Closeness Centrality in Neural and Interconnection Networks

Jane Olive Sharon[1], T.M. Rajalaxmi[2]
[1]Department of Computer Science,
SSN College of Engineering, Chennai-603 110, India
[2]Department of Mathematics,
SSN College of Engineering, Chennai-603 110, India
janeolivesharon@gmail.com

Abstract

In graph theory and network analysis, centrality measures identify the most important vertices within a graph. In a connected graph, closeness centrality of a node is a measure of centrality, calculated as the reciprocal of the sum of the lengths of the shortest paths between the node and all other nodes in the graph. In this paper, we compute closeness centrality for a class of neural networks and the sibling trees, classified as a family of interconnection networks.

Keywords: Centrality, Network, Closeness centrality, Neural network, Transmission of a vertex, Sibling trees.

1 Introduction

Centrality is a key concept in network studies. As the everyday use of the term implies, it means that a person or organization is in some way a focal point or main figure in whatever group of people or organizations is being considered. Based on studies of small groups and the flow of information in hypothetical networks of different shapes and sizes, some network analysts hypothesize that centrality may be an indicator of power if it is assumed that the person or organization is a gathering point for information, with the information contributing to power because of its importance [1].

1.1 Closeness Centrality

In a connected graph, closeness centrality of a node is a measure of centrality and is calculated as the sum of the lengths of the shortest paths between the node and all other nodes in the graph. Thus the more central a node is, the closer it is to all other nodes.

For a graph $G(V, E)$, the closeness of a vertex x in G is defined as

$$C(x) = \frac{1}{\sum_{y \in V} d(y, x)},$$

where $d(y, x)$ is the distance between the vertices y and x. When speaking of closeness centrality, we usually refer to its normalized form which represents the average length of the shortest paths instead of their sum. It is generally given by the previous formula multiplied by -1 , where N is the number of nodes in the graph. For large graphs this difference becomes inconsequential, so the -1 is dropped resulting in:

$$C(x) = \frac{N}{\sum_{y \in V} d(y, x)}.$$

This adjustment allows comparisons between nodes of graphs of different sizes. Taking distances from or to all other nodes is irrelevant in undirected graphs, whereas it can produce totally different results in directed graphs. For example, a website can have a high closeness centrality from outgoing link, but low closeness centrality from incoming links.

1.2 Betweenness Centrality

In graph theory, betweenness centrality is a measure of centrality in a graph based on shortest paths. For every pair of vertices in a connected graph, there exists at least one shortest path between the vertices such that either the number of edges that the path passes through or the sum of the weights of the edges is minimized. The betweenness centrality for each vertex is the number of these shortest paths that pass through the vertex. Betweenness centrality finds wide applications in network theory: it represents the degree of which nodes stand between each other. For example, in a telecommunications network, a node with higher betweenness centrality would have more control over the network, because more information will pass through that node. Betweenness centrality was devised as a general measure of centrality: it applies to a wide range of problems in network theory, including problems related to social networks, biology, transport and scientific cooperation. The betweenness centrality of a node v is given by the expression:

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass through v.

1.3 Transmission in Networks

The transmission of a vertex $u\sigma V(G)$, also called farness or vertex Wiener value in the literature, is defined as the sum of the lengths of all shortest paths between a chosen vertex and all other vertices in G [13,14,15]. Using transmission, we can define a well known topological index of a graph G called the Weiner index $W(G)$, introduced by Wiener. This graph index is defined for a connected graph G as the sum of the lengths of shortest paths between all unordered pairs of vertices in G. It is the oldest topological index related to molecular branching and based on its success, many other topological indices correlating to distance matrix of chemical graphs have been developed subsequently to Wiener's work. Wiener index was at first used for predicting the boiling points of paraffins, but later a strong correlation between Wiener index and other chemical or physical properties of a compound was found, such as critical points in general, the density, surface tension, and viscosity of compounds liquid phase and the van der Waals surface area of the molecule. In theoretical computer science, Wiener index is known as transmission index and is considered as one of the basic descriptors of fixed interconnection networks because it provides the average distance between any two nodes of the network. The transmission $T(u)$ of a vertex $u \in V(G)$ is a concept closely related to the Wiener index but localized to the selected vertex.

Definition 1.1 For a vertex u in G, a u-vertex routing R_u in G is a set of $n1$ paths between u and v, where n is the number of vertices and v belongs to V(G). The congestion or load on an edge e in a given routing R_u of G is the number of paths of R_u which go through it, and is denoted by $\phi(G, R_u, e)$ or simply as $\phi(R_u, e)$. A u-vertex routing R_u in G is a minimum routing if it is comprised of shortest paths from u to every other vertex of G.

Definition 1.2 [2] Let $u \in V(G)$ and $S = \{S_1, S_2, \ldots, S_k\}$ be a partition of $E(G)$ such that each S_i is an edge cut of G for $1 \leq i \leq k$. Then S is said to be transmission partition rooted at u if for any given minimum routing R_u, every member P of R_u passes through at most one edge of each S_i, $1 \leq i \leq k$.

Theorem 1.1 [9] (Transmission Lemma) Let G be a graph on n vertices and let $u \in V(G)$. Let $\{S_1, S_2, \ldots, S_m\}$ be a transmission partition of $E(G)$ rooted at u, such that G_i is the component of $G \backslash S_i$ which does not contain u, $1 \leq i \leq m$. Then $d_G(u) = \sum_{i=1}^{m} |V(G_i)|$.

We use the notation $[2(E(G))]$ to represent the collection of all the edge is of G such that each edges repeated exactly twice.

Theorem 1.2 (2-Transmission Lemma) Given $u \in V(G)$, let $S = \{S_1, S_2, \ldots, S_m\}$ be a transmission partition of $[2E(G)]$ rooted at u such that G_i is the component of $G \backslash S_i$ which does not contain u. Then $T(u) = \frac{1}{2}\sum_{(i=1)}^{m} |V(G_i)|$.

2 Neural Networks

In machine learning, convolutional neural networks (CNN, or ConvNet) are artificial neural networks that have successfully been applied to outperform conventional methods in modeling the sequence specificity of DNAprotein [3], analyzing visual imagery, analyzing handwritten numeral [4]. Artificial neural networks are statistical learning models, inspired by biological neural networks such as the brain, that are used in machine learning. These networks are represented as systems of interconnected neurons, which send messages to each other [5]. The basic idea behind a neural network is to simulate lots of densely interconnected brain cells inside a computer to learn things, recognize patterns, and make decisions in a humanlike way.

CNNs use a variation of multilayer perceptrons designed to require minimal pre-processing. In between the input units and output units are one or more layers of hidden units, which, together, form the majority of the artificial brain. Most neural networks are fully connected, which means each hidden unit and each output unit is connected to every unit in the layers either side. See Figure 1.

Steganography, the art of hiding information inside host media like pictures and movies, and steganalysis, its countermeasure attempting to detect the presence of a hidden information within an innocent-looking document, are information security techniques for telemedicine. A key knowledge of image steganalyzer can be produced by convolutional neural networks (CNN) [6].

A typical neural network has anything from a few dozen to hundreds, thousands, or even millions of artificial neurons called units arranged in a series of layers, each of which connects to the layers on either side. Some of them, known as input units, are designed to receive various forms of information from the outside world that the network will attempt to learn

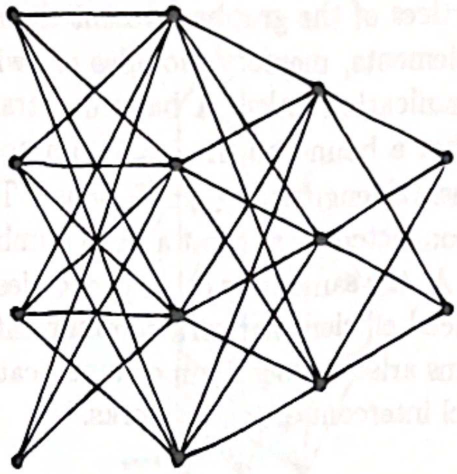

Figure 1: Neural Network

about, recognize, or otherwise process. Other units sit on the opposite side of the network and signal how it responds to the information it has learned; those are known as output units. In between the input units and output units are one or more layers of hidden units, which, together, form the majority of the artificial brain. Most neural networks are fully connected, which means each hidden unit and each output unit is connected to every unit in the layers either side. All in all, neural networks have made computer systems more useful by making them more human. [7]. Eric Goles et al. [8] study how convolutional neural networks can be used to bear on a large class of graph-based learning problems.

3 Interconnection Networks

An interconnection network consists of hardware and software entities that are interconnected to facilitate efficient computation and communication. These entities can be in the form of processors, processes, memory modules or computer systems. Due to the recent developments in parallel and distributed computing, the design and analysis of various interconnection networks has been a main topic of research for the past few years. An essential component of a supercomputer based on large-scale parallel processing is the interconnection network. It provides communication among the processors and memories. Interconnection networks also play a key role in the design and implementation of communication networks and the recent advent of optical communication technology adds more design problems. This explains a growing number of research articles and congresses devoted to interconnection networks as also special issues of journals being published on the subject [10, 11]. Interconnection networks are often modeled by finite graphs. The vertices of the graph represent the nodes of the network, that is, processing elements, memory modules or switches, and the edges correspond to communication links. A basic constraint in many network design problems is that a bound on the maximum node degree is imposed by cost and fundamental engineering limitations. That is, the nodes of the network can be connected by at most a fixed number of communication lines to other nodes. At the same time other properties are crucial for many applications which need efficient network communications. Combinatorial isoperimetric problems arise frequently in communications engineering and in the field of parallel interconnection networks.

4 Transmission in Neural Nerworks

We formally define an architecture of a convolution neural network N. It consists of k levels of vertices, k odd, with exactly one vertex v at level 1, n_i vertices at level , $2 \leq i \leq k$, such that v is adjacent to all the n_2 vertices in level 2; number of vertices in levels i and $i+1$, i even, are equal and induce a perfect matching; the $n_{(i-1)}$ vertices at level $i-1$ and the n_i vertices at level i induce a complete bipartite graph, $2 \leq i \leq k-2$. The vertices at level $k-2$ and $k-1$ induce a complete bipartite graph. Finally the vertices at level $k-1$ and k also induce a complete bipartite graph. We denote this architecture by, $N\{k; n_1, n_2, \ldots, n_k\}$, where $n_1 = 1$

Let S_i be the set of edges with one end in level i and the other end in level $(i+1)$, $1 \leq i \leq k-1$. Each S_i is an edge cut; the end vertices of the edges in S_i induce a complete bipartite graph when i is odd and the end vertices of edges in S_i induce a perfect matching when i is even, $1 \leq i \leq k-1$. Let t_j be the number of vertices in the component of $G \backslash S_j$ which does not contain u, $1 \leq j \leq k-1$.

Theorem 1.3 Let G be the neural network $N\{k; n_1, n_2, \ldots, n_k\}$, where $n_1 = 1$. For $u \in V(G)$ in level i, let $T_i(u)$ denote the transmission of u in G, $1 \leq i \leq k$. Then for $1 \leq i \leq k$,

$$T_i(u) = \begin{cases} \sum_{j=1}^{k-1} t_j, & i = 1 \\ \sum_{j=1}^{k-1} t_j + 2(n_i - 1) + 2(n_{i-1} - 1), & i \text{ } odd, i < k-1 \\ \sum_{j=1}^{k-1} t_j + 2(n_i - 1) + 2(n_{i+1} - 1), & i \text{ } even, i < k-1 \\ \sum_{j=1}^{k-1} t_j + 2(n_i - 1), & i = k-1 \text{ } or \text{ } k. \end{cases}$$

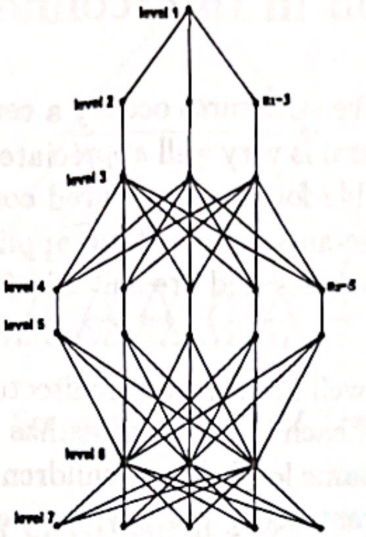

Figure 2: A Convolution Neural Network

Proof: Let $u \in V(G)$. Suppose u is in level i, i odd, $i \neq 1$. Label all the vertices other than u in the same level as 2 and all the vertices in the preceeding level $(i-1)$ except the vertex adjacent to u as 2. Then $T_i(u) = \sum_{(j=1)}^{(k-1)} t_j + 2(n_i - 1) + 2(n_{(i-1)} - 1)$. Suppose u is in level i, i even. Then label all the vertices in level i except u as 2, and all the vertices in level $(i+1)$ except the vertex adjacent to u as 2. Then $T_i(u) = \sum_{(j=1)}^{(k-1)} t_j + 2(n_i - 1) + 2(n_{(i+1)} - 1)$.

Corollary: The closeness of a vertex u in G is obtained by taking the reciprocal of $T_i(u)$. See Figure 3.

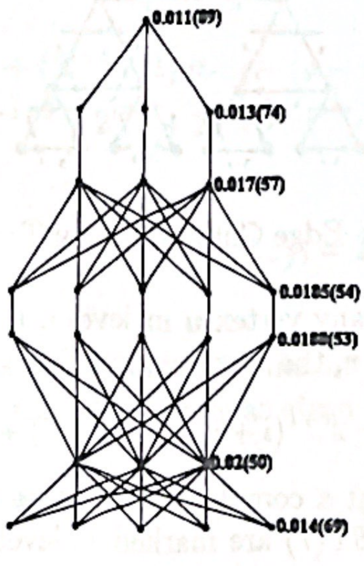

Figure 3: Closeness of vertices in $N(7;1,3,3,5,5,3,5)$ with transmission in parenthesis

5 Transmission in Interconnection Networks

In network world, tree-like structures occupy a central place, and the value of tree structures in general is very well appreciated [12]. The tree interconnection network is suitable for tree structured computations (multi-input, single-output) and divide-and-conquer type applications. Tree-based networks have fixed degree nodes and are suitable for massively parallel systems.

The binary tree is a well known tree architecture. A binary tree is said to be a full binary tree if each internal vertex has exactly two children and all the leaves are at the same level. These children are described as left and right children of the parent node. For any non-negative integer r, the full binary tree of height r, denoted by T_r has r levels and level i, $0 \leq i \leq r$, contains 2^i vertices. Thus T_r has exactly $2^{(r+1)} - 1$ vertices. In the sequel we compute the transmission for each of the vertices in a sibling tree.

6 Sibling Tree Networks

Definition 1.3 The r-dimensional sibling tree $ST(r)$, is obtained from the complete binary tree of height r, by adding edges called sibling edges between left and right children of the same parent node of $ST(r)$. We define horizontal and oblique cuts as shown in Figure 4.

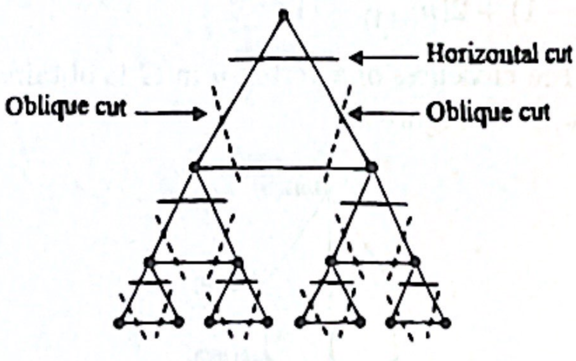

Figure 4: Edge Cuts of Sibling Tree $ST(3)$

Theorem 1.4. For any vertex u in level i, $0 \leq i \leq r$, of the sibling tree $ST(r)$ of dimension r, the transmission $Ti(u)$ of u is given by

$$T_i(u) = 2^{r+1}(i + r - 4 + 3.2^{-i}) + 2i + 2$$

Proof: Let T_r represent a complete binary tree of height r, $r \geq 1$. The levels of the vertices of $ST(r)$ are marked as level 0, level 1,..., level r as shown in Figure 5. Let $n_i = |V(T_i)|$, $1 \leq i \leq r$. Then $n_i = 2^{i+1} - 1$, $1 \leq i \leq r$. Let u be a vertex of $ST(r)$ at level i, $0 \leq i \leq r$. By 2-dimension lemma, we obtain the following expression for the transmission of u.

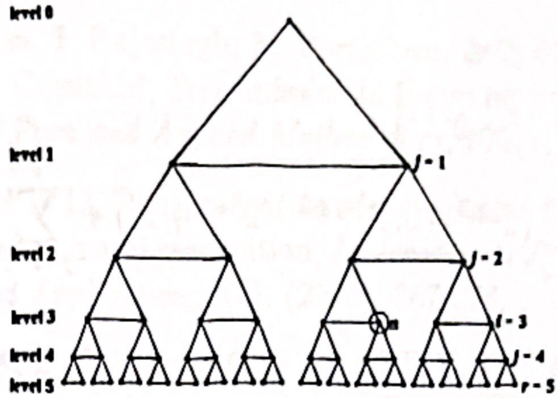

Figure 5: Steps in the proof of Theorem 1.4

Let n be the number of vertices in STr, $r \geq 1$. Then $n = 2^{r+1} - 1$. For $1 \leq i \leq r$, there are 2^{r-i+1} oblique cuts which divide the graph G into 2 components G_1 and G_2 with $|G_1| = 2^{i-1}$ and $|V(G_2)| = 2^{r+1} - 2^i$. Further for $1 \leq i \leq r$, there are 2^{r-i} horizontal cuts which divide the graph into 2 components G_1 and G_2 with $|V(G_1)| = 2(2^i - 1)$ and $|V(G_2)| = 2^{r+1} - 2^{i+1} + 1$. Let X_j be the contribution to $T_i(u)$ by the edge cut S_j comprising of edges between the levels (j-1) and j, $1 \leq j \leq r$.

Therefore for $j = 1$,

$$x_1 = (n - 2n_{r-1}) + (n - n_{r-1}) + n_{r-1} = 2^{r+2} - 2^{r+1}$$

For $1 < j < i$,

$$\begin{aligned} x_j &= (n - 2n_{r-j}) + (2^{j-1} - 1)\, 2n_{r-j} + (n - n_{r-j}) + (2^j - 1)\, n_{r-j} \\ &= 2^{r+3} - 3.2^{r-j+2} - 2^{j+1} + 2^2 \end{aligned}$$

For $j = i$,

$$\begin{aligned} x_i &= (n - 2n_{r-i}) + (2^{i-1} - 1)\, n_{r-i} + (n - n_{r-i}) + (2^i - 1)\, n_{r-i} \\ &= 2^{r+3} - 3.2^{r-i+2} - 2^{i+1} + 2^2 \end{aligned}$$

For $i < j \leq r$,

$$x_j = 2^{j-1}(2n_{r-j}) + 2^j(n_{r-j}) = 2^{r+2} - 2^{j+1}$$

Thus by 2-Transmission lemma, when u lies in level i, $0 \leq i \leq r$, the contribution x_j to $T_i(u)$ for $1 \leq j \leq r$, is as given below:

$$x_j = \begin{cases} 2^{r+2} - 2^{r+1} & ; \quad j = 1 \\ 2^{r+3} - 3.2^{r-j+2} - 2^{j+1} + 2^2 & ; \quad 1 < j \leq i \\ 2^{r+2} - 2^{j+1} & ; \quad i < j \leq r, \end{cases}$$

Therefore, for $1 \leq i \leq r$,

$$\begin{aligned} 2T_i(u) &= (2^{r+2} - 2^{r+1}) \\ &+ \sum_{j=2}^{i} (2^{r+3} - 3.2^{r-j+2} - 2^{j+1} + 2^2) + \sum_{j=i+1}^{r} (2^{r+2} - 2^{j+1}) \end{aligned}$$

and hence,

$$\begin{aligned} T_i(u) &= (2^{r+1} - 2^r) + \sum_{j=2}^{i} (2^{r+2} - 3.2^{r-j+1} - 2^j + 2) + \sum_{j=i+1}^{r} (2^{r+1} - 2^j) \\ &= 2^{r+1}\left(i + r - 4 + 3.2^{-i}\right) + 2i + 2 \end{aligned}$$

Remark: By symmetry of $ST(r)$, the transmission of any vertex in a level is the same. Further the closeness of vertex u in level i is obtained by taking the reciprocal of $T_i(u)$, $0 \leq i \leq r$. See Figure 6.

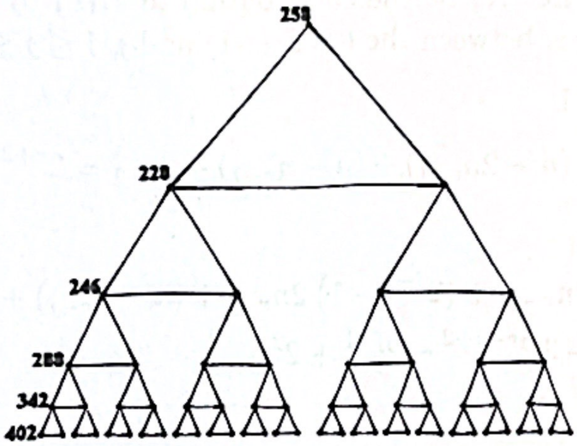

Figure 6: Transmission of vertices in $ST(5)$ with equal values in every level

7 Conclusion

Closeness of vertices in a class of neural networks and in a family of interconnection networks have been computed. It would be an interesting line of research to explore these centrality measures in social networks.

References

[1] S.P. Borgatti, Centrality and network flow, *Elsevier Journal Social Networks*, 27, (2005), 55-71.

[2] D.A. Jemilet, I. Rajasingh, N. Parthiban, A.G. Karnam, R. Prabhat and V. Gayathiri, Transmission in Benes network, *International Journal of Pure and Applied Mathematics*, 109(8), (2016), 153-159.

[3] X. Han and Y. Li, The application of convolution neural networks in handwritten numeral recognition, *International Journal of Database Theory and Application*, 8(3), (2015), 367-376.

[4] H. Zeng, M.D. Edwards, G. Liu and D.K. Gifford, Convolutional neural network architectures for predicting DNA protein binding, *Bioinformatics*, 32(12), (2016), 121-127.

[5] M. Salomon, R. Couturier, C. Guyeux, J.-F. Couchot and J.M. Bahi, Steganalysis via a convolutional neural network using large convolution filters for embedding process with same stego key: A deep learning approach for telemedicine, *European Research in Telemedicine*, 6(2), (2017), 79-92.

[6] C. Woodford, Neural networks, Woodford, *Neural networks*. Retrieved from http://www.explainthatstuff.com/introduction-to-neural-networks (2017).

[7] https://stevenmiller888.github.io/mind-how-to-build-a-neural-network/ Mind: How to Build a Neural Network (2015).

[8] E. Goles, G.A. Ruz, Dynamics of neural networks over undirected graphs, *Neural Networks*, 63, (2015), 156-169.

[9] I. Rajasingh, P. Manuel, N. Parthiban, D. A. Jemilet and R. S. Rajan, Transmission in Butterfly networks, Section A: Computer Science Theory, Methods and Tools, *The Computer Journal*, 59(8), (2016), 1174-1179.

[10] D.F. Hsu, Introduction to a special issue on interconnection networks, *Networks*, 23, (1993), 211-213.

[11] Y. L. Lai and K. Williams, A survey of solved problems and applications on bandwidth, edgesum, and profile of graphs, *Journal of Graph Theory*, 31, (1999), 75-94.

[12] J.C. Bermond, F. Comellas and D.F. Hsu, Distributed loop computer networks, *A survey: Journal of Parallel and Distributed Computing*, 24(1), (1995), 2 - 10.

[13] A. Abiad, B. Brimkov, A. Erey, L. Leshock, X.M.-Rivera and S.-Y. Song, J. Williford, On the Wiener index, distance cospectrality and transmission-regular graphs, *Discrete Applied Mathematics*, 230, (2017), 1-10.

[14] M. Krnc and R. Skrekovski, Centralization of transmission in networks, *Discrete Mathematics*, 338, (2015), 2412-2420.

[15] L. Soltes, Transmission in graphs: A bound and vertex removing, *Mathematica Slovaca*, 41(1), (1991), 11-16.