

Poincaré Embeddings for Visualizing Eigenvector Centrality

by

Alena Chang

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2020 by the
Graduate Supervisory Committee:

Guoliang Xue, Chair
Dejun Yang
Yezhou Yang

ARIZONA STATE UNIVERSITY

August 2020

ABSTRACT

Hyperbolic geometry, which is a geometry which concerns itself with hyperbolic space, has caught the eye of certain circles in the machine learning community as of late. Lauded for its ability to encapsulate strong clustering as well as latent hierarchies in complex and social networks, hyperbolic geometry has proven itself to be an enduring presence in the network science community throughout the 2010s, with no signs of fading into obscurity anytime soon. Hyperbolic embeddings, which map a given graph to hyperbolic space, have particularly proven to be a powerful and dynamic tool for studying complex networks. Hyperbolic embeddings are exploited in this thesis to illustrate centrality in a graph. In network science, centrality quantifies the influence of individual nodes in a graph. Eigenvector centrality is one type of such measure, and assigns an influence weight to each node in a graph by solving for an eigenvector equation. A procedure is defined to embed a given network in a model of hyperbolic space, known as the Poincaré disk, according to the influence weights computed by three eigenvector centrality measures: the PageRank algorithm, the Hyperlink-Induced Topic Search (HITS) algorithm, and the Pinski-Narin algorithm. The resulting embeddings are shown to accurately and meaningfully reflect each node's influence and proximity to influential nodes.

TABLE OF CONTENTS

	Page
LIST OF TABLES	iii
LIST OF FIGURES	iv
CHAPTER	
1 INTRODUCTION	1
1.1 Outline of Problem	1
1.2 Aims and Objectives	3
2 BACKGROUND	5
2.1 PageRank Algorithm	5
2.2 HITS Algorithm	11
2.3 Pinski-Narin Algorithm	19
2.4 Hyperbolic Geometry	26
2.5 Hyperbolic Embeddings	33
3 IMPLEMENTATION	39
4 RESULTS AND OBSERVATIONS	51
5 CONCLUSION	73
REFERENCES	75

LIST OF TABLES

Table	Page
2.1 Characteristic Properties of Euclidean, Spherical, and Hyperbolic Geometries	29
4.1 PageRank Weights After Iteration $t = 1$	58
4.2 HITS Authority Weights After Iteration $t = 1$	61
4.3 HITS Hub Weights After Iteration $t = 1$	63
4.4 Pinski-Narin Weights After Iteration $t = 1$	66
4.5 Five Most Influential Nodes Identified by PageRank, HITS, and Pinski-Narin	67

LIST OF FIGURES

Figure	Page
2.1 PageRank Transfer of Influence in a Backlink Chain	7
2.2 A Set of Straight Lines in the Poincaré Disk that All Pass Through a Given Point and are All Parallel to the Blue Line.	28
2.3 A Hyperbolic Tiling in the Poincaré Disk Model.	33
2.4 An Embedding of a Regular Tree in the Poincaré Disk so that All Connected Nodes are Spaced Equally Far Apart	36
3.1 Graph of Radial Coordinate Function $f(x) = e^{-x}$	41
4.1 Graph Used in Experimentation.	51
4.2 Initial PageRank Poincaré Embedding	54
4.3 Initial HITS Authority Weight Poincaré Embedding	55
4.4 Initial HITS Hub Weight Poincaré Embedding	55
4.5 Initial Pinski-Narin Poincaré Embedding	56
4.6 PageRank Poincaré Embedding After Iteration $t = 1$	57
4.7 HITS Authority Weight Poincaré Embedding After Iteration $t = 1$	60
4.8 HITS Hub Weight Poincaré Embedding After Iteration $t = 1$	62
4.9 Pinski-Narin Poincaré Embedding After Iteration $t = 1$	65
4.10 PageRank Poincaré Embedding After Iteration $t = 2$	68
4.11 PageRank Poincaré Embedding After Iteration $t = 3$	68
4.12 PageRank Poincaré Embedding After Iteration $t = 4$	69
4.13 PageRank Poincaré Embedding After Iteration $t = 5$	69
4.14 PageRank Poincaré Embedding After Iteration $t = 6$	70
4.15 PageRank Poincaré Embedding After Iteration $t = 7$	70
4.16 PageRank Poincaré Embedding After Iteration $t = 8$	71
4.17 Pinski-Narin Poincaré Embedding After Iteration $t = 28$	71

Figure	Page
4.18 HITS Authority Weight Poincaré Embedding After Iteration $t = 17$...	72
4.19 HITS Hub Weight Poincaré Embedding After Iteration $t = 17$	72

Chapter 1

INTRODUCTION

1.1 Outline of Problem

Complex networks encapsulate many of the intricate relationships that characterize the natural world. In particular, an elegant and mathematically rigorous procedure to quantify individual node influence in complex networks has long been the subject of study. The multitude of attempts at formalizing such a system throughout the twentieth century like Pinski and Narin (1976) and Page *et al.* (1998) demonstrates the endless utility of a computationally sound ranking scheme. Dubbed centrality measures, these measures quantify the importance of a node in a complex network in terms of traits involving the network as a whole, like connectedness or number of shortest paths.

This thesis is concerned with one type of centrality measure: eigenvector centrality. Eigenvector centrality quantifies a node's influence based on the number of links the node has to other nodes in the network, while taking into account the connectedness of the node, the connectedness of said node's neighbors, and so on. Computing the eigenvector centrality for a given network entails solving some eigenvector equation, but algorithms to compute eigenvector centrality are typically iterative in practice. As the number of iterations increases, because a complex network can be viewed as a closed system, the influence of a given node based on its neighbors' collective influence eventually "stabilizes", and the centrality score for each node converges to its respective value, similar to the way a Markov chain converges to a steady state distribution (Durrett (2012)).

As of late, hyperbolic geometry has become somewhat of a hot button topic among certain circles in the computer science research community. It has been applied to a myriad of fields like link prediction, most notably in Kitsak *et al.* (2019), whose work has been expanded to include multiplex networks (multilayer networks with the same set of nodes in each layer) in Samei and Jalili (2019). Hyperbolic geometry has even been extended to neural networks, as shown by Ganea *et al.* (2018) at NeurIPS. Chami *et al.* (2019) and Liu *et al.* (2019) further stoked the fire by continuing the aforementioned work on hyperbolic neural networks at NeurIPS the following year. It is apparent more and more computer scientists are finding useful applications for hyperbolic geometry, particularly via hyperbolic embeddings (Chamberlain *et al.* (2017), Nickel and Kiela (2017)), with no signs of slowing down.

Hyperbolic embeddings entail embedding, or mapping, a given complex or social network in hyperbolic space, wherein the postulates of hyperbolic geometry are adhered to instead of those of Euclidean geometry. Doing so can not only preserve intrinsic properties like distance, but also elucidate latent features like hierarchies and cliques. It is no wonder hyperbolic embeddings have become a subject of intrigue in the machine learning community, as they bear the potential to illuminate idiosyncrasies of complex and social networks while respecting the integrity of the original network, as opposed to embeddings in traditional Euclidean space.

In a hyperbolic embedding, each node in a given network is assigned coordinates to be mapped to in the specified hyperbolic space. The challenge in performing hyperbolic embeddings lies in producing a mapping for each node in the network such that the (hyperbolic) distance between any given pair of nodes accurately reflects their relationship in the greater context of the actual network itself.

This thesis exclusively focuses on three existing eigenvector centrality measures: PageRank (Page *et al.* (1998)), the HITS algorithm (Kleinberg (1998)), and the

Pinski-Narin algorithm (Pinski and Narin (1976)). Each of these algorithms differs in how they initialize and compute the nodes' influence weights. As such, if the computation of a complex network's influence weight eigenvector were to be represented in terms of a hyperbolic embedding, the computation of the nodes' weights until convergence should (ideally) manifest differently in hyperbolic space for each algorithm. However, there is a noticeable lack of data visualization techniques for eigenvector centrality measures in the realm of hyperbolic embeddings. This is an issue that can be ameliorated. The problem this thesis aims to address is as follows: Given a complex network, how can each node in the network be embedded in hyperbolic space according to its eigenvector centrality score in a way that meaningfully reflects both its respective influence and its relationship with influential nodes in the broader network as a whole?

1.2 Aims and Objectives

This thesis marries eigenvector centrality measures with hyperbolic embeddings by devising a mathematically rigorous and sound mapping of a complex network to hyperbolic space based on the influence weight output computed by PageRank, HITS, and Pinski-Narin. This endeavor is ultimately a data visualization task using hyperbolic geometry.

Recall that algorithms to calculate eigenvector centrality are usually iterative, and the three algorithms this thesis is concerned with are no exception. Each iteration of each algorithm until convergence is accompanied by an appropriate hyperbolic embedding. The objective is to devise a procedure to map each node in a given network to hyperbolic space for a given iteration so that the hyperbolic distance between any pair of nodes accurately reflects each node's influence in terms of the

entire network at that moment. The goal is to, hopefully, offer a novel and fresh way to visualize a number of (relatively) old algorithms.

Chapter 2

BACKGROUND

This chapter covers the background knowledge necessary to comprehend the proposed hyperbolic embedding scheme in the following section. First the three eigenvector centrality algorithms to be studied in this thesis are discussed: PageRank, HITS, and Pinski-Narin. Then the reader is familiarized with hyperbolic geometry with a crash course in the topic. Finally, the reader's background in hyperbolic geometry is extended to the concept of hyperbolic embeddings in the context of complex and social network analysis.

2.1 PageRank Algorithm

Probably the most well-known eigenvector centrality measure, PageRank was published in 1998 by Larry Page, Sergey Brin, Rajeev Motwani, and Terry Winograd (Page *et al.* (1998)). No explanation is necessary to convey PageRank's significance and long-term impact on network science. Only a discussion of the set-up and implementation of PageRank is needed for this thesis.

The authors acknowledge the large body of literature on academic citation analysis at the time, but point out that measuring the influence of a web page in the World Wide Web poses a different set of challenges compared to measuring the influence of a publication in a citation network. For one, academic papers are painstakingly reviewed before being published, while web pages are completely unregulated in terms of quality control. This discrepancy means the reader is reasonably rest assured of the quality of an academic publication with many citations, while the reader has no systematic way of verifying whether a web page with many citations is a reliable

source. One may easily generate astronomical numbers of web pages to artificially inflate citation counts. On the other hand, researchers in select circles who purposely cite each other's papers in disproportionate numbers (sometimes known as citation cartels) are more easily detected, as academic publications are peer-reviewed. Secondly, the features by which the influence of web pages is measured must not be replicable, or else profit seeking ventures can easily exploit said features to increase their citation numbers. For example, if one of the criteria for an influential web page is number of clicks, a greedy individual can simply produce a large number of web pages inundated with attention-grabbing features to attract hits, although the pages themselves may have very little informative value.

Furthermore, academic papers are generally consistent in quality and their intent: to widen the knowledge of their respective field. A paper in a biochemistry journal is different from a paper in an astrophysics journal, but both are nonetheless held in high regard. In contrast, web pages vary widely in quality, usage, citations, and length. An obscure user question on a niche forum about an IBM computer should be treated differently than the IBM home page. Most web pages, in fact, are of poor quality and receive very few citations, with a few select pages dominating in terms of influence for a given search term.

Like their predecessors Pinski and Narin, the authors of PageRank realize the importance of not only including citation count in quantifying influence, but also the influence of the actual citations. The authors realize that a large number of citations for a web page does not necessarily indicate said page is highly influential, which can be attributed to factors which are not present in academic citation databases. They claim PageRank is capable of distinguishing a citation from a highly influential page from a number of citations from insignificant pages via propagation of influence through links. That is, a page has high rank if the sum of the ranks of its citations,

or backlinks, is high. In turn, if another page has a backlink from the aforementioned page, then its own rank is appropriately rewarded. A consecutive series of backlinks can thus be viewed as a domino effect in terms of transfer of influence, as long as the backmost page in the backlink chain has high rank. This concept is illustrated in Figure 2.1 below.

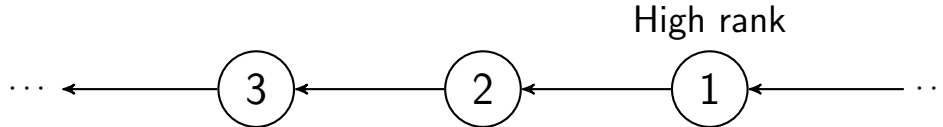


Figure 2.1: PageRank Transfer of Influence in a Backlink Chain

If node 1 above has high rank, then node 2’s rank in turn increases accordingly because of node 2’s backlink from node 1, and likewise node 3’s rank receives a boost from node 3’s backlink from node 2. Alternatively, a node may have high rank by having many backlinks in general. PageRank allows a page to have high rank if it has a few highly ranked backlinks and also if it has many backlinks.

The original PageRank paper offers two versions of the algorithm: a simplified version and the full version. Both are defined, starting with the simplified version. If u is a web page, define F_u as the set of pages u points to, and B_u as the set of pages that point to u . Then define $N_u = |F_u|$ as the number of links from u as well as a normalization factor c (the significance of c is elaborated on shortly). The rank of node u is defined as

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}.$$

So the rank of node u , $R(u)$, is the sum of the ranks of all of its backlinks, with each rank divided by the number of outgoing links corresponding to the node of said rank, and with the entire sum normalized by the factor c . A page’s rank is divided by its number of forward links so its rank is evenly distributed among all of its forward links. So if a page u links to four other pages, $R(u)$ is divided into quarters

when computing the rank of each of u 's outgoing neighbors: $R(u)/4$. Note that the normalization factor $c < 1$ because there may exist pages with no forward links, and so their weight is lost from the system. Define dangling links, which are links that point to a page with no outgoing links. A page with no outgoing links is known as a sink: once a web surfer reaches a sink, the surfer has no way of leaving the page via link. Dangling links are problematic in the aforementioned model because it is unclear where or how their weight should be distributed. In other words, the rank of a sink gets stuck in a vacuum, and cannot be transferred to other nodes in the system. To remedy this issue, and since dangling links do not directly affect other pages' ranks, the authors simply remove any dangling links from the system when applying PageRank, and then restore them after all of the PageRanks are calculated.

Before explaining the role of the normalization factor c , better known as the damping factor, the random surfer model must be introduced. The intuition behind PageRank is based on a random walk on the World Wide Web. The actor in this random walk is dubbed a "random surfer". The random surfer in the model explores the World Wide Web by clicking on successive links at random, and theoretically eventually stops clicking. The damping factor can be interpreted as the probability that the random surfer continues clicking, which is now often assumed to be 0.85, which is also the value used for c in the implementation of PageRank in this thesis. Multiplying the sum of a node u 's backlink ranks divided by their respective number of forward links by c in $R(u)$ above somewhat lessens any adverse effects of the existence of dangling links.

The simplified version of PageRank can be modeled as an eigenvector equation. If the given network consists of n nodes, define the $n \times n$ square matrix A , where entry

$$A_{u,v} = \begin{cases} 1/N_u & \text{if there is an edge from } u \text{ to } v \\ 0 & \text{else} \end{cases}$$

is defined. The value $1/N_u$ can be interpreted as the probability the random surfer jumps to page v if they are at page u and randomly choose one of u 's outgoing links. A is a row-stochastic matrix, as each row sums to one. Define the $n \times 1$ column vector R , where each entry of R corresponds to the rank of a node (which was defined earlier). Then the equation $R = cAR$ can be formulated. It is seen that R is an eigenvector of A with eigenvalue c . The dominant eigenvalue of A is wanted, and this value can be found by repeatedly multiplying A by any non-degenerate start vector R . The primary shortcoming of the simplified version of PageRank is that it cannot account for the case where two web pages point to each other but to no other page. If there is a page which points to one of these two pages, this loop accumulates rank but never distributes any rank throughout iterations of simplified PageRank. This loop ends up forming a trap which is dubbed a rank sink. This downfall segues over to the full version of PageRank, which overcomes the problem of rank sinks.

The full version of PageRank is defined as follows. Let E be some $n \times 1$ column vector over the web pages which corresponds to a source of rank. Then the PageRank of a set of pages is an assignment, R' , to the pages where

$$R'(u) = c \sum_{v \in B_u} \frac{R'(v)}{N_v} + cE(u)$$

such that c is maximized and $\|R'\|_1 = 1$, where $\|R'\|_1 = \sum_u |R'(u)|$ is the L_1 norm of R' . The latter condition means the ranks of the pages can be viewed collectively as a probability distribution: the rank of a page also denotes the probability the random surfer clicks on said page. The authors point out that if E is all positive, c must be reduced to balance the aforementioned equation. This technique corresponds to

a decay factor. In matrix notation, the expression above can be written as $R' = c(AR' + E)$, or alternatively $R' = c(A + E \times \mathbf{1})R'$ where $\mathbf{1}$ is the $1 \times n$ row vector consisting of all ones, since $\|R'\|_1 = 1$. To be specific, $E \times \mathbf{1}$ is simply an $n \times n$ matrix where all of the entries in the row corresponding with some node u are $E(u)$. Because $\|R'\|_1 = 1$, multiplying $E \times \mathbf{1}$ by R' is simply equivalent to E . This equation is an eigenvector equation, as R' is an eigenvector of $(A + E \times \mathbf{1})$ with eigenvalue c . Recall the random surfer model. If the case occurs where the surfer gets stuck in a loop of web pages, the behavior where the surfer randomly jumps to some other page to escape the loop is modeled via the additional factor E in the full version of PageRank. The purpose of the vector E is to enable the random surfer to jump to a random page chosen based on the distribution in E whenever the surfer gets “bored”. Note that there is a slight discrepancy between the original definition of the full version of PageRank and implementations of PageRank in practice, as seen shortly.

The full version of PageRank as an eigenvector equation was just defined, but only the iterative procedure was implemented for this thesis, which differs from the formal definition above. The iterative procedure to compute PageRank is as follows. Given a graph of n nodes, with the nodes enumerated from 1 to n , at iteration $t = 0$, assign the graph an initial probability distribution of $\mathbf{R}(0) = [\frac{1}{n}, \dots, \frac{1}{n}]^T$, where \mathbf{R} is an $n \times 1$ column vector denoting the influence weight of every node in the graph, with $R_i(t)$ representing the influence weight of node i after iteration t , or just the i th entry of $\mathbf{R}(t)$. For following iterations $t \geq 1$, the PageRank of a page i is computed using the formula

$$R_i(t) = c \sum_{j \in B_i} \frac{R_j(t-1)}{N_j} + \frac{1-c}{n}, \quad 1 \leq i \leq n$$

where c is the damping factor, B_i is the set of pages that point to node i , and N_j is the number of outgoing links from node j . In matrix notation, this procedure is

$$\mathbf{R}(t) = cM\mathbf{R}(t-1) + \frac{1-c}{n}\mathbf{1}$$

where $\mathbf{1}$ is the $n \times 1$ column vector consisting of all ones and M is the $n \times n$ square matrix where

$$M_{ij} = \begin{cases} 1/N_j & \text{if there is an edge from } j \text{ to } i \\ 0 & \text{else} \end{cases}$$

is the i th row, j th column entry of M , $1 \leq i, j \leq n$. Another way to define M is $M = (K^{-1}A)^T$, where A is the adjacency matrix of the graph and K is the diagonal matrix with the outdegrees in the diagonal. The definition of A here may be confusing, as A in this context is the adjacency matrix of the graph, whereas in the prior eigenvector equation defined for the full version of PageRank, A is equivalent to the transpose of M itself, M^T . It may be beneficial to the reader to make this clarification. \mathbf{R} was continually computed until

$$|\mathbf{R}(t) - \mathbf{R}(t-1)| < \epsilon$$

where $\epsilon > 0$ is arbitrary but fixed. This concludes this section on PageRank.

2.2 HITS Algorithm

Published in the same time frame as PageRank (with PageRank being published shortly after HITS in 1998), Jon Kleinberg's Hyperlink-Induced Topic Search algorithm, better known by the acronym HITS, also defines a systematic method of quantifying web page influence in a hyperlinked environment (Kleinberg (1998)). The intent of the paper is to construct a focused subgraph of the World Wide Web with respect to a broad search topic. In other words, the objective is to distill the vast expanses of the World Wide Web into a concise network of pages deemed relevant to a given search query in terms of their authority. The entirety of this endeavor is predicated on a

user input query. Kleinberg delineates different categories of queries: specific queries (e.g., “Does Netscape support the JDK 1.1 code-signing API?”), broad-topic queries (e.g., “Find information about the Java programming language”), and similar-page queries (e.g., “Find pages ‘similar’ to java.sun.com”). The paper discusses the first two query types comparatively, only to hone in on the second type for the proposed algorithm: broad-topic queries. The challenge in identifying relevant pages for specific queries lies in the fact that there exist very few pages that hold the required information, and it is often difficult to actually find these pages. On the other hand, identifying relevant pages for broad-topic queries poses the opposite problem: there are simply too many candidate pages for a human user to reasonably intake at once. In this scenario, the most “authoritative” pages must be filtered from the collection of relevant pages for a given broad-topic query.

Kleinberg strives to attach a specific metric to the concept of authority in a hyperlinked environment. A hyperlink from one page to another ideally represents an endorsement from one page to another. This endorsement, according to Kleinberg, involves some degree of latent human judgment, which is what the notion of authority is formulated on. When the creator of a web page p includes a link to page q , the creator has in some measure conferred authority on q via their “endorsement”. This logic means links enable the identification of authoritative pages through the pages that point to them. Some potential downsides include the fact that links do not necessarily indicate conferral of authority. For example, a link can exist for purely navigational purposes (“Click here to return to the main menu”) or represent paid advertisements. Another downside is finding an equilibrium between relevance and popularity in the definition of authority.

The procedure as defined in the original paper is a link-based model for the conferral of authority. It highlights the relationship between authoritative pages for a

broad-topic query and the pages that link to many related authorities, dubbed *hubs*, noting that there exists a sort of organic equilibrium between hubs and authorities based on the link structure of the graph. A focused subgraph of the World Wide Web is constructed for a specified broad-topic query by identifying both hubs and authorities in the hyperlinked environment simultaneously. To do so, a collection V of hyperlinked pages is first established as a directed graph G , which is defined by the pair $G = (V, E)$, where V is the set of vertices, denoting web pages, and E is the set of directed edges, where $(p, q) \in E$ indicates a link from page p to page q . The out-degree of a node p is the number of nodes p has links to, and the in-degree of p is the number of nodes that have links to p . The aim of this algorithm is to define a subgraph of G as follows: if $W \subseteq V$ is a subset of the pages, $G[W]$ is called the graph induced on W , where $G[W]$'s nodes are the pages in W and $G[W]$'s edges are the links between pages in W .

Let σ be an arbitrary but fixed broad-topic query string. Before identifying authoritative pages in the hyperlinked environment based on the link structure, the induced subgraph on the World Wide Web on which the algorithm is applied must be determined first. The purpose of this initial step is to lessen the computational burden of the task by isolating the pages relevant to the query σ . This specific collection of pages is given a variable name, S_σ , which satisfies the following properties: (1) S_σ is relatively small; (2) S_σ is rich in relevant pages; (3) S_σ contains most (or many) of the strongest authorities. Keeping S_σ small in turn keeps computational costs reasonable and facilitates the task of finding authoritative pages. To find S_σ , a parameter t is given, which is usually set to around 200. The t highest-ranked pages for the query σ from a text-based search engine are collected. These t pages are called the root set R_σ , and satisfy properties (1) and (2) above, but usually not (3). So these t pages in R_σ are not enough. Note that the top t pages returned by the text-based

search engines all contain the query string σ , so $R_\sigma \subseteq Q_\sigma$, where Q_σ is the set of all pages containing σ . However, Q_σ itself often does not satisfy condition (3), and also there are often very few links between pages in R_σ , so the current subgraph lacks the sufficient structure to identify authoritative pages. From here, the root set R_σ is expanded to produce S_σ , which satisfies the three required conditions. Say there exists an authoritative page p for the query σ . p may not be in the root set R_σ , but there is likely a page in R_σ that links to p . So starting with R_σ , the subgraph can be expanded to include more authoritative pages via the links that enter and leave R_σ . This procedure is formalized as follows.

Subgraph($\sigma, \mathcal{E}, t, d$)

σ : a query string

\mathcal{E} : a text-based search engine

t, d : natural numbers

Let R_σ denote the top t results of \mathcal{E} on σ .

Set $S_\sigma := R_\sigma$

For each page $p \in R_\sigma$

Let $\Gamma^+(p)$ denote the set of all pages p points to.

Let $\Gamma^-(p)$ denote the set of all pages pointing to p .

Add all pages in $\Gamma^+(p)$ to S_σ .

If $|\Gamma^-(p)| \leq d$ then

Add all pages in $\Gamma^-(p)$ to S_σ .

Else

Add an arbitrary set of d pages from $\Gamma^-(p)$ to S_σ .

End

Return S_σ

In short, S_σ is constructed by growing R_σ to include any page pointed to by a page in R_σ and any page that points to a page in R_σ , with the restriction that a single page in R_σ is allowed to bring at most d pages pointing to it into S_σ . This d ceiling value is necessary to avoid blowing up the size of S_σ in the case where a page in R_σ is pointed to by a huge number of pages.

S_σ is called the base set for a broad-topic query σ . In Kleinberg’s experiments, he uses the (now defunct) search engine AltaVista and sets $t = 200$ and $d = 50$. He finds that S_σ typically satisfies conditions (1), (2), and (3) above, and its size is usually in the range 1000-5000. His experiment results are not discussed here, as only comprehension of the implementation of HITS is necessary for this thesis. The actual HITS algorithm is set up for computing hubs and authorities in the newly-acquired base set S_σ .

Just as the authors of PageRank offer a solution for handling dangling links, Kleinberg offers a way to offset the effect of links that solely exist for navigational purposes, and do not contribute to the notion of authority here. Let $G[S_\sigma]$ denote the subgraph induced on the pages in the base set S_σ . Partition the links in $G[S_\sigma]$ into two categories: traverse links and intrinsic links. Traverse links connect pages with different domain names, while intrinsic links connect pages with the same domain name. Intrinsic links often exist for purely navigational purposes, and so do not necessarily confer authority in any meaningful way in this study. Hence, delete all intrinsic links from the subgraph $G[S_\sigma]$, keeping only traverse links. This action bears the final desired graph G_σ .

Both PageRank and HITS recognize that a large number of incoming links do not necessarily translate into a high influence weight. The influence of the pages attached to the incoming links must be considered in the weighting scheme. Indeed, Kleinberg points out that pages with high in-degree values (a large number of links pointing to them) even in G_σ , which was so painstakingly defined, are not necessarily authoritative regarding the query σ . A page p in G_σ with a large in-degree can still have many incoming links from pages which are not relevant to σ . Such endorsements are thus hollow and should not be counted in p ’s influence weight. The distinction between genuinely authoritative pages and pages that are simply “universally popular” lends

itself to the symbiotic relationship between authoritative pages and hub pages. Authoritative pages relevant to the query σ should have both large in-degree and many incoming links from pages that have links to multiple relevant authoritative pages. The latter pages are called *hub pages*. Hub pages point to many authoritative pages, while authoritative pages are pointed to by many hub pages. This idea seems to employ circular logic, but this equilibrium actually comprises a mutually reinforcing relationship in the formal definition of HITS.

Finally, now that the necessary background knowledge is amassed, an iterative algorithm for HITS can be defined as follows. Each page p is assigned two numerical weights: a non-negative authority weight $x^{(p)}$ and a non-negative hub weight $y^{(p)}$. Normalize the weights of each type for all of the pages so their squares sum to one:

$$\sum_{p \in S_\sigma} (x^{(p)})^2 = 1 \quad \text{and} \quad \sum_{p \in S_\sigma} (y^{(p)})^2 = 1.$$

A larger x -value denotes a strong authority, while a larger y -value denotes a strong hub. The reciprocal relationship between hubs and authorities can be numerically described as follows: If a page p points to many pages with large x -values, then p should receive a large y -value, and if p is pointed to by many pages with large y -values, then p should receive a large x -value. Two operations are defined, labeled \mathcal{I} and \mathcal{O} . Given the authority weights $\{x^{(p)}\}$ and hub weights $\{y^{(p)}\}$ of the graph, the \mathcal{I} operation updates the x -values as follows:

$$x^{(p)} \leftarrow \sum_{q:(q,p) \in E} y^{(q)}$$

while the \mathcal{O} operation updates the y -values as follows:

$$y^{(p)} \leftarrow \sum_{q:(p,q) \in E} x^{(q)}.$$

The \mathcal{I} and \mathcal{O} operations are how hubs and authorities reinforce one another. The iterative procedure to compute the authority and hub weights for the nodes entails

applying operations \mathcal{I} and \mathcal{O} in an alternating fashion until (ideally) a fixed state is reached.

Represent the set of authority weights $\{x^{(p)}\}$ as a vector x , where each coordinate corresponds with the authority weight of its respective page in G_σ . Likewise, represent the set of hub weights $\{y^{(p)}\}$ as a vector y . Now define the procedure

Iterate(G, k)

G : a collection of n linked pages

k : a natural number

Let z denote the vector $(1, 1, 1, \dots, 1) \in \mathbb{R}^n$.

Set $x_0 := z$

Set $y_0 := z$

For $i = 1, 2, \dots, k$

 Apply the \mathcal{I} operation to (x_{i-1}, y_{i-1}) , obtaining new x -weights x'_i .

 Apply the \mathcal{O} operation to (x'_i, y_{i-1}) , obtaining new y -weights y'_i .

 Normalize x'_i , obtaining x_i .

 Normalize y'_i , obtaining y_i .

End

Return (x_k, y_k)

This procedure can be applied to pinpoint the top c authorities and top c hubs via the procedure

Filter(G, k, c)

G : a collection of n linked pages

k, c : natural numbers

$(x_k, y_k) := \text{Iterate}(G, k)$

Report the pages with the c largest coordinates in x_k as authorities.

Report the pages with the c largest coordinates in y_k as hubs.

For his experiments, Kleinberg applies the Filter procedure with G_σ for the G parameter and with $c \approx 5 - 10$.

As with PageRank, HITS in theory involves solving some eigenvector equation, which is how the value to pass for k in the Filter procedure above can be determined. It can be shown that the sequences of vectors $\{x_k\}$ and $\{y_k\}$ converge to fixed points x^* and y^* , respectively, when the Iterate procedure is applied with arbitrarily large values of k . However, for this thesis, solely the iterative versions of PageRank, HITS, and Pinski-Narin are implemented. Unlike the explanation of PageRank, which necessarily entails setting up its respective eigenvector equation first before segueing into its iterative version, the matrix equation formulation for HITS is omitted. Instead of solving for an eigenvector equation to determine what value to use for k in the Filter procedure, for the experimentation phase, the Iterate procedure was simply repeatedly applied until $|x_i - x_{i-1}| < \epsilon$ and $|y_i - y_{i-1}| < \epsilon$, where $\epsilon > 0$ is arbitrary but fixed. And with that, this section on the HITS algorithm is concluded.

2.3 Pinski-Narin Algorithm

Somewhat more obscure than the latter two algorithms, but nonetheless pivotal in its contribution to network science, the Pinski-Narin algorithm was published in

the *Information Processing and Management* journal by Gabriel Pinski and Francis Narin in 1976 (Pinski and Narin (1976)). Originally intended for citation networks of scientific publications, the authors make three observations on the limitations of the centrality measures that occupied the network science zeitgeist of the time. Firstly, they do not account for the average length of individual papers appearing in a given scientific journal, so centrality measures are skewed in favor of journals which publish longer papers, namely review journals. Secondly, all citations are considered equal: they grant equal weight to all citations, regardless of the citing journal. Pinski and Narin posit that higher weight should be granted to a citation from a more prestigious journal over one from a less prestigious journal. This idea has also been advanced by Manfred Kochen in his book *Principles of Information Retrieval* (Kochen (1974)). Finally, centrality measures at the time did not have a normalization procedure for the discrepancies between different subsets of the scientific literature community. Biochemistry, for example, is noted for its high publication frequency, while astronomy tends to output much fewer publications in general and at a much slower rate. Hence, a citation from a biochemistry journal may differ greatly in value compared to a citation from an astronomy journal.

Pinski and Narin argue their algorithm ameliorates the three aforementioned shortcomings by considering the following three factors: (1) the (size-independent) influence weight of the journal, which both weights the citations received from other journals and also normalizes itself using the number of references given to other journals; (2) the influence of the individual publications which comprise the journal, which consists of the weighted number of citations each publication receives; (3) the influence of the journal as a whole which is the product of the influence per publication and the total number of publications. In short, Pinski and Narin deem a journal influential if it is frequently cited by other influential journals, and they wish to reflect

this weighting scheme in their algorithm. This recursive intuition is also exercised by Kleinberg in his HITS algorithm when computing the authority and hub weights.

Before defining the actual Pinski-Narin algorithm, the problem is set up. Say a citation network consisting of n nodes is given, with the nodes enumerated from 1 to n , where each node denotes a member of some set of publishing entities (a journal, a publication, etc.). Avoid attaching a specific type of publication to a node to allow for generality in the universe of scientific publications. Each edge signifies a citation from one unit to another. The network is represented by its respective $n \times n$ citation matrix C , which is defined as follows.

$$C = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}$$

where the i th row, j th column entry C_{ij} denotes the number of citations node j receives from node i , and alternatively the number of references node i gives to node j , $1 \leq i, j \leq n$. Note the difference in the definition of “citation” versus that of “reference”. A simple way to distinguish between the two terms is to remember that a citation is *received*, while a reference is *given*.

Define the row sum

$$S_i = \sum_{j=1}^n C_{ij}$$

which is the sum of the entries in the i th row of the citation matrix C . S_i may be interpreted as the total number of references given out by node i . Then the influence weight, W_i , of node i is defined as

$$W_i = \sum_{k=1}^n \frac{W_k C_{ki}}{S_i}, \quad i = 1, \dots, n. \quad (2.1)$$

In other words,

$$W_i = \sum_{k=1}^n \frac{W_k C_{ki}}{S_i} = \frac{\sum_{k=1}^n W_k C_{ki}}{\sum_{j=1}^n C_{ij}},$$

so the influence weight of node i can be thought of as the weighted sum of its citations divided by the number of its references. If all nodes are considered equally, if node i receives more citations than it gives out references, then i has a higher weight than a node that has fewer citations than references. However, Pinski-Narin does not weigh all nodes equally, which is precisely the role of weighting the sum of citations in the numerator of W_i , $\sum_{k=1}^n W_k C_{ki}$. The idea is that a citation from a more influential node is considered more highly than a citation from a less influential node. So if node i receives a citation from a very prestigious node l , node i 's influence weight W_i is rewarded accordingly. Likewise, if node i receives a citation from an obscure node m , W_i is not as handsomely compensated, so $W_l C_{li} > W_m C_{mi}$, so long as $C_{li} = C_{mi}$. An exception may occur when $C_{mi} \gg C_{li}$. It may be the case that a significant number of citations from a less influential node may be weighed equally as or even more highly than a small number of citations from a more influential node.

The aforementioned definition of W_i can be viewed as a special case of the general system of equations

$$\sum_{k=1}^n W_k \gamma_{ki} - \lambda W_i = 0, \quad i = 1, \dots, n \quad (2.2)$$

where $\gamma_{ki} = C_{ki}/S_i$. Equation (2.1) is an instance of equation (2.2) where $\lambda = 1$. The values of λ that satisfy the equation immediately above are called the eigenvalues of the system. $\lambda = 1$ itself is an eigenvalue, so it is guaranteed there exists at least one non-zero solution for equation (2.2). γ_{ki} denotes the number of citations node i receives from node k divided by the total number of references given by node i .

Define the $n \times n$ matrix

$$\boldsymbol{\gamma} = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1n} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2n} \\ \vdots & \vdots & & \vdots \\ \gamma_{n1} & \gamma_{n2} & \cdots & \gamma_{nn} \end{bmatrix}.$$

Then the column sum i , $1 \leq i \leq n$, represents the ratio between the number of citations received by node i and the number of references given by node i . In particular,

$$\begin{aligned} \gamma_{1i} + \gamma_{2i} + \cdots + \gamma_{ni} &= \sum_{k=1}^n \gamma_{ki} \\ &= \sum_{k=1}^n \frac{C_{ki}}{S_i} \\ &= \frac{\sum_{k=1}^n C_{ki}}{\sum_{j=1}^n C_{ij}} \\ &= \frac{\text{total number of citations to node } i \text{ from other nodes}}{\text{total number of references from node } i \text{ to other nodes}} \end{aligned}$$

Let $\boldsymbol{\gamma}^T$ denote the transpose of $\boldsymbol{\gamma}$:

$$\boldsymbol{\gamma}^T = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1n} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2n} \\ \vdots & \vdots & & \vdots \\ \gamma_{n1} & \gamma_{n2} & \cdots & \gamma_{nn} \end{bmatrix}$$

Then $\gamma_{ik}^T = \gamma_{ki}$. Employ the Kronecker delta symbol

$$\delta_{ik} = \begin{cases} 1 & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases}$$

so it is possible to write

$$\sum_{k=1}^n (\gamma_{ik}^T - \lambda \delta_{ik}) W_k = 0, \quad i = 1, \dots, n. \quad (2.3)$$

Equation (2.3) is simply another way to write equation (2.2). Equation (2.3) bears a system of n homogeneous equations for the influence weights. To find a solution for this system, solve for eigenvalue λ in the n th order equation

$$\det \begin{bmatrix} \gamma_{11} - \lambda & \gamma_{21} & \cdots & \gamma_{n1} \\ \gamma_{12} & \gamma_{22} - \lambda & \cdots & \gamma_{n2} \\ \vdots & \vdots & & \vdots \\ \gamma_{1n} & \gamma_{2n} & \cdots & \gamma_{nn} - \lambda \end{bmatrix} = 0 \quad (2.4)$$

known as the characteristic equation. A non-zero solution for the weights can be found by finding values of λ which satisfy equation (2.4). Given the $n \times 1$ column vector $\mathbf{W} = [W_1, \dots, W_n]^T$ of influence weights, the eigenvalue equation above can be expressed as a vector equation

$$\boldsymbol{\gamma}^T \cdot \mathbf{W} = \lambda \mathbf{W}. \quad (2.5)$$

Now the normalization procedure of Pinski-Narin is described. The condition is imposed that the size-weighted average of the weights is one:

$$\frac{\sum_{k=1}^n S_k W_k}{\sum_{k=1}^n S_k} = 1.$$

This condition grants the weights both an absolute and a relative meaning, since the value one represents an average value. Each root λ of the characteristic equation (2.4) (or alternatively, equation (2.5)) determines an eigenvector—a potential solution vector—of the equation, but only the eigenvector corresponding with the largest eigenvalue is desired.

The iterative procedure to compute Pinski-Narin, which is what is actually implemented to produce the experimentation results, is as follows. Start with equal weights of one for all of the nodes, just like HITS. The values $W_i^{(0)} = 1, i = 1, \dots, n$, can be considered as zeroth order approximations to the weights. The first order weights are

$$\begin{aligned}
W_i^{(1)} &= \frac{\sum_{k=1}^n C_{ki}}{S_i} \\
&= \frac{\text{total number of citations to node } i \text{ from other nodes}}{\text{total number of references from node } i \text{ to other nodes}},
\end{aligned}$$

which are precisely the column sums of the matrix γ defined earlier. To acquire the next order of approximation, substitute the values $W_i^{(1)}$ into the right hand side of equation (2.1). In general, the m th order approximation is

$$W_i^{(m)} = \sum_{k=1}^n \frac{W_k^{(m-1)} C_{ki}}{S_i} = \sum_{k=1}^n W_k^{(m-1)} \gamma_{ki} = \sum_{j=1}^n (\gamma^m)_{ji}.$$

To find the exact weights, essentially continually square the γ matrix to obtain

$$W_i = W_i^{(\infty)} = \sum_{j=1}^n \left(\lim_{m \rightarrow \infty} \gamma^m \right)_{ji}.$$

$\lambda = 1$ is the dominant eigenvalue (that is, the largest eigenvalue) for the characteristic equation (2.4), so instead of actually solving the eigenvector equation, just repeated squarings of the γ matrix can be performed. For this thesis, γ was continually squared until $|W_i^{(t)} - W_i^{(t-1)}| < \epsilon$, $i = 1, \dots, n$, where $\epsilon > 0$ is arbitrary but fixed.

A follow-up paper by Nancy Geller was published in 1978 (Geller (1978)), in which Geller sheds further insight into the Pinski-Narin algorithm using Markov chain theory. Geller observes that the influence weights computed by Pinski-Narin correspond to the stationary distribution of the following random process: starting with an arbitrary academic publication j , randomly choose one of j 's references and move to said reference, and so on. This process mimics the random surfer model for PageRank, except instead of exploring web pages in the World Wide Web, publications in a citation network are being explored.

As is the case with PageRank and HITS, Pinski and Narin's experiment results are not discussed. This section on the Pinski-Narin algorithm is concluded with the

following observation: while Kleinberg cites Pinski and Narin's paper in his HITS paper, Page, Brin, Motwani, and Winograd do not do so in their PageRank paper. The latter party, however, cite Kleinberg's HITS paper in their PageRank paper.

2.4 Hyperbolic Geometry

At this point, the reader has a general understanding of the three eigenvector centrality algorithms implemented in this thesis. Now the fundamental principles this entire thesis is predicated on must be explained: hyperbolic geometry. The literature on hyperbolic geometry is boundless, but the information provided in this section is primarily derived from Birger Iversen's book *Hyperbolic Geometry* (Iversen (1992)). In short, hyperbolic geometry is a geometry which concerns itself with non-Euclidean space. Euclidean space is governed by Euclid's five axioms, which are as follows.

1. A straight line can be drawn joining any two points.
2. Any straight line segment can be extended indefinitely in a straight line.
3. Given any straight line segment, a circle can be drawn having the segment as radius and one endpoint as center.
4. All right angles are congruent.
5. If two lines are drawn which intersect a third in such a way that the sum of the inner angles on one side is less than two right angles, then the two lines inevitably must intersect each other on that side if extended far enough. This postulate is equivalent to what is known as the parallel postulate.

The fifth postulate can be reworded as follows: Given a straight line and a point not on the line, there exists exactly one straight line intersecting the point that is parallel to the given line. Investigations of the parallel postulate by mathematicians like

János Bolyai (1802-1860), Carl Friedrich Gauss (1777-1855), and Nikolai Lobachevsky (1793-1856) have revealed the axiom is independent of the preceding four axioms in that there exists a space, now called hyperbolic space, which satisfies the first four of Euclid's axioms, but not the fifth.

The discovery of hyperbolic space arose from attempts to prove the parallel postulate independently of the other four postulates. Giovanni Saccheri (1667-1733) was the first to make serious headway in this pursuit, followed by Johann Heinrich Lambert (1728-77), who made notable speculations on the nature of a non-Euclidean geometry. By 1850 most of the properties of such a new geometry were known to Bolyai, Gauss, and Lobachevsky. The main issue at the time was proving the existence of this geometry, but twenty years later this mystery was resolved, thanks to the work of Eugenio Beltrami (1835-1900), Arthur Cayley (1821-95), and Felix Klein (1849-1925). Before the reader falls asleep from this history lecture, the reader is beseeched to push onward for the exciting details on hyperbolic space and its geometry. This concludes the historical background of hyperbolic geometry (pinky promise).

In hyperbolic space, given a straight line and a point not on the line, it is possible for there to be more than one straight line intersecting the point that is parallel to the given line. In fact, there are infinitely many such lines. This phenomenon can be seen in the following Figure 2.2, which exhibits a set of straight lines in the Poincaré disk that intersect a given point but are also all parallel to the bolded blue line (Chamberlain *et al.* (2017)).

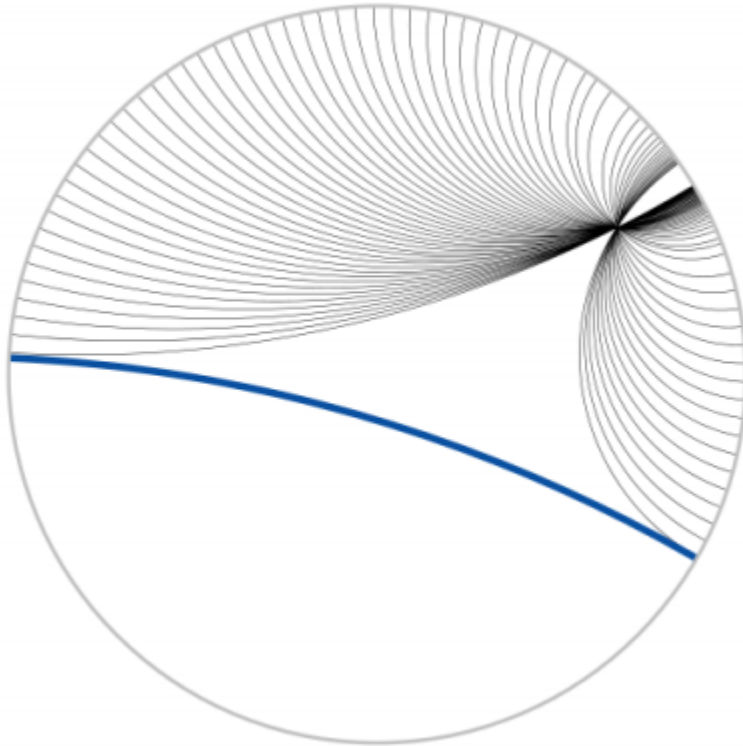


Figure 2.2: A Set of Straight Lines in the Poincaré Disk that All Pass Through a Given Point and are All Parallel to the Blue Line.

What the Poincaré disk is and its relation to hyperbolic space are explained shortly. If the reader is confused by how the straight lines in Figure 2.2 appear to be curved, this is also clarified shortly.

To spare the reader the tedious journey of trying to navigate mathematical jargon, in short, hyperbolic space is the space dictated by the axioms of hyperbolic geometry, which are identical to the five postulates of Euclidean geometry listed above, but with the fifth postulate relaxed so that given a line and a point that does not lie on it, there is more than one line going through the given point that is parallel to the given line. There are only three types of isotropic spaces: Euclidean space, spherical space, and hyperbolic space. Isotropy means “uniformity in all orientations”, and in this context means the axioms which govern each space hold true in any orientation in the respective space. The notion of curvature is used to differentiate between the

isotropic spaces. As the name implies, the curvature of a surface is a metric which numerically quantifies how curved the surface is. Euclidean geometry concerns itself with spaces of zero curvature (also known as flat spaces), spherical geometry handles spaces of constant positive curvature, and hyperbolic geometry deals with spaces of constant negative curvature. The following Table 2.1 explaining basic differences among Euclidean, spherical, and hyperbolic geometries is pulled from Krioukov *et al.* (2010).




Property	Euclidean	Spherical	Hyperbolic
Curvature K	0	> 0	< 0
Parallel lines	1	0	∞
Triangles are	normal	thick	thin
Shape of triangles			
Sum of Δ angles	π	$> \pi$	$< \pi$
Circle length	$2\pi r$	$2\pi \sin \zeta r$	$2\pi \sinh \zeta r$
Disk area	$2\pi r^2/2$	$2\pi(1 - \cos \zeta r)$	$2\pi(\cosh \zeta r - 1)$

Table 2.1: Characteristic Properties of Euclidean, Spherical, and Hyperbolic Geometries

Each entry in the row for “Parallel lines” indicates the number of lines that are parallel to a given line and intersect a point not belonging to the given line for the specified geometry. The variable $\zeta = \sqrt{|K|}$. In the implementation of this thesis, $\zeta = 1$.

Hyperbolic space and hyperbolic geometry are the focus for the remainder of this section. The reasons why hyperbolic geometry is preferred in this context are discussed in the following section on hyperbolic embeddings. In practice, hyperbolic geometry is rarely explored in hyperbolic space alone. This is frightening, unfamiliar territory. Rather, one of various models that embed hyperbolic space in Euclidean

space is selected. This way, hyperbolic geometry can be physically observed in action in a familiar, warm, friendly environment. Such models include the Poincaré ball model, the Poincaré half-space model, the hyperboloid model, and the Klein model, among others. The Poincaré ball model is opted for for this thesis. In particular, two-dimensional hyperbolic space \mathbb{H}^2 , also known as the hyperbolic plane, is the sole concern. As such, the two-dimensional case of the Poincaré ball model, which is called the Poincaré disk model, is defined.

The Poincaré disk model represents the entire infinite hyperbolic plane \mathbb{H}^2 inside the finite unit disk $\mathbb{D} = \{z \in \mathbb{R}^2 : \|z\| < 1\}$, where $\|\cdot\|$ denotes the L_2 norm. The boundary of the unit disk, the Euclidean circle $\partial\mathbb{D} = \{z \in \mathbb{R}^2 : \|z\| = 1\}$, is not part of the hyperbolic plane \mathbb{H}^2 , but rather represents its infinitely distant points, known as the boundary at infinity $\partial\mathbb{H}^2$. The distance between two points (x, y) and (x', y') in the Euclidean plane \mathbb{R}^2 is determined by the two-dimensional Euclidean metric $\|(x, y) - (x', y')\| = \sqrt{(x - x')^2 + (y - y')^2}$. Although the Poincaré disk resides in \mathbb{R}^2 , the distance between points $z_i = (x_i, y_i)$ and $z_j = (x_j, y_j)$ is dictated by the two-dimensional hyperbolic metric, the hyperbolic distance

$$\cosh d(z_i, z_j) = 1 + 2 \frac{\|z_i - z_j\|^2}{(1 - \|z_i\|^2)(1 - \|z_j\|^2)} \quad (2.6)$$

where $d(z_i, z_j)$ denotes the hyperbolic distance between points z_i and z_j . To actually compute $d(z_i, z_j)$, simply take the \cosh^{-1} of both sides of the above expression. Note that the Poincaré disk model can easily be generalized to the Poincaré ball model of n dimensions by adjusting for the unit ball \mathbb{D}^n . Also note that equation (2.6) is expressed in terms of rectangular coordinates. Equation (2.6) can be rewritten in terms of polar coordinates (r, θ) , where r is the radial coordinate and θ is the angular coordinate. Say z_i has polar coordinates (r_i, θ_i) , while z_j has polar coordinates (r_j, θ_j) .

Then it may be written

$$\cosh d(z_i, z_j) = \cosh r_i \cosh r_j - \sinh r_i \sinh r_j \cos \Delta\theta \quad (2.7)$$

where $\Delta\theta = \pi - |\pi - |\theta_i - \theta_j||$ is the angle between z_i and z_j .

The circumference and the area of a circle in hyperbolic space are defined as functions of the circle's radius r ,

$$C(r) = 2\pi \sinh r \quad \text{and} \quad A(r) = 2\pi(\cosh r - 1),$$

respectively. Recall that

$$\cosh r = \frac{e^r + e^{-r}}{2} \quad \text{and} \quad \sinh r = \frac{e^r - e^{-r}}{2}.$$

An important characteristic of the hyperbolic plane \mathbb{H}^2 can be observed when the derivative of $A(r)$, $A'(r)$, is taken:

$$\begin{aligned} A'(r) &= \frac{dA}{dr} = 2\pi \frac{d}{dr}(\cosh r - 1) \\ &= 2\pi \frac{d}{dr} \left(\frac{e^r + e^{-r}}{2} - 1 \right) \\ &= \pi(e^r - e^{-r}) \\ &= 2\pi \sinh r = C(r). \end{aligned}$$

When the value of r blows up (that is, $r \rightarrow \infty$), $A'(r) \approx e^r$. That is, the rate of change in the area of a circle is exponential in terms of its radius r . This growth rate is in stark contrast to the rate of change in the area of a circle in conventional \mathbb{R}^2 : since $A(r) = \pi r^2$ and $C(r) = 2\pi r$ in this case, $A'(r) = 2\pi r = C(r)$. In other words, the rate of change in the area of a circle in \mathbb{R}^2 is polynomial (linear, to be exact) in terms of r . \mathbb{H}^2 is then, in some sense, “larger” than \mathbb{R}^2 : the area of a circle grows exponentially with its radius, rather than linearly. More generally, hyperbolic

spaces expand faster than Euclidean spaces: the former expand exponentially, while the latter expand polynomially.

Every model of hyperbolic space has its pros and cons. The Poincaré disk model for \mathbb{H}^2 is conformal, meaning it preserves angles, so Euclidean angles between lines in the Poincaré disk are equal to the hyperbolic angles between said lines. However, areas and distances are warped in the Poincaré disk. In fact, hyperbolic distances grow exponentially towards the boundary $\partial\mathbb{D}$. If r_e equals the Euclidean distance from the disk center (also known as the origin of \mathbb{H}^2) and r_h equals the hyperbolic distance from the disk center, then the two quantities are related by

$$r_e = \tanh \frac{r_h}{2}.$$

This distortion explains why straight lines in the Poincaré disk, like in Figure 2.2, appear curved: when seeking the shortest path (in terms of hyperbolic distance) from one point on the disk to another, it is faster to move close to the disk center, where distances are shorter, than closer to the edge. In fact, hyperbolic geodesic lines in the Poincaré disk, i.e., shortest paths between two points at the boundary $\partial\mathbb{D}$, consist of disk diameters and arcs of Euclidean circles intersecting $\partial\mathbb{D}$ perpendicularly. The image in Figure 2.3 below illustrates what traversing geodesics in the Poincaré disk is like. Each tile is of constant area in hyperbolic space, but vanishes in Euclidean space at the boundary $\partial\mathbb{D}$.

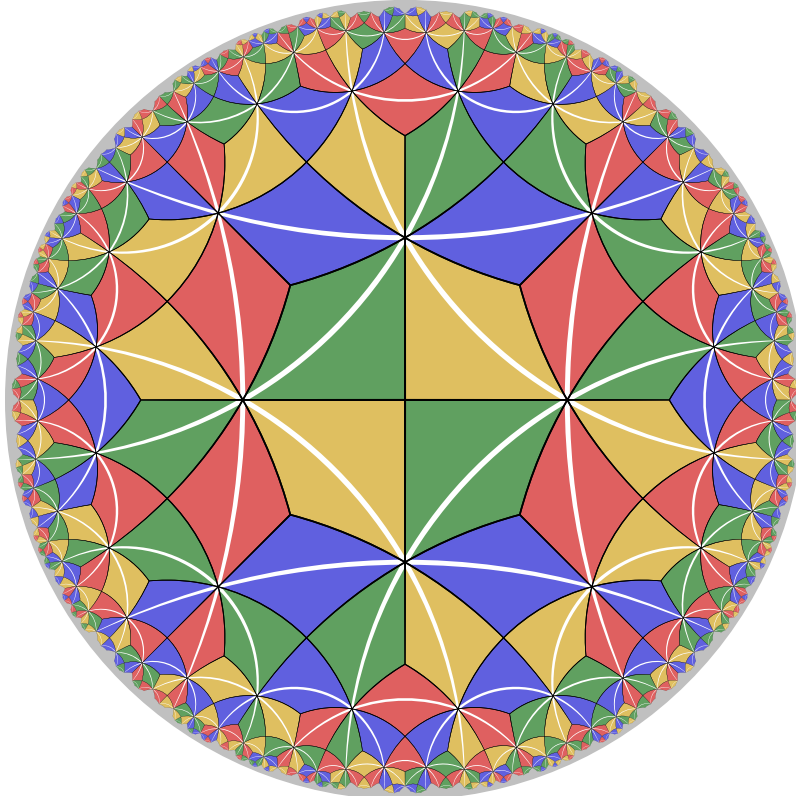


Figure 2.3: A Hyperbolic Tiling in the Poincaré Disk Model

Hyperbolic geometry is a beautifully dynamic and fascinating way to make sense of space. In the next section, it is explained how to exploit this powerful set of tools to study complex and social networks in the form of hyperbolic embeddings.

2.5 Hyperbolic Embeddings

Now hyperbolic geometry may be contextualized for this thesis specifically. Geometry naturally lends itself to network science as a way to quantify and compare characteristics of complex and social networks. A primary objective of network geometry is to describe and define the continuous hidden metric behind the inherently discrete structure of complex networks, and researchers are increasingly finding hyperbolic geometry to be the golden ticket (Bianconi and Rahmede (2017)). Chapter one of the book *Big Data in Complex and Social Networks* provides an overview of big

data analytics based on hyperbolic space as well as recent developments in the field (Thai *et al.* (2016)). Hyperbolic geometry has not only been used to perform data embeddings, but also to perform greedy routing and decision-making optimization, among other applications. This thesis is only concerned with embeddings of complex networks in hyperbolic space.

Recall from the prior section that hyperbolic space expands exponentially. In particular, that the area of a circle grows exponentially in terms of its radius r . This property easily lends hyperbolic space to modeling complex networks because such networks are often scale-free. If the degree of a node in a given network denotes its number of incident edges, the degree distribution of the network indicates the probability distribution of the node degrees over the entire network. A scale-free network is one where its degree distribution follows a power law. A power law is a functional relationship between two quantities x, y that can be modeled as $y \propto x^a$, where a is a constant and \propto stands for “proportional to”. Because scale-free networks expand exponentially, such networks contain latent hierarchies, i.e., tree-like structures (Boguna *et al.* (2009)), and since hyperbolic geometry is known as the geometry of trees, the marriage between them is an amicable one. In fact, the exponent of the power law function modeling the degree distribution of a scale-free complex network ends up being a function of the hyperbolic space curvature (Krioukov *et al.* (2010)). DeSa *et al.* (2018) shows how to embed arbitrary graphs as trees into hyperbolic space, specifically the Poincaré ball.

Hyperbolic space models are well-suited for modeling hierarchical data. Recall that a tree in graph theory is defined as an undirected, connected, and acyclic graph. In other words, a connected graph that contains zero cycles. Say a b -ary tree is given, which is a tree with branch factor b . The branch factor of a tree indicates the number of children at each node, the outdegree. Then the tree has $(b + 1)b^{\ell-1}$ nodes at level

ℓ and $\frac{(b+1)b^{\ell-2}}{b-1}$ nodes on a level less than or equal to ℓ . Note that as $\ell \rightarrow \infty$, the aforementioned quantities $(b+1)b^{\ell-1} \approx b^\ell$ and $\frac{(b+1)b^{\ell-2}}{b-1} \approx b^\ell$, respectively. It is seen that the number of children grows exponentially with their distance from the root of the tree. This tree-like behavior can be easily modeled using a circle of radius r , modeling the hyperbolic plane \mathbb{H}^2 , by mapping nodes which are exactly ℓ levels below the root directly on the circle's boundary, with $r \propto \ell$, and mapping nodes which are less than ℓ levels below the root within the circle (Nickel and Kiela (2017)). Since hyperbolic space expands exponentially, hyperbolic space can be viewed as just a continuous version of a tree. Likewise, trees can be viewed as “discrete hyperbolic spaces”, in a way. Figure 2.4 below, pulled from Nickel and Kiela (2017), shows an embedding of a regular tree (a tree where every node has equal degree, in this case three) in the Poincaré disk so that all connected nodes are spaced equally far apart in terms of hyperbolic distance. Krioukov *et al.* (2010) even proves that assuming an underlying hyperbolic geometry for a complex network yields heterogeneous degree distributions (huge variability between node degrees) and strong clustering in the network topology.

Faloutsos *et al.* (1999) and Boguna *et al.* (2010) further demonstrate the utility of hyperbolic space for modeling complex networks. Faloutsos *et al.* (1999) shows that the growth rate of the Internet can be modeled in terms of power laws. In other words, the Internet, when considered as a complex network, increases in size exponentially. Boguna *et al.* (2010) extends the aforementioned work considerably by devising an efficient greedy routing scheme for autonomous systems (AS's) in the Internet which lessens overhead costs via hyperbolic embeddings. AS in this paper is a name for a part of the Internet owned and administered by the same organization. Because the complex network comprising the Internet expands exponentially, handling rapidly growing overhead costs, which in this case deal with routing information packets be-

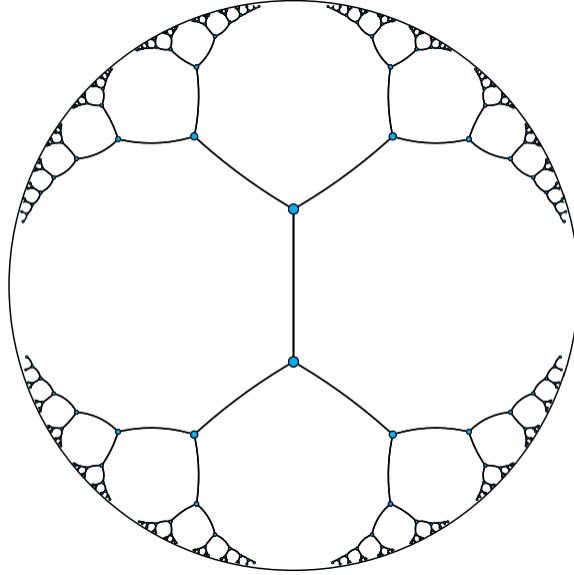


Figure 2.4: An Embedding of a Regular Tree in the Poincaré Disk so that All Connected Nodes are Spaced Equally Far Apart

tween computers, is a recurring headache for the people in charge of Internet routing. The procedure outlined in this paper consists of mapping the Internet to hyperbolic space and exploiting its properties to perform greedy forwarding efficiently. More specifically, the authors use statistical inference techniques to find coordinates for each AS in the hyperbolic space underlying the Internet. An AS holding a packet in-transit reads its destination AS's coordinates, calculates the hyperbolic distance between the destination and each of its neighbors, and forwards the packet to the neighbor closest to the destination (i.e., greedy forwarding). The authors show their embedding achieves both efficiency and robustness using a number of performance metrics. The benefits of using hyperbolic geometry to study complex networks are endless.

This thesis focuses exclusively on Poincaré embeddings, which are hyperbolic embeddings in the Poincaré disk model. The Poincaré disk model is the second most commonly used model for the hyperbolic plane \mathbb{H}^2 among mathematicians, behind the Poincaré half-plane model (the two-dimensional case of the Poincaré half-space

model) (Iversen (1992)). This order of preference among hyperbolic space models is contrary to that employed by computer scientists, who, in most of the network science papers encountered, opt for the Poincaré disk model. This discrepancy can be attributed to the visual compactness of the Poincaré disk, making it more attractive to computer scientists when performing visualization tasks. On the other hand, the Poincaré half-plane model has certain properties which are useful for applications of hyperbolic geometry to number theory, making it the first choice among mathematicians. ~~Or perhaps mathematicians and computer scientists are mortal enemies.~~

Two such Poincaré embeddings are the framework proposed in Papadopoulos *et al.* (2012) and the HyperMap embedding algorithm (Papadopoulos *et al.* (2015)). Papadopoulos *et al.* (2012) offers an algorithm which actively grows a complex network in the Poincaré disk model based on the nodes' popularity and similarity with one another. With every iteration, a new node is added to the network embedded in the model, forming links with existing nodes in the embedding, taking into account the popularity of the existing nodes and the similarity between the new node and the existing nodes. The procedure is formally defined as follows, assuming an empty network is initially given.

1. At time $t \geq 1$, add a new node t to the embedded network by assigning it the polar coordinates (r_t, θ_t) where the angular coordinate, θ_t , is sampled uniformly at random from $[0, 2\pi]$ and the radial coordinate, r_t , relates to the birth date of node t via the relation $r_t(t) = \ln t$. Then increase the radial coordinate of every existing node $s < t$ to $r_s(t) = \beta r_s(t) + (1 - \beta)r_t(t)$, $\beta \in [0, 1]$.
2. Connect the new node t with a subset of existing nodes $\{s\}$, where $s < t, \forall s$, consisting of the m nodes with the m smallest values of product $s \cdot \theta_{st}$, where m is a parameter controlling the average node degree (i.e., the extent of the

correlations among nodes), and θ_{st} is the angular distance between nodes s and t .

It is found that when constructing a network using the steps outlined above, the nodes simply connect to their closest m nodes in terms of hyperbolic distance. To summarize the intent of this algorithm, the radial coordinate represents the popularity of a node, while the angular coordinate represents the similarity between a node and other nodes in the network. HyperMap is an extended version of the previous algorithm in that it permits adding links between existing nodes, while the other algorithm only allows links to be added between a newcomer and an existing node. It should be noted that a network conceived from either Papadopoulos *et al.* (2012) or Papadopoulos *et al.* (2015) exhibits two characteristics: (1) the nodes appear to be very clustered since the links added between close nodes lead to the formation of a large number of triangles; (2) the degree distribution of the network adheres to a power law (the network is scale-free!).

The core of this thesis is in a similar vein as Papadopoulos *et al.* (2012) and Papadopoulos *et al.* (2015), although the hyperbolic coordinates in the proposed embedding are tailored to reflect the respective objective, which is to illustrate eigenvector centrality. At this point, the reader has accumulated the background knowledge necessary for understanding the essence of this thesis: a hyperbolic embedding of a complex network in the Poincaré disk model based on the PageRank, HITS, and Pinski-Narin eigenvector centrality algorithms. The purpose of explaining the two aforementioned hyperbolic embedding algorithms is to get the reader's toes wet in regards to Poincaré embeddings. Now an original embedding is introduced.

Chapter 3

IMPLEMENTATION

Each node in the network is assigned a pair of hyperbolic coordinates (r, θ) in the Poincaré disk model, where r is the radial coordinate, denoting the node's distance from the origin, and θ is the angular coordinate, denoting the node's angle from the positive x -axis. A node's hyperbolic coordinates should be chosen so that said node's position in the Poincaré disk in relation to the other nodes in the network accurately reflects the node's respective influence.

PageRank initializes the nodes' influence weights differently from HITS and Pinski-Narin. As such, PageRank's initial hyperbolic embedding should vary from those for HITS and Pinski-Narin. In particular, while PageRank and Pinski-Narin offer only one general, all-encompassing categorization of the nodes' influence weights, HITS produces two types of weights: hub weights and authority weights. Each iteration of HITS is hence accompanied by two influence weight vectors, so each iteration of HITS is accompanied by two corresponding hyperbolic embeddings as well. After each iteration of one of the algorithms PageRank, HITS, and Pinski-Narin, the nodes' influence weights are updated accordingly, along with the hyperbolic embeddings.

The network in the experimentation is defined as an unweighted, directed graph, which is defined as a pair $G = (V, E)$. V is the set of vertices, while E is the set of edges. G is represented in terms of an adjacency matrix A . If the network in question consists of n nodes (that is, $|V| = n$), and the nodes are enumerated from 1 to n , the adjacency matrix A is defined as an $n \times n$ square matrix where

$$\begin{aligned}
a_{ij} &= \begin{cases} 1 & \text{if there is a directed edge from node } i \text{ to node } j \\ 0 & \text{else} \end{cases} \\
&= \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{else} \end{cases}
\end{aligned}$$

is the i th row, j th column entry in A , $1 \leq i, j \leq n$.

Before formally defining the implementation of the proposed hyperbolic embedding, the logic behind the hyperbolic coordinate mappings is first explained. Two such mappings are defined: a radial mapping for the radial coordinate r , and an angular mapping for the angular coordinate θ . Both are determined based on the specified node's computed influence weight after a given iteration of one of PageRank, HITS, and Pinski-Narin. Now the setup for each mapping is explained.

The radial mapping for some node i , $1 \leq i \leq n$, after iteration t of a given algorithm is $r_i(t) = e^{-x_i(t)}$, where $x_i(t)$ is the influence weight of node i after iteration t . When the nodes' influence weights are initialized by one of PageRank, HITS, and Pinski-Narin, the nodes' radial coordinates would then be $r_i(0) = e^{-x_i(0)}$, $i = 1, \dots, n$. The reason why the radial coordinate is defined using this function is because its rate of growth as $x_i(t) \rightarrow \infty$ is well-suited for the Poincaré disk model, as seen in the graph in Figure 3.1 below.

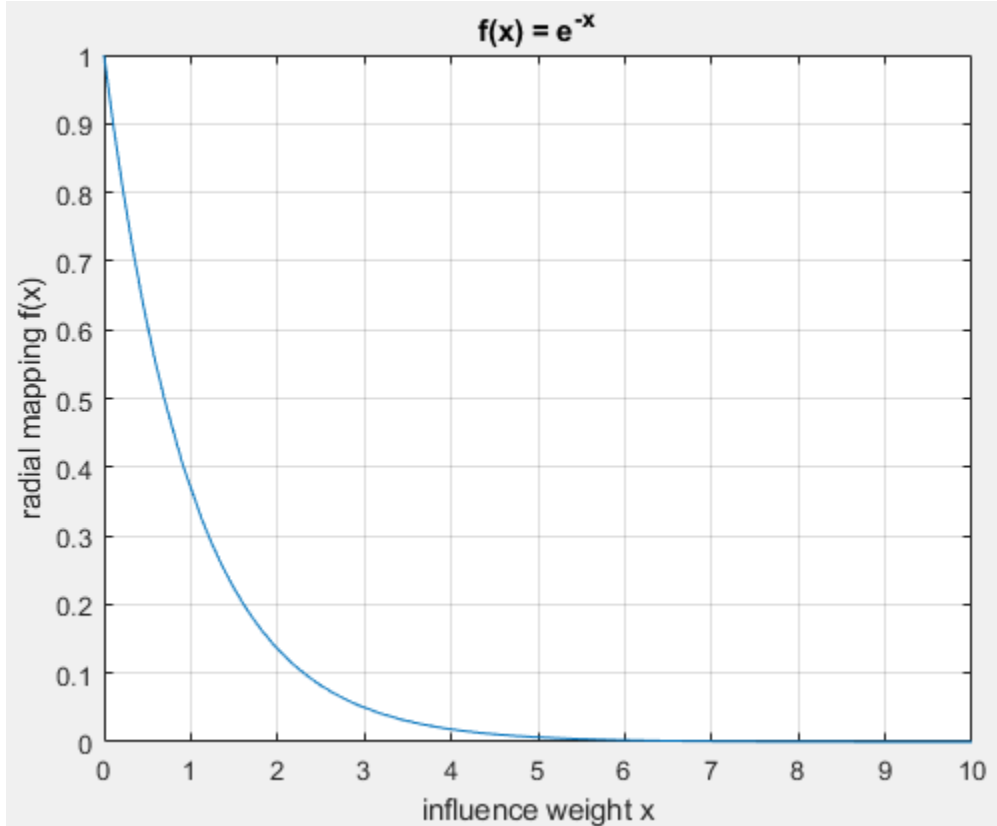


Figure 3.1: Graph of Radial Coordinate Function $f(x) = e^{-x}$

In this graph, it can be seen that as $x \rightarrow \infty$ (in other words, as a given node’s influence increases without bound), the corresponding node’s radial coordinate (or distance from the origin) $r \rightarrow 0$. If a node is more influential, it is ideally granted a higher influence weight by its respective eigenvector centrality algorithm, and hence is rewarded by being assigned a smaller radial coordinate, so it is located closer to the origin of the Poincaré disk. Likewise, if a node is less influential, it should be granted a lower influence weight by the given algorithm, and subsequently is penalized by being assigned a larger radial coordinate, so it is located farther away from the origin of the Poincaré disk. The idea is that more influential nodes are mapped closer to the origin, while less influential nodes are mapped farther away from the origin. The hope is that this discrepancy determined by the function $r_i(t) = e^{-x_i(t)}$ allows “more

important” nodes to be easily distinguished from “less important” nodes in a network. When one of PageRank, HITS, and Pinski-Narin is applied to a network’s adjacency matrix, until convergence, the algorithm produces hyperbolic embeddings where more influential nodes eventually migrate towards the origin, while less influential nodes gradually migrate away from the origin, all in real time.

All of the graphs worked with in this thesis are assumed to be strongly connected, which is when every vertex is reachable from every other vertex. That is, there exists a directed path between any pair of vertices in the graph. Hence, a node’s influence weight is always nonzero, as each node is guaranteed to have at least one outgoing and one incoming edge. This assumption is necessary because the Poincaré disk consists of the unit disk dictated by the hyperbolic metric. Recall that the unit disk is defined as $\mathbb{D} = \{z \in \mathbb{R}^2 : \|z\| < 1\}$, which comprises the points in \mathbb{R}^2 occupying the interior of $x^2 + y^2 = 1$. The fact that the network is guaranteed to be strongly connected further strengthens the case for $r_i(t) = e^{-x_i(t)}$ as the radial mapping because no matter how insignificant a node may be in terms of influence, said node always resides within the unit disk. An unimportant node can get arbitrarily close to the border of the unit disk, but never actually touches it. Figure 3.1 shows that the radial coordinate is 1 in the case where a node’s influence weight is 0. In other words, a node with absolutely no influence would be mapped to the border of the unit disk. Rest assured this scenario never occurs, as only strongly connected graphs are provided.

Another motivation for the choice of radial mapping $r_i(t) = e^{-x_i(t)}$ is the fact that the hyperbolic sine and cosine functions, which are used to calculate angles and distances in hyperbolic geometry, are defined in terms of the exponential functions e^x and e^{-x} , as seen below.

$$\cosh x = \frac{e^x + e^{-x}}{2} \qquad \sinh x = \frac{e^x - e^{-x}}{2}$$

The behavior of a node’s migration towards or away from the origin in the Poincaré disk as the node’s influence waxes or wanes, respectively, thus closely mimics what such a journey would actually look like in hyperbolic space.

Now the angular mapping $\theta_i(t)$ is defined and elaborated on, $1 \leq i \leq n$, where t is the number of iterations of one of PageRank, HITS, and Pinski-Narin after initialization. This mapping entails considerably more effort computation-wise than the radial mapping. While the radial mapping is easily and directly calculated, the angular mapping demands comparative deliberation based on the specified node’s hyperbolic distance from its neighbors. In the initial hyperbolic embeddings for PageRank, HITS, and Pinski-Narin, the angular coordinates are selected as follows: node 1’s angular coordinate is chosen uniformly at random from the interval $[0, 2\pi]$. From here, node i ’s angular coordinate is defined as $\theta_i(t) = \theta_{i-1}(t) + 2\pi/n$, $i = 2, \dots, n$. This way, all of the nodes are initially equiangular based on their enumeration. That is, the angle between nodes 1 and 2 is equal to the angle between nodes 2 and 3, and so on. Choosing only the first node’s initial angular coordinate randomly, instead of choosing all of the nodes’ initial angular coordinates randomly, somewhat alleviates the errors that may stem from random initialization, which curse many machine learning algorithms.

For iterations $t \geq 1$, a node’s angular coordinate must be selected in a way that is sensitive to both the node’s computed influence weight and the node’s distance to more influential nodes in the network. After iteration t of one of PageRank, HITS, and Pinski-Narin, the radial coordinate of node i is updated using the function $r_i(t) = e^{-x_i(t)}$. If node i becomes more influential after this iteration, its influence weight $x_i(t)$ increases accordingly, so its radial mapping value $r_i(t)$ decreases, and hence the node migrates towards the origin of the Poincaré disk to indicate its elevated importance in the network. Analogously, if node i becomes less influential after this

iteration, $x_i(t)$ decreases, and so $r_i(t)$ increases in value, and the node migrates away from the origin in the embedding.

As for the angular coordinate of node i , it is adjusted as follows. Fix a node i without loss of generality. Parse through node i 's neighbors and identify two quantities: the neighbor with the highest influence weight, called node k , and the neighbor with the smallest hyperbolic distance from node i , called node l . It should be emphasized that that the latter quantity is computed using the hyperbolic coordinates from the prior iteration. Given node i and its neighbors, which are represented via the set $N_i = \{1 \leq m \leq n : (i, m) \in E\}$, these two quantities are formally defined as follows.

$$k := \arg \max_{j \in N_i} x_j(t) \qquad l := \arg \min_{j \in N_i} d_{ij}$$

where d_{ij} is the hyperbolic distance between node i and node j , which recall is found using the formula

$$\cosh d_{ij} = \cosh(r_i(t-1)) \cosh(r_j(t-1)) - \sinh(r_i(t-1)) \sinh(r_j(t-1)) \cos(\theta_{ij}(t-1))$$

where $\theta_{ij}(t-1) = \pi - |\pi - |\theta_i(t-1) - \theta_j(t-1)||$ is the angular distance between nodes i and j , and $(r_i(t-1), \theta_i(t-1))$ and $(r_j(t-1), \theta_j(t-1))$ are the hyperbolic coordinates of nodes i and j , respectively. Recall that node i 's closest neighbor l is identified using the hyperbolic coordinates from the prior iteration.

With node i 's most influential neighbor k and node i 's closest neighbor l , define the value

$$d_{\text{new}} := \min\{d_{il}, \delta\} = \min \left\{ \min_{j \in N_i} d_{ij}, \delta \right\}$$

where $\delta > 0$ is an arbitrarily small positive number chosen by the user. After iteration t of one of PageRank, HITS, and Pinski-Narin, if node i 's influence weight is higher than that of all of node i 's neighbors (to be specific, $x_i(t) > x_j(t)$ for all $j \in N_i$, or simply $x_i(t) > x_k(t)$), do **not** update $\theta_i(t)$ and leave it with the same value as that of

the previous iteration. Otherwise, choose node i 's updated angular coordinate, $\theta_i(t)$, to be the angle such that the hyperbolic distance between node i and node k is equal to d_{new} . In other words, the value of $\theta_i(t) \in [0, 2\pi]$ where

$$\begin{aligned}\cosh d_{ik} &= \cosh(r_i(t-1)) \cosh(r_k(t-1)) - \sinh(r_i(t-1)) \sinh(r_k(t-1)) \cos(\theta_{ik}) \\ &= \cosh d_{\text{new}}\end{aligned}$$

is desired, if $\theta_{ik} = \pi - |\pi - |\theta_i(t) - \theta_k(t-1)||$ is the angular distance between nodes i and k . Note that all hyperbolic coordinates in the expression above are derived from the prior iteration, save for $\theta_i(t)$, which is the variable being solved for.

The reason why the δ value is included in the definition of d_{new} is to ensure the new hyperbolic distance between node i and its most influential neighbor k is small, in the case where d_{il} is not a small value. So if node i 's neighbors are all relatively far away, δ guarantees that node i migrates towards its most influential neighbor in the case where node i is not more influential than all of its neighbors.

Now the way to compute $\theta_i(t)$ is outlined as follows.

$$\begin{aligned}\cosh d_{\text{new}} &= \cosh(r_i(t-1)) \cosh(r_k(t-1)) - \sinh(r_i(t-1)) \sinh(r_k(t-1)) \cos(\theta_{ik}) \\ \sinh(r_i(t-1)) \sinh(r_k(t-1)) \cos(\theta_{ik}) &= \cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}} \\ \cos(\theta_{ik}) &= \frac{\cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}}}{\sinh(r_i(t-1)) \sinh(r_k(t-1))} \\ \theta_{ik} &= \cos^{-1} \left(\frac{\cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}}}{\sinh(r_i(t-1)) \sinh(r_k(t-1))} \right)\end{aligned}$$

Now $\theta_{ik} = \pi - |\pi - |\theta_i(t) - \theta_k(t-1)||$ is substituted into the left-hand side, giving

$$\begin{aligned}\pi - |\pi - |\theta_i(t) - \theta_k(t-1)|| &= \cos^{-1} \left(\frac{\cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}}}{\sinh(r_i(t-1)) \sinh(r_k(t-1))} \right) \\ |\pi - |\theta_i(t) - \theta_k(t-1)|| &= \pi - \cos^{-1} \left(\frac{\cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}}}{\sinh(r_i(t-1)) \sinh(r_k(t-1))} \right)\end{aligned}$$

For convenience, set

$$(*) = \frac{\cosh(r_i(t-1)) \cosh(r_k(t-1)) - \cosh d_{\text{new}}}{\sinh(r_i(t-1)) \sinh(r_k(t-1))}.$$

Then there are multiple ways to solve for $\theta_i(t)$, as seen below.

1.

$$\pi - |\theta_i(t) - \theta_k(t - 1)| = \pi - \cos^{-1}(*)$$

$$|\theta_i(t) - \theta_k(t - 1)| = \cos^{-1}(*)$$

There are two ways to solve for $\theta_i(t)$ from here.

(a)

$$\theta_i(t) - \theta_k(t - 1) = \cos^{-1}(*)$$

$$\theta_i(t) = \theta_k(t - 1) + \cos^{-1}(*) \quad (1a)$$

(b)

$$\theta_i(t) - \theta_k(t - 1) = -\cos^{-1}(*)$$

$$\theta_i(t) = \theta_k(t - 1) - \cos^{-1}(*) \quad (1b)$$

2.

$$\pi - |\theta_i(t) - \theta_k(t - 1)| = \cos^{-1}(*) - \pi$$

$$|\theta_i(t) - \theta_k(t - 1)| = 2\pi - \cos^{-1}(*)$$

There are another two ways to solve for $\theta_i(t)$ from here.

(a)

$$\theta_i(t) - \theta_k(t - 1) = 2\pi - \cos^{-1}(*)$$

$$\theta_i(t) = 2\pi - \cos^{-1}(*) + \theta_k(t - 1) \quad (2a)$$

(b)

$$\theta_i(t) - \theta_k(t - 1) = \cos^{-1}(*) - 2\pi$$

$$\theta_i(t) = \cos^{-1}(*) - 2\pi + \theta_k(t - 1) \quad (2b)$$

Define the set

$$\Theta := \{(1a), (1b), (2a), (2b)\}$$

to be used later. Now there are four possible values for $\theta_i(t)$. Which should be chosen to be node i 's angular coordinate? The first constraint is that $\theta_i(t) \in [0, 2\pi]$. Note that since node i 's most influential neighbor k 's hyperbolic coordinates, $(r_k(t-1), \theta_k(t-1))$, are fixed, there are two possible ways to shift node i : to the left or to the right side of node k . The second constraint, then, is that the value of $\theta_i(t)$ that is closer (in terms of hyperbolic distance) to $\theta_i(t-1)$ using $r_i(t-1)$ is chosen. The latter constraint is particularly important because this would mean the shift node i would have to make to its new position in the Poincaré disk after iteration t is more intuitive regarding the behavior of a node's migration towards more influential nodes. Imagine that a given network represents a social network of people, where each person, represented by a node, is a social climber: they crave to be near popular and powerful people, and also desire to be popular and powerful themselves. If each node in the network was sentient in the most primitive, greediest sense, a node is expected to make the shortest trip possible to be closer to its most influential neighbor. Unless, of course, node i is more influential than all of its neighbors, in which case node i stays put and does not move at all angular-wise.

Recall the definition of the set Θ above. If node i is not more influential than all of its neighbors, for the coordinate $\theta_i(t)$, choose the element $\theta \in \Theta$ that lies in the interval $[0, 2\pi]$ and minimizes the hyperbolic distance between $(r_i(t-1), \theta_i(t-1))$ and $(r_i(t-1), \theta)$:

$$\arg \min_{\substack{\theta \in \Theta \\ \theta \in [0, 2\pi]}} \cosh^{-1} (\cosh(r_i(t-1)) \cosh(r_i(t-1)) - \sinh(r_i(t-1)) \sinh(r_i(t-1)) \cos(\pi - |\pi - \theta - \theta_i(t-1)|))$$

For convenience, set the aforementioned expression equal to (**). Thus, the formal definition of node i 's angular coordinate after iteration $t \geq 1$ of one of PageRank,

HITS, and Pinski-Narin, $\theta_i(t)$, is as follows.

$$\theta_i(t) := \begin{cases} \theta_i(t-1) & \text{if } x_i(t) > x_k(t) \\ (**) & \text{else} \end{cases} \quad 1 \leq i \leq n.$$

The idea behind this definition of $\theta_i(t)$ is to move node i closer to its most influential neighbor, if node i itself is not the most influential node out of all of its neighbors. Furthermore, the hyperbolic distance between node i and its most influential neighbor k should be at least as small as that between node i and its closest neighbor l , in the case where node i is not more influential than all of its neighbors.

Ideally, as the nodes' influence weights approach their "steady state" values as the number of iterations increases, the objective of a given node's journey until convergence is twofold: migrate towards the origin and migrate towards influential nodes. The hope is that this greedy behavior from the nodes (if the analogy of the network of social climbers from earlier is recalled) not only echos the greedy behavior of nodes in a network in studies like Boguna *et al.* (2010), Boyd *et al.* (2006), and Kleinberg (2007), but also highlights any clusters in the embeddings themselves. By the time a given algorithm has converged, the nodes have "settled" into their respective cliques, with the most popular and powerful nodes at or near the center of said cliques.

To conclude the definition of the hyperbolic coordinate mappings, the functionality of the two mappings can be summarized by viewing the radial coordinate r as a reflection of a node's influence, and the angular coordinate θ as a reflection of a node's proximity to more influential nodes in the network. The three eigenvector centrality algorithms implemented here all attempt to include both influence and proximity to influential nodes in their respective influence weighting schemes. Likewise, both of these quantities are reflected in the hyperbolic embeddings. At last, the formal definition of the hyperbolic embedding can be outlined.

Say a complex network consisting of n nodes, enumerated from 1 to n , is given. Let $\epsilon > 0$ and $\delta > 0$ be arbitrary but fixed. The procedure to produce hyperbolic embeddings of the network for some algorithm until convergence is as follows.

- Initialization phase ($t = 0$): All of the nodes are initialized with centrality score $x_i(0)$, $1 \leq i \leq n$, which varies for PageRank, HITS, and Pinski-Narin. In the case of HITS, each node is initialized with an authority weight $x_i(0) = 1$ as well as a hub weight $y_i(0) = 1$. Compute the initial hyperbolic coordinates $(r_i(0), \theta_i(0))$ for node i as follows.

1. Radial coordinate r_i : $r_i(0) = e^{-x_i(0)}$, $1 \leq i \leq n$.

2. Angular coordinate θ_i :

$$\theta_i(0) := \begin{cases} \text{uniform random value in } [0, 2\pi] & \text{if } i = 1 \\ \theta_{i-1}(0) + 2\pi/n & \text{if } 2 \leq i \leq n \end{cases}.$$

- $t \geq 1$: Compute iteration t of the given algorithm. Update node i 's hyperbolic coordinates as follows.

1. Radial coordinate r_i : $r_i(t) = e^{-x_i(t)}$, where $x_i(t)$ is the influence weight of node i after iteration t , $1 \leq i \leq n$.

2. Angular coordinate θ_i : If N_i is the set of node i 's neighbors, identify i 's most influential neighbor k and closest neighbor (hyperbolic distance-wise) l ,

$$k := \arg \max_{j \in N_i} x_j(t) \qquad l := \arg \min_{j \in N_i} d_{ij}$$

where d_{ij} is the hyperbolic distance between node i and node j , computed via the formula

$$\cosh d_{ij} = \cosh(r_i(t-1)) \cosh(r_j(t-1)) - \sinh(r_i(t-1)) \sinh(r_j(t-1)) \cos(\theta_{ij}(t-1))$$

where $\theta_{ij}(t-1) = \pi - |\pi - |\theta_i(t-1) - \theta_j(t-1)||$ is the angular distance between nodes i and j , and $(r_i(t-1), \theta_i(t-1))$ and $(r_j(t-1), \theta_j(t-1))$ are the hyperbolic coordinates of nodes i and j after iteration $t-1$, respectively.

Compute

$$d_{\text{new}} := \min\{d_{il}, \delta\}.$$

Compute (1a), (1b), (2a), and (2b) and define the set

$$\Theta := \{(1a), (1b), (2a), (2b)\}.$$

Set

$$\theta_i(t) := \begin{cases} \theta_i(t-1) & \text{if } x_i(t) > x_k(t) \\ (**) & \text{else} \end{cases} \quad 1 \leq i \leq n$$

where

$$(**) = \arg \min_{\substack{\theta \in \Theta \\ \theta \in [0, 2\pi]}} \cosh^{-1}(\cosh(r_i(t-1)) \cosh(r_i(t-1)) - \sinh(r_i(t-1)) \sinh(r_i(t-1)) \cos(\pi - |\pi - |\theta - \theta_i(t-1)||)).$$

- Repeat the last step until $|x_i(t) - x_i(t-1)| < \epsilon$ for all $i = 1, \dots, n$. In the case of HITS, repeat until both $|x_i(t) - x_i(t-1)| < \epsilon$ and $|y_i(t) - y_i(t-1)| < \epsilon$ for all $i = 1, \dots, n$.

RESULTS AND OBSERVATIONS

The graph used in the experimentation is the following Figure 4.1.

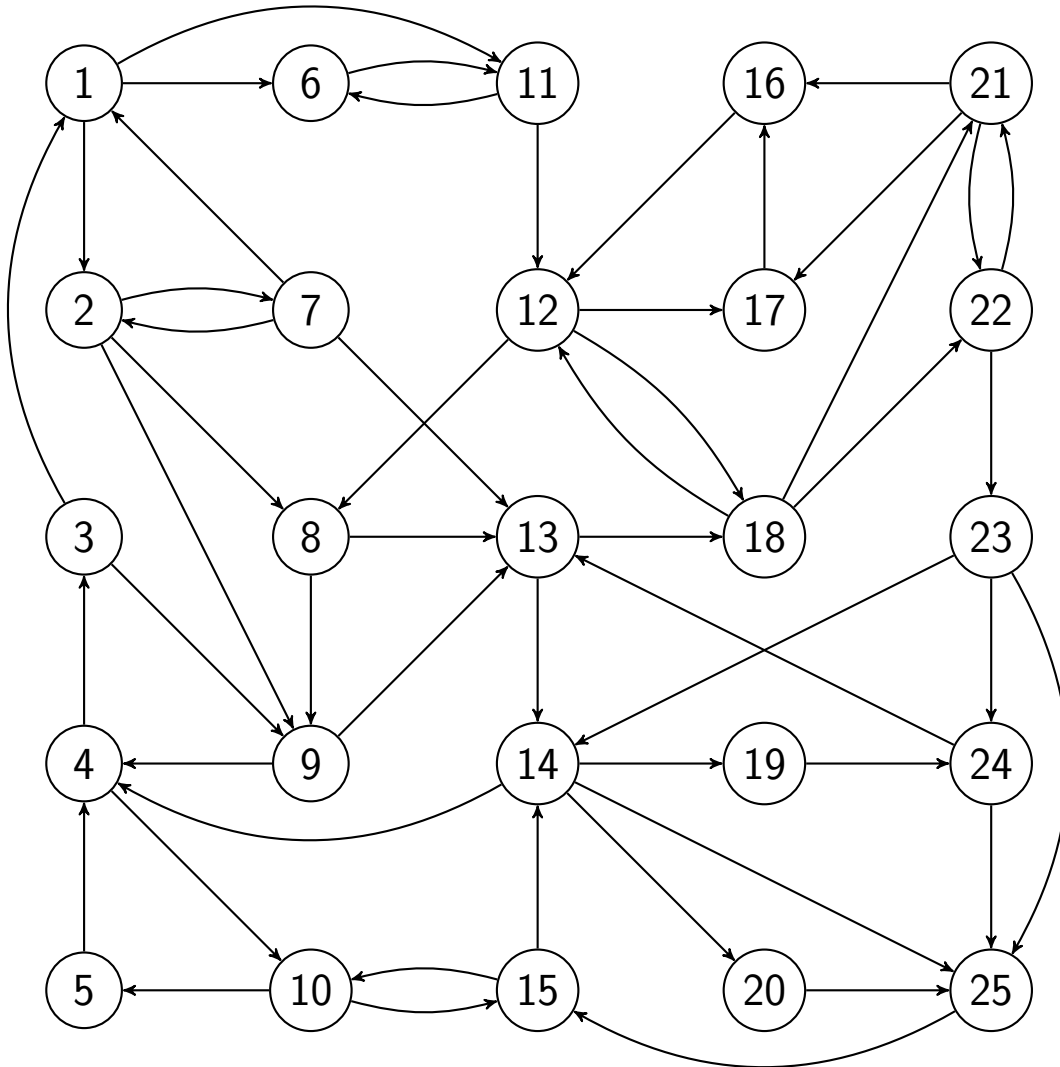


Figure 4.1: Graph Used in Experimentation

This graph comprises 25 nodes and 52 edges. The procedure outlined in the previous section was applied with $\epsilon = 0.001$ and $\delta = 0.1$ to this graph's adjacency matrix for

PageRank, HITS, and Pinski-Narin, producing a series of hyperbolic embeddings for all three algorithms. The results are as follows.

PageRank initializes all the nodes' influence weights with $1/n$, where n is the number of nodes. In this case, the 25 nodes are each initialized with the weight $1/25$. Since PageRank normalizes the nodes' weights so they always sum to one for any given iteration, the collective weights can be viewed as a probability distribution. This means initially, each of the 25 nodes has a $1/25$ chance of being visited by the actor in the random surfer model. In subsequent iterations, each node's weight reflects the node's influence, but can also be viewed as the probability the node is visited by the random surfer. The more influential a node is (so in the embedding, the closer they are to the origin of the Poincaré disk), the higher their chance is of being chosen.

HITS initializes the nodes' authority and hub weights with one. For all subsequent iterations, the authority weights in HITS are normalized so their squares sum to one, as are the hub weights, so the weights cannot be viewed as a probability distribution as with PageRank. Pinski-Narin also initializes the nodes' influence weights with one, but its normalization method is markedly different from those of the other two algorithms: Pinski-Narin fixes the condition that the size-weighted average of the weights is one, as explained in Section 2. One may notice by now that out of the three algorithms, Pinski-Narin is the only algorithm that permits a node's weight to exceed one. This distinction is a driving factor behind the choice of radial mapping $r_i(t) = e^{-x_i(t)}$. Nodes in Pinski-Narin can get arbitrarily close to the origin of the Poincaré disk, because $\lim_{x \rightarrow \infty} e^{-x} = 0$ (as can be seen in Figure 3.1), while the nodes in PageRank and HITS can never exceed a weight of one, and so the nodes can never have a radial coordinate smaller than $e^{-1} \approx 0.3679$. This discrepancy means nodes in embeddings for PageRank and HITS typically never get very close to

the origin at all. This distance is exacerbated when the given network is very large, and so each node inevitably ends up with a very small value for its weight due to the normalization methods in PageRank and HITS. To clarify, “very small value” means when a node’s weight is considered as a numeric value in general. A more influential node has a larger weight compared to other nodes in the network, but if the network is very large and dense, said power player’s weight is still a mere drop in the bucket. Lastly, note that although PageRank, HITS, and Pinski-Narin have very different ways of treating the nodes’ influence weights, the definition of the proposed embedding permits comparative analysis of the three algorithms.

Now that the differences in how PageRank, HITS, and Pinski-Narin initialize and normalize the nodes’ influence weights have been discussed, these differences can be viewed in the physical embeddings. All of the following embeddings have arrows between nodes to denote outgoing edges, and are not related with hyperbolic distance quantities in any way. The following image in Figure 4.2 is an initial embedding for PageRank on the graph in Figure 4.1. It is known that the 25 nodes are each initialized with the weight $1/25$, so each node’s initial radial coordinate is $r_i(0) = e^{-1/25} \approx 0.9608$, $i = 1, \dots, n$, mapping each node relatively close to the border of the Poincaré disk. As per the embedding definition, node 1’s initial angular coordinate is chosen uniformly at random from the interval $[0, 2\pi]$, while every other node i ’s angular coordinate is $\theta_i(0) = \theta_{i-1}(0) + 2\pi/25$, $i = 2, \dots, n$. As such, the nodes are all initially equiangular by their enumeration. The angle between nodes 1 and 2 is identical to the angle between nodes 2 and 3, which is also identical to the angle between nodes 3 and 4, and so on.

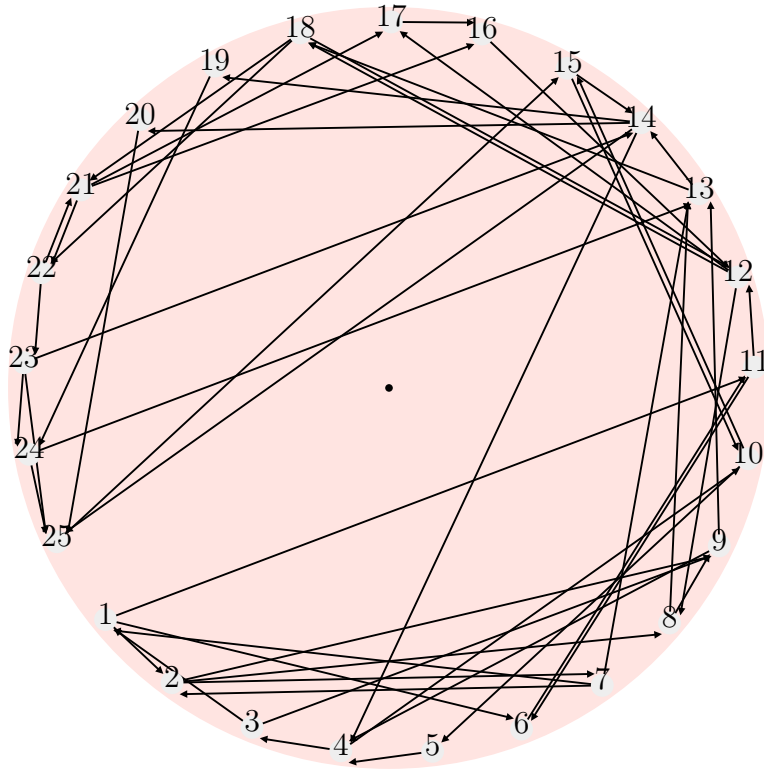


Figure 4.2: Initial PageRank Poincaré Embedding

The following images Figure 4.3 and Figure 4.4 are initial embeddings for HITS on the same graph. HITS initializes the authority weights and hub weights for all nodes to one, so each node has an initial radial coordinate of $r_i(0) = e^{-1} \approx 0.3679$, $i = 1, \dots, n$, placing the nodes considerably closer to the origin compared to the initial PageRank embedding. Because node 1's initial angular coordinate is chosen uniformly at random from $[0, 2\pi]$, the orientation of node 1 (as well as that of the remaining 24 nodes) may vary for the authority weight and hub weight embeddings, as well as for different executions of HITS.

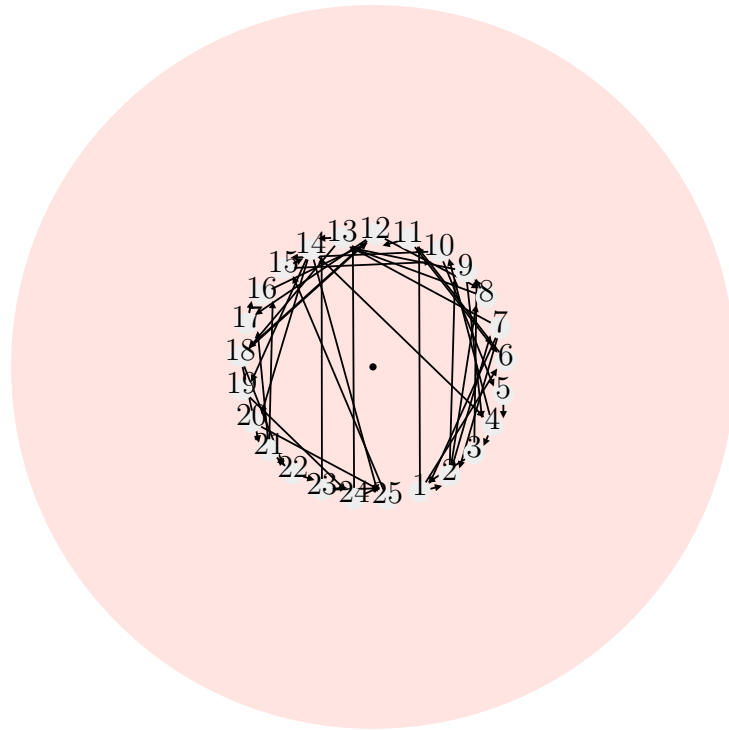


Figure 4.3: Initial HITS Authority Weight Poincaré Embedding

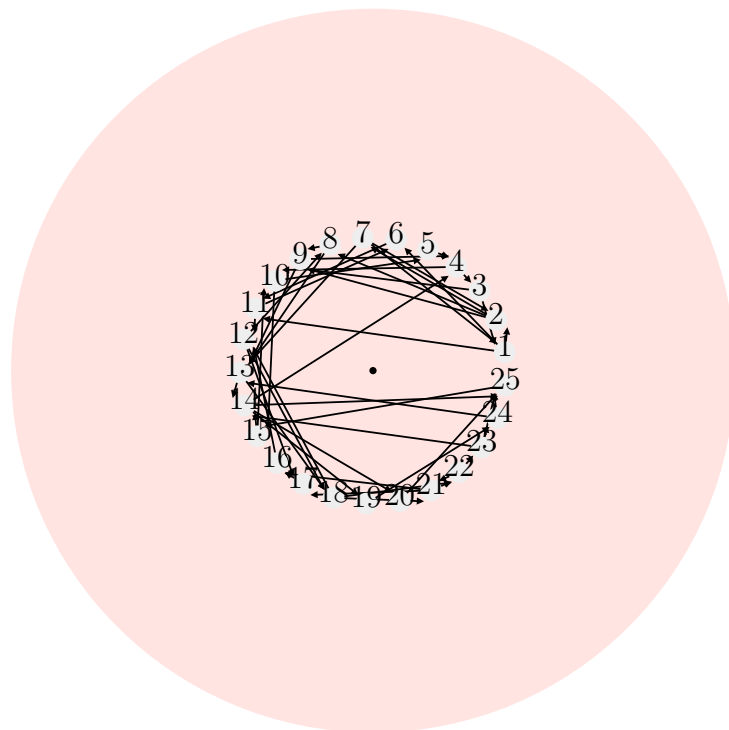


Figure 4.4: Initial HITS Hub Weight Poincaré Embedding

The image following the initial HITS embeddings, Figure 4.3 and Figure 4.4, is Figure 4.5, an initial embedding for Pinski-Narin on the same graph.

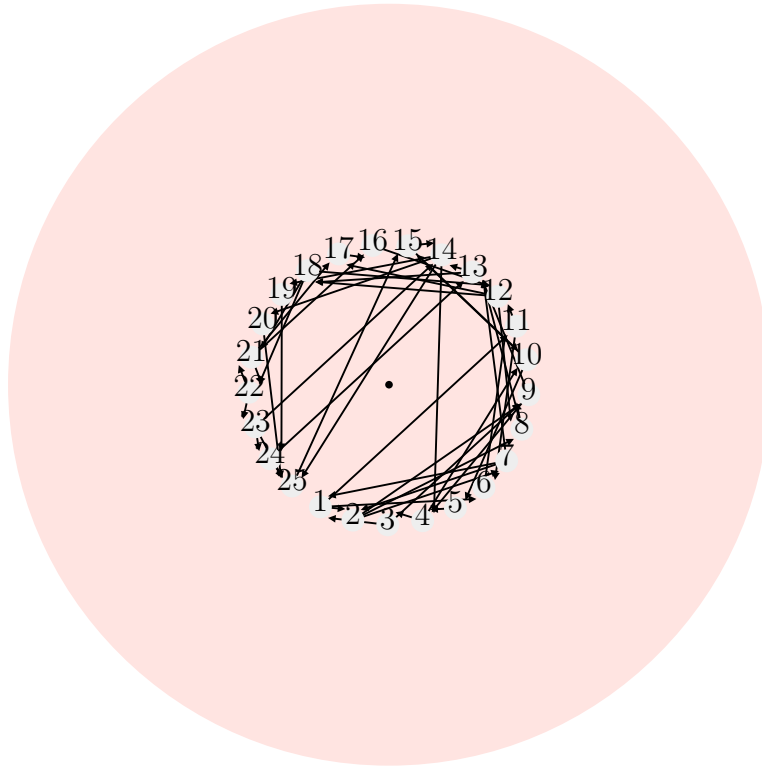


Figure 4.5: Initial Pinski-Narin Poincaré Embedding

Pinski-Narin’s initialization procedure is identical to that of HITS. Like HITS, all of the nodes’ initial radial coordinate is $r_i(0) = e^{-1} \approx 0.3679$, $i = 1, \dots, n$. Like HITS, because node 1’s angular coordinate is chosen uniformly at random from $[0, 2\pi]$, the orientation of node 1 may vary for different executions of the algorithm. Figures 4.2-4.5 all illustrate what was already explained in words earlier: that PageRank initializes the nodes’ weights differently than HITS and Pinski-Narin. It should be clarified that the black dot in the center of all of the embeddings denotes the origin $(0, 0)$ of the Poincaré disk.

Now the embedding images for the first-order weights computed by all three algorithms are provided. The following Figure 4.6 is the embedding for PageRank after one iteration.

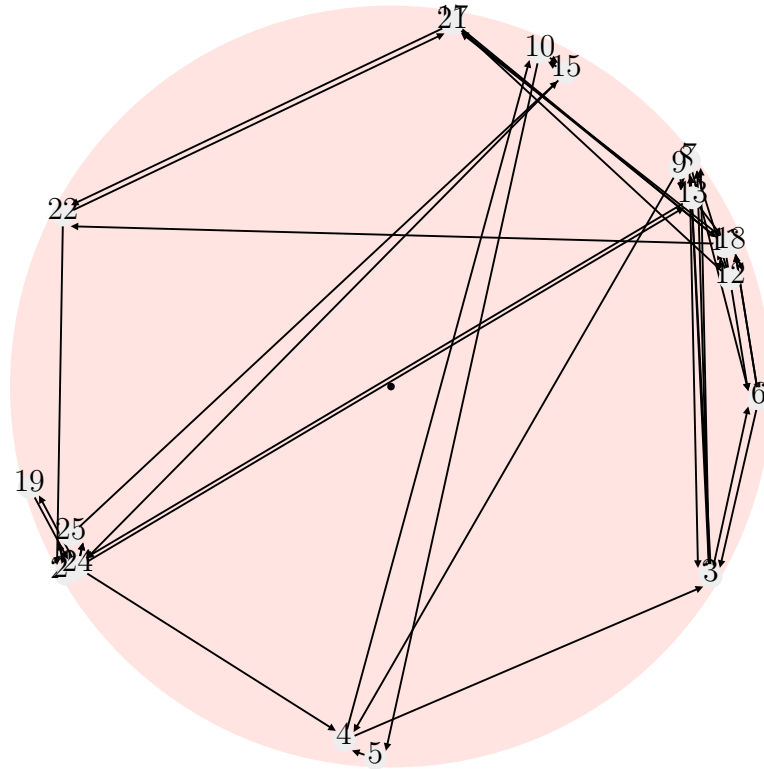


Figure 4.6: PageRank Poincaré Embedding After Iteration $t = 1$

Already a dramatic shift in the positions of the nodes is visible, but the new positions of the nodes are not erratic or random at all. They are only reflecting the changes in the calculated influence weights after one iteration. Clustering behavior based on the new computed weights is also now evident. All of the nodes were initialized with weight $1/25$, but after one iteration of PageRank the weights are now as follows in Table 4.1. The top five most influential nodes, in non-increasing order, are nodes 25, 12, 13, 4, and 15. Nodes 12 and 13 in particular are equally influential. Node 25's new position and the positions of its neighbors are specifically discussed.

Node	Weight	Node	Weight
1	0.03433333333333334	14	0.05133333333333333
2	0.02866666666666667	15	0.05699999999999995
3	0.023000000000000003	16	0.05133333333333333
4	0.06549999999999999	17	0.02866666666666667
5	0.023000000000000003	18	0.03433333333333334
6	0.03433333333333334	19	0.014500000000000002
7	0.01733333333333333	20	0.014500000000000002
8	0.02866666666666667	21	0.03433333333333334
9	0.05133333333333333	22	0.02866666666666667
10	0.04	23	0.023000000000000003
11	0.05133333333333333	24	0.05133333333333333
12	0.06833333333333333	25	0.07683333333333335
13	0.06833333333333333		

Table 4.1: PageRank Weights After Iteration $t = 1$

Node 25's (rectangular) coordinates changed very little after one iteration: $(-0.8718796578751445, -0.403661006997402)$ initially and then $(-0.8403496657973324, -0.38906331769728947)$ afterwards. This slight shift is easily explained by the fact that node 25's weight increased from $1/25 = 0.04$ to 0.07683333333333335 , so node 25's radial coordinate decreased and hence node 25 migrated closer to the origin. Furthermore, node 25's only outgoing neighbor is node 15, which is clearly not more influential than node 25 based on Table 4.1, so node 25 has no reason to migrate towards another node. Node 25 has incoming edges from four other nodes: nodes 14, 20, 23, and 24. Nodes 14 and 24 increased in influence, while nodes 20 and 23 decreased in influence. Regardless of

the changes, none of the four nodes are at this point more influential than node 25, so they gravitate towards their more influential neighbor node 25. Node 14's initial rectangular coordinates were $(0.6620556767794402, 0.6962748216263124)$, and are now $(-0.8228329057755754, -0.47473544397836376)$. Node 20's initial rectangular coordinates were $(-0.6533300624123761, 0.7044687189186446)$, and are now $(-0.8537056675036964, -0.4925475588597237)$. Node 23's initial rectangular coordinates were $(-0.958499141602944, 0.06630039165084191)$, and are now $(-0.8464799222521894, -0.4883786475826334)$. Node 24's initial rectangular coordinates were $(-0.9448743654314925, -0.17415160044360792)$, and are now $(-0.8228329057755754, -0.47473544397836376)$.

Node 14 in particular made quite the pilgrimage, migrating from nearly the opposite end of the Poincaré disk as seen in the initial PageRank embedding Figure 4.2. If the embeddings for PageRank reflect the actions of greedy social climbers in a social network, the first-order embedding Figure 4.6 above would be showing that node 14 made the trek purely to get closer to its more popular and powerful neighbor node 25. The journeys made by nodes 14, 20, 23, and 24 to get closer to node 25 after one iteration form a strong cluster in the lower left-hand side of the Poincaré disk. It is visible from Figure 4.6 that node 19 migrated towards node 24, its most influential (and only) neighbor after one iteration of PageRank. The migration behavior seen in the nodes thus far is what was originally hoped for when creating the embedding.

Figures 4.7 and 4.8 below are the embeddings for the authority and hub weights after one iteration of HITS, respectively. Like the first-order PageRank embedding Figure 4.6, many of the nodes have shifted greatly compared to the initial HITS embeddings Figures 4.3 and 4.4. Both the authority weights and hub weights were initialized to one, but after one iteration Tables 4.2 and 4.3 below show they have decreased, not because the nodes became less influential (which would be the case for

PageRank), but because of the normalization scheme implemented by HITS, which means the authority and hub weights cannot be interpreted as a probability distribution as with PageRank. Recall that HITS normalizes the weights by having the squares of the authority weights sum to one, and the squares of the hub weights sum to one. Since all of the nodes' authority and hub weights decreased, Figures 4.7 and 4.8 above show that all of the nodes have migrated farther away from the origin. However, only one iteration of HITS has been executed, and as HITS is executed until convergence, eventually the nodes “settle” into their final authority and hub weights.

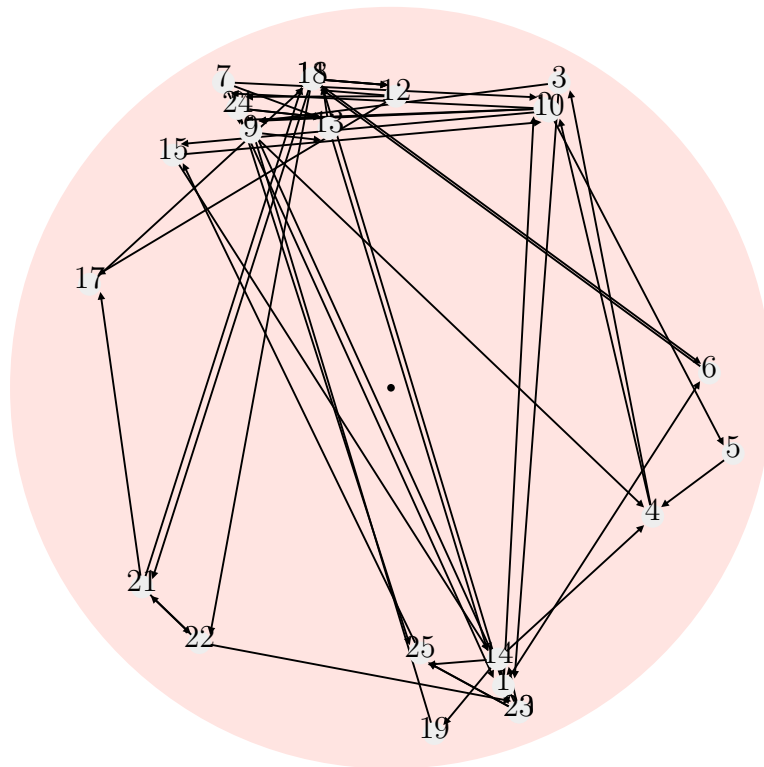


Figure 4.7: HITS Authority Weight Poincaré Embedding After Iteration $t = 1$

First examine the change in the authority weights and how they are reflected in Figure 4.7. Based on Table 4.2 nodes 13 and 25 are tied for most influential authority node, while nodes 4, 9, 12, and 14 are tied for second place. Figure

Node	Weight	Node	Weight
1	0.1781741612749496	14	0.2672612419124244
2	0.1781741612749496	15	0.1781741612749496
3	0.0890870806374748	16	0.1781741612749496
4	0.2672612419124244	17	0.1781741612749496
5	0.0890870806374748	18	0.1781741612749496
6	0.1781741612749496	19	0.0890870806374748
7	0.0890870806374748	20	0.0890870806374748
8	0.1781741612749496	21	0.1781741612749496
9	0.2672612419124244	22	0.1781741612749496
10	0.1781741612749496	23	0.0890870806374748
11	0.1781741612749496	24	0.1781741612749496
12	0.2672612419124244	25	0.3563483225498992
13	0.3563483225498992		

Table 4.2: HITS Authority Weights After Iteration $t = 1$

4.7 exhibits clustering behavior, but is noticeably different from that in the first-order PageRank embedding Figure 4.6. To highlight this distinction, again examine node 25. Node 25’s rectangular coordinates in the initial HITS authority weight embedding were $(0.04008763440535052, -0.3656887540031764)$, but after one iteration of HITS are now $(0.07630356001007793, -0.6960588770079115)$. Despite this migration away from the origin, when node 25’s only outgoing neighbor node 15 is examined, node 15 is not more influential than node 25, so node 25 does not migrate towards node 15, and hence it is seen in Figure 4.7 that node 25 has only migrated outwards to reflect its “reduced” influence. Examine the change in the coordinates for node 25’s incoming neighbors: nodes 14, 20, 23, and 24. Node 14’s coor-

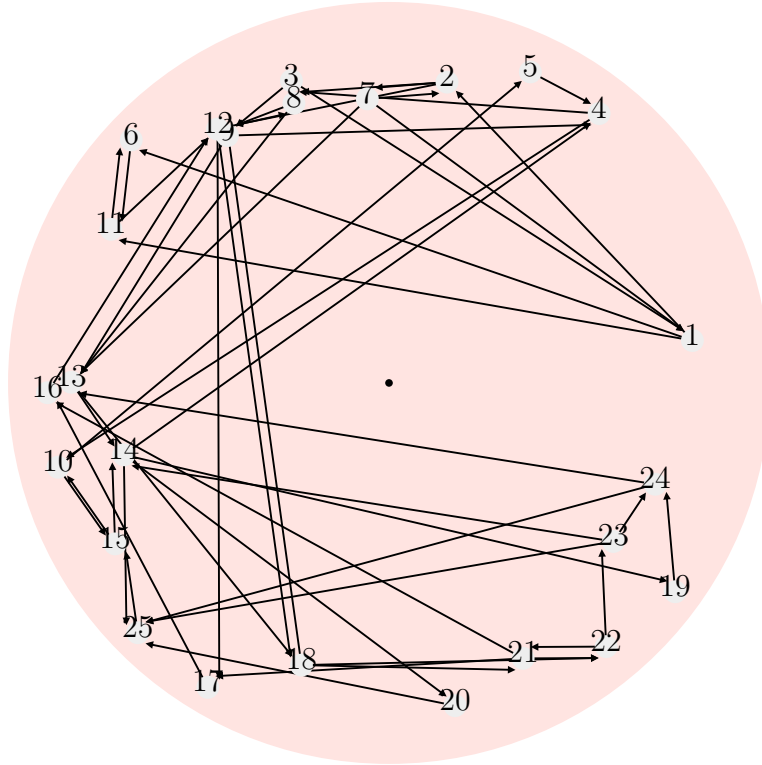


Figure 4.8: HITS Hub Weight Poincaré Embedding After Iteration $t = 1$

Coordinates in the initial embedding were $(-0.1718915488342151, 0.3252515621422707)$, but are now $(0.2809977590291964, -0.7120317990848298)$. Node 20's coordinates in the initial embedding were $(-0.3354029121350334, -0.15112964556284691)$, but are now $(0.33580171626085464, -0.8509018043099291)$. Node 23's coordinates in the initial embedding were $(-0.14104283900214393, -0.3397678631107245)$, but are now $(0.33580171626085464, -0.8509018043099291)$. Node 24's coordinates in the initial embedding were $(-0.05211488731600311, -0.3641693586185595)$, but are now $(-0.40231413396498206, 0.7337383797558849)$.

Compared to the first-order PageRank embedding, three of node 25's incoming neighbors, nodes 14, 20, and 23, have migrated towards node 25, but node 25's other incoming neighbor, node 24, has instead migrated towards its neighbor node 13, despite nodes 13 and 25 having the same authority weight. This is because when a

Node	Weight	Node	Weight
1	0.21792981368764439	14	0.32689472053146656
2	0.21792981368764439	15	0.181608178073037
3	0.181608178073037	16	0.10896490684382219
4	0.10896490684382219	17	0.0726432712292148
5	0.10896490684382219	18	0.2542514493022518
6	0.0726432712292148	19	0.0726432712292148
7	0.2905730849168592	20	0.1452865424584296
8	0.2542514493022518	21	0.21792981368764439
9	0.2542514493022518	22	0.10896490684382219
10	0.10896490684382219	23	0.32689472053146656
11	0.181608178073037	24	0.2905730849168592
12	0.21792981368764439	25	0.0726432712292148
13	0.181608178073037		

Table 4.3: HITS Hub Weights After Iteration $t = 1$

given node’s most influential neighbor is computed, the nodes are iterated through in increasing order $1, \dots, 25$, so node 24 gravitated towards node 13 over node 25. This may seem like an error of negligence, but only one iteration of HITS has been executed thus far, and until the authority weights converge the nodes continue shifting.

Now discuss the change in the hub weights and how they are reflected in Figure 4.8. From Table 4.3 nodes 14 and 23 are tied for most influential hub node, while nodes 7 and 24 are tied for second place. Nodes 8, 9, and 18 are tied for third place. Compared to the first-order HITS authority weight embedding just discussed, there is noticeably weaker clustering for the hub weights. It is seen that nodes 14 and 23, the most influential hub nodes after one iteration, occupy opposite ends of the

Poincaré disk, and seem to have each loosely attracted a number of nodes, but not to the point where strong cluster membership can be pinpointed. Note that there is a directed edge from node 23 to node 14, but since node 14’s weight is tied with node 23’s weight, there is no reason for node 23 to migrate towards node 14.

Figure 4.9 directly below depicts the embedding for Pinski-Narin after one iteration, and Table 4.4 shows the computed weights after one iteration of Pinski-Narin. Compared to the other first-order embeddings encountered thus far, the nodes are significantly closer to the origin. It is also much more difficult to single out clusters in Figure 4.9. Node 25 is the most influential node by far, so much so that node 25 is covering the black dot denoting the origin of the Poincaré disk in Figure 4.9 above. This is fitting, as $x_{25}(1) = 4$, so node 25’s radial coordinate is $r_{25}(1) = e^{-x_{25}(1)} = e^{-4} \approx 0.0183$. Node 23’s position in this embedding should be clarified. Node 23’s weight is tied for the lowest with node 7, so node 23 is placed relatively far away from the origin. Node 23 has outgoing edges to two nodes: node 14 and node 25. Since node 25 is more influential, node 23 migrates as close as possible to node 25. However, in the embedding, it appears node 23 is closer to node 14 than node 25! This is because node 14 also has an outgoing edge to node 25, and so node 14 has migrated as close as possible to node 25. Both nodes 14 and 23 have migrated as close to node 25 as their radial coordinates allow, which in turn projects the appearance that node 23 has chosen to gravitate towards node 14. But only one iteration of Pinski-Narin has been examined at this point. Until the Pinski-Narin algorithm converges, the nodes shift greatly until they “settle”.

Recall that Pinski-Narin is the only algorithm of the three being worked with that permits a node’s weight to exceed one. The first-order weights for Pinski-Narin especially mark its departure from PageRank and HITS in this regard: the fraction consisting of the number of incoming edges over the number of outgoing edges. In

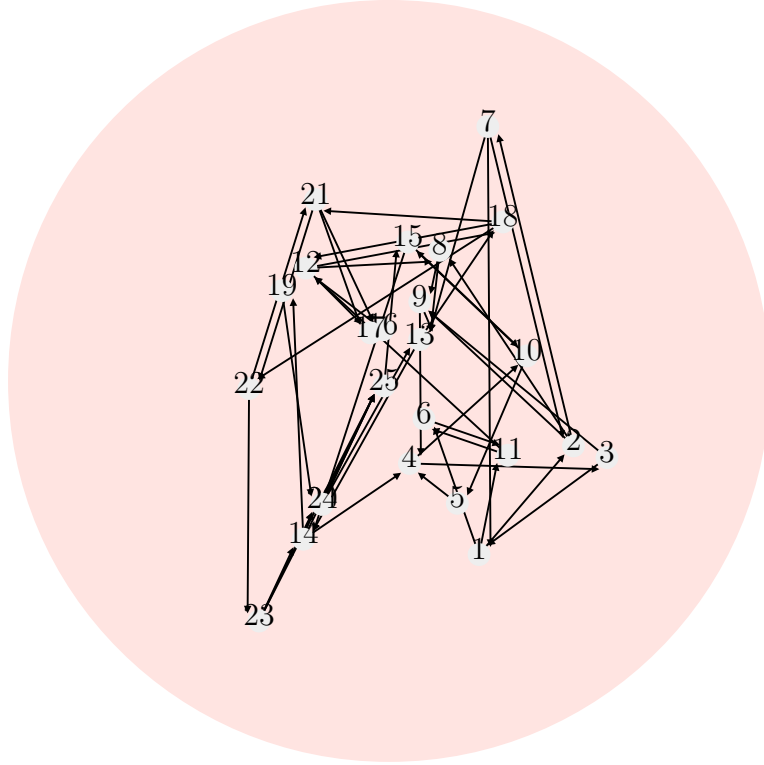


Figure 4.9: Pinski-Narin Poincaré Embedding After Iteration $t = 1$

a citation network, a node's first-order weight can be viewed as the ratio between the number of citations received versus the number of references given. For instance, if a node has 1 incoming edge and 5 outgoing edges, then said node's first-order influence weight according to Pinski-Narin is $1/5$. If a node has 10 incoming edges and 2 outgoing edges, then said node's first-order weight is $10/2$. A node can have an influence weight of 0 in Pinski Narin, which occurs when a node has no incoming edges. So if some node called i has 7 outgoing edges and no incoming edges, said node's first-order weight is $x_i(1) = 0/7 = 0$, and one would say node i has absolutely no influence in the network. In such a case, the node in question would have a first-order radial coordinate of $r_i(1) = e^{-x_i(1)} = e^0 = 1$, which would map node i to the border of the Poincaré disk. However, recall from earlier that it is assumed all of the graphs worked with are strongly connected. Then each node is guaranteed to have

Node	Weight	Node	Weight
1	0.6666666666666666	14	0.75
2	0.6666666666666666	15	1.0
3	0.5	16	2.0
4	1.5	17	2.0
5	1.0	18	0.6666666666666666
6	2.0	19	1.0
7	0.3333333333333333	20	1.0
8	1.0	21	0.6666666666666666
9	1.5	22	1.0
10	1.0	23	0.3333333333333333
11	1.0	24	1.0
12	1.0	25	4.0
13	2.0		

Table 4.4: Pinski-Narin Weights After Iteration $t = 1$

an incoming edge as well as an outgoing edge, and so a node is guaranteed to have a nonzero weight, not only in Pinski-Narin, but in any of the three algorithms being worked with.

Of the three algorithms, PageRank required the fewest number of iterations until convergence: 8 iterations total. HITS converged after 17 iterations, while Pinski-Narin converged after 28 iterations. Interestingly, the three algorithms differed in the top five influential nodes they each identified, which are as follows.

PageRank			Pinski-Narin		
Order	Node	Weight	Order	Node	Weight
1	15	0.08129473823704451	1	25	2.7831786498186375
2	12	0.07540667108305212	2	15	2.573060827549414
3	10	0.06883967017942033	3	10	2.362637537084204
4	14	0.06778651427574965	4	5	2.3625832962390425
5	4	0.06634011281712254	5	16	2.2205277043232896
HITS authorities			HITS hubs		
Order	Node	Weight	Order	Node	Weight
1	25	0.5676995100014323	1	14	0.505372124891322
2	13	0.5165609851460752	2	24	0.41269543983013635
3	4	0.37578080799433755	3	23	0.35858963111958697
4	14	0.2149641603324868	4	9	0.3396465982547808
5	9	0.21152366028751235	5	7	0.3270669931493054

Table 4.5: Five Most Influential Nodes Identified by PageRank, HITS, and Pinski-Narin

For the rest of this section, the remaining embeddings until convergence for PageRank are provided, and then only the final embeddings for HITS and Pinski-Narin are listed.

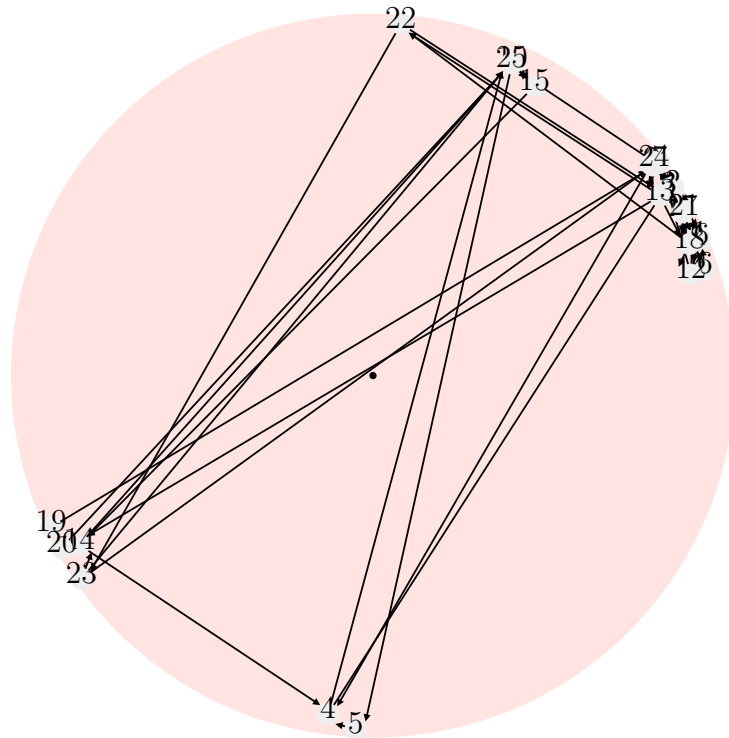


Figure 4.10: PageRank Poincaré Embedding After Iteration $t = 2$

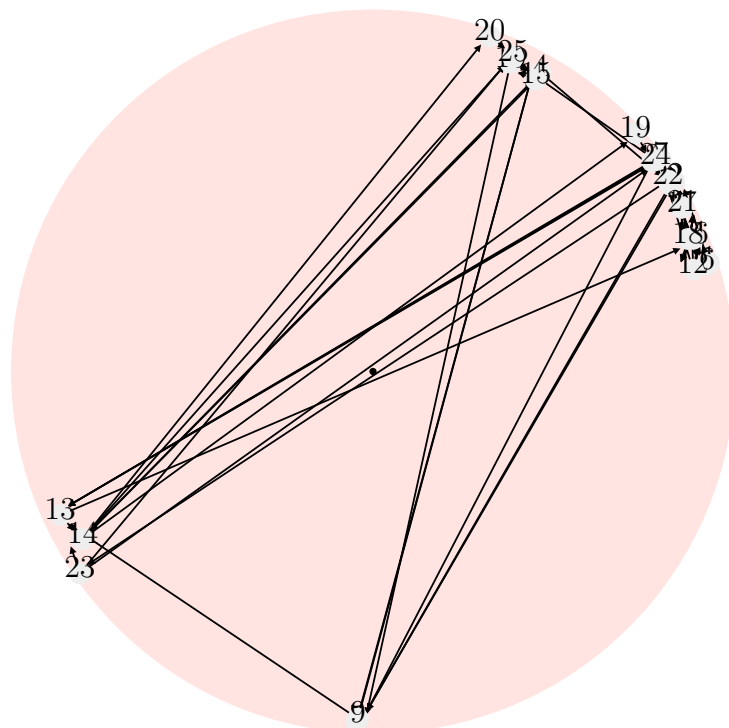


Figure 4.11: PageRank Poincaré Embedding After Iteration $t = 3$

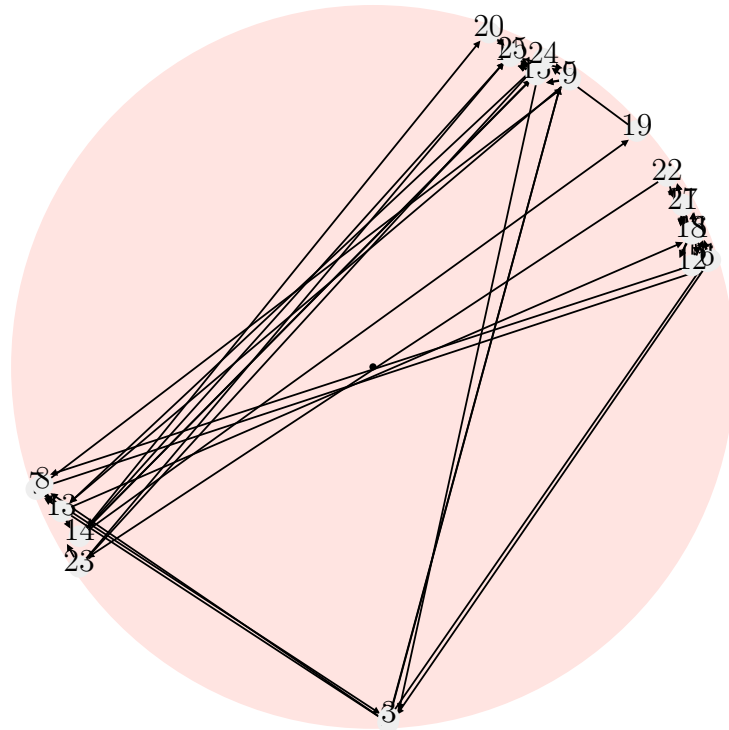


Figure 4.12: PageRank Poincaré Embedding After Iteration $t = 4$

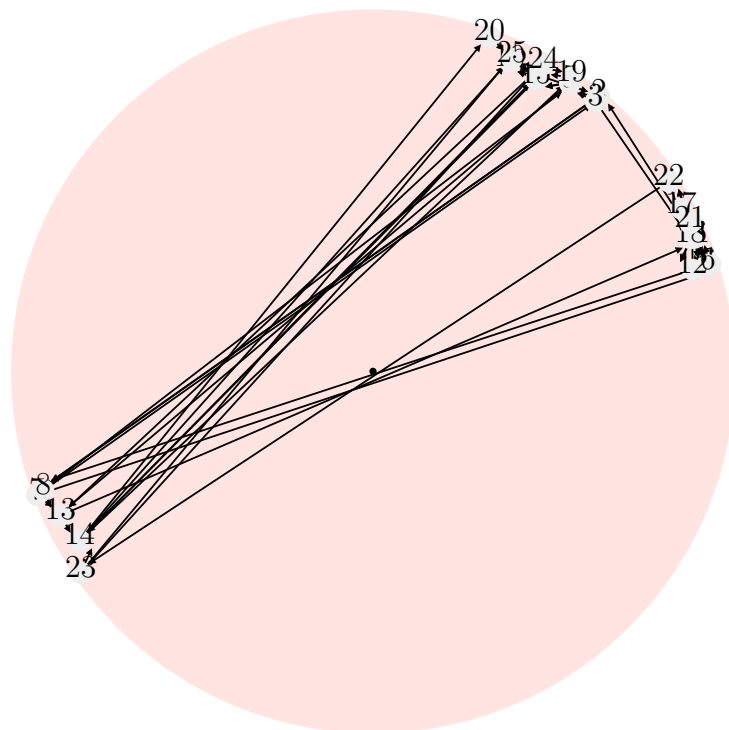


Figure 4.13: PageRank Poincaré Embedding After Iteration $t = 5$

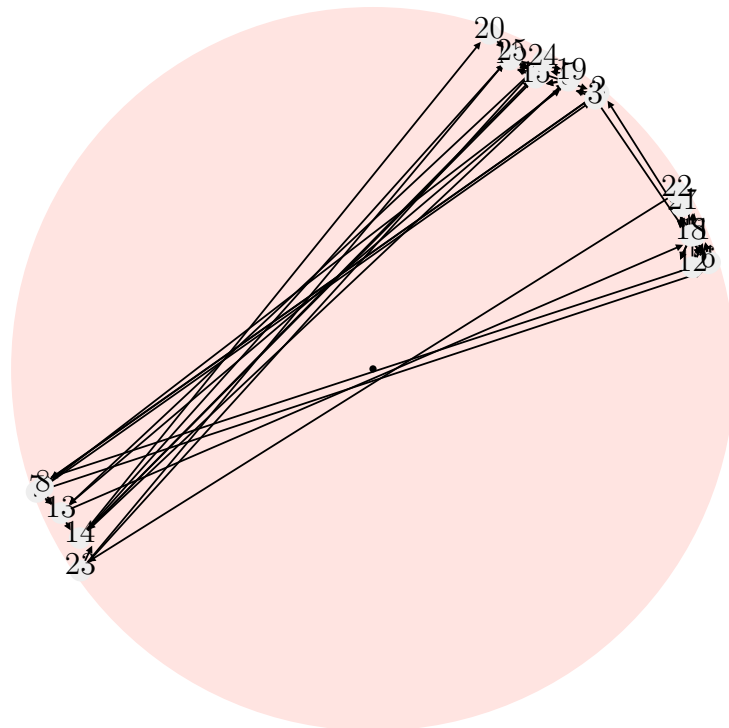


Figure 4.14: PageRank Poincaré Embedding After Iteration $t = 6$

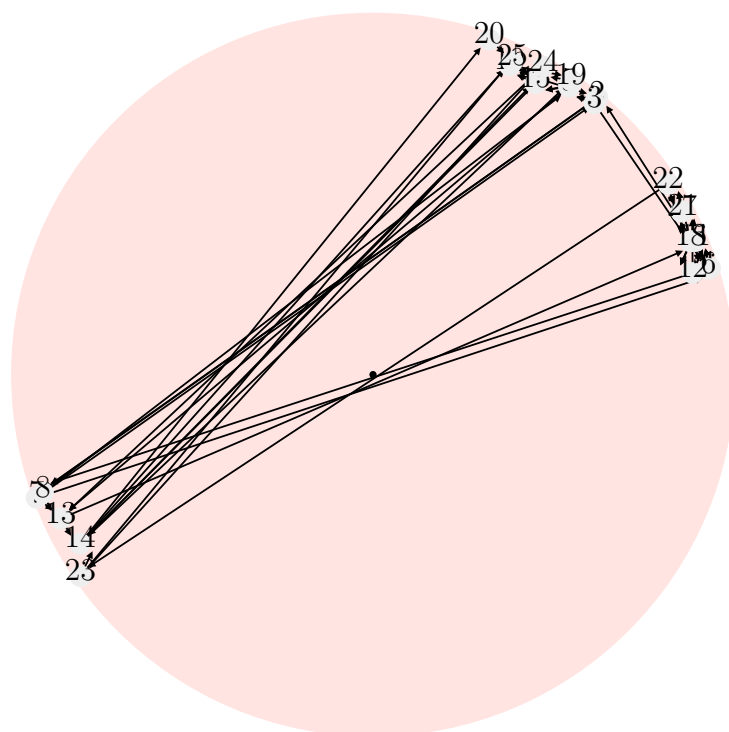


Figure 4.15: PageRank Poincaré Embedding After Iteration $t = 7$

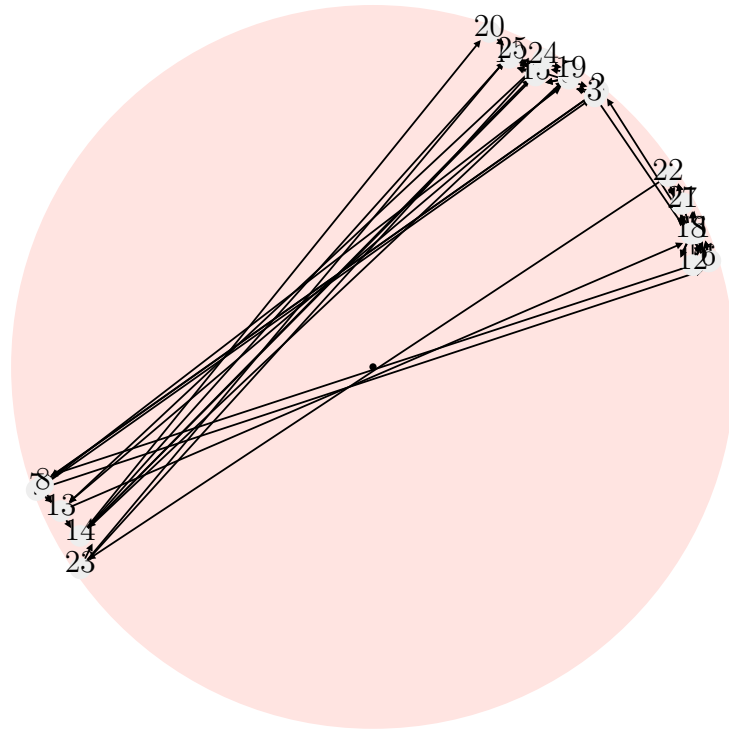


Figure 4.16: PageRank Poincaré Embedding After Iteration $t = 8$

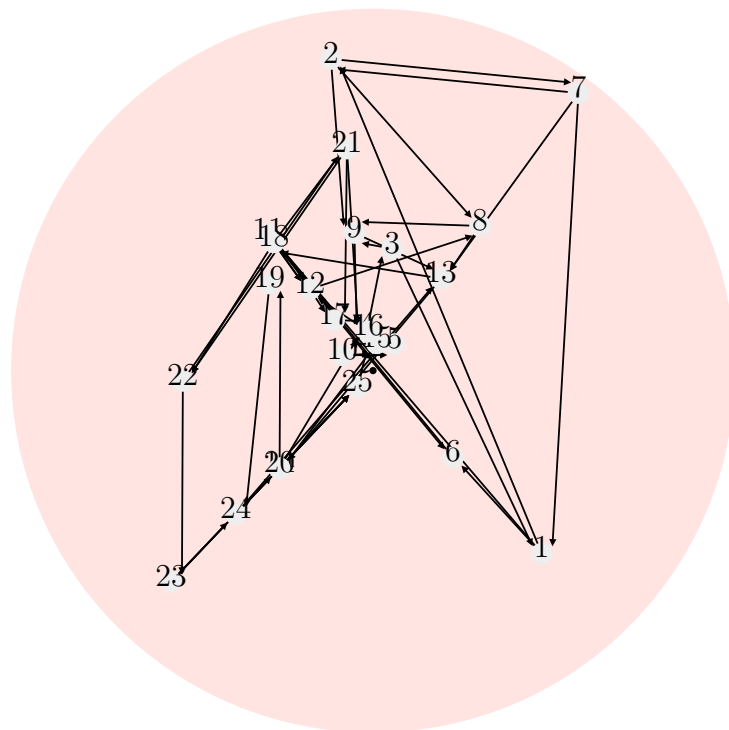


Figure 4.17: Pinski-Narin Poincaré Embedding After Iteration $t = 28$

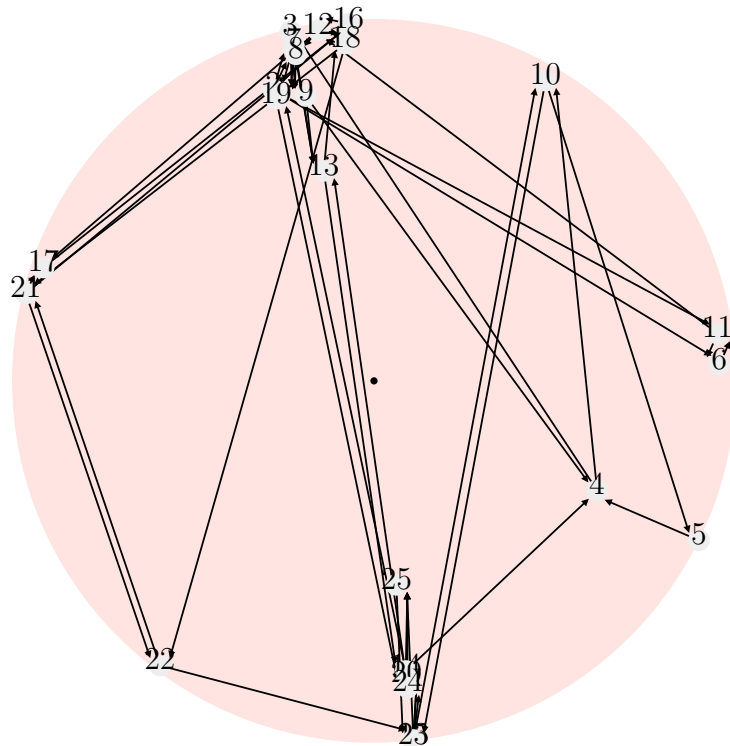


Figure 4.18: HITS Authority Weight Poincaré Embedding After Iteration $t = 17$

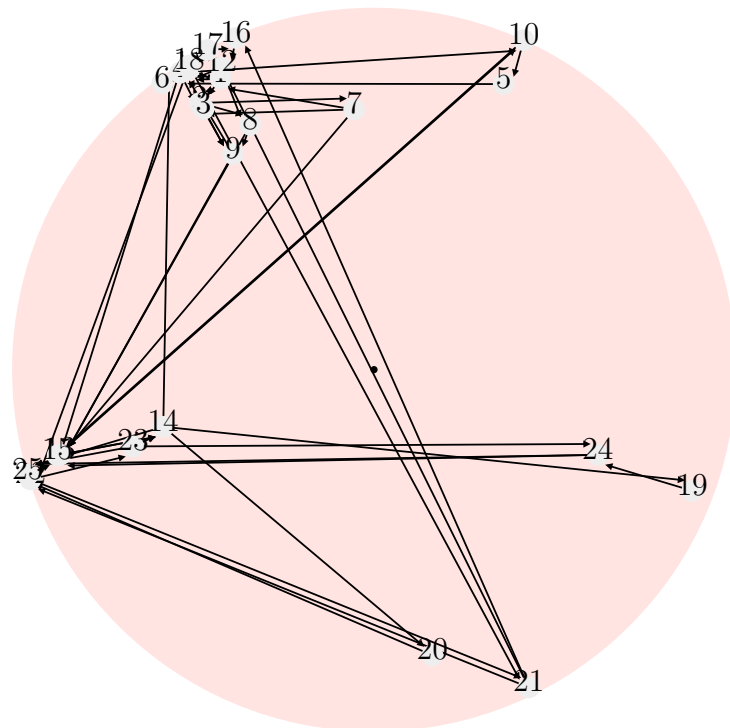


Figure 4.19: HITS Hub Weight Poincaré Embedding After Iteration $t = 17$

Chapter 5

CONCLUSION

This thesis offers a hyperbolic embedding scheme that embeds the nodes in a given complex network in the Poincaré disk based on their respective influence weights computed by the PageRank, HITS, and Pinski-Narin algorithms. The embedding procedure consists of computing a pair of hyperbolic coordinates (r, θ) for each node after each iteration of a given algorithm. The radial coordinate r reflects a node's influence in the network as a whole, while the angular coordinate θ indicates a node's proximity to more influential nodes in the network. A more influential node is assigned a smaller value for r , and hence the node is mapped closer to the origin of the Poincaré disk. A less influential node is assigned a larger value for r , and so the node is mapped farther away from the origin. The angular coordinate θ is chosen so that the node in question is mapped relatively close to its most influential neighbor. After formally defining the embedding procedure, the process was applied to the three aforementioned eigenvector centrality algorithms and the results were discussed.

The intent of the embedding is to physically illustrate any greedy behavior from the nodes as they compete for influence in the network. The nodes can be viewed as self-serving social climbers in a social network: every node wishes to be influential as well as to be near influential nodes. Each node's influence weight computation until convergence manifests in the form of a journey throughout the Poincaré disk as they migrate towards or away from the origin and more influential nodes, depending on their computed weight.

The work done for this thesis is far from complete, or even thorough for that matter. There are many potential avenues to explore from here. Future work may

include applying the defined embedding procedure to the Poincaré half-plane model, or to more eigenvector centrality algorithms, like the Katz centrality Katz (1953). The original intent was to apply this embedding procedure to a significantly larger and more complex citation network (one with thousands of nodes and edges), but constructing the network in addition to detecting and eliminating any sinks (nodes with no outgoing edges) proved to be an insurmountable challenge given the time constraints. An even more ambitious effort is to construct an entirely original eigenvector centrality.

The following final remark is marked by a shift to the first person. When the general idea for this thesis was first pitched to me, my advisor suggested I find a centrality measure that prioritizes quality over quantity: the influence of the node attached to the incoming edge is prioritized over the number of incoming edges when calculating a given node's influence. In particular, my advisor suggested I find a centrality measure that would deem a paper with fewer citations but more prestigious citations more influential than a paper with more citations overall. This discrepancy in quantifying influence can be traced back to the initialization stage: the nodes' influence weights should be initialized so that nodes with more incoming edges are considered more influential in the beginning, but over time the influence of the citations themselves are taken into account in the weight computation. As seen in the different initialization methods for PageRank, HITS, and Pinski-Narin, as well as the flood of network science research that followed since the inception of these algorithms, this distinction between quality and quantity in measuring node centrality is an ongoing challenge. This goal still unfortunately eludes me, so this thesis may be considered a conciliatory prize for a failed mission. But this thesis has sparked a driving curiosity and enthusiasm for network science that I am sure will follow me throughout my PhD studies.

REFERENCES

- Bianconi, G. and C. Rahmede, “Emergent hyperbolic network geometry”, *Scientific Reports* **7**, 41974 (2017).
- Boguna, M., D. Krioukov and K. C. Claffy, “Navigability of Complex Networks”, *Nature Physics* **5**, 74–80 (2009).
- Boguna, M., F. Papadopoulos and D. Krioukov, “Sustaining the Internet with hyperbolic mapping”, *Nature Communications* **1**, 6 (2010).
- Boyd, S., A. Ghosh, B. Prabhakar and D. Shah, “Randomized Gossip Algorithms”, *IEEE Transactions on Information Theory* **52**, 6, 2508–2530 (2006).
- Chamberlain, B., J. Clough and M. Deisenroth, “Neural embeddings of graphs in hyperbolic space”, arXiv preprint arXiv:1705.10359 (2017).
- Chami, I., R. Ying, C. Re and J. Leskovec, “Hyperbolic graph convolutional neural networks”, *NeurIPS* (2019).
- DeSa, C., A. Gu, C. Re and F. Sala, “Representation tradeoffs for hyperbolic embeddings”, *Proceedings of machine learning research* **80**, 4460 (2018).
- Durrett, R., *Essentials of Stochastic Processes, 2nd edition* (Springer, New York, 2012).
- Faloutsos, M., P. Faloutsos and C. Faloutsos, “On power-law relationships of the internet topology”, *ACM SIGCOMM Computer Communication Review* **29**, 4, 251–262 (1999).
- Ganea, O., G. Becigneul and T. Hofmann, “Hyperbolic neural networks”, *NeurIPS* pp. 5345–5355 (2018).
- Geller, N., “On the citation influence methodology of Pinski and Narin”, *Information Processing & Management* **14**, 93–95 (1978).
- Iversen, B., *Hyperbolic Geometry* (Cambridge University Press, Cambridge, 1992).
- Katz, L., “A new status index derived from sociometric analysis”, *Psychometrika* **18**, 1 (1953).
- Kitsak, M., I. Voitalov and D. Krioukov, “Link prediction with hyperbolic geometry”, arXiv preprint arXiv:1903.08810 [physics.soc-ph] (2019).
- Kleinberg, J., “Authoritative sources in a hyperlinked environment”, *JACM* **46**, 5, 604–632 (1998).
- Kleinberg, R., “Geographic routing using hyperbolic space”, *Proceedings of the IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications* pp. 1902–1909 (2007).

- Kochen, M., *Principles of Information Retrieval* (Wiley, New York, 1974).
- Krioukov, D., F. Papadopoulos, M. Kitsak, A. Vahdat and M. Boguna, “Hyperbolic geometry of complex networks”, *Phys. Rev. E* **82**, 3, 036106 (2010).
- Liu, Q., M. Nickel, and D. Kiela, “Hyperbolic graph neural networks”, *NeurIPS* (2019).
- Nickel, M. and D. Kiela, “Poincare embeddings for learning hierarchical representations”, *NeurIPS* (2017).
- Page, L., S. Brin, R. Motwani and T. Winograd, “The PageRank citation ranking: Bringing order to the Web”, (1998).
- Papadopoulos, F., M. Kitsak, M. A. Serrano, M. Boguna and D. Krioukov, “Popularity versus similarity in growing networks”, *Nature* **489**, 7417, 537–540 (2012).
- Papadopoulos, F., C. Psomas and D. Krioukov, “Network mapping by replaying hyperbolic growth”, *IEEE/ACM Transactions on Networking* **23**, 1, 198–211 (2015).
- Pinski, G. and F. Narin, “Citation influence for journal aggregates of scientific publications: Theory, with application to the literature of physics”, *Information Processing & Management* **12**, 297–312 (1976).
- Samei, Z. and M. Jalili, “Application of hyperbolic geometry in link prediction of multiplex networks”, *Scientific Reports* **9**, 1 (2019).
- Thai, M. T., W. Wu and H. Xiong, *Big Data in Complex and Social Networks* (CRC Press, Boca Raton, 2016).