Cognitive Coverage in Multi-Attribute Graph

Ishani Gupta, Sai Nikhil Maram

December 16, 2017

Contents

1 Introduction **3**
1.1 Applications . . . 3
1.2 Algorithm Design Criteria . . . 3
1.3 Data Set . . . 3

2 Algorithms **4**
2.1 Brute Force . . . 4
2.2 Max Coverage with k diversification Algorithm . . . 4

3 Data Analysis and Discussions **5**
3.1 Scenario 1 . . . 5
3.1.1 Test Scenario . . . 5
3.1.2 Execution Time Analysis . . . 5
3.1.3 Coverage Analysis . . . 6
3.2 Test Scenario 2 . . . 7
3.2.1 Test Scenario . . . 7
3.2.2 Coverage Analysis . . . 7
3.3 Test Scenario 3 . . . 7
3.3.1 Test Scenario . . . 7
3.3.2 Coverage Analysis . . . 7

4 Future Improvements **8**

5 Conclusions **8**

6 References **9**

1 Introduction

1.1 Applications

The focus of the project is choosing a committee in a multi-attributed network which ensures maximum diversification among the committee and improving coverage in the network over repeated committee formations. The algorithm we are aiming is a version of maximum coverage algorithm which ensures "k" level of diversification.A major application of the algorithm is to choose an optimal committee for non routine tasks. They are a group of individuals with largest pool of unique memory records.

Another application which can be explored keeping the algorithm in mind is marketing i.e. in case of a promotional event, who all to give a informational about the event so that they cover maximum people in a community.

Consider a promotional event for triathlon which requires you to pick few people to give a free pass so that event is publicized in the networks. So picking people whose specific interests are cycling or running or swimming would ensure us more coverage for the event rather than picking people who are only interested in running. By picking people with diversified interests you can reach to neighborhood of a diversified committee.

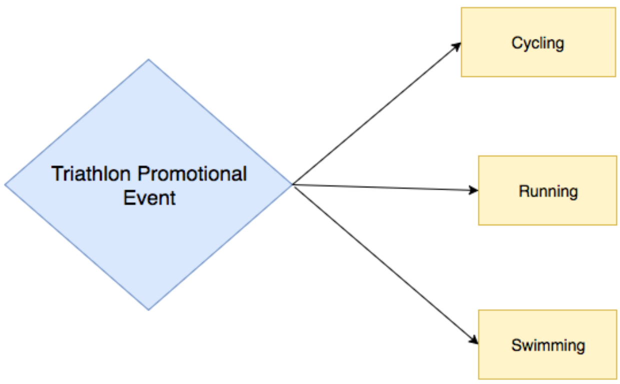


Figure 1: Promotional event marketing strategy

1.2 Algorithm Design Criteria

For the design of our algorithm we recognized 3 dimensions to the problem as following:

- Selection of size of committee (x)
- Choosing the needed attributes (k)
- Degree of representation of an attribute

1.3 Data Set

We picked our network from a number of networks sampled through the Attributed Graph Model (AGM) framework[5]. AGM consists of samples where every node has set of attributes which correlate across the edges.The network we worked with had 5906 nodes with each node having three attributes.We picked 50 nodes initially and scaled up our network up till 350 nodes to see the time our proposed algorithm takes vs the brute force algorithm.

2 Algorithms

The point of interest is choosing a committee among possible committees such that the selected committee contains all required attributes needed and ensures maximum coverage. This can be translated into maximizing equation presented below. Here the committee refers to group of nodes picked.

$$f = max(\sum_{c \epsilon C} coverage(c) * \alpha)$$

$$\alpha = \begin{cases} 1 & if\, all\, attributes\, are\, present \\ 0 & else \end{cases}$$

$$C = different\, committess\, possible$$

2.1 Brute Force

This is the basic approach we applied to set our baseline for testing our algorithm. Let x be the size of committee.This is a general solving technique which includes all the possible combination of x people from the total bulk of individuals.After the configuration is chosen we apply the filter of having all the attributes represented.

Hypothesis 1: *Given that the required attributes and degree of representation of each attribute are known, we hypothesize the following for brute force algorithm: Brute Force is an exhaustive algorithm and as the network spreads(number of nodes increases), choosing all different possible committees is an NP-Hard problem. Moreover, if we look at the terms of improving on the committee formation over time, no improvement can be seen if brute force algorithm is employed and hence it will be difficult to reach small components in the network.*

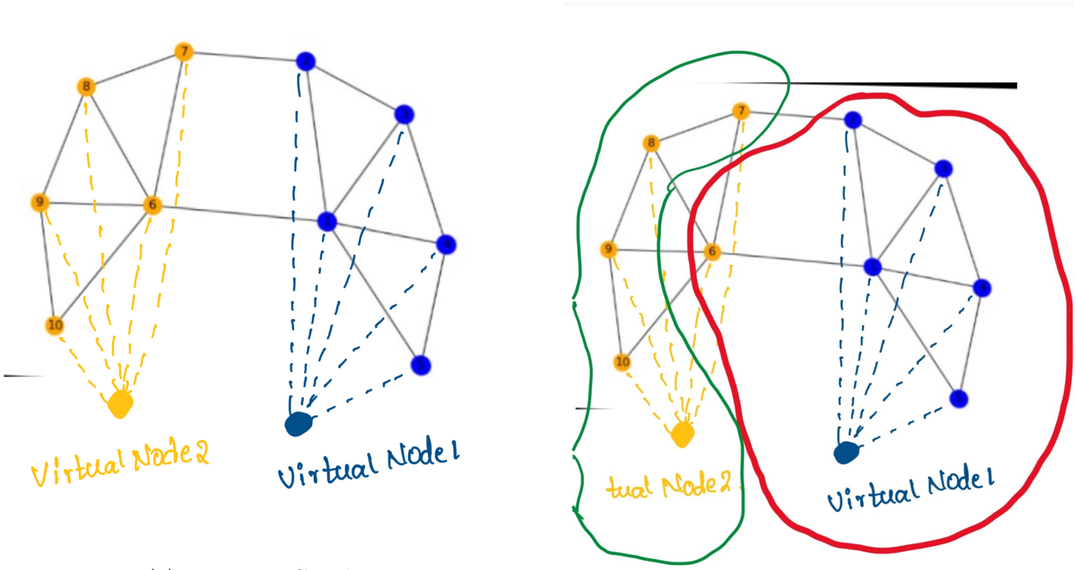
2.2 Max Coverage with k diversification Algorithm

Our main idea is based on greedy approach. The algorithm always picks a node with maximum coverage in the subgraph considered. The following are the steps involved in algorithm.

1. We first build a bipartite graph by introducing the attributes as virtual nodes and create an edge between each node and virtual node if corresponding attribute is present in the node.
2. Pick a virtual node and create a subgraph that includes all the vertices the virtual node is connected to. Pick a node with highest degree in the subgraph and remove the node and its neighbors from the graph.
3. Repeat step(2) for all the virtual nodes present.
4. We get a committee member from each subgraph created.

Hence,The proposed algorithm applies filter for diversification prior to selection and hence reduces the load for selecting the next member.This makes the problem solvable in real time.Another dimension of the problem we are concentrating on is reaching maximum coverage by making many committees over time. We assume that once we form a committee , there is enough time for people in committee to form a connection and next time we form a committee we can consider edges between them.Hence, we expect to get a huge coverage too over iterating over the committee with time.

Hypothesis 2: *Given the required attributes is known and degree of representation of each attribute , we hypothesize the following for our proposed algorithm: Max Coverage with k diversification Algorithm is going to give us approximate solution in real time for the best combination of x individuals for forming committee.We also hypothesis in reaching maximum coverage over time.*



(a) Bipartite Graph

(b) Sub-graphs w.r.t to each virtual node

Figure 2: Implementation of proposed algorithm

Implementation Details:

- 1. When picking the committee, we iterate over all different combinations of virtual nodes that can be picked. For example if v1,v2,v3 are virtual nodes present. Then different combinations of virtual nodes are considered like v1,v3,v2, v2,v1,v3 etc.. which gives us committee for each combination and committee with maximum coverage is selected among them.
- 2. If the committee size is less than the feature size, then the subgraph considered will be nodes that are connected to multiple virtual nodes and ensuring the node selected in the subgraph has both attributes corresponding to the virtual node and increasing the coverage in the network.

Code can be found here

3 Data Analysis and Discussions

Both the algorithms are employed on our dataset starting from 50 nodes and stepping up till 350 nodes. We varied our requirements of number of attributes and size of committee.The easy cases were when size of committee was greater than or equal to attributes needed to represent by the committee.

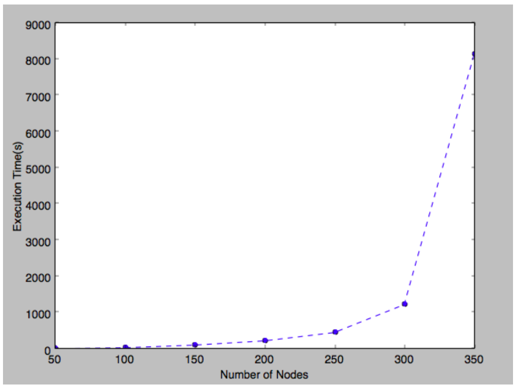
3.1 Scenario 1

3.1.1 Test Scenario

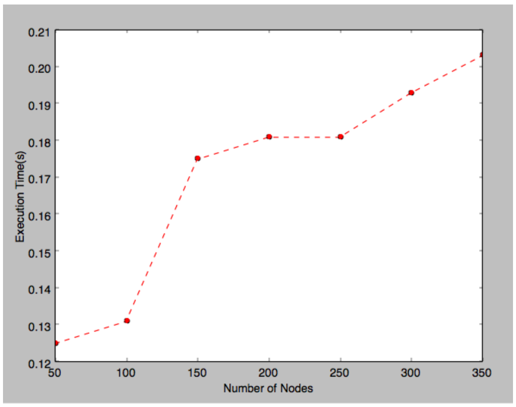
This is a simple case used to verify our hypothesis mentioned in Section 2. The scenario includes selection of 3 individuals for committee (x=3) and 3 features(k=3) mandatory to be represented by the committee. We use 50 to 350 nodes' network to find how much load our proposed algorithm can bear when compared to baseline algorithm.Another feature we tried to explore was how the committee improves over time with keeping 20 iterations.

3.1.2 Execution Time Analysis

Brute Force Algorithm We started with a network of 50 nodes and increased 50 nodes a time to reach 350 nodes.



(a) Brute Force



(b) Max Coverage with k diversification

Figure 3: Increase in execution time with increase in number of nodes in the network

As it can be seen in the Figure 3(a) , time to run brute force algorithm for 50 nodes is 4.2 seconds and it increases drastically to 8200 seconds(2.2 Hours) which is very high execution time and hence makes the problem NP Hard which complies with our **Hypothesis 1**.

Max Coverage with k diversification Algorithm For the purpose of comparing the performance of our proposed algorithm with the brute force algorithm, we employ similar conditions i.e. We started with a network of 50 nodes and increased 50 nodes a time to reach 350 nodes. As it can be seen in the Figure 3(b), time to run the proposed algorithm for 50 nodes is 0.125 seconds and it increases gradually to 0.2031 seconds which is executing the problem in real time. Hence, verifying our **Hypothesis 2**.

3.1.3 Coverage Analysis

Next, we worked on checking the improvement in coverage (i.e. maximum reach out) as we iterate over the committee keeping **Assumption 1** intact.We have 100 nodes in the network with size of committee =3 and number of needed attributes=3.

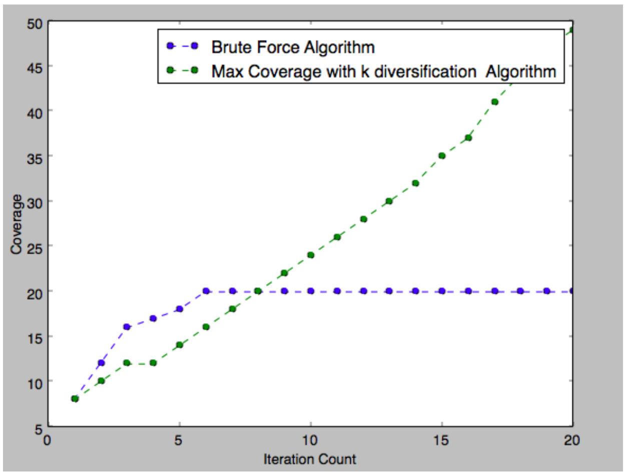


Figure 4: Increase in coverage with iterations in the network for test scenario 1

Figure 4 gave us very positive results. As expected, our proposed algorithm improved drastically over each iteration.Brute Force algorithm stagnates at 20% coverage whereas the proposed algorithm started with coverage of 8 nodes and rose up to 49 nodes out of 100 with 20 iterations.Thus, Increasing coverage from 8% to almost 50% which are great results to start with !

3.2 Test Scenario 2

3.2.1 Test Scenario

In this case the committee size is more than the number of features that the committee need to represent. The scenario includes selection of 4 individuals for committee (x=4) and 3 features(k=3) mandatory to be represented by the committee. A feature can be given extra weight in selecting the committee such that 2 members in the committee can posses that feature. In our current implementation we have given priority to the feature which ensures us the maximum coverage in the network. We use 50 to 350 nodes' network to find how much load our proposed algorithm can bear when compared to baseline algorithm.Another feature we tried to explore was how the committee improves over time with keeping 20 iterations.

3.2.2 Coverage Analysis

In this test, we compare the coverage reached by our proposed algorithm in comparison to one reached by brute force algorithm.We do 20 iterations with the vision on improving the coverage with each iteration.

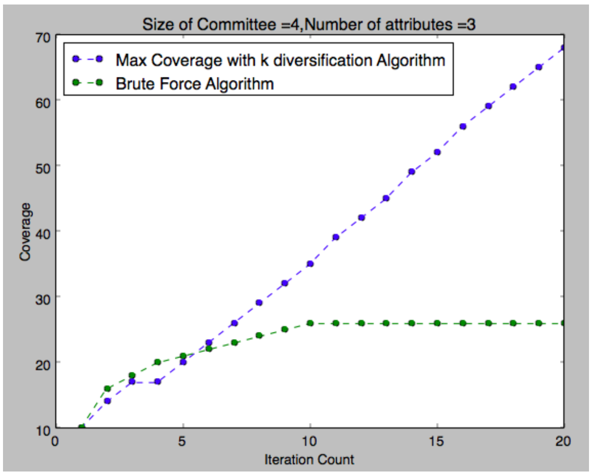


Figure 5: Increase in coverage with iterations in the network for test scenario 2

Figure 5 gave us very positive results. As expected, our proposed algorithm improved drastically over each iteration.Brute Force algorithm stagnates at 26% coverage whereas the proposed algorithm started with coverage of 10 nodes and rose upto 68 nodes out of 100 with 20 iterations.Thus, Increasing coverage from 10% to almost 68%.

3.3 Test Scenario 3

3.3.1 Test Scenario

In this case the committee size is less than the number of features that the committee need to represent. The scenario includes selection of 2 individuals for committee (x=2) and 3 features(k=3) mandatory to be represented by the committee. In our current implementation, we chose the committee members such that one member possess two features and second member posses the other feature while ensuring the maximum coverage in the network. We use 50 to 350 nodes' network to find how much load our proposed algorithm can bear when compared to baseline algorithm.Another feature we tried to explore was how the committee improves over time with keeping 20 iterations.

3.3.2 Coverage Analysis

In this test, we compare the coverage reached by our proposed algorithm in comparison to one reached by brute force algorithm when the committee size =2.We do 20 iterations with the vision on improving the coverage with each iteration.

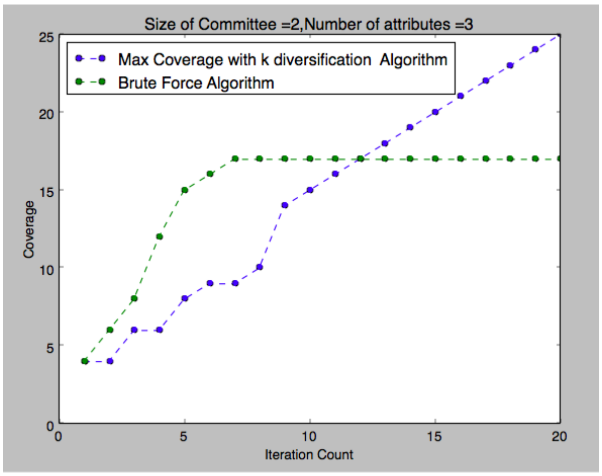


Figure 6: Increase in coverage with iterations in the network for test scenario 3

As seen in Figure 6, The proposed algorithm provides with a coverage up to 25% where as brute force provides up to 17%. This is the hypothesized behavior.

4 Future Improvements

There are several domains which can be explored in order to improve our proposed algorithm. Across our tests, we have worked with different committee sizes. We have yet to explore other two design criteria.

Firstly, deciding upon the important attributes which are needed to be represented by the committee(for the case where size of committee $<$ number of attributes).Although not studied throughly yet, we vision to use machine learning to decide the rank various features on basis upon their importance for the specific case of committee formation.

Secondly,interrogating the degree of representation of the attributes in the selected committee(for the case when size of committee $>$ number of attributes) will be an interesting area to explore.

5 Conclusions

Proof of Correction As we can see in Figure 7 , Increasing the committee size improves the coverage.For committee size =2, coverage reached is 25% , for committee size =3 , coverage reached is 49% and for committee size =4, coverage reached is 68%.This is an expected behavior and a strong argument for correctness of our algorithm.

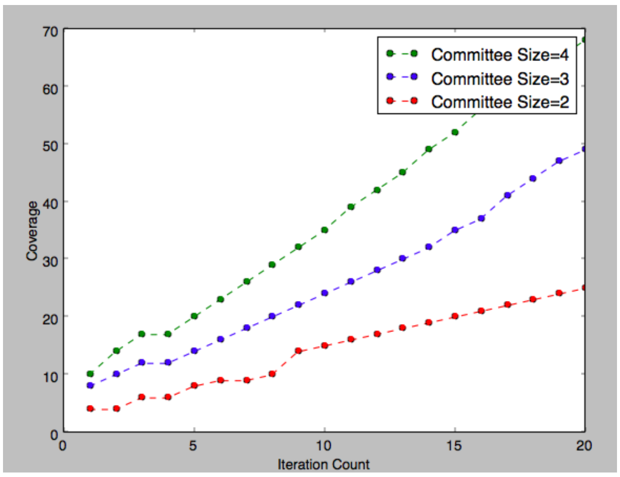


Figure 7: Increase in coverage with iterations in the network for all 3 test cases

6 References

References

[1] Alex.T.Jones, Noah.E.Friedkin and Ambuj.K.Singh *Modeling the Co-evolution of Committee Formation and Awareness Networks in Organizations.*

[2] Cecile Bothorel, Juan David Cruz, Matteo Magnani and Barbora Micenkova *Clustering attributed graphs: Models, measures and methods*

[3] Ashwin Bahulkar and Boleslaw K. Szymanski and Kevin Chan and Omar Lizardo *Co-evolution of two networks representing different social relations in NetSense*

[4] Daniel M. Wegner, Paula Raymond, Ralph Erber *Transactive Memory in Close Relationships*

[5] Joseph J. Pfeiffer III and Sebastian Moreno and Timothy La Fond and Jennifer Neville and Brian Gallagher,
= Attributed Graph Models: Modeling network structure with correlated attributes.
`https://www.cs.purdue.edu/homes/jpfeiff/agm/agm.html`