Fast Shortest Path Distance Estimation in Large Networks

Network Analysis

Abhishek Akshat

ABSTRACT

Many real-world applications rely on computing distances between node pairs. In this paper, we explored methods to estimate shortest path distance between two nodes in large networks fast using various strategies proposed. We used landmark-selection strategies to estimate distance estimation on real-world graphs. We implemented approximate landmark-based methods for point-to-point distance estimation. The central idea was to select a subset of nodes as landmarks and compute the distances offline from each node in the graph to those landmarks. In the course of run-time, we can use these pre-computed distances from landmarks to estimate distance between two nodes. We tested the robustness and efficiency of these techniques and strategies with five large real-world network datasets. We extended our work by applying these methods on directed graphs as well. We also explored a new landmark selection strategy based on approximate betweenness centrality. Our experiments suggest that optimal landmark selection can yield more accurate results faster than the traditional approach of selecting landmarks at random. We evaluate the efficiency of these strategies using approximation error and discuss the results obtained.

KEYWORDS

Graphs, shortest-paths, landmarks methods, social network analysis, network science

ACM Reference Format:

Abhishek Akshat. 2022. Fast Shortest Path Distance Estimation in Large Networks: Network Analysis. In *Proceedings of Network Analysis.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the past few years, we have observed an increase in the availability of large networks such as social media networks, communication network, etc. As the popularity for such social media keeps on increasing, so does the size of large networks. As the graph size increases, the complexity to run even basic algorithms such as computation of shortest path from one node to another also increases which results in consumption of more time and resources.

Many applications rely on computing distances between node pairs. For example, shortest paths are used in social networks to predict new links, i.e. friends that haven't added each other yet, and similarly in protein-protein interaction networks, the distance is a measure of how much influence one protein has on another. The complexity for such computations increases with increase in the size of network graphs. Computing the shortest distance between a pair of nodes is an essential operation that can take a long time in large graphs using traditional algorithms. Therefore, we need fast and smart measures to tackle this issue. To tackle this problem, various methods has been introduced by many researchers in the past years.

In this paper, we present methods to compute shortest path distance between two nodes using landmark-based approaches using the methods and algorithms proposed in the research paper by Potamias et al. [5] on five different large real-world network datasets and present the analysis of the results. The authors [5] applied these methods only on undirected graphs on five large real-world network datasets, therefore, we are investigated these methods on directed as well as undirected graphs and present our results.

Our main contributions are summarized as follows:

- We explore and experiment with multiple landmark selection strategies.
- We investigate a new approach for selecting landmarks i.e. using approximate betweenness centrality of the nodes.
- We test the efficiency and accuracy of our methods on five real-world dataset out of which, we have three undirected and two directed graphs.
- We use approximation error to check the discrepancy between the actual distance and approximated distance.

For our experiments, we used five real world networks: Two undirected social graphs from GitHub and Twitch, a undirected road network of Pennsylvania, a directed web graph from Stanford and finally a directed communication network graph from European Union Email. Furthermore, we explored a new landmark selection strategy based on approximate betweenness centrality.

Paper Structure. In Section 2 we introduce our notations and definitions. In Section 3 we outline the related work. In Section 4 we present our approaches used. In Section 5 we introduce the datasets we have used. In Section 6 we discussed the experiments performed followed by results and discussion in sections 7 and 8,respectively. Finally, Section 9 presents some concluding remarks.

2 PRELIMINARIES

In this section, we describe the notations and definitions along with our problem statement.

2.1 Graph

Graphs consist of nodes also known as vertices or actors that represents the unit of analysis, and links also known as relationships or edges are connections that connect the mentioned nodes in some significant way. For example: In a social network, the nodes represent users and the edges may represent friendship, acquaintance or communication. Let us consider a graph $G(V, E)$ with n vertices

Network Analysis, Report, 2022

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

and m edges, where n may represent users and m may represent mutual relationships.

2.2 Shortest Path

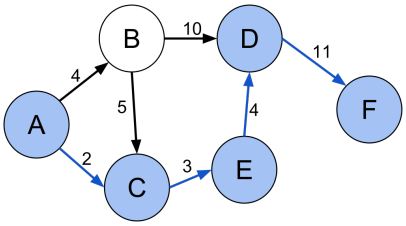
A shortest path between two nodes in a graph is a path with the minimum number of edges. If the graph is weighted, it is a path with the minimum sum of edge weights. Given two vertices $x, y \in V$, let $d_G(x, y)$ be the length of the shortest path between any two vertices, we refer to this as the geodesic distance or shortest path distance between such vertices. The shortest-path distance in graphs satisfies the triangle inequality i.e. For any three nodes a, b, and c, the following inequalities holds true:

$$d_G(a, c) \leq d_G(a, b) + d_G(b, c) \tag{1}$$

$$d_G(a, c) \geq \mid d_G(a, b) - d_G(b, c) \mid \tag{2}$$

For example, In figure 1, we have a weighted directed graph, the shortest path from node A to node F is A > C > E > D > F.

Figure 1: Weighted Directed Graph with 6 Nodes



2.3 Landmarks

Landmarks are a small subset of nodes from the complete set of nodes. These landmarks are used to estimate the distance between two points in a similar way to how one might estimate the distance between two points using a landmark in real life.

A set of landmark nodes is selected and distances or actual shortest paths are pre-computed from each vertex to all of the landmarks reachable from the given vertex. We can then quickly compute approximate distances between any two vertices in *O(k)* time, where k is the number of landmarks. We can use these estimates as a component of a graph traversal in order to obtain a shortest path from one node to another.

Consider a graph G with n nodes and m edges, and a set of d landmarks D. We pre-compute the distances between each nodes in G and each landmark offline. The cost of this computation is d BFS traversals of the graph i.e. $O(md)$.

From equation 1 and 2, we know that for any two vertices $x, y \in V$, $d_G(x, y)$ lies in the range of lower Bound (L) and Upper Bound (U) i.e.

$$L \leq d_G(x, y) \leq U \tag{3}$$

where,

$$L = max_i \mid y_i - x_i \mid \tag{4}$$

$$U = min_j\{y_j + x_j\} \tag{5}$$

Potamias et al. [5] experiments and results indicate that the upper-bound (U) estimates work much better than the other types of estimates. Therefore, we focused our work on upper-bound estimates. The estimation takes only $O(d)$ operations where d is a constant, namely the number of landmarks used.

The algorithm does need to store the precomputed distances in memory. This has a worst case space complexity of $O(md)$. As memory is not very expensive anymore, this is not a very big cost for smaller networks, however as the size of the network increases, this can become a significant factor.

2.4 Good Landmarks

According to Potamias et al. [5], a good landmark is a node that is very central in any given graph and has many shortest paths passing through it i.e. node with highest betweenness centrality. Mathematical Expression of Betweenness Centrality [2]:

$$C_b(u) = \sum_{v,w \in V} \frac{\sigma_u(v, w)}{\sigma(v, w)} \tag{6}$$

where, $\sigma(v, w)$ is the number of shortest paths from v to w and $\sigma_u(v, w)$ is the number of such shortest paths that run through u.

Potamias et al. [5] used two approaches for selecting central nodes i.e. (i) nodes with high-degree, (ii) nodes with low closeness centrality.

Closeness centrality is defined as the average distance of a given node to other nodes in the graph. Mathematical Expression of Closeness Centrality:

$$C_c(v) = \frac{1}{n} \sum_{w \in V} d_G(v, w) \tag{7}$$

2.5 Problem Statement

Potamias et al. [5] used these methods on undirected, unweighted graphs for the simplicity of exposition. In this paper, we used the strategies described by the original paper [5] for landmark selection and applied it to undirected as well as directed networks on five different large real-world network datasets and we also explore a new landmark selection strategy based on approximate betweenness centrality.

3 RELATED WORK

This section describes related work with respect to our project.

For unweighted graphs with n nodes and m edges, the cost of computing the Single Shortest Path from a node to all others using Breadth First Search is $O(m + n)$. For weighted graphs, Dijkstra [3] proposed an algorithm that computes Single Shortest Path in time $O(n^2)$.

The landmark methods have been widely used in the application of network graphs [1, 5, 8–10]. In the paper by Zhao et al. [10], the authors introduced Orion, a prototype graph coordinate system that uses landmarks for applications such as computing node separation, centrality computation, mutual friend detection, and community detection. However, the challenge of optimal selection of landmarks is still being widely addressed by many researchers.

Potamias et al. [5] provided evidence in their paper that the problem of optimal landmark selection is NP-hard problem. The authors

of the paper [5] investigated various strategies to choose landmarks that scale well to huge graphs. The application of these methods to social search queries showed high precision and accuracy.

F. W. Takes and W. A. Kosters [8] also suggested methods that outperform landmark selection techniques based on centrality. The authors proposed a new adaptive set of rules for selecting landmarks that picks central nodes and also ensures that the landmarks selected properly covers different areas of the graph.

Akiba et al. [1] proposed a exact method for shortest-path distance queries on large-scale networks. The authors method precomputes distance from selected landmarks by performing a BFS from every vertex while simultaneously pruning during breadth-first searches.

Tretyakov et al. [9] also suggested an improvement in landmark selection strategy. The authors propose a greedy approach to select landmarks that provide the best coverage of all shortest paths in a random sample of vertex pairs. The authors were able to achieve notable accuracy improvements with their proposed method. The authors of the paper only experimented with undirected graphs.

In our paper, we are investigating the methods proposed by [5] and applied these methods on directed as well as undirected graphs. Furthermore, we explored a new landmark selection strategy based on approximate betweenness centrality and present our results and conclusions.

4 APPROACH

In this section, we describe the approaches and techniques used in this project.

4.1 Landmark-Selection Strategies

The central idea in the paper by Potamias et al. [5] is to use a small set of nodes as landmarks to quickly estimate the distance between any two points in the graph. This is achieved by first calculating the distance from each node to all of the landmarks. These precomputed distances can then be used to give an estimate of the actual distance.

In order to calculate the distance for any node to a given landmark on an undirected graph we perform a BFS from the landmark d. For directed landmarks we perform a BFS on the original graph, this returns the distance from the landmark to all nodes reachable form d. The distance to the landmark from all points that can reach d can be calculated by performing BFS on the reversed graph.

After the distances have been computed it is very simple to give an upper bound on the distance between two different points. For two points s and t, $U = min\{d_G(s,d) + d_G(d,t)\}$. where $d \in D$ This is an upper bound on the distance between the two points. If a landmark is (indirectly) connected to both nodes, then there exist a path of length $d_G(s,d) + d_G(d,t)$ through the landmark. The shortest path between two points trough a landmark is then an upper bound on the actual shortest path. This upper bound is equal to the upper bound if the landmark is on the shortest path between the two nodes.

Potamias et al. [5] noted that selecting good landmarks is instrumental in obtaining good results with this method. In fact, selecting the best landmark is related to finding the node with the highest betweenness centrality. The strategies to choose landmarks are discussed in this section. The computation of centrality for all nodes

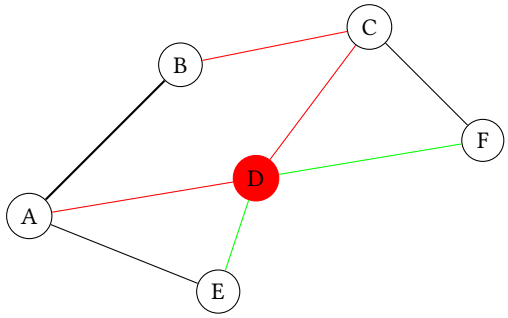


Figure 2: Visualisation of landmarks. Vertex D is a landmark. The distance between points A and B are estimated. $d_G\{A,B\}$ is 1. By using landmark D, the estimated distance is 3. The distance between E and F is estimated correctly.

is not strategy scalable for very large graphs with millions of nodes and edges. Therefore, we are computing approximate centralities for this task. As descried by [5], we can divide landmark-selection strategies in following categories namely basic and constrained strategies. We are also exploring a new landmark selection strategy based on approximate betweenness centrality in this paper.

4.1.1 *Basic Strategies:*

To select a set of d landmarks, three baseline strategies that were suggested by Potamias et al. [5] namely, Random, Degree and Centrality based on closeness centrality. We are also exploring this basic landmark selection strategy based on approximate betweenness centrality. Brief details about each strategies is discussed below:

- RANDOM: In this landmark selection strategy, we select a set of d nodes uniformly at random from the graph.
- DEGREE: In this landmark selection strategy, we sort the nodes of the graph by decreasing degree and then we choose the top d nodes. The intuition of the authors of [5] behind this was that the more connected a node is in a graph, the higher the chance that node participates in many shortest paths.
- CLOSENESS CENTRALITY: In this landmark selection strategy, we select a set of d landmark nodes with the lowest closeness centrality. The intuition of the authors of [5] behind this was that the closer a node appears to the rest of the nodes, the chances that it is part of many shortest paths is further more.
- BETWEENNESS CENTRALITY: In this landmark selection strategy, we select a set of d landmark nodes with the lowest approximate betweenness centrality. This centrality measure estimates the betweenness using a small set of nodes.

4.1.2 *Constrained Strategies:*

The central idea suggested by Potamias et al. [5] is to choose nodes that cover as many pairs as possible. While using the above basic landmark selection strategies, the selected landmarks might have a lot of pairs that are common among them. Therefore, the choice of landmarks in these cases is not always optimum. We also explored constrained method using betweenness centrality. The constrained

variant of the strategies suggested by Potamias et al. [5] depends on a depth parameter h.

Steps involved in the selection of a set of d landmark:

- Rank the nodes according to one of the following strategies:
  - Random
  - Highest Degree
  - Lowest Closeness Centrality
  - Lowest Betweenness Centrality
- Select landmarks iteratively based to their ranks.
- For each selected landmark l, we remove all nodes that are at distance h or less from l.
- Repeat the process until d landmarks are selected.

For the experiments in this project we have only considered $h = 1$ as suggested by [5].

4.2 Working with Directed Networks

Extending the algorithm to work with directed networks requires calculating two different sets of distances for each landmark. The distance from the landmark to any vertex, and the distance from that same vertex to the landmark. Estimating the distance between two different vertices then uses both the distances to calculate an upper bound on the real distance.

4.3 Approximate Betweenness Centrality

Another extension to the original paper was the implementation of the approximate betweenness landmark selection strategy. The authors of the original paper mentioned that the selection of good landmarks was similar to finding nodes with a high betweenness centrality, but they did not implement such a strategy in their paper. Calculating the exact betweenness centrality for large networks is extremely expensive, but there are implementations for the approximate betweenness in several network analysis packages.

5 DATA

In this section, we briefly describe the data we have used for our experiments. We used five different real-world network datasets out of which three are undirected graphs and two are directed graphs. Table 1 displays the description of the datasets used. We briefly describe the datasets used during this project below. We selected three undirected graphs:

- **Github Social Network[6]:** This dataset is a large social network of Github. Nodes are individual users, and vertices represent mutual follower relationships. The network contains 37,700 nodes, and 289,003 edges.
- **Twitch Streamers Social Network[7]:** This is a social network of Twitch streamers active on the site in the spring of 2018. Nodes are individual users, and users are linked if they follow each other. An interesting property of this network is that it has a single connected component. It consists of 168,114 nodes and 6,797,557 edges.
- **Pennsylvania Road Network:** The road network of Pennsylvania, USA. Nodes are intersections and dead ends, edges represent the roads. There are 1,088,092 nodes, and 1,541,898 edges. The diameter of the graph is 786, and the effective diameter, the diameter if the longest 10% of the shortest paths are ignored, is $\tilde{5}30$. This is obviously not a small world network, but this network seemed relevant and interesting considering the objective of this paper.

In order to validate the results for our extensions to the algorithm we also selected the following two directed networks:

- **EU email communication network[4]:** This dataset is an anonymised email graph of a large European research institute from October 2003 to May 2005. Each node is an email address and a directed link from i to j means that the email address i sent at least one message to j. There are 265,214 nodes and 420,045 edges in the graph.
- **Stanford web graph[4]:** This is a graph of web pages from Stanford University in 2002. Directed links are hyperlinks between the different web pages. There are 281,903 nodes, and 2,312,497 edges.

6 EXPERIMENTS

In this section, we describe the experimental setup and results obtained by applying the our approach on the selected datasets in the preceding section.

The algorithms described in the previous section were implemented using Networkx in python. The precomputed distances were stored in dictionaries for fast retrieval of the values. The experiments were run using the full graphs present in the datasets, without selecting the largest connected components. The program was run on one of the LIACS DSlab servers (Uridium).

The approximate betweenness was calculated using the betweenness function in networkx. This function can also calculate the approximate betweenness on a subset of n nodes, estimating the betweenness using a BFS. The number of nodes used for this estimation was $\sqrt[3]{N}$ where N is the number of nodes in the network. This function was chosen heuristically such that the number of times BFS was used would not increase dramatically with the size of the network, as the number of nodes for the largest dataset was very large.

The error of the algorithm was measured using the approximation error $|\hat{d} - d|/d$ where d is the actual distance between the vertices, and $\hat{d}$ is the estimated distance. In the case that there are no landmarks that can be used to calculate the distance between the two points, i.e. there is no landmark that can reach both of the required nodes, then a distance of -1 is returned. This will result in an error of ≤ 2, dependent on the distance between the two points.

In order to generate pairs of points, 20 random points in the graph were selected, and a BFS was used to find all reachable points in the graph from these nodes, and to find the distances to all the other points. All of these distances were then used in evaluating the performance of the algorithm. The same set of node pairs were used in evaluating each selection strategy for a given dataset and number of landmarks.

We evaluate the performance of the landmark distance estimations using several different numbers of landmarks, 1, 10, 25, 50, 75, 100, 125, and 150 landmarks respectively. Each selection strategy is evaluated using these numbers of landmarks, and each combination of selection strategy and number of landmarks is evaluated on each dataset. The exception to this is the closeness and constrained closeness strategy. Due to time constraints, these selection strategies

Dataset	Type	Directed/ Undirected	Nodes	Edges
GitHub	Social Network	Undirected	37 K	289 K
Twitch Gamers	Social Network	Undirected	168 K	6 M
Pennsylvania Road	Road Network	Undirected	1 M	1.5 M
Stanford Hyperlinks	Web	Directed	281 K	2.3 M
EU Email	Communication Network	Directed	265 K	420 K

Table 1: Description of Datasets Used

were only evaluated on the Github, and Email datasets. The estimation errors for all pairs of points mentioned above are calculated for each of the evaluations of the algorithm, and the mean of this error is then added to the results.

7 RESULTS

The estimation error for all experiments run are graphed in figure 3. The numerical results of the runs with 10 and 75 landmarks are shown in tables 2 and 3 respectively. Several of the strategies implemented yielded very accurate results. Degree and Betweenness were the best in all but one of the datasets across all numbers of landmarks. In the two datasets where constrained closeness was evaluated, this strategy too performed very well. The other constrained strategies did not perform as well as expected, most of them performing significantly worse compared to even the random selection strategies. Some of the strategies and datasets had some errors during the evaluation, which is why some datapoints are missing.

We do not see any differences in the performance of the algorithm between the directed and undirected algorithms. The Road network in some of the evaluation metrics also scored similarly to other datasets, indicating that the algorithm could be quite robust, even for networks that do not possess scale free qualities.

8 DISCUSSION

Some of the results do support the conclusion reached by the authors of Potamias et al. however some results do not. In particular a lot of the constrained strategies, and the closeness selection strategy have errors that are higher than the random selection strategy.

There could be several reasons for these abnormal results. One possibility could be that the method used to generate vertex pairs for evaluation did not cover enough nodes, as only a relatively small number of nodes was selected as a starting point.

Second, the presence of multiple connected components has in some occasions meant that no path was found in the landmark set. This problem should however become less likely as the size of the landmark set becomes larger, although this problem will be compounded by the previous problem. This is because if one of these nodes is in a different component to all the landmarks, then all of the results of that node will be invalid. In future studies both of these problems should be taken into account, either by only considering the giant component, and/or by changing the evaluation of the algorithm. The second point, of disconnected components, however does not account for the very high error by the Closeness strategy in the Github dataset.

One other explanation for the abnormal results regarding the constrained degree and constrained betweenness might be found in the structure of scale free networks themselves. Nodes with a very high degree are mostly hubs that are very well connected within the network. These nodes usually also have a low closeness, and a high betweenness in the network. In the constrained strategies every vertex connected to already selected landmarks are removed from consideration, and as the nodes with high degree and betweenness are likely connected to these nodes. If a lot of nodes exist with a path through these discarded nodes and not through the point selected as a landmark, then the overall quality of the estimations could decrease potentially. On the other hand, spreading landmarks throughout the network can increase the coverage of the landmark set, increasing the accuracy. This would have to be investigated further, and it is very likely that the structure of each individual network would have a large effect on the presence and strength of these effects. A thourough investigation of the data in the original paper did not yield any significant insights to these results.

Even though the constrained betweenness strategy did not perform very well, this strategy performed similarly to the constrained degree strategy, indicating that this lower performance is likely due to the same mechanism responsible for the poor performance of the constrained degree strategy. The betweeenness did not yield significantly better results compared to degree, performing about equally. The betweenness did perform worse on the road network. This network is not a scale free network, which is likely the cause of the worse performance. In a scale free network the nodes are connected through a relatively densly connected core, which is were the betweenness would select most of the nodes. In the road network there likely are not as many nodes functioning as a hub where a lot of shortest paths pass through, which means that the betweenness strategy would have more difficulties in selecting the best nodes. High degree nodes however are likely relatively central in cities as these represent big, significant intersections or interchanges.

9 CONCLUSION

Many applications rely on computing distances between node pairs. The main purpose of our project is to estimate shortest path distance between two nodes in large networks fast using various strategies proposed. We used landmark-selection strategies to estimate distance estimation on real-world graphs. We implemented approximate landmark-based methods for point-to-point distance estimation in very large networks as suggest by Potamias et al. [5]. The central idea was to select a subset of nodes as landmarks and compute the distances offline from each node in the graph to those landmarks. In the course of run-time, we can use these pre-computed distances from landmarks to estimate distance between two nodes. We tested

10 landmarks									
Dataset	betweenness	betweenness constrained	closeness	closeness constrained	degree	degree constrained	random	random constrained	
Email	0.1115	1.2427	1.2427	0.1496	0.2795	1.2510	1.0196	1.2510	
GitHub	0.0406	1.0478	2.9987	0.2534	0.0496	0.9659	0.6806	0.6918	
Road	1.0402	1.6396	0.5069	1.8447	0.3774	0.4330			
Twitch	0.1028	1.7047	0.1053	1.7789	0.6680	0.6312			
Stanford	0.0495	1.0895	0.0338	1.1006	0.2062	0.7522			

Table 2: Mean estimation error for all datasets with 10 landmarks

75 landmarks								
Dataset	betweenness	betweenness constrained	closeness	closeness constrained	degree	degree constrained	random	random constrained
Email	0.0282	1.2465	1.2465	0.1098	0.0358	1.2425	0.8429	0.7476
GitHub	0.0119	0.8905	2.0319	0.1951	0.0136	0.9302	0.5071	0.5097
Road	0.6709	1.4638	0.0850	1.4765	0.0796	0.0904	0.3774	0.4330
Twitch	0.0361	1.5031	0.0519	1.4510	0.4971	0.5374	0.6680	0.6312
Stanford	0.0227	1.0918	0.0127	1.0940	0.3645	0.3051	0.2062	0.7522

Table 3: Mean estimation error for all datasets with 75 landmarks

the robustness and efficiency of these techniques and strategies with five large real-world network datasets, out of which three datasets were undirected and two were directed graphs. We also explored a new landmark selection strategy based on approximate betweenness centrality. Some of the results matched the original paper, and we were able to sucesfully extend the algorithms to directed networks. The approximate betweenness selection strategy performed similarly to the degree selection strategy. There were som odd results in the results as well, these were likely a result of the evaluation method beeing less than perfect, although other causes can not yet be excluded.

The selection of optimal landmark remains NP-hard problem. However, various methods proposed by many researchers showed that proper selection of landmarks show increase in efficiency and accuracy compared to traditional methods. In future work, we can use various combinations of the aforementioned strategies which might result in even more improved efficiency and accuracy.

REFERENCES

[1] Takuya Akiba, Yoichi Iwata, and Yuichi Yoshida. 2013. Fast Exact Shortest-Path Distance Queries on Large Networks by Pruned Landmark Labeling. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data* (New York, New York, USA) *(SIGMOD '13)*. Association for Computing Machinery, New York, NY, USA, 349–360. https://doi.org/10.1145/2463676.2465315

[2] Ulrik Brandes. 2001. A Faster Algorithm for Betweenness Centrality. *Journal of Mathematical Sociology* 25 (2001), 163–177.

[3] Edsger W Dijkstra. 1959. A note on two problems in connexion with graphs. *Numerische mathematik* 1, 1 (1959), 269–271.

[4] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2007. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)* 1, 1 (2007), 2–es.

[5] Michalis Potamias, Francesco Bonchi, Carlos Castillo, and Aristides Gionis. 2009. Fast Shortest Path Distance Estimation in Large Networks. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management* (Hong Kong, China) *(CIKM '09)*. Association for Computing Machinery, New York, NY, USA, 867–876. https://doi.org/10.1145/1645953.1646063

[6] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. 2019. Multi-scale Attributed Node Embedding. arXiv:1909.13021 [cs.LG]

[7] Benedek Rozemberczki and Rik Sarkar. 2021. Twitch Gamers: a Dataset for Evaluating Proximity Preserving and Structural Role-based Node Embeddings. arXiv:2101.03091 [cs.SI]

[8] Frank W. Takes and Walter A. Kosters. 2014. Adaptive Landmark Selection Strategies for Fast Shortest Path Computation in Large Real-World Graphs. In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, Vol. 1. 27–34. https://doi.org/10.1109/WI-IAT.2014.13

[9] Konstantin Tretyakov, Abel Armas-Cervantes, Luciano García-Bañuelos, Jaak Vilo, and Marlon Dumas. 2011. Fast Fully Dynamic Landmark-Based Estimation of Shortest Path Distances in Very Large Graphs. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (Glasgow, Scotland, UK) *(CIKM '11)*. Association for Computing Machinery, New York, NY, USA, 1785–1794. https://doi.org/10.1145/2063576.2063834

[10] Xiaohan Zhao, Alessandra Sala, Christo Wilson, Haitao Zheng, and Ben Y Zhao. 2010. Orion: shortest path estimation for large social graphs. *networks* 1 (2010), 5.

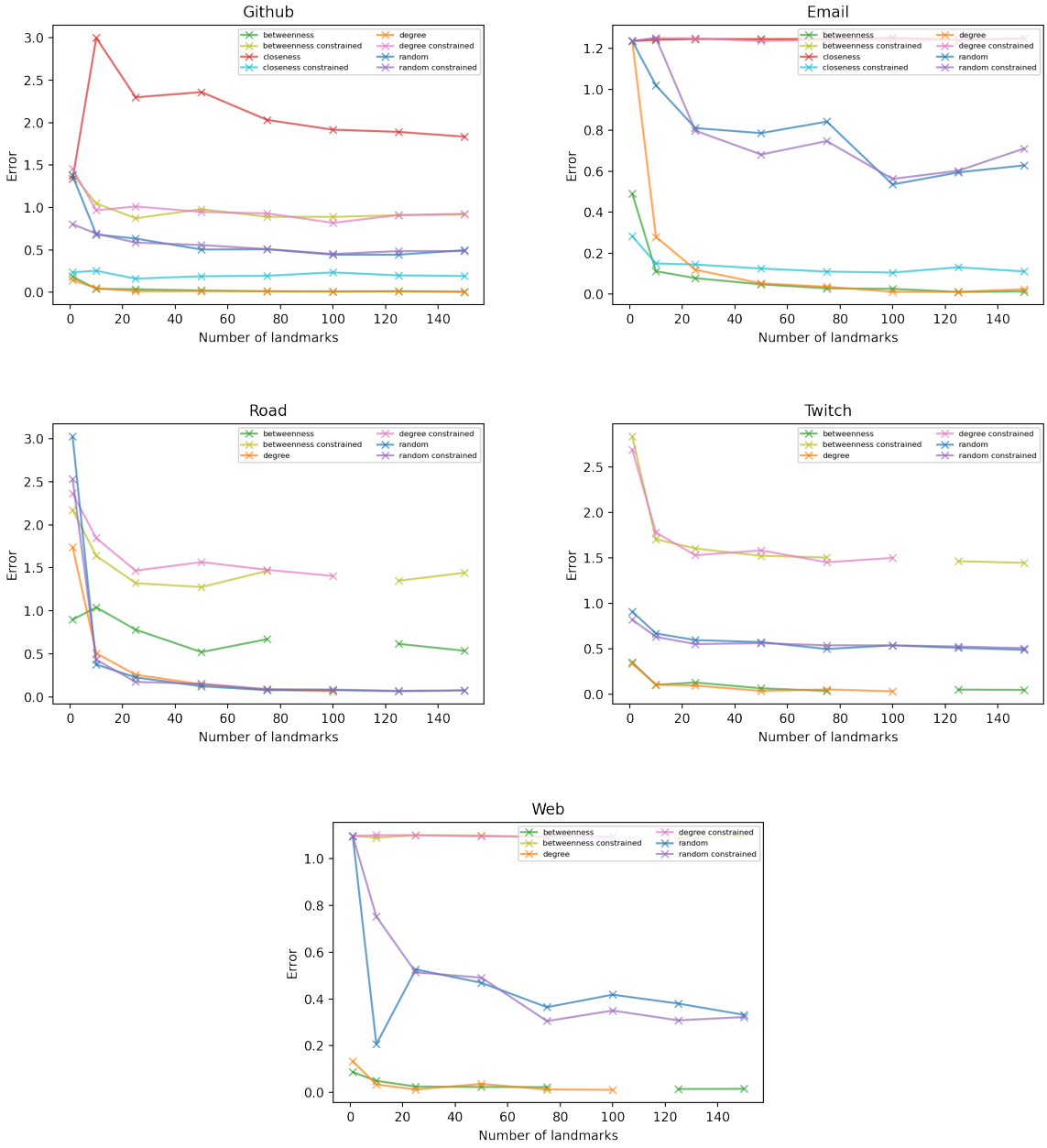


Figure 3: Estimation error for all datasets and all selection strategies.