Metric Value 0.6666666666666666  
 Final Metric Value 0.4444444444444444  
 Clustering data into 7 clusters  
 Cost1 Metric Value 0.5714285714285714  
 Cost2 Metric Value 0.5714285714285714  
 Final Metric Value 0.32653061224489793  
 Clustering data into 8 clusters  
 Cost1 Metric Value 0.5  
 Cost2 Metric Value 0.5  
 Final Metric Value 0.25  
 Clustering data into 9 clusters  
 Cost1 Metric Value 0.4444444444444444  
 Cost2 Metric Value 0.4444444444444444  
 Final Metric Value 0.19753086419753085  
 Clustering data into 10 clusters  
 Cost1 Metric Value 0.4  
 Cost2 Metric Value 0.4  
 Final Metric Value 0.16000000000000003  
 Clustering data into 11 clusters  
 Cost1 Metric Value 0.36363636363636365  
 Cost2 Metric Value 0.36363636363636365  
 Final Metric Value 0.1322314049586777  
 Clustering data into 12 clusters  
 Cost1 Metric Value 0.3333333333333333  
 Cost2 Metric Value 0.3333333333333333  
 Final Metric Value 0.1111111111111111  
 Clustering data into 13 clusters  
 Cost1 Metric Value 0.3076923076923077  
 Cost2 Metric Value 0.3076923076923077  
 Final Metric Value 0.09467455621301776  
 Clustering data into 14 clusters  
 Cost1 Metric Value 0.2857142857142857  
 Cost2 Metric Value 0.2857142857142857  
 Final Metric Value 0.08163265306122448  
 Clustering data into 15 clusters  
 Cost1 Metric Value 0.26666666666666666



Cost2 Metric Value 0.266666666666666666  
Final Metric Value 0.071111111111111111  
Clustering data into 16 clusters  
Cost1 Metric Value 0.25  
Cost2 Metric Value 0.25  
Final Metric Value 0.0625  
Clustering data into 17 clusters  
Cost1 Metric Value 0.23529411764705882  
Cost2 Metric Value 0.23529411764705882  
Final Metric Value 0.05536332179930796  
Clustering data into 18 clusters  
Cost1 Metric Value 0.222222222222222222  
Cost2 Metric Value 0.222222222222222222  
Final Metric Value 0.04938271604938271  
Clustering data into 19 clusters  
Cost1 Metric Value 0.21052631578947367  
Cost2 Metric Value 0.21052631578947367  
Final Metric Value 0.04432132963988919  
Clustering data into 20 clusters  
Cost1 Metric Value 0.2  
Cost2 Metric Value 0.2  
Final Metric Value 0.040000000000000001  
Clustering data into 21 clusters  
Cost1 Metric Value 0.19047619047619047  
Cost2 Metric Value 0.19047619047619047  
Final Metric Value 0.03628117913832199  
Clustering data into 22 clusters  
Cost1 Metric Value 0.18181818181818182  
Cost2 Metric Value 0.18181818181818182  
Final Metric Value 0.03305785123966942  
Clustering data into 23 clusters  
Cost1 Metric Value 0.17391304347826086  
Cost2 Metric Value 0.17391304347826086  
Final Metric Value 0.030245746691871453  
Clustering data into 24 clusters  
Cost1 Metric Value 0.166666666666666666  
Cost2 Metric Value 0.166666666666666666  
Final Metric Value 0.027777777777777776  
Clustering data into 25 clusters  
Cost1 Metric Value 0.16  
Cost2 Metric Value 0.16  
Final Metric Value 0.0256  
Clustering data into 26 clusters  
Cost1 Metric Value 0.15384615384615385  
Cost2 Metric Value 0.15384615384615385  
Final Metric Value 0.02366863905325444  
Clustering data into 27 clusters  
Cost1 Metric Value 0.14814814814814814

Cost2 Metric Value 0.14814814814814814  
Final Metric Value 0.02194787379972565  
Clustering data into 28 clusters  
Cost1 Metric Value 0.14285714285714285  
Cost2 Metric Value 0.14285714285714285  
Final Metric Value 0.02040816326530612  
Clustering data into 29 clusters  
Cost1 Metric Value 0.13793103448275862  
Cost2 Metric Value 0.13793103448275862  
Final Metric Value 0.019024970273483946  
Clustering data into 30 clusters  
Cost1 Metric Value 0.13333333333333333  
Cost2 Metric Value 0.13333333333333333  
Final Metric Value 0.01777777777777778  
Clustering data into 31 clusters  
Cost1 Metric Value 0.12903225806451613  
Cost2 Metric Value 0.12903225806451613  
Final Metric Value 0.016649323621227886  
Clustering data into 32 clusters  
Cost1 Metric Value 0.125  
Cost2 Metric Value 0.125  
Final Metric Value 0.015625  
Clustering data into 33 clusters  
Cost1 Metric Value 0.12121212121212122  
Cost2 Metric Value 0.12121212121212122  
Final Metric Value 0.014692378328741967  
Clustering data into 34 clusters  
Cost1 Metric Value 0.11764705882352941  
Cost2 Metric Value 0.11764705882352941  
Final Metric Value 0.01384083044982699  
Clustering data into 35 clusters  
Cost1 Metric Value 0.11428571428571428  
Cost2 Metric Value 0.11428571428571428  
Final Metric Value 0.013061224489795917  
Clustering data into 36 clusters  
Cost1 Metric Value 0.11111111111111111  
Cost2 Metric Value 0.11111111111111111  
Final Metric Value 0.012345679012345678  
Clustering data into 37 clusters  
Cost1 Metric Value 0.10810810810810811  
Cost2 Metric Value 0.10810810810810811  
Final Metric Value 0.011687363038714392  
Clustering data into 38 clusters  
Cost1 Metric Value 0.10526315789473684  
Cost2 Metric Value 0.10526315789473684  
Final Metric Value 0.011080332409972297  
Clustering data into 39 clusters  
Cost1 Metric Value 0.10256410256410256

Cost2 Metric Value 0.10256410256410256  
Final Metric Value 0.010519395134779749  
Clustering data into 40 clusters  
Cost1 Metric Value 0.1  
Cost2 Metric Value 0.1  
Final Metric Value 0.010000000000000002  
Clustering data into 41 clusters  
Cost1 Metric Value 0.0975609756097561  
Cost2 Metric Value 0.0975609756097561  
Final Metric Value 0.009518143961927425  
Clustering data into 42 clusters  
Cost1 Metric Value 0.09523809523809523  
Cost2 Metric Value 0.09523809523809523  
Final Metric Value 0.009070294784580497  
Clustering data into 43 clusters  
Cost1 Metric Value 0.09302325581395349  
Cost2 Metric Value 0.09302325581395349  
Final Metric Value 0.00865332612222823  
Clustering data into 44 clusters  
Cost1 Metric Value 0.09090909090909091  
Cost2 Metric Value 0.09090909090909091  
Final Metric Value 0.008264462809917356  
Clustering data into 45 clusters  
Cost1 Metric Value 0.08888888888888889  
Cost2 Metric Value 0.08888888888888889  
Final Metric Value 0.007901234567901235  
Clustering data into 46 clusters  
Cost1 Metric Value 0.08695652173913043  
Cost2 Metric Value 0.08695652173913043  
Final Metric Value 0.007561436672967863  
Clustering data into 47 clusters  
Cost1 Metric Value 0.0851063829787234  
Cost2 Metric Value 0.0851063829787234  
Final Metric Value 0.007243096423721141  
Clustering data into 48 clusters  
Cost1 Metric Value 0.08333333333333333  
Cost2 Metric Value 0.08333333333333333  
Final Metric Value 0.006944444444444444  
Clustering data into 49 clusters  
Cost1 Metric Value 0.08163265306122448  
Cost2 Metric Value 0.08163265306122448  
Final Metric Value 0.006663890045814243  
Clustering data into 50 clusters  
Cost1 Metric Value 0.08  
Cost2 Metric Value 0.08  
Final Metric Value 0.0064  
Clustering data into 51 clusters  
Cost1 Metric Value 0.0784313725490196

Cost2 Metric Value 0.0784313725490196  
Final Metric Value 0.006151480199923107  
Clustering data into 52 clusters  
Cost1 Metric Value 0.07692307692307693  
Cost2 Metric Value 0.07692307692307693  
Final Metric Value 0.00591715976331361  
Clustering data into 53 clusters  
Cost1 Metric Value 0.07547169811320754  
Cost2 Metric Value 0.07547169811320754  
Final Metric Value 0.005695977216091135  
Clustering data into 54 clusters  
Cost1 Metric Value 0.07407407407407407  
Cost2 Metric Value 0.07407407407407407  
Final Metric Value 0.0054869684499314125  
Clustering data into 55 clusters  
Cost1 Metric Value 0.07272727272727272  
Cost2 Metric Value 0.07272727272727272  
Final Metric Value 0.005289256198347107  
Clustering data into 56 clusters  
Cost1 Metric Value 0.07142857142857142  
Cost2 Metric Value 0.07142857142857142  
Final Metric Value 0.00510204081632653  
Clustering data into 57 clusters  
Cost1 Metric Value 0.07017543859649122  
Cost2 Metric Value 0.07017543859649122  
Final Metric Value 0.0049245921822099106  
Clustering data into 58 clusters  
Cost1 Metric Value 0.06896551724137931  
Cost2 Metric Value 0.06896551724137931  
Final Metric Value 0.0047562425683709865  
Clustering data into 59 clusters  
Cost1 Metric Value 0.06779661016949153  
Cost2 Metric Value 0.06779661016949153  
Final Metric Value 0.004596380350474002  
Clustering data into 60 clusters  
Cost1 Metric Value 0.06666666666666667  
Cost2 Metric Value 0.06666666666666667  
Final Metric Value 0.0044444444444444444  
Clustering data into 61 clusters  
Cost1 Metric Value 0.06557377049180328  
Cost2 Metric Value 0.06557377049180328  
Final Metric Value 0.004299919376511691  
Clustering data into 62 clusters  
Cost1 Metric Value 0.06451612903225806  
Cost2 Metric Value 0.06451612903225806  
Final Metric Value 0.004162330905306971  
Clustering data into 63 clusters  
Cost1 Metric Value 0.06349206349206349

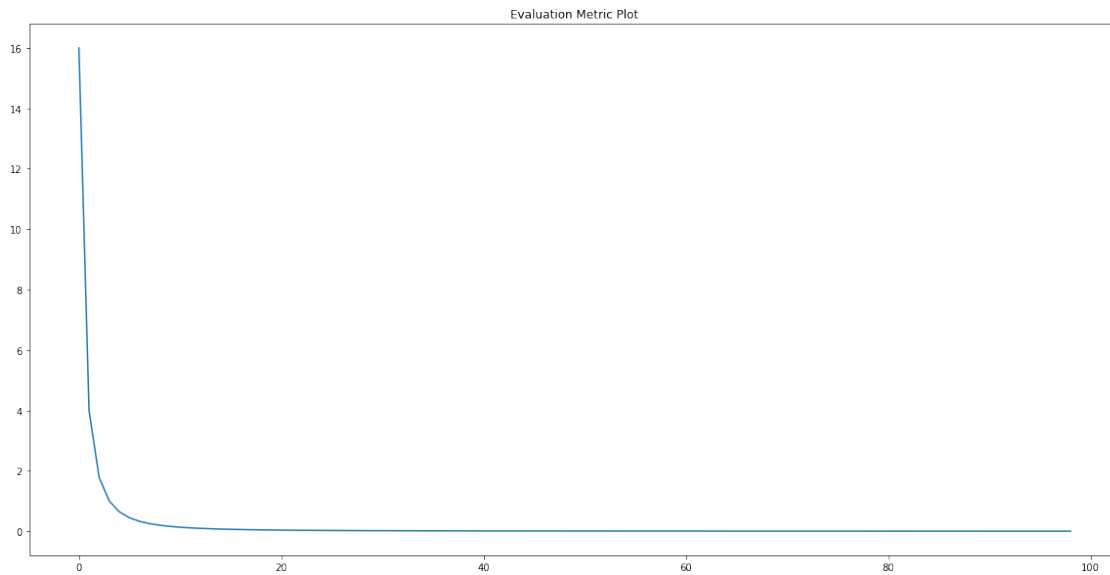
Cost2 Metric Value 0.06349206349206349  
Final Metric Value 0.004031242126480221  
Clustering data into 64 clusters  
Cost1 Metric Value 0.0625  
Cost2 Metric Value 0.0625  
Final Metric Value 0.00390625  
Clustering data into 65 clusters  
Cost1 Metric Value 0.06153846153846154  
Cost2 Metric Value 0.06153846153846154  
Final Metric Value 0.0037869822485207105  
Clustering data into 66 clusters  
Cost1 Metric Value 0.06060606060606061  
Cost2 Metric Value 0.06060606060606061  
Final Metric Value 0.0036730945821854917  
Clustering data into 67 clusters  
Cost1 Metric Value 0.05970149253731343  
Cost2 Metric Value 0.05970149253731343  
Final Metric Value 0.0035642682111828913  
Clustering data into 68 clusters  
Cost1 Metric Value 0.058823529411764705  
Cost2 Metric Value 0.058823529411764705  
Final Metric Value 0.0034602076124567475  
Clustering data into 69 clusters  
Cost1 Metric Value 0.057971014492753624  
Cost2 Metric Value 0.057971014492753624  
Final Metric Value 0.003360638521319051  
Clustering data into 70 clusters  
Cost1 Metric Value 0.05714285714285714  
Cost2 Metric Value 0.05714285714285714  
Final Metric Value 0.0032653061224489793  
Clustering data into 71 clusters  
Cost1 Metric Value 0.056338028169014086  
Cost2 Metric Value 0.056338028169014086  
Final Metric Value 0.0031739734179726245  
Clustering data into 72 clusters  
Cost1 Metric Value 0.05555555555555555  
Cost2 Metric Value 0.05555555555555555  
Final Metric Value 0.0030864197530864196  
Clustering data into 73 clusters  
Cost1 Metric Value 0.0547945205479452  
Cost2 Metric Value 0.0547945205479452  
Final Metric Value 0.003002439482079189  
Clustering data into 74 clusters  
Cost1 Metric Value 0.05405405405405406  
Cost2 Metric Value 0.05405405405405406  
Final Metric Value 0.002921840759678598  
Clustering data into 75 clusters  
Cost1 Metric Value 0.05333333333333333

Cost2 Metric Value 0.05333333333333334  
Final Metric Value 0.0028444444444444445  
Clustering data into 76 clusters  
Cost1 Metric Value 0.05263157894736842  
Cost2 Metric Value 0.05263157894736842  
Final Metric Value 0.0027700831024930744  
Clustering data into 77 clusters  
Cost1 Metric Value 0.05194805194805195  
Cost2 Metric Value 0.05194805194805195  
Final Metric Value 0.002698600101197504  
Clustering data into 78 clusters  
Cost1 Metric Value 0.05128205128205128  
Cost2 Metric Value 0.05128205128205128  
Final Metric Value 0.0026298487836949372  
Clustering data into 79 clusters  
Cost1 Metric Value 0.05063291139240506  
Cost2 Metric Value 0.05063291139240506  
Final Metric Value 0.0025636917160711424  
Clustering data into 80 clusters  
Cost1 Metric Value 0.05  
Cost2 Metric Value 0.05  
Final Metric Value 0.0025000000000000005  
Clustering data into 81 clusters  
Cost1 Metric Value 0.04938271604938271  
Cost2 Metric Value 0.04938271604938271  
Final Metric Value 0.002438652644413961  
Clustering data into 82 clusters  
Cost1 Metric Value 0.04878048780487805  
Cost2 Metric Value 0.04878048780487805  
Final Metric Value 0.002379535990481856  
Clustering data into 83 clusters  
Cost1 Metric Value 0.04819277108433735  
Cost2 Metric Value 0.04819277108433735  
Final Metric Value 0.0023225431847873424  
Clustering data into 84 clusters  
Cost1 Metric Value 0.047619047619047616  
Cost2 Metric Value 0.047619047619047616  
Final Metric Value 0.0022675736961451243  
Clustering data into 85 clusters  
Cost1 Metric Value 0.047058823529411764  
Cost2 Metric Value 0.047058823529411764  
Final Metric Value 0.002214532871972318  
Clustering data into 86 clusters  
Cost1 Metric Value 0.046511627906976744  
Cost2 Metric Value 0.046511627906976744  
Final Metric Value 0.0021633315305570576  
Clustering data into 87 clusters  
Cost1 Metric Value 0.04597701149425287

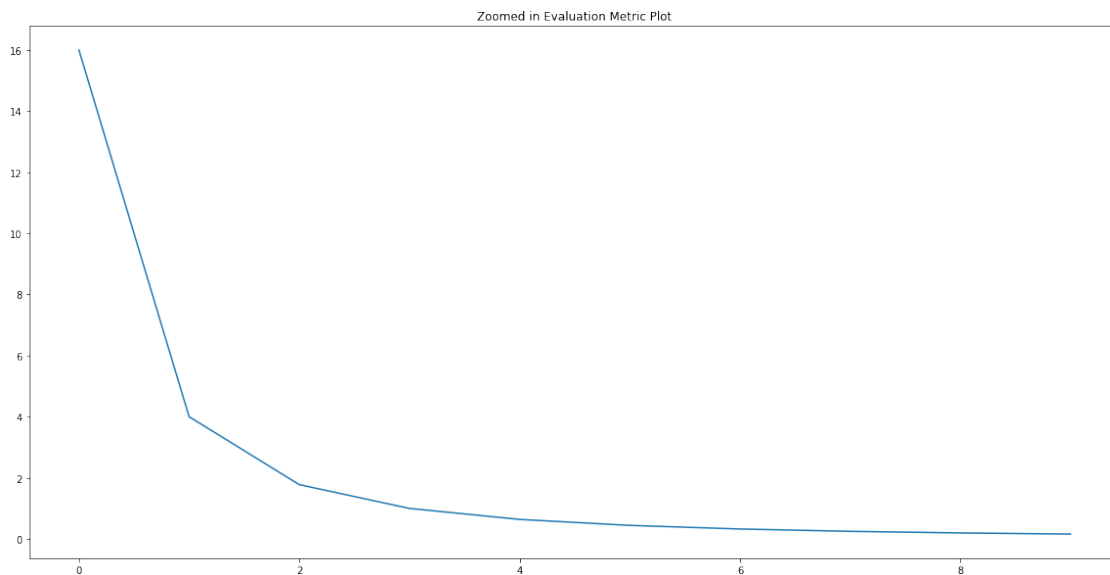
Cost2 Metric Value 0.04597701149425287  
Final Metric Value 0.002113885585942661  
Clustering data into 88 clusters  
Cost1 Metric Value 0.045454545454545456  
Cost2 Metric Value 0.045454545454545456  
Final Metric Value 0.002066115702479339  
Clustering data into 89 clusters  
Cost1 Metric Value 0.0449438202247191  
Cost2 Metric Value 0.0449438202247191  
Final Metric Value 0.0020199469763918696  
Clustering data into 90 clusters  
Cost1 Metric Value 0.044444444444444446  
Cost2 Metric Value 0.044444444444444446  
Final Metric Value 0.0019753086419753087  
Clustering data into 91 clusters  
Cost1 Metric Value 0.04395604395604396  
Cost2 Metric Value 0.04395604395604396  
Final Metric Value 0.0019321338002656688  
Clustering data into 92 clusters  
Cost1 Metric Value 0.043478260869565216  
Cost2 Metric Value 0.043478260869565216  
Final Metric Value 0.0018903591682419658  
Clustering data into 93 clusters  
Cost1 Metric Value 0.043010752688172046  
Cost2 Metric Value 0.043010752688172046  
Final Metric Value 0.001849924846803099  
Clustering data into 94 clusters  
Cost1 Metric Value 0.0425531914893617  
Cost2 Metric Value 0.0425531914893617  
Final Metric Value 0.0018107741059302852  
Clustering data into 95 clusters  
Cost1 Metric Value 0.042105263157894736  
Cost2 Metric Value 0.042105263157894736  
Final Metric Value 0.0017728531855955678  
Clustering data into 96 clusters  
Cost1 Metric Value 0.041666666666666664  
Cost2 Metric Value 0.041666666666666664  
Final Metric Value 0.0017361111111111111  
Clustering data into 97 clusters  
Cost1 Metric Value 0.041237113402061855  
Cost2 Metric Value 0.041237113402061855  
Final Metric Value 0.0017004995217345094  
Clustering data into 98 clusters  
Cost1 Metric Value 0.04081632653061224  
Cost2 Metric Value 0.04081632653061224  
Final Metric Value 0.0016659725114535606  
Clustering data into 99 clusters  
Cost1 Metric Value 0.04040404040404041

Cost2 Metric Value 0.04040404040404041  
Final Metric Value 0.0016324864809713297

```
[97]: _, ax = plt.subplots(1,1, figsize=(20, 10))  
      ax.plot(final_metrics)  
      ax.set_title('Evaluation Metric Plot')  
      plt.show()
```



```
[96]: _, ax = plt.subplots(1,1, figsize=(20, 10))  
      ax.plot(final_metrics[:10])  
      ax.set_title('Zoomed in Evaluation Metric Plot')  
      plt.show()
```





```
[120]: number_of_clusters = 4
print('Clustering data into {} clusters'.format(number_of_clusters))
kms = KMeans(n_clusters=number_of_clusters, random_state=42)
kms.fit(actor_embeddings)
pred = kms.predict(actor_embeddings)
# https://stackoverflow.com/questions/60935289/ego-graph-in-networkx
cost1_metric = 0
cost2_metric = 0
for ele in np.random.choice(actor_nodes, number_of_cluster):
    temp_sub_graph = A.subgraph(ele)
    cost1_metric += cost1(temp_sub_graph, number_of_clusters)
    cost2_metric += cost2(temp_sub_graph, number_of_clusters)
final_cost = cost1_metric * cost2_metric
print('Cost1 Metric Value',cost1_metric)
print('Cost2 Metric Value',cost2_metric)
print('Final Metric Value',final_cost)
```

Clustering data into 4 clusters

Cost1 Metric Value 1.0

Cost2 Metric Value 1.0

Final Metric Value 1.0

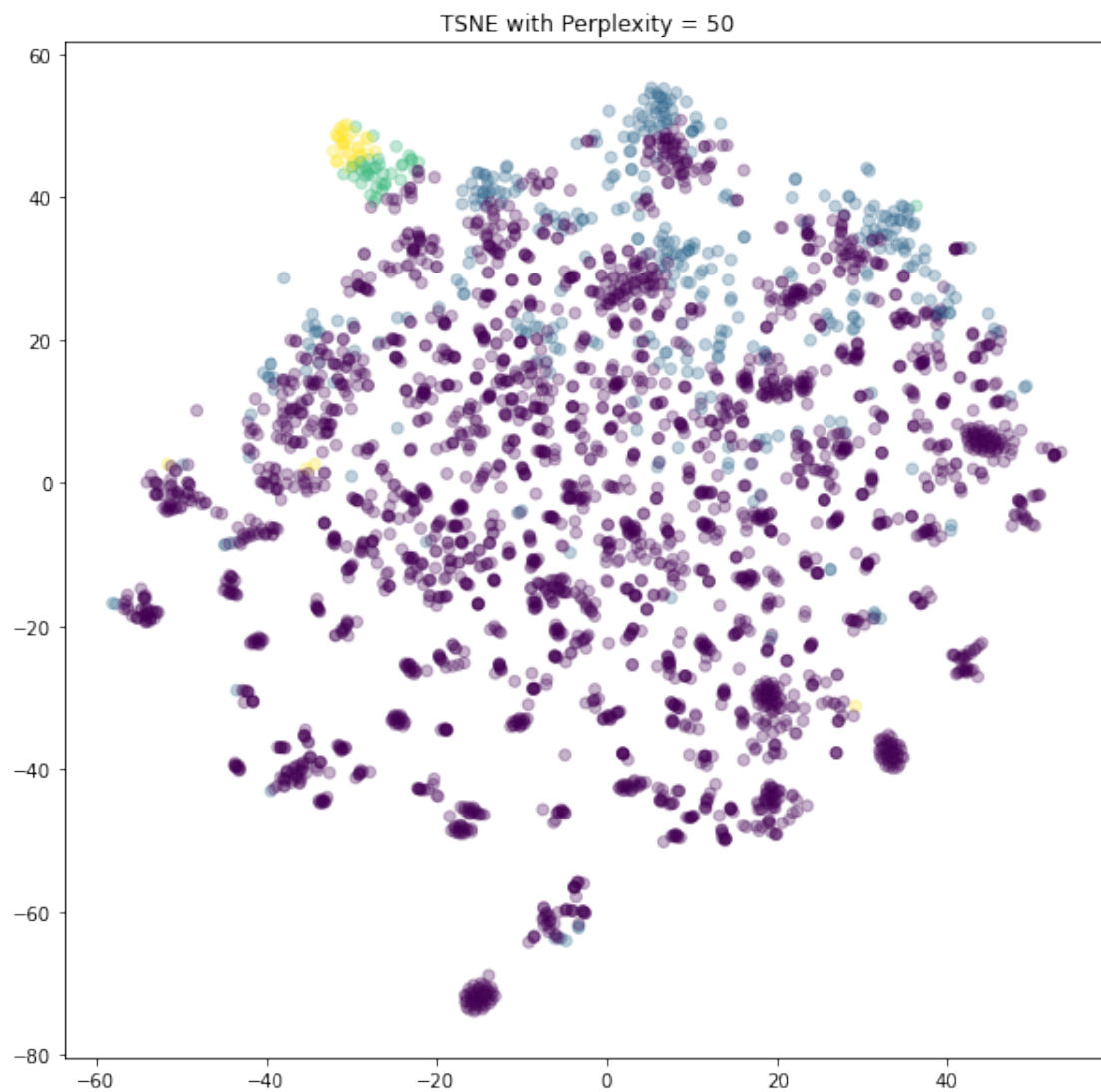
Displaying similar actor clusters

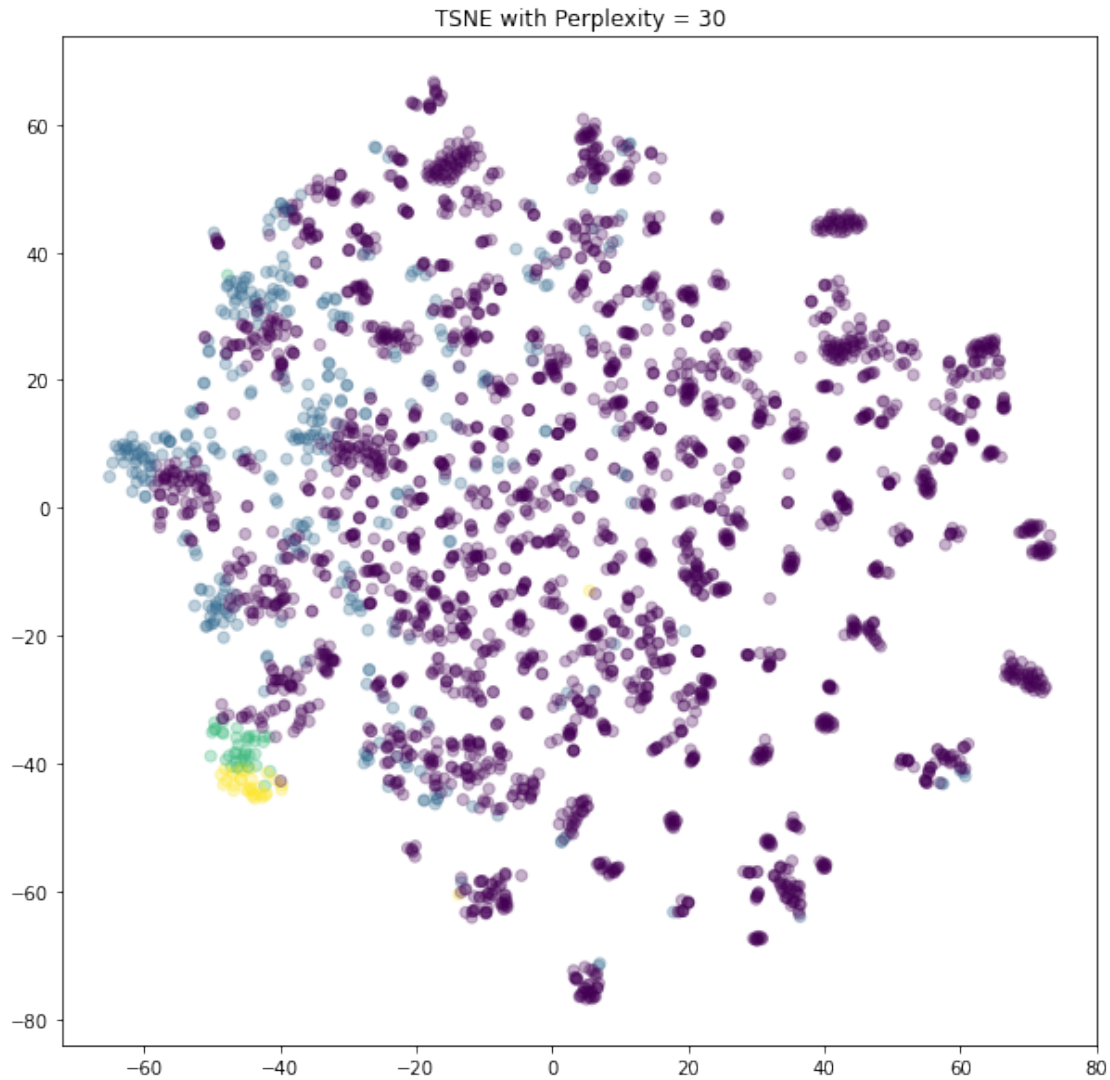
```
[121]: tsne = TSNE(n_components=2, perplexity=50)
actor_2d_transformation = tsne.fit_transform(actor_embeddings)

_, ax = plt.subplots(1, 1, figsize=(10, 10))
ax.scatter(actor_2d_transformation[:, 0], actor_2d_transformation[:, 1],
           c = pred, alpha=0.3)
ax.set_title('TSNE with Perplexity = 50')
plt.show()

tsne = TSNE(n_components=2, perplexity=30)
actor_2d_transformation = tsne.fit_transform(actor_embeddings)

_, ax = plt.subplots(1, 1, figsize=(10, 10))
ax.scatter(actor_2d_transformation[:, 0], actor_2d_transformation[:, 1],
           c = pred, alpha=0.3)
ax.set_title('TSNE with Perplexity = 30')
plt.show()
```





Grouping similar movies

```
[122]: clusters = list(range(2,101))
final_metrics = []
for k in clusters:
    number_of_clusters = k
    print('Clustering data into {} clusters'.format(k))
    kms = KMeans(n_clusters=number_of_clusters, random_state=42)
    kms.fit(actor_embeddings)
    pred = kms.predict(movie_embeddings)
    cost1_metric = 0
    cost2_metric = 0
    for ele in np.random.choice(actor_nodes, number_of_cluster):
        temp_sub_graph = A.subgraph(ele)
```

```

        cost1_metric += cost1(temp_sub_graph, number_of_clusters)
        cost2_metric += cost2(temp_sub_graph, number_of_clusters)
    final_cost = cost1_metric * cost2_metric
    final_metrics.append(final_cost)
    print('Cost1 Metric Value',cost1_metric)
    print('Cost2 Metric Value',cost2_metric)
    print('Final Metric Value',final_cost)

```

```

Clustering data into 2 clusters
Cost1 Metric Value 2.0
Cost2 Metric Value 2.0
Final Metric Value 4.0
Clustering data into 3 clusters
Cost1 Metric Value 1.3333333333333333
Cost2 Metric Value 1.3333333333333333
Final Metric Value 1.7777777777777777
Clustering data into 4 clusters
Cost1 Metric Value 1.0
Cost2 Metric Value 1.0
Final Metric Value 1.0
Clustering data into 5 clusters
Cost1 Metric Value 0.8
Cost2 Metric Value 0.8
Final Metric Value 0.6400000000000001
Clustering data into 6 clusters
Cost1 Metric Value 0.6666666666666666
Cost2 Metric Value 0.6666666666666666
Final Metric Value 0.4444444444444444
Clustering data into 7 clusters
Cost1 Metric Value 0.5714285714285714
Cost2 Metric Value 0.5714285714285714
Final Metric Value 0.32653061224489793
Clustering data into 8 clusters
Cost1 Metric Value 0.5
Cost2 Metric Value 0.5
Final Metric Value 0.25
Clustering data into 9 clusters
Cost1 Metric Value 0.4444444444444444
Cost2 Metric Value 0.4444444444444444
Final Metric Value 0.19753086419753085
Clustering data into 10 clusters
Cost1 Metric Value 0.4
Cost2 Metric Value 0.4
Final Metric Value 0.16000000000000003
Clustering data into 11 clusters
Cost1 Metric Value 0.36363636363636365
Cost2 Metric Value 0.36363636363636365

```

Final Metric Value 0.1322314049586777  
Clustering data into 12 clusters  
Cost1 Metric Value 0.3333333333333333  
Cost2 Metric Value 0.3333333333333333  
Final Metric Value 0.1111111111111111  
Clustering data into 13 clusters  
Cost1 Metric Value 0.3076923076923077  
Cost2 Metric Value 0.3076923076923077  
Final Metric Value 0.09467455621301776  
Clustering data into 14 clusters  
Cost1 Metric Value 0.2857142857142857  
Cost2 Metric Value 0.2857142857142857  
Final Metric Value 0.08163265306122448  
Clustering data into 15 clusters  
Cost1 Metric Value 0.2666666666666666  
Cost2 Metric Value 0.2666666666666666  
Final Metric Value 0.0711111111111111  
Clustering data into 16 clusters  
Cost1 Metric Value 0.25  
Cost2 Metric Value 0.25  
Final Metric Value 0.0625  
Clustering data into 17 clusters  
Cost1 Metric Value 0.23529411764705882  
Cost2 Metric Value 0.23529411764705882  
Final Metric Value 0.05536332179930796  
Clustering data into 18 clusters  
Cost1 Metric Value 0.2222222222222222  
Cost2 Metric Value 0.2222222222222222  
Final Metric Value 0.04938271604938271  
Clustering data into 19 clusters  
Cost1 Metric Value 0.21052631578947367  
Cost2 Metric Value 0.21052631578947367  
Final Metric Value 0.04432132963988919  
Clustering data into 20 clusters  
Cost1 Metric Value 0.2  
Cost2 Metric Value 0.2  
Final Metric Value 0.04000000000000001  
Clustering data into 21 clusters  
Cost1 Metric Value 0.19047619047619047  
Cost2 Metric Value 0.19047619047619047  
Final Metric Value 0.03628117913832199  
Clustering data into 22 clusters  
Cost1 Metric Value 0.18181818181818182  
Cost2 Metric Value 0.18181818181818182  
Final Metric Value 0.03305785123966942  
Clustering data into 23 clusters  
Cost1 Metric Value 0.17391304347826086  
Cost2 Metric Value 0.17391304347826086

Final Metric Value 0.030245746691871453  
Clustering data into 24 clusters  
Cost1 Metric Value 0.16666666666666666  
Cost2 Metric Value 0.16666666666666666  
Final Metric Value 0.027777777777777776  
Clustering data into 25 clusters  
Cost1 Metric Value 0.16  
Cost2 Metric Value 0.16  
Final Metric Value 0.0256  
Clustering data into 26 clusters  
Cost1 Metric Value 0.15384615384615385  
Cost2 Metric Value 0.15384615384615385  
Final Metric Value 0.02366863905325444  
Clustering data into 27 clusters  
Cost1 Metric Value 0.14814814814814814  
Cost2 Metric Value 0.14814814814814814  
Final Metric Value 0.02194787379972565  
Clustering data into 28 clusters  
Cost1 Metric Value 0.14285714285714285  
Cost2 Metric Value 0.14285714285714285  
Final Metric Value 0.02040816326530612  
Clustering data into 29 clusters  
Cost1 Metric Value 0.13793103448275862  
Cost2 Metric Value 0.13793103448275862  
Final Metric Value 0.019024970273483946  
Clustering data into 30 clusters  
Cost1 Metric Value 0.13333333333333333  
Cost2 Metric Value 0.13333333333333333  
Final Metric Value 0.017777777777777778  
Clustering data into 31 clusters  
Cost1 Metric Value 0.12903225806451613  
Cost2 Metric Value 0.12903225806451613  
Final Metric Value 0.016649323621227886  
Clustering data into 32 clusters  
Cost1 Metric Value 0.125  
Cost2 Metric Value 0.125  
Final Metric Value 0.015625  
Clustering data into 33 clusters  
Cost1 Metric Value 0.12121212121212122  
Cost2 Metric Value 0.12121212121212122  
Final Metric Value 0.014692378328741967  
Clustering data into 34 clusters  
Cost1 Metric Value 0.11764705882352941  
Cost2 Metric Value 0.11764705882352941  
Final Metric Value 0.01384083044982699  
Clustering data into 35 clusters  
Cost1 Metric Value 0.11428571428571428  
Cost2 Metric Value 0.11428571428571428

Final Metric Value 0.013061224489795917  
 Clustering data into 36 clusters  
 Cost1 Metric Value 0.1111111111111111  
 Cost2 Metric Value 0.1111111111111111  
 Final Metric Value 0.012345679012345678  
 Clustering data into 37 clusters  
 Cost1 Metric Value 0.10810810810810811  
 Cost2 Metric Value 0.10810810810810811  
 Final Metric Value 0.011687363038714392  
 Clustering data into 38 clusters  
 Cost1 Metric Value 0.10526315789473684  
 Cost2 Metric Value 0.10526315789473684  
 Final Metric Value 0.011080332409972297  
 Clustering data into 39 clusters  
 Cost1 Metric Value 0.10256410256410256  
 Cost2 Metric Value 0.10256410256410256  
 Final Metric Value 0.010519395134779749  
 Clustering data into 40 clusters  
 Cost1 Metric Value 0.1  
 Cost2 Metric Value 0.1  
 Final Metric Value 0.010000000000000002  
 Clustering data into 41 clusters  
 Cost1 Metric Value 0.0975609756097561  
 Cost2 Metric Value 0.0975609756097561  
 Final Metric Value 0.009518143961927425  
 Clustering data into 42 clusters  
 Cost1 Metric Value 0.09523809523809523  
 Cost2 Metric Value 0.09523809523809523  
 Final Metric Value 0.009070294784580497  
 Clustering data into 43 clusters  
 Cost1 Metric Value 0.09302325581395349  
 Cost2 Metric Value 0.09302325581395349  
 Final Metric Value 0.00865332612222823  
 Clustering data into 44 clusters  
 Cost1 Metric Value 0.09090909090909091  
 Cost2 Metric Value 0.09090909090909091  
 Final Metric Value 0.008264462809917356  
 Clustering data into 45 clusters  
 Cost1 Metric Value 0.08888888888888889  
 Cost2 Metric Value 0.08888888888888889  
 Final Metric Value 0.007901234567901235  
 Clustering data into 46 clusters  
 Cost1 Metric Value 0.08695652173913043  
 Cost2 Metric Value 0.08695652173913043  
 Final Metric Value 0.007561436672967863  
 Clustering data into 47 clusters  
 Cost1 Metric Value 0.0851063829787234  
 Cost2 Metric Value 0.0851063829787234

Final Metric Value 0.007243096423721141  
Clustering data into 48 clusters  
Cost1 Metric Value 0.08333333333333333  
Cost2 Metric Value 0.08333333333333333  
Final Metric Value 0.006944444444444444  
Clustering data into 49 clusters  
Cost1 Metric Value 0.08163265306122448  
Cost2 Metric Value 0.08163265306122448  
Final Metric Value 0.006663890045814243  
Clustering data into 50 clusters  
Cost1 Metric Value 0.08  
Cost2 Metric Value 0.08  
Final Metric Value 0.0064  
Clustering data into 51 clusters  
Cost1 Metric Value 0.0784313725490196  
Cost2 Metric Value 0.0784313725490196  
Final Metric Value 0.006151480199923107  
Clustering data into 52 clusters  
Cost1 Metric Value 0.07692307692307693  
Cost2 Metric Value 0.07692307692307693  
Final Metric Value 0.00591715976331361  
Clustering data into 53 clusters  
Cost1 Metric Value 0.07547169811320754  
Cost2 Metric Value 0.07547169811320754  
Final Metric Value 0.005695977216091135  
Clustering data into 54 clusters  
Cost1 Metric Value 0.07407407407407407  
Cost2 Metric Value 0.07407407407407407  
Final Metric Value 0.0054869684499314125  
Clustering data into 55 clusters  
Cost1 Metric Value 0.07272727272727272  
Cost2 Metric Value 0.07272727272727272  
Final Metric Value 0.005289256198347107  
Clustering data into 56 clusters  
Cost1 Metric Value 0.07142857142857142  
Cost2 Metric Value 0.07142857142857142  
Final Metric Value 0.00510204081632653  
Clustering data into 57 clusters  
Cost1 Metric Value 0.07017543859649122  
Cost2 Metric Value 0.07017543859649122  
Final Metric Value 0.0049245921822099106  
Clustering data into 58 clusters  
Cost1 Metric Value 0.06896551724137931  
Cost2 Metric Value 0.06896551724137931  
Final Metric Value 0.0047562425683709865  
Clustering data into 59 clusters  
Cost1 Metric Value 0.06779661016949153  
Cost2 Metric Value 0.06779661016949153



Final Metric Value 0.004596380350474002  
Clustering data into 60 clusters  
Cost1 Metric Value 0.06666666666666667  
Cost2 Metric Value 0.06666666666666667  
Final Metric Value 0.004444444444444444  
Clustering data into 61 clusters  
Cost1 Metric Value 0.06557377049180328  
Cost2 Metric Value 0.06557377049180328  
Final Metric Value 0.004299919376511691  
Clustering data into 62 clusters  
Cost1 Metric Value 0.06451612903225806  
Cost2 Metric Value 0.06451612903225806  
Final Metric Value 0.004162330905306971  
Clustering data into 63 clusters  
Cost1 Metric Value 0.06349206349206349  
Cost2 Metric Value 0.06349206349206349  
Final Metric Value 0.004031242126480221  
Clustering data into 64 clusters  
Cost1 Metric Value 0.0625  
Cost2 Metric Value 0.0625  
Final Metric Value 0.00390625  
Clustering data into 65 clusters  
Cost1 Metric Value 0.06153846153846154  
Cost2 Metric Value 0.06153846153846154  
Final Metric Value 0.0037869822485207105  
Clustering data into 66 clusters  
Cost1 Metric Value 0.06060606060606061  
Cost2 Metric Value 0.06060606060606061  
Final Metric Value 0.0036730945821854917  
Clustering data into 67 clusters  
Cost1 Metric Value 0.05970149253731343  
Cost2 Metric Value 0.05970149253731343  
Final Metric Value 0.0035642682111828913  
Clustering data into 68 clusters  
Cost1 Metric Value 0.058823529411764705  
Cost2 Metric Value 0.058823529411764705  
Final Metric Value 0.0034602076124567475  
Clustering data into 69 clusters  
Cost1 Metric Value 0.057971014492753624  
Cost2 Metric Value 0.057971014492753624  
Final Metric Value 0.003360638521319051  
Clustering data into 70 clusters  
Cost1 Metric Value 0.05714285714285714  
Cost2 Metric Value 0.05714285714285714  
Final Metric Value 0.0032653061224489793  
Clustering data into 71 clusters  
Cost1 Metric Value 0.056338028169014086  
Cost2 Metric Value 0.056338028169014086

Final Metric Value 0.0031739734179726245  
Clustering data into 72 clusters  
Cost1 Metric Value 0.05555555555555555  
Cost2 Metric Value 0.05555555555555555  
Final Metric Value 0.0030864197530864196  
Clustering data into 73 clusters  
Cost1 Metric Value 0.0547945205479452  
Cost2 Metric Value 0.0547945205479452  
Final Metric Value 0.003002439482079189  
Clustering data into 74 clusters  
Cost1 Metric Value 0.05405405405405406  
Cost2 Metric Value 0.05405405405405406  
Final Metric Value 0.002921840759678598  
Clustering data into 75 clusters  
Cost1 Metric Value 0.05333333333333334  
Cost2 Metric Value 0.05333333333333334  
Final Metric Value 0.002844444444444445  
Clustering data into 76 clusters  
Cost1 Metric Value 0.05263157894736842  
Cost2 Metric Value 0.05263157894736842  
Final Metric Value 0.0027700831024930744  
Clustering data into 77 clusters  
Cost1 Metric Value 0.05194805194805195  
Cost2 Metric Value 0.05194805194805195  
Final Metric Value 0.002698600101197504  
Clustering data into 78 clusters  
Cost1 Metric Value 0.05128205128205128  
Cost2 Metric Value 0.05128205128205128  
Final Metric Value 0.0026298487836949372  
Clustering data into 79 clusters  
Cost1 Metric Value 0.05063291139240506  
Cost2 Metric Value 0.05063291139240506  
Final Metric Value 0.0025636917160711424  
Clustering data into 80 clusters  
Cost1 Metric Value 0.05  
Cost2 Metric Value 0.05  
Final Metric Value 0.0025000000000000005  
Clustering data into 81 clusters  
Cost1 Metric Value 0.04938271604938271  
Cost2 Metric Value 0.04938271604938271  
Final Metric Value 0.002438652644413961  
Clustering data into 82 clusters  
Cost1 Metric Value 0.04878048780487805  
Cost2 Metric Value 0.04878048780487805  
Final Metric Value 0.002379535990481856  
Clustering data into 83 clusters  
Cost1 Metric Value 0.04819277108433735  
Cost2 Metric Value 0.04819277108433735

Final Metric Value 0.0023225431847873424  
 Clustering data into 84 clusters  
 Cost1 Metric Value 0.047619047619047616  
 Cost2 Metric Value 0.047619047619047616  
 Final Metric Value 0.0022675736961451243  
 Clustering data into 85 clusters  
 Cost1 Metric Value 0.047058823529411764  
 Cost2 Metric Value 0.047058823529411764  
 Final Metric Value 0.002214532871972318  
 Clustering data into 86 clusters  
 Cost1 Metric Value 0.046511627906976744  
 Cost2 Metric Value 0.046511627906976744  
 Final Metric Value 0.0021633315305570576  
 Clustering data into 87 clusters  
 Cost1 Metric Value 0.04597701149425287  
 Cost2 Metric Value 0.04597701149425287  
 Final Metric Value 0.002113885585942661  
 Clustering data into 88 clusters  
 Cost1 Metric Value 0.045454545454545456  
 Cost2 Metric Value 0.045454545454545456  
 Final Metric Value 0.002066115702479339  
 Clustering data into 89 clusters  
 Cost1 Metric Value 0.0449438202247191  
 Cost2 Metric Value 0.0449438202247191  
 Final Metric Value 0.0020199469763918696  
 Clustering data into 90 clusters  
 Cost1 Metric Value 0.044444444444444446  
 Cost2 Metric Value 0.044444444444444446  
 Final Metric Value 0.0019753086419753087  
 Clustering data into 91 clusters  
 Cost1 Metric Value 0.04395604395604396  
 Cost2 Metric Value 0.04395604395604396  
 Final Metric Value 0.0019321338002656688  
 Clustering data into 92 clusters  
 Cost1 Metric Value 0.043478260869565216  
 Cost2 Metric Value 0.043478260869565216  
 Final Metric Value 0.0018903591682419658  
 Clustering data into 93 clusters  
 Cost1 Metric Value 0.043010752688172046  
 Cost2 Metric Value 0.043010752688172046  
 Final Metric Value 0.001849924846803099  
 Clustering data into 94 clusters  
 Cost1 Metric Value 0.0425531914893617  
 Cost2 Metric Value 0.0425531914893617  
 Final Metric Value 0.0018107741059302852  
 Clustering data into 95 clusters  
 Cost1 Metric Value 0.042105263157894736  
 Cost2 Metric Value 0.042105263157894736

```

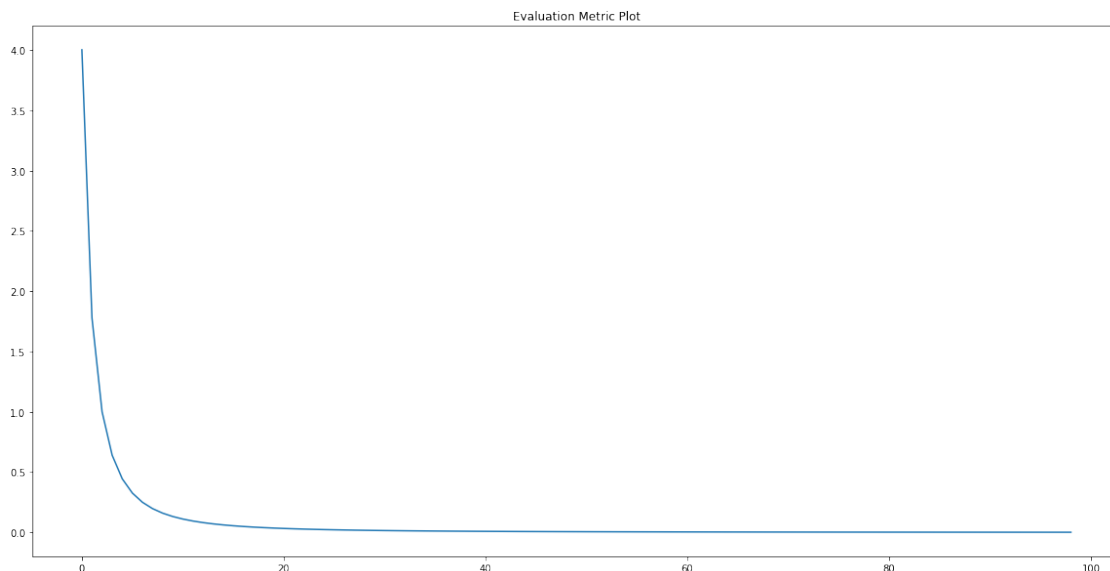
Final Metric Value 0.0017728531855955678
Clustering data into 96 clusters
Cost1 Metric Value 0.041666666666666664
Cost2 Metric Value 0.041666666666666664
Final Metric Value 0.0017361111111111111
Clustering data into 97 clusters
Cost1 Metric Value 0.041237113402061855
Cost2 Metric Value 0.041237113402061855
Final Metric Value 0.0017004995217345094
Clustering data into 98 clusters
Cost1 Metric Value 0.04081632653061224
Cost2 Metric Value 0.04081632653061224
Final Metric Value 0.0016659725114535606
Clustering data into 99 clusters
Cost1 Metric Value 0.04040404040404041
Cost2 Metric Value 0.04040404040404041
Final Metric Value 0.0016324864809713297
Clustering data into 100 clusters
Cost1 Metric Value 0.04
Cost2 Metric Value 0.04
Final Metric Value 0.0016

```

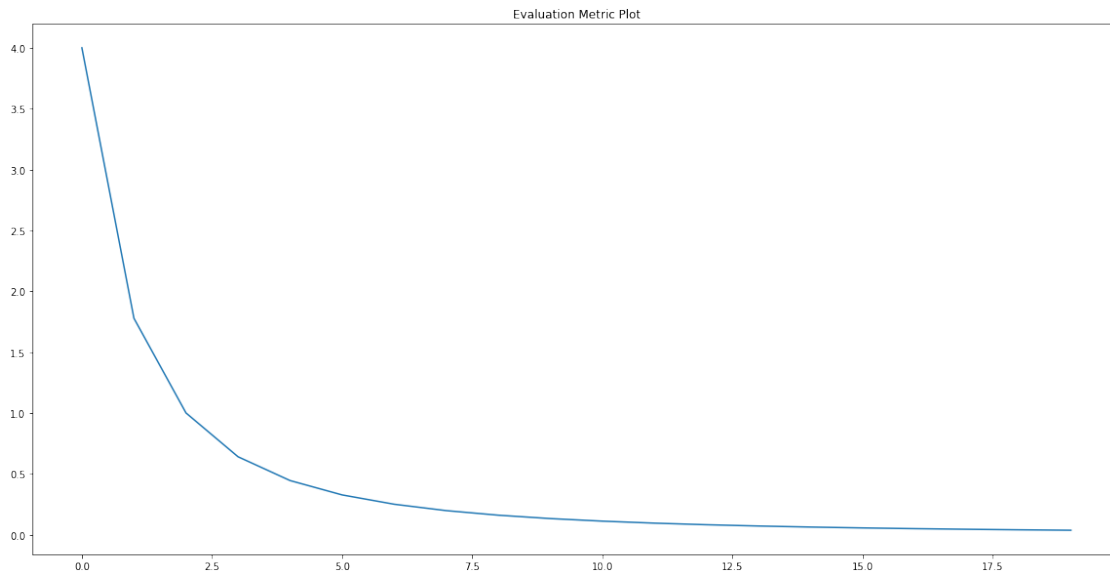
```

[123]: _, ax = plt.subplots(1,1, figsize=(20, 10))
        ax.plot(final_metrics)
        ax.set_title('Evaluation Metric Plot')
        plt.show()

```



```
[124]: _, ax = plt.subplots(1,1, figsize=(20, 10))
ax.plot(final_metrics[:20])
ax.set_title('Evaluation Metric Plot')
plt.show()
```



Displaying similar movie clusters

```
[130]: number_of_clusters = 4
print('Clustering data into {} clusters'.format(number_of_clusters))
kms = KMeans(n_clusters=number_of_clusters, random_state=42)
kms.fit(movie_embeddings)
pred = kms.predict(movie_embeddings)
# https://stackoverflow.com/questions/60935289/ego-graph-in-networkx
cost1_metric = 0
cost2_metric = 0
for ele in np.random.choice(movie_nodes, number_of_cluster):
    temp_sub_graph = A.subgraph(ele)
    cost1_metric += cost1(temp_sub_graph, number_of_clusters)
    cost2_metric += cost2(temp_sub_graph, number_of_clusters)
final_cost = cost1_metric * cost2_metric
print('Cost1 Metric Value',cost1_metric)
print('Cost2 Metric Value',cost2_metric)
print('Final Metric Value',final_cost)
```

```
Clustering data into 4 clusters
Cost1 Metric Value 1.0
Cost2 Metric Value 1.0
Final Metric Value 1.0
```

```
[131]: tsne = TSNE(n_components=2, perplexity=50)
movie_2d_transformation = tsne.fit_transform(movie_embeddings)

_, ax = plt.subplots(1, 1, figsize=(10, 10))
ax.scatter(movie_2d_transformation[:, 0], movie_2d_transformation[:, 1],
           c = pred, alpha=0.3)
ax.set_title('TSNE with Perplexity = 50')
plt.show()

tsne = TSNE(n_components=2, perplexity=30)
movie_2d_transformation = tsne.fit_transform(movie_embeddings)

_, ax = plt.subplots(1, 1, figsize=(10, 10))
ax.scatter(movie_2d_transformation[:, 0], movie_2d_transformation[:, 1],
           c = pred, alpha=0.3)
ax.set_title('TSNE with Perplexity = 30')
plt.show()
```



