IMPERIAL

IMPERIAL COLLEGE LONDON

DEPARTMENT OF MATHEMATICS

SECOND-YEAR GROUP RESEARCH PROJECT

Random walks on graphs

Authors:
Kassam, Adam (CID: 2374184)
Li, Daniel (CID: 2374487)
Soni, Hridita (CID: 2374160)
Watson, William (CID: 2386679)

Supervisor:
Professor Mauricio Barahona

September 25, 2025

Abstract

Random walks on graphs provide a powerful mathematical framework for modelling stochastic processes, with applications spanning web search algorithms, social network analysis, and semantic discovery. First, we review the necessary **graph-theoretic** and **Markov-chain foundations**, introducing basic notions of adjacency and transition matrices, exploring time-reversibility and harmonic functions, and examining key walk parameters—access and cover times, mixing rates, symmetry, ordering, and monotonicity—that quantify how walkers traverse network structures. Then, we delve into **asymptotic behaviour**, establishing existence and uniqueness of the stationary distribution, characterising convergence rates through spectral gaps, and presenting a Monte Carlo interpretation that leverages ergodic and central-limit theorems. We turn to **PageRank** as a seminal global centrality measure: we outline its backlink-driven formulation, address structural challenges via teleportation and damping, describe efficient computation, and survey extensions. Finally, we introduce **severability**, a local coherence metric defined by the retention and mixing of walks on candidate subgraphs, and illustrate its practical use in semantic dictionary expansion and in uncovering Twitter interaction communities among members of the U.S. Congress. These sections collectively showcase the theoretical depth of random walks on graphs. At the same time, they demonstrate their broad applicability to real-world problems in data analysis and network science.

Contents

1 Introduction

A *random walk*, or *drunkard's walk*, follows a walker starting at a node on the graph at time $t = 0$, who jumps to any out-neighbour at time $t = 1$ with a probability proportional to the edge weight of the graph. A random sequence of steps has many applications in probability theory, spanning from electrical networks and particle movement in physics to the fluctuations of stock prices in financial markets. The theory of random walks on graphs is closely related to other branches of graph theory as basic properties are related to the spectral graph and Markov chain theory and can be used for graph exploration, to traverse large or complicated networks.

An intuitive example of random walks is the *simple 8 puzzle*, a popular sliding tile game on a frame of numbered squares. In the graph below, each node is a valid configuration of the tiles and two nodes are connected by an edge only if one configuration can be transformed into another by one tile slide. At every step, a random transformation can be chosen from the out-edges of the current node and the process of randomly sliding tiles forms a random walk on the permutation graph. The structure of the graph reflects the state space of the puzzle.

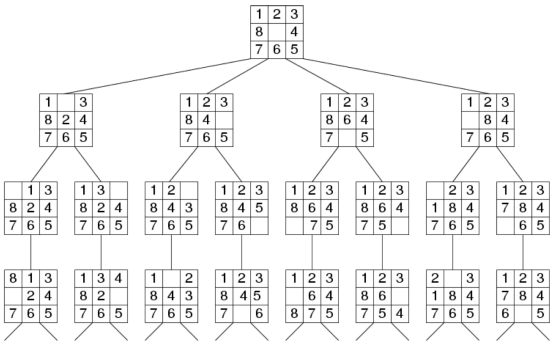

Figure 1 Example of part of a Permutation Graph for an 8 puzzle configuration [1]

Random walks have been comprehensively studied on regular structures, such as lattices, giving rise to the most commonly-used properties. But using the richer topology of generic graphs, especially infinite graphs, can have a considerable effect on these properties and their asymptotic behaviours can reveal universal features. These are limited to the Euclidean dimension on lattices but otherwise, these features allow the dimensions of inhomogeneous structures to be generalised. This grants the splitting between local and average properties, producing an elementary framework to investigate complex systems even from an experimental perspective. [2]

In the sections that follow, we delve into how these systems mature over time, observing how their long-term behaviour reveals deep structural insights about the graph considered. Although applications of random walks are vast and diverse, we focus on PageRank as random walks underpin the method for ranking pages based on the graph of their link structure. Finally, we investigate the subject of severability, illustrating how random walks can be used to identify and isolate communities or weakly connected regions within the graph. Overall, these perspectives allow us to highlight the importance and practical versatility of random walks in graph theory.

1.1 Graph Theory

A random walk defines a **stochastic process** on the vertices of a finite, connected graph G, which is comprised of a vertex-set V and an associated edge-set E, such that $G = (V, E)$. The graph is often assumed to contain no loops (i.e. no edges of the form (u, u)) and no parallel

edges (i.e. no two edges share the same vertex pair), as these can result in infinite loops across two edges. Two vertices u and v (called *end-vertices*) are connected if the edge $(u, v) \in E$, with this structure reflected below in the adjacency matrix. The number of edges incident on a vertex v is called its degree and is denoted by $d(v)$ (for a weighted graph, this is influenced by the weight function w so $d(v) = \sum_u w(u, v)$). A path is a sequence of alternating edges and vertices denoted by v_0/v_n if it travels from v_0 to v_n and has length n. This relation between nodes is an equivalence relation and its equivalence classes are called components. The graph being connected is equivalent to saying it has a unique component. [3]

Starting at the v_0 node, the walk travels to v_t at time t, moving to a neighbour with the probability of $1/d(v_t)$. Given the initial distribution P_0, we can draw the initial node v_0. If the graph is unweighted, the probability of travel between nodes is uniform but if the graph is weighted, it is reflected in the probability transition matrix. Let $P_t(i) = \mathbb{P}(v_t = i)$, where $p_{ij} = 1/d(i)$ if $(i, j) \in E$ and the graph is unweighted, otherwise $p_{ij} = w(i, j)/d(i)$ if the graph is weighted by weight function w. Then $M = (p_{ij})_{i,j \in V}$ is the matrix of transition probabilities. A_G is the adjacency matrix of the graph G, detailing the edges linking the nodes, where the i, jth term is 1 if the nodes i and j are connected and 0 otherwise. D is the diagonal matrix with its elements $(D)_{ii} = 1/d(i)$. From these matrices, we have that $M = DA_G$ and for a d-regular graph, where all vertices have a degree of d, we have that $M = (1/d)A_G$.

Definition 1.1. The transition probabilities between the vertices can be defined by the equation called the rule of the walk:

$$P_{t+1} = M^T P_t \implies P_t = (M^T)^t P_0$$

It follows that the probability p_{ij}^t, which is the ij-entry of M^t , is the probability that a walk which starts at i will reach j in t steps. [4]

1.2 Markov Chains

The successive jumps of the walk define a finite *Markov chain* on the graph G, where the states correspond to the vertices V. The probability of the presence of the random walker evolves with time and is defined by the transition matrix P.

If G is a **regular** graph, such that all the vertices have the same degree, the Markov chain it defines is **symmetric**, which means that the probability of the jump from the node x to node y is equivalent to the probability from y to x. However, a non-regular graph G gives rise to a Markov chain that is *time reversible*, so the corresponding random walk is also a random walk when considered backward. Therefore, for all walks where v_0 is drawn for P_0, we get a probability distribution P_t on v_t, but we also get the probability distribution Q on (v_0, v_1, ...,v_t) and if we reverse the sequence, we get Q'. Time-reversibility means that the distribution Q' is the same as the distribution obtained by looking at random walks starting from P_t.

Another concept that can be used to study the evolution of random walks is harmonic functions. A function is harmonic at a node if its value at that node is the expected value of its neighbours, weighted by transition probabilities.

Definition 1.2. Consider a Markov Chain Z on the vertex-set V with transition matrix P, which is reversible with respect to a positive function π. A function f is **harmonic** on $U \subseteq V$ if

$$\forall u \in U,\ f(u) = \sum_{v \in V} p_{u,v} f(v)$$

This is equivalent to $f(u) = \mathbb{E}(f(Z_1) \big| Z_0 = u)$, where $\mathbb{E}$ refers to the expectation of a random variable. Here, f is harmonic with respect to P. [3]

The hitting probabilities are examples of harmonic functions for the chain.

Theorem 1.1. *Let $W = V \setminus U$, $s \in U$ and for $v \in V$, $g(v)$ be the probability that the chain hits s before a node in W, having started at v.*

$$g(v) = \mathbb{E}_v(Z_n = s : n < T_W)$$

where

$$T_W = \inf\{n \geq 0 : Z_n \in W\}$$

T_W is the first-passage time to W, and $\mathbb{E}_v(\cdot) = \mathbb{E}(\cdot|Z_0 = v)$ denotes the probability measure of the chain starting at u. Then, the function g is harmonic on U. [3]

This theory establishes a significant relationship between the structure of the graph and the probabilistic behaviour of random walks on it. The notions of time reversibility and harmonic functions are basic tools for examining how information traverses through the network. Harmonic functions, such as hitting probabilities, allow us to study the dynamics of random walks, especially in relation to their asymptotic behaviour, which we will see later.

1.3 Spectral Properties

The **spectral theory of matrices** specifies when a matrix can be diagonalised under a transformation using its corresponding eigenvalues and eigenvectors. Historically, matrix theory and linear algebra were used to analyse adjacency matrices of graphs and specific eigenvalues were titled the "algebraic connectivity" of the graph. Developments over the years have caused spectral theory to follow a more geometric path, leading to powerful, new techniques emerging. One of the most fundamental applications of spectral theory is in astronomy, where it is used to distinguish the make-up of distant stars, identify the gases in their atmosphere and their temperature and density profiles. The powerful methods of spectral theory can similarly be applied to random walks as the probability p_{ij}^t (a walk starting at i will be at node j after t steps) is an entry of matrix M^t, and so we can understand the walk deeper through M's eigen-properties.

The matrix M has *largest eigenvalue* 1, with corresponding *left eigenvalue* π and *right eigenvalue* $\mathbf{1}$ (the all-ones vector on V). From this, we know that exactly one step is made from each node, as we see that $M\mathbf{1} = \mathbf{1}$.

To bring M to a *symmetric* form (assuming it is not already symmetric), we consider the matrix $N = D^{1/2}AD^{1/2} = D^{1/2}MD^{1/2}$. This matrix N is symmetric and can be written in spectral form $N = \sum_{k=1}^{n} \lambda_k v_k v_k^t$ where $\lambda_1 \geq \lambda_2 \geq ...$ are eigenvalues of N and $v_1, v_2...$ are the corresponding eigenvectors of unit length. [4]

We transform the transition matrix into a symmetric form, in order to allow spectral analysis, which provides a path for understanding the behaviour of random walks. The eigenvalues and eigenvectors of the symmetric matrix convey critical information about the graph's structure and these insights deepen our understanding of the walk we are studying.

1.4 Main Parameters and Properties

There are some crucial properties which have the most important role in the quantitative theory surrounding random walks.

1.4.1 Access Time and Commute Time

Access time (also known as *hitting time*) is denoted by H_{ij} and is defined as the expected number of steps the walker takes from node i before visiting node j. Using this property, we can find the sum:

$$\kappa(i, j) = H(i, j) + H(j, i)$$

This summation is called the *commute time* and it represents the expected steps of the walker who, after starting at i, visits j and then reaches i again.

1.4.2 Cover Time

The *cover time* describes the expected number of steps a walker would take to reach every node in the graph. Often a start node is given and the associated cover time can be computed, but if no start node is chosen, the commute time reflects the maximum expected cover time over all possible starting nodes.

1.4.3 Mixing Rate

The *mixing rate* is a measure of how quickly a walk converges to its limiting distribution. If the graph G is **non-bipartite**, then we have that $p_{ij}^t \to \frac{d_j}{2m}$ as $t \to \infty$ and so this property can be calculated as:

$$\mu = \limsup_{t\to\infty} \max_{i,j} \left| p_{ij}^t - \frac{d_j}{2m} \right|^{\frac{1}{t}}$$

However, if the graph is **bipartite** and can be partitioned into $\{V_1, V_2\}$, then the results become far more complicated. Hence, for now we can ignore them. [4]

1.4.4 Symmetry

The *time reversibility* nature of random walks guarantees that all two-step circular trips are symmetric. If we consider two nodes u and v, which have the same degree, then the probability that a random walk which begins at u travels to v and then back to u (which is computed as $\frac{1}{\kappa(u,v)\pi(u)}$) is equivalent to the probability that the walk starts at v, travels to u and then back to v. Similarly, the *cover time* of the first time the random walk returns to u after visiting v is equal to the probability that random walk starting at v visits u before returning to v.

Theorem 1.2. *For any three nodes u, v, and w,*

$$H(u, v) + H(v, w) + H(w, u) = H(u, w) + H(w, v) + H(v, u)$$

1.4.5 Ordering

If the node u is visited before the node v, then we have that $H(u, v) \leq H(v, u)$. This *ordering* can be derived from fixing any node t and ordering nodes according to the value of $H(u, t) - H(t, u)$. This property is a consequence of the symmetry of a random walk but the orderings are not unique. We can partition nodes in the graph so that classes are formed of nodes when $H(u, v) = H(v, u)$, and then there is a well-defined ordering of the equivalence classes (independent of the reference node t).

Theorem 1.3. *If graph has a vertex transitive automorphism (for any two vertices, there exists an automorphism that maps one to the other), then:*

$$\forall i, j : H(i, j) = H(j, i).$$

1.4.6 Monotonicity

Let G' be a copy of the graph G, which has m edges, but with an additional edge (a, b). We would expect the random walk to return back to a previous node less frequently as the graph is more dense. However, in reality, we do not always see the access time, commute times and cover times decrease as expected. In this case, the commute time of the graph G' is at most $1 + 1/m$ times the commute time of G. Equivalently, we can say the value of $\kappa(s, t)/m$ does not decrease.

2 Asymptotic properties

Generally, for the random walk on a graph, we study the **stationary distribution** on the graph, as a way to study how the system evolves over time.

2.1 Stationary distribution

Definition 2.1. A **right stochastic** matrix, or a **row stochastic** matrix, is a square matrix of non-negative real numbers, with each row summing to 1.

Theorem 2.1 (Existence of Stationary distribution)**.** *If P is an $n \times n$ row stochastic matrix, then there exists a **stationary distribution** π, such that*

$$P^T \pi = \pi.$$

Proof. Since P is row stochastic, we have $P\mathbf{1_n} = \mathbf{1_n}$, which means that 1 is an eigenvalue of P. The stationary distribution exists since P and P^T have the same eigenvalues. □

Although there is at least one stationary distribution, it does not guarantee that it is *unique*, nor does it guarantee that the random walk converges to a distribution given an initial distribution. Consider the following example:

$$P_1 = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 1 \end{pmatrix}, P_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Notice that since $P_1^k = P_1$, the distribution to which the random walk converges completely depends on the initial distribution. For P_2, the random walk does not converge at all.

Thus, to obtain the desired stationary distribution, we need to condition the property of G. Note that since the uniqueness of the stationary distribution is related to the eigenspace associated with the eigenvalue 1, the following definition and result would be helpful.

Definition 2.2. A square matrix P is **primitive** if there exists a positive integer k such that $P^k > 0$.

Remark. If P is a probability transition matrix of a random walk on the corresponding graph G, then P being *primitive* is equivalent to G being *strongly connected* and *aperiodic*.

Theorem 2.2 (Perron–Frobenius Theorem [5])**.** *Suppose that P is a primitive matrix. Then, there exists an eigenvalue r such that*

(i) $r \in \mathbb{R}$ *and* $r > 0$*;*

(ii) $r > |\lambda|$ *for any eigenvalue* $\lambda \neq r$*;*

(iii) The eigenvectors associated with r are unique up to constant multiples.

Corollary 2.2.1 (Uniqueness of Stationary distribution)**.** *Let P be a $n \times n$ row stochastic matrix. If P is primitive, then the stationary distribution is unique.*

Proof. Let $x \in \mathbb{C}^n$ and x_k be the entry of P of the largest modulus.

$$|\lambda x_k| = \left| \sum_{i=1}^{n} p_{ki} x_i \right| \leq \sum_{i=1}^{n} p_{ki} |x_i| \leq |x_k| \sum_{i=1}^{n} p_{ki} = |x_k| \quad \Longrightarrow \quad |\lambda| \leq 1.$$

Since P is primitive and 1 is an eigenvalue of P, according to the Perron-Frobenius theorem, the eigenvector x associated with 1 is unique, as it is a probability distribution. □

Remark. As this also suggests that -1 is not an eigenvalue of P if P is primitive, it is also worth mentioning that -1 is an eigenvalue of P if and only if G is a bipartite graph (period of 2). To see this, consider writing the transition matrix P as

$$\begin{pmatrix} 0 & A \\ B & 0 \end{pmatrix},$$

where both A and B are row-stochastic.

If a graph has multiple stationary distributions, the eigenvalue 1 is **not simple**. Given a graph G, we consider each of its *communicating classes* (strongly connected components) as a single node, form a graph G', and add a self-loop for each terminal node, representing the transition between components. We can write the probability transition matrix on G' as

$$\begin{pmatrix} I_k & 0 \\ A & B \end{pmatrix}.$$

As $A \neq \mathbf{0}$, B cannot have an eigenvalue of 1. In this sense, the algebraic multiplicity of 1 is exactly k, again by the Perron-Frobenius theorem, where k equals the number of *absorbing communicating classes*. Any stationary distribution appears to be a linear combination of the stationary distributions within communicating classes, representing the probability of *sinking* into different sets of nodes.

Finally, along with the condition that P is primitive, we further show that the random walk, given any initial distribution, converges to the unique stationary distribution.

Theorem 2.3 (Convergence of Random Walk)**.** *Let P be a primitive probability transition matrix and π be the stationary distribution. Then, for any initial distribution x, $x^T P^t$ converges to π^T*

Proof. We apply the Jordan decomposition to P over $\mathbb{C}$. Also, recalling that 1 is a simple eigenvalue of P with a one-dimensional eigenspace, we have $P = VJV^{-1}$ where

$$J = [1] \oplus (\bigoplus_{i=2}^{k} J_{m_k}(\lambda_i))$$

For each Jordan Block, we have

$$J_{m_k}(\lambda_i)^t = (\lambda_i I + J_{m_k}(0))^t = \sum_{i=0}^{m_k - 1} \binom{t}{i} \lambda_i^{t-i} J_{m_k}(0)^i$$

Since $\binom{t}{i}\lambda_i^{t-i} \to 0$ as $t \to \infty$ for $|\lambda_i| < 1$, each entry of $J_{m_k}(\lambda_i)^t$ converges to 0 as $t \to \infty$. When $t \to \infty$, P^t converges to

$$VJ^tV^{-1} = (V\mathbf{e}_1)(\mathbf{e}_1^TV^{-1}) = (V\mathbf{e}_1)(\mathbf{e}_1^TV^{-1}) = \mathbf{1}_n\pi$$

Note that $(\mathbf{e}_1^TV^{-1})P = (\mathbf{e}_1^TV^{-1})(VJV^{-1}) = (\mathbf{e}_1^TJ)V^{-1} = \mathbf{e}_1^TV^{-1}$ is in fact the left-eigenvector of P. Thus, $\mathbf{e}_i^TP^t$ converges to $\mathbf{e}_i^T\mathbf{1}_n\pi = \pi^T$ □

However, in many cases, P is not primitive. We can define

$$P' = (1-\alpha)P + \alpha\mathbf{1}_{n\times n}$$

so that the graph becomes connected, and also aperiodic due to the added self-loops. The technique is called 'teleportation' and will be introduced in more detail in later sections.

2.2 More on spectral properties

As we established the *existence of a stationary distribution*, conditioned on the graph to impose uniqueness, and proved the convergence of the random walk, it can be seen that the eigenvalues of the probability transition matrix play an important role and contain rich information, including the relationship between communicating classes and the algebraic multiplicity of 1, which we have already discussed.

To simplify the discussion and computation, we assume that the Markov chain structures being studied are all *time-reversible*.

Lemma 2.4 (adapted from [6]). *Let P be a time-reversible transition matrix with a unique stationary distribution π. Then we have*

$$P^T = \mathbf{1}_n\pi^T + \sum_{i=2}^{n} \lambda_i D^{-\frac{1}{2}} v_i v_i^T D^{\frac{1}{2}}$$

where $D = \mathrm{diag}(\pi_1, \ldots, \pi_n)$, $\lambda_i \in \mathbb{R}$ are eigenvalues of P, $\{v_1, v_2, \ldots, v_n\}$ is an orthonormal basis.

Proof. Consider the matrix $S = D^{\frac{1}{2}}P^TD^{-\frac{1}{2}}$, which is symmetric by time reversibility, and apply the spectral theorem similar to [6]. □

This not only again establishes the convergence of the random walk, but also gives a tight connection with the spectrum of P. For example, we may directly study the rate of convergence as follows. Let P be primitive and time-reversible. Without loss of generality, let $\lambda_i \geq \lambda_{i+1}$. By Lemma 2.4,

$$\left|((P^T)^tx - \pi)_k\right| \leq \lambda^t \left(\sum_{i=2}^{n} \mathbf{e}_k^T D^{-\frac{1}{2}} v_i v_i^T D^{\frac{1}{2}} x\right)$$

where $\lambda = \max\{|\lambda_2|, |\lambda_n|\}$ and the convergence rate is governed by λ^t. We call the value $\gamma = 1 - \lambda$ the **spectral gap** of P.

2.3 A Monte Carlo View

One way to think about the stationary distribution of the random walk on the graph is that it gives the *frequency with which a node is visited in the long run*. Such an interpretation is important, as it regards the stationary distribution as a centrality measure in networks. For

instance, the PageRank algorithm developed by Google (see section 3) is used to rank web pages in their search engine, treating the stationary distribution as a measure of importance.

To formally state this idea, the following result is essential.

Theorem 2.5 (Ergodic theorem [7]). *Let the $n \times n$ transition matrix P be primitive, $\{X_i : i = 0, 1, \dots\}$ be a Markov chain with P as transition matrix. Then, for any bounded function, $f : I \to \mathbb{R}$, we have*

$$P\left(\frac{1}{n}\sum_{i=0}^{m-1} f(X_i) \to \mathbb{E}_\pi[f(X)] \ \ as \ \ m \to \infty\right) = 1,$$

where I is the state space. In particular, if we take $f(x) = \mathbf{1}_{x=k}$, we have $\mathbb{E}_\pi[f(X)] = \pi_k$.

With this idea formally stated, we can estimate the stationary distribution of a finite irreducible matrix P by counting the occurrence of nodes by sampling random walks. Not only can we eventually get an accurate result, but we can also estimate how long it takes to get a sufficiently accurate estimation by extending the central limit theorem to a Markov chain setting.

Theorem 2.6 (Markov Chain Central Limit Theorem [8]). *Let $\{X_i : i = 0, 1, \dots\}$ be a finite irreducible and aperiodic Markov chain with stationary distribution π, $f : I \to \mathbb{R}$ be a Borel measurable function, $\mathbb{E}_\pi[f(X)^2] < \infty$. Then we have*

$$\sqrt{m}(\mu_m - \mu) \xrightarrow{d} \mathcal{N}(0, \sigma^2)$$

where

$$\mu_m = \frac{1}{m}\sum_{i=1}^{m} f(X_i), \quad \mu = \mathbb{E}_\pi[f(X)]$$

$$\sigma^2 = Var_\pi(X) + 2\sum_{i=1}^{\infty} cov_\pi(f(X_0), f(X_i))$$

We try to calculate the variance when $f(x) = \mathbf{1}_{x=k}$, which yields the following result.

Theorem 2.7. *Suppose that P is primitive and time-reversible. Then we have*

$$\left|\sigma^2 - \pi_k(1 - \pi_k)\right| = \mathcal{O}\left(\frac{1}{\gamma}\right)$$

where γ is the spectral gap of P.

Proof.

$$Var(X_0) = \pi_k(1 - \pi_k).$$
$$\begin{aligned} cov_\pi(f(X_0), f(X_t)) &= \mathbb{E}[f(X_0)f(X_t)] - \mathbb{E}[f(X_0)]\mathbb{E}[f(X_t)], \\ &= \pi_k P_{K,k}(t) - \pi_k^2. \end{aligned}$$

where $P_{k,k}(t)$ is the probability that the random walk starting from k returns to k after t time steps. To compute the sum $\sum P_{k,k}(t) - \pi_k$, we again use Lemma 2.4:

$$\begin{aligned} \left|\sum_{t=1}^{\infty} P_{kk}(t) - \pi_k\right| &= \left|\sum_{t=1}^{\infty} \mathbf{e}_k^T((P^T)^t - Q)\mathbf{e}_k\right| \\ &= \sum_{i=2}^{n} v_{ik}^2 \left|\sum_{t=1}^{\infty} \lambda_i^t\right| \\ &\le \frac{n\lambda}{1-\lambda} = \mathcal{O}\left(\frac{1}{\gamma}\right) \end{aligned}$$

□

We take a look at the sum that is part of the covariance closely:

$$\sum_{i=2}^{n} v_{ik}^2 \frac{\lambda_i}{1-\lambda_i}$$

Assume that *all eigenvalues are non-negative*. Then we have $\sigma^2 = \pi_k(1-\pi_k) + \mathcal{O}(\frac{1}{\gamma})$. Again, we relate the variance to the eigenvalues, specifically the spectral gap, but why does it appear to be related?

The *smaller* the spectral gap, the *higher* the variance of the estimation. As the estimation is based on the frequency of visiting, it turns out that the variance would be high if the random walker tends to visit the node too much or rarely visits the node within a time period. If we think of the behaviour of the random walker on the graph, it is possible that the random walker is trapped in a set of nodes within this time period.

From such intuition, we can make a reasonable guess that if the spectral gap is small, there are likely to exist components of the graph that it is hard for the random walker to escape from. Recall that the absorbing components contribute to the algebraic multiplicity of 1, which can be considered an extreme case for this.

What about the negative eigenvalues, as they would decrease the variance? To answer this, we can also think of the case that P has an eigenvalue of -1, that is, when the graph G is bipartite, as discussed earlier. An eigenvalue closer to -1 can therefore indicate a strong periodicity, so that each node can be visited on a more regular basis.

Example (Connecting two graphs). Consider the graph formed by connecting two graphs that are complete (K_n) or circular (C_n) by a single undirected edge. We list λ_2 and λ_n for a few pair of graphs.

G_1	G_2	λ_2	λ_n
K_5	K_1	0.282	-0.532
K_5	K_3	0.860	-0.515
K_5	K_5	0.927	-0.377

G_1	G_2	λ_2	λ_n
C_5	K_1	0.545	-0.869
C_5	K_3	0.855	-0.809
C_5	K_5	0.910	-0.809

Table 1 Eigenvalues for various (G_1, G_2).

Motivated by this, it is tempting to define a measure on such 'connectivity' of the graph. If a component of the graph is 'isolated' from the other part, we should expect few edges toward the outside. Following on from this, we can introduce **conductance** as a measure of connectivity.

Definition 2.3 (Conductance)**.** Let $G = (V, E)$ be a graph and A be its adjacency matrix. The *conductance* of a subset of nodes $S \subset V$ is defined as

$$\Phi(S) = \frac{\nabla(S, V \setminus S)}{\min(\mathrm{vol}(S), \mathrm{vol}(V \setminus S))}$$

where we have

$$\nabla(A, B) = \sum_{i \in A} \sum_{j \in B} a_{ij}, \quad \mathrm{vol}(S) = \nabla(S, V).$$

Intuitively, it can be interpreted as the *ease of escape* from a given set of nodes. To measure the overall conductance of the graph, we can select the set $S \subset V$ such that the conductance is minimized:

$$\Phi(G) = \min_{S \subset V} \Phi(S)$$

First introduced by Cheeger [9], this is often called the **Cheeger's constant** which is originally used in differential geometry for isoperimetric problems. The conductance is a *discrete analogue* of Cheeger's constant on a graph, considering the separation of the graph to subgraphs. The following result is significant on the relationship of the conductance and spectral property of the graph:

Theorem 2.8. *Suppose that P is a primitive time-reversible transition matrix on graph G. Then we have the following inequality:*

$$\frac{\Phi(G)^2}{2} < 1 - \lambda_2 < 2\Phi(G)$$

Remark. This theorem is adapted from Chung[10]; we have modified the statement to fit our setup, since the original formulation uses the Laplacian matrix, which we will not develop further here.

The inequality provides evidence on the relationship between the spectral gap and the almost absorbing components of the graph, as we have already discussed.

In this section, we have characterised the relations between the variance of the estimation of stationary distribution, defined the spectral gap of the transition matrix, discussed the behaviour of a random walker on different types of graphs, introduced the conductance of the graph and related it to the spectral gap of the transition matrix and the discussion we have previously. Along with all the insights and knowledge we have with the theory of random walks on graphs, we are stepping into their practical applications.

3 PageRank

3.1 Background

We now discuss a fundamental application of random walks concerning the relative importance of web pages. Throughout this section, we refer to a *directed* graph of the web $G = (V, E)$ with nodes V representing $|V| = n$ web pages and directed edges E representing forward links between pages. "Rank" is also referred to as a sort of substance, due to it being handled in a context simulating diffusion. We begin with a definition:

Definition 3.1. A **centrality measure** on a set S is an importance-based ranking of the elements of S, with respect to some context.

A centrality is used to obtain a hierarchical ranking for V. A very straightforward **local** centrality measure is degree counting: nodes with the greatest degrees receive the highest rankings (corresponding to many pages holding links to them). In the context of the web, this is very prone to "exploitation". For example, if a legion of sites L is created with no purpose other than having links to a page P, P will receive *fraudulent rank* from the useless sites in L. So, not only should the number of links to a page be considered, but also the quality of such links. Naturally, a global measure is likely to be more effective against exploitation as each ranking is inferred from the ranking of every node before it.

PageRank is an acclaimed **global** centrality measure introduced by Brin and Page[11], designed to be used on web pages. It is *backlink-driven*:

Definition 3.2 (Backlink)**.** For $v \in V$, if E contains an edge $e = (u, v)$ for some $u \in V$, we saythe edge e is a *backlink* of v. Informally, the node u is interchangeably called a backlink of v.

The **rank** of a page depends on that of its backlinks. Decentralised in nature, this gives rise to a core quality of PageRank —- it is not easily susceptible to exploitation. Using this centrality in our example above, a negligible amount of fraudulent rank is propagated from L to P because pages in L receive very low rankings. In contrast, if an important page such as Wikipedia's front page links to P, then it propagates high rank to P.

Intuitively, PageRank can be modelled by a "random surfer" – a simulated progression through web pages by clicking links at random. That is, a random walk traversing G. The computed ranking is the *limiting distribution* of this walk. Note that the surfer may visit a page spontaneously without using a link, which is treated as an external influence of sorts (e.g. visiting via direct URL). This practical aspect of the surfer proves useful and is addressed before we state the PageRank definition.

The associated search engine implementation of PageRank was named **Google**, which excelled in general accuracy, particularly for *underspecified* queries. Engines using PageRank typically use filters or tuning to alter the distribution to be more relevant to the query. PageRank has since become a staple for computing diffusion-like rankings and there are countless adaptations of it in every STEMM field and more, such as media and sociology[12].

3.2 Structural challenges and teleportation

3.2.1 Simple PageRank

For B_u denoting the set of backlinks of a page u and N_p the number of backlinks of p, a simple ranking is an assignment R satisfying

$$R(u) = c \sum_{p \in B_u} \frac{R(p)}{N_p},$$

for all pages $u \in V$, where $c < 1$ is a normalisation constant. A page propagates an *equal fraction* of its rank through each of its forward links, which parallels the random choice made by the surfer. This property allows us to use the transition matrix defined earlier: $P = D^{-1}A$, where A is the adjacency matrix of G and D the diagonal matrix of *out-degrees*. The entries of P comprise the values $\frac{1}{N_p}$. Treating R as a vector over V, we want to analyse the dominant eigenvector $R = cP^T R$ to determine our results.

3.2.2 Sinks and dangling links

The above definition does not appear to tackle the exploitation problem we discussed. It admits a few problems in addition to this.

Firstly, this equation's sole reliance on direct recursion causes it not to branch across *disconnected components* of G, so global reach is not achieved by iterating a surfer from one seed node. The formula thus risks components receiving disproportionate rankings; satisfying assignments to disjoint components can be scaled differently in the overall assignment. This implies it may be useful to include a type of *automatic re-seeding* in an improved definition, to enforce each node's global influence.

Secondly, the presence of **rank sinks** is an issue:

Definition 3.3 (Rank sink)**.** A subset $S \subset V$ such that there is no edge in E from a node in S to a node in $V \setminus S$ is called a *rank sink*. That is, $\forall u \in S, \forall v \in V \setminus S : (u, v) \notin E$.

Remark. This is a broad definition, but captures the property we are concerned about.

Rank sinks are problematic asymptotically because rank accumulates in them and never escapes. Consequently, the walk converges to a degenerate distribution concentrated in the sinks.

Thirdly, **dangling links** are also problematic:

Definition 3.4 (Dangling link)**.** A page in V with no outgoing edges is called a *dangling node*. Edges in E with dangling nodes as endpoints are called *dangling links*.

Remark. In essence, a sink is a module of nodes acting as a collective dangling node. This structural pattern is discussed in Section 4.

As dangling nodes do not propagate rank, it is unclear what their relative rankings to other nodes should be. They violate the stochastic property of P as they correspond to zero rows.

While it is possible for a graph to evade all three of these problems by virtue of its structure, a large, noisy, realistic graph of the web will certainly exhibit them. Thus, the PageRank definition must handle them.

3.2.3 Teleportation

The first two issues above are handled by **teleportation**. This denotes the surfer occasionally jumping to a page in V according to some distribution even if no edge in E represents such a transition. All nodes are effectively interconnected through teleportation, ensuring global reach from any starting node. Moreover, rank diffuses from sinks via teleportation, with an intensity dependent on how frequently the surfer teleports.

A common solution to the third issue is to modify P's dangling-node rows to *point uniformly* to each of the n pages by replacing them with normalised $\mathbf{1}^T$ vectors[13] (where $\mathbf{1} = (1, ..., 1)^T$). This effectively removes all dangling links and minimally affects computed ranks in the body of G. Note that this is analogous to a forced teleportation according to a uniform distribution whenever the surfer is on a dangling node. By all means, the choice of distribution can be more nuanced than uniform.

3.3 Expressing PageRank

3.3.1 Original statement and computation

Revising our system with the concept of *teleportation* equips us with a virtual *strongly connected* graph G' for our surfer and a (properly) row-stochastic transition matrix P' after dangling node adjustment. PageRank assumes this revision and incorporates teleportation by expressing $R(u)$ as a weighted sum using *Simple PageRank* (3.2.1) and an additional source of rank $E(u)$ for jumps. We are now ready to state the authentic definition of PageRank.

Definition 3.5 (**PageRank**)**.** As mentioned, let $E \in \mathbb{R}^n_{\geq 0}$ be a **fixed** source of rank and define $E(u) \in \mathbb{R}_{\geq 0}$ as its page-indexed entry for $u \in V$. Let $\|R\|_1 = \sum_{i=1}^n |R_i|$ refer to the *L1-norm* of R. The original publication[11] defines the PageRank of V as an assignment R' to V satisfying

$$R'(u) = c \sum_{p \in B_u} \frac{R'(p)}{N_p} + cE(u), \tag{1}$$

such that c is maximised and $\|R'\|_1 = 1$.

Its *asymptotic* matrix form is $R' = c(P'^T R' + E)$. Note that by $\|R\|_1 = 1$ and $R_i \geq 0$ we have $E = E\sum_1^n R_i = E\mathbf{1}^T R'$. Then it can be factorised as $R' = c(P'^T + E\mathbf{1}^T)R'$, where $E\mathbf{1}^T$ is a rank-one outer product with each column equal to E. The ranking can thus be obtained by computing the dominant eigenvector of $(P'^T + E\mathbf{1}^T)$, which we know to be unique by the Perron-Frobenius Theorem (2.2).

Nevertheless, PageRank is not computed by explicitly calculating the dominant eigenvector in practice. An elementary question to tackle is, why not? Simply, the graph of the web is too large. Eigendecompositions are typically $\mathcal{O}(n^3)$, making it a computationally infeasible task for the web graph. The PageRank algorithm delivered alongside the definition uses *power iteration* – repeatedly applying the transition matrix to an initial vector over V and normalising the result until it is sufficiently close to the dominant eigenvector. See the algorithm (1).

Algorithm 1: PageRank by power iteration [11]

Input: Vector S over the web pages V,
Transition matrix P',
Teleportation vector E,
Tolerance ϵ.
Output: PageRank vector R'.
$R'_0 \leftarrow S$;
repeat
  $R'_{i+1} \leftarrow P'^T R'_i$;
  $d \leftarrow \|R'_i\|_1 - \|R'_{i+1}\|_1$;
  $R'_{i+1} \leftarrow R'_{i+1} + dE$;
  $\delta \leftarrow \|R'_{i+1} - R'_i\|_1$;
until $\delta \leq \epsilon$;

In this case, normalisation happens by construction in the algorithm: $\|dE\|_1$ accounts for the norm difference between R'_i and R'_{i+1}. Its time complexity is roughly linear in $\log n$ and it is reasonable to expect 30-100 iterations for convergence in most practical uses.

3.3.2 Damping form for Personalised PageRank

In the original definition, the magnitude of E constrains the value of c. Since $\|R'\|_1 = 1$, if E is all positive or the sum of its components is large in magnitude, clearly c must be reduced. So, the size of E dictates the proportions of contributions to $R'(u)$ from random link propagation versus teleportation.

Alternatively, modern publications tend to delegate this weighting to the PageRank definition rather than that of E by using a *damping form*[14], defining the required satisfying assignment to be

$$R'(u) = \alpha \sum_{p \in B_u} \frac{R'(p)}{N_p} + (1 - \alpha)E(u),$$

where $\alpha \in (0, 1)$. E in this case is a *teleportation distribution vector*, with non-negative entries and a component sum of 1, and α is called the *damping factor* or *teleportation parameter*. This formulation decouples the probability that the surfer teleports from the actual teleportation distribution, simplifying parameter interpretations and giving finer parameter tuning.

The damping factor α must be chosen appropriately so that teleportation is frequent enough to diffuse rank from sinks, but also does not inhibit the random walk process from providing insightful structural data about the graph. The convention has been to use the battle-tested

$\alpha = 0.85$ since the original publication (equivalently $\|E\|_1 = 0.15$ in equation 1), expecting a jump every 6.7 links clicked.

Personalised PageRank (PPR) is an umbrella term for variations of PageRank that reshape E from the standard PageRank's uniform distribution. It is motivated by the desire to make PageRank context-sensitive; realistic surfers have a pool of sites that are more likely to be jumped to, such as bookmarked sites or homepages. Intuitively, the teleportation distribution has a powerful effect on the produced ranking since the nodes in a neighbourhood of this pool are hit more frequently. E is the main parameter adjusted for different applications, so having it normalised makes for good comparisons.

3.4 Practicality

For one search query, a realistic surfer would only traverse a subset of web pages that are of interest. What if we want to *efficiently* compute a ranking and discard irrelevant nodes, leaving a *dense* subset of relevant ones? An attempt could be to lump the teleportation distribution to a small set of nodes, apply PageRank, and pluck low-rank nodes. But, there is no guarantee that the results are tightly interconnected, rather that they are just related to the seed. We can use a standardised approach to ensure that the subset we determine is of **good quality** and **interconnection**.

All quoted results are to be referenced from Andersen, Chung and Lang [15]. We discuss an application of a PPR assuming that the E vector and starting vector are equally a *single node* – a one-hot vector for $v \in V$ ($\mathbf{e}_v$ in the standard basis). Denote this by $s := e_v$. Their PPR has the formula $\mathrm{pr}_\alpha(s) = ts + (1-t)\mathrm{pr}_\alpha(W^T s)$ where $W = \frac{1}{2}(I + D^{-1}A)$, which they state is equivalent to our PPR definition up to a change in t. Note that they use t as "teleport probability" – that is, $t = 1 - \alpha$ from our damping form definition.

3.4.1 Local approximation of a PageRank vector

Many applications of PPRs apply the PageRank algorithm to the whole of G' before further analysis but here we do not because efficiency is desired. Their paper states and proves an ϵ-tolerant **local approximation** algorithm `ApproximatePR`$(s \in V, t, \epsilon)$ to a PageRank vector with a runtime of $\mathcal{O}(\frac{1}{\epsilon t})$. It is a *push algorithm*.

Briefly, two vectors are maintained, p of final probabilities (initially $\mathbf{0}$) and a temporary store r of probability mass (initially s). A `push`(u) operation adds $t.r(u)$ to $p(u)$ and then spreads half of the remaining $(1-t)$ fraction within r to adjacent nodes via a lazy random walk. Pushing is repeated until entries of r have been sufficiently minimised. The resulting p is ϵ-close to $R'(s)$ (the actual PageRank value). The intuition is that because E is one-hot, the push algorithm's localised spread of probability mass around v is very similar to the PageRank vector from repeated teleportation to v.

Remark. The order of nodes chosen for pushing *does* affect the output. Computations typically use a FIFO queue or a greedy choice (choosing to push u with largest $r(u)$) for the order decision. Output variation is fine given a small tolerance ϵ (as outputs will be similar).

By the semantics of `push()`, nonzero ranks are given to nodes orbiting u. The network representation of connections in such an orbit is called an *ego network*. Once we compute an approximation u' of a PageRank vector, we acknowledge that the rankings are intentionally *biased* near u. We will keep this in mind while we construct some sets that are candidates for a local community.

3.4.2 Bounding conductance for a credible community

Definition 3.6 (Sweep cut)**.** Given a PageRank vector p with $N_p = |\text{Supp}(p)|$, consider an ordering on its nodes $v_1, ..., v_N$ based on *probability per degree*: $p(v_i)/d(v_i) \geq p(v_{i+1})/d(v_{i+1})$. A degree-normalised **sweep cut** over p is the collection of sets:

$$\{v_1\} \subset \{v_1, v_2\} \subset \cdots \subset \{v_1, v_2, \ldots, v_{N_p}\}.$$

We now have an approximate PageRank vector v and candidate sets from a sweep cut. The reason we consider sets from a sweep cut is for the proposition below. A sensible way to evaluate whether these sets are ideal communities is by observing their conductances: recall that a *low* conductance is desirable here, and it is not biased (unlike v) because it is a graph property.

Proposition 3.1. *For chosen positive t and ϵ, let C be a set with conductance $\mathcal{O}(t)$. There exists a subset $C_t \subseteq C$ such that for any $v \in C_t$, performing a sweep cut over* `ApproximatePR`(v, t, ϵ) *produces a set with conductance $\mathcal{O}(\sqrt{t \log(\text{vol}(C))})$.*

That is, for *many* of the nodes v in C, consider the approximated PageRank vector v' using v as the seed node. This says that at least one of the sets in the sweep cut of v' will have a conductance of the order stated above.

Proof. See the referenced paper. □

Under "nice" assumptions, Proposition 3.1 shows that for appropriate t and ϵ, the set in the sweep cut with *minimal conductance* is provably credible as a community. It can be found in $\mathcal{O}(\text{vol}(\text{Supp}(p)) + N_p \log N_p)$.

Remark. The seed v need not be in the community. That is, v_1 is not necessarily equal to v.

In practice, there are imperfections. The procedure should be applied to multiple seed nodes for comparison and safety, since Prop 3.1 does not apply to all nodes in C, meaning it can fail for "bad" nodes – the results hinge on the seed node. Also, the bound $\mathcal{O}(\sqrt{t \log(\text{vol}(C))})$ is loose and could be worse than $\mathcal{O}(t)$. Lastly, it relies solely on conductance, so more sophisticated measures would outperform it. It does, however, outline the measures that can be taken for model reduction and efficient computation.

We have compared *centralities*, stated the original *PageRank* formulation after addressing realistic and structural challenges, expanded on that with *Personal PageRank*, and showed practicalities with approximations and a PPR for performing *local community detection*. Importantly, this section develops an appreciation for the contrast between local and global uses of random walks. Something we leave to explore is the discrepancy between the precise definition of "community" for directed and undirected graphs! PageRank is a global centrality – its local applications typically use it as a preprocessor before doing analysis on the *produced vector*.

In the next section, quantities are defined meticulously to capture layers of locality in the *graph* structure.

4 Severability

4.1 Motivation

While PageRank is a good **global** measure of a node's importance within a graph, acting as a centrality measure for the web page nodes within the larger 'Web graph', there is also a need for **local** quality measures that can be used to evaluate subsets of nodes.

Severability, as introduced by Yu *et al.*[16], can be described as a quantifier for how 'coherent' a subset of nodes is under a random walk. Intuitively, a severable subset (or 'component') is one where a random walker mixes quickly *inside* it but escapes only slowly to the *outside*.

In the authors' words, severability "measures how similarly the entire component exchanges probability flows with its surroundings, compared to the component aggregated into a single node". In other words, if we treated the whole subset as one macro-state, the random-walk behaviour should be nearly the same as in the original network.

This captures the idea of local time-scale separation: the subset behaves almost autonomously for some time interval (a 'local well' in the Markov landscape), even if the overall graph is complex. Unlike global clustering methods, severability is defined for any candidate subset without the need to partition the entire graph.

4.2 Defining severability via random walk dynamics

We consider the following thought experiment.

Suppose we release a random walker into a small region A of a graph $G = (V, E)$. The region is densely connected internally, but only weakly linked to the rest of the graph. How long will the walker stay inside A? And, once inside, how quickly will its movement become unpredictable or uniform across A?

We assume throughout that G is a finite undirected graph with positive edge weights, and that every node has at least one outgoing edge. Let P be the row-stochastic transition matrix defining the random walk on G (see 1.1). That is, P_{ij} gives the probability of transitioning from node i to node j in one step.

Let $A \subset V$ be a candidate subgraph, and let Q denote the submatrix of P corresponding to transitions between nodes in A (i.e., transitions that begin and end in A).

4.2.1 Retention

Intuitively, **retention** measures how likely a walker is to remain within A over a given number of time steps τ, assuming it starts uniformly at random in A.

Definition 4.1 (Retention)**.** Let $A \subset V$ with $|A| = k$, and let $Q \in \mathbb{R}^{k \times k}$ be the transition submatrix induced on A. The *retention* over time τ is defined as

$$\rho_\tau(A) = \frac{1}{k} \mathbf{1}^\top Q^\tau \mathbf{1},$$

where $\mathbf{1} \in \mathbb{R}^k$ is the all-ones vector.

This quantity represents the expected probability that the walker, starting from a uniform distribution on A, remains in A after τ steps — i.e., how "leaky" the set is over time.

4.2.2 Mixing

While retention tracks escape behavior, it doesn't tell us whether the walk has become "well-distributed" inside A. For that, we look at **mixing**: the extent to which the walk loses memory of its starting point — again, conditioned on it staying in A.

To compute this, we normalize each row of Q^τ (the τ-step transition probabilities for each start node in A) into a probability distribution over A, and measure how similar those distributions are to each other.

Definition 4.2 (Mixing)**.** Let $q_i^{(\tau)} \in \mathbb{R}^k$ be the i-th row of Q^τ. Define the conditional distribution $\tilde{q}_i^{(\tau)} = q_i^{(\tau)}/q_i^{(\tau)}\mathbf{1}$. Let $\bar{q} = \frac{1}{k}\sum_{i=1}^k \tilde{q}_i^{(\tau)}$ be the average conditional distribution.

Then the *mixing* of A at time τ is given by

$$\mu_\tau(A) = 1 - \frac{1}{k}\sum_{i=1}^{k} \left\| \tilde{q}_i^{(\tau)} - \bar{q} \right\|_{\mathrm{TV}},$$

where the total variation norm is defined by

$$\|\mathbf{v}\|_{\mathrm{TV}} = \frac{1}{2}\|\mathbf{v}\|_1 = \frac{1}{2}\sum_{j=1}^{k} |v_j|.$$

A high mixing score means that no matter where the walker started in A, its distribution over A has converged (conditioned on not leaving). This reflects strong internal connectivity and fast equilibration.

4.2.3 Severability

We are now ready to define **severability**. It combines both retention and mixing into a single quantity that captures the overall 'coherence' of A as a mesoscale component[17], one that traps the walk and mixes internally before the walker escapes.

Definition 4.3 (Severability)**.** The *severability* of $A \subset V$ at time τ is defined as the average of its retention and mixing:

$$\sigma_\tau(A) = \frac{\rho_\tau(A) + \mu_\tau(A)}{2}.$$

Values of $\sigma_\tau(A)$ close to 1 indicate that A behaves like an isolated "basin" of diffusion: the walker stays confined for many steps and equilibrates inside before leaking out. Such sets are of particular interest in identifying communities, local modules, or coherent features in complex networks.

Example (Triangle in a Path Graph). Consider a graph formed by a triangle $A = \{v_1, v_2, v_3\}$ attached via a weak edge to a long path. Starting from any node in A, a random walker is likely to stay in A for a long time due to its high internal connectivity and weak external links. Moreover, due to the triangle's symmetry, the distribution of the walker quickly becomes uniform over A, regardless of the starting node. Thus, $\rho_\tau(A)$ and $\mu_\tau(A)$ are both high for moderate τ, making A highly severable at that time scale.

4.3 Applications

4.3.1 Local Graph-based dictionary expansion

Building on severability's capability to identify coherent local substructures by utilising the random walk on a graph, Schindler *et al.*propose **Local Graph-based dictionary expansion** (LGDE)[18] for semantic keyword discovery. This method leverages severability to extract contextually relevant word communities from a geometric graph of word embeddings.

First, given a vocabulary $V \subseteq \mathbb{R}^r$ of word embeddings, LGDE constructs a semantic similarity graph $G^{(k)} = (V, E^{(k)})$ using the *Continuous k-Nearest Neighbours* method[19].

Then, for each seed keyword $w \in W_0$, LGDE computes its semantic community:

$$C^{(k,t)}(w) = \underset{C \subseteq V}{\operatorname{argmax}}\, \sigma_t(C),$$

where $\sigma_t(C) = \frac{\rho_t(C)+\mu_t(C)}{2}$ is the severability of C at Markov time t (Definition 4.3).

The expanded dictionary is the union of overlapping communities: $W(k,t) = \bigcup_{w \in W_0} C^{(k,t)}(w)$.

By utilising severability, LGDE addresses key limitations of prior methods:

- **Chain of association**: Unlike thresholding or kNN, which are reliant on direct similarity, severability captures multi-step paths between words.
- **Adaptive scaling**: Communities automatically adjust size/sparsity via k (graph density).
- **Weighted semantics**: The retention-mixing balance leverages edge weights, unlike combinatorial methods.

As a result, LGDE outperforms baselines across two different benchmarks: *hate speech detection* and *4chan conspiracy analysis*. This stems from severability's ability to isolate *semantically autonomous* word groups.

4.3.2 Twitter Interaction Graph for the US Congress

We can apply LGDE's notion of a 'semantic community' to analysing a Twitter interaction graph for the 117th United States Congress. The given dataset, based off a six-month period of interactions, represents a *directed*, *weighted* graph, where edge weights are "probabilities of influence"[20] between all pairs of Congresspeople. Essentially, these probabilities of influence are equal to the normalised sum of 'influence' — the frequency with which each member of Congress retweeted, quote tweeted, replied to, or mentioned other Congressional members.

By converting the given graph to its undirected form (see A), we were able to apply **severability**. Our goal was to first find severable components of Congresspeople, and secondly to determine the most influential seed nodes within the graph, occurring frequently in other nodes' severable components, to establish which Congresspeople most 'dominate' Twitter interactions.

For the combinations of Markov time and seed nodes (Congresspeople) where we were able to find an optimal severable component, those components had severability values **exclusively in the range 0.4 to 0.7**, with the distribution becoming more concentrated around 0.5 over time, as seen in figure 2. This indicates a decent level of retention and mixing for random walks starting at a variety of seed nodes.

To deduce the Congresspeople with the most 'far-reaching' interaction components, we constructed a (symmetric) matrix J, where the element J_{ij} represents the *Jaccard similarity* between the severable component with Congressperson/node i as a seed, and the corresponding copmponent with Congressperson/node j as a seed, averaged out over the **first 5 Markov timesteps**.

Due to the large size of our severable components (31.28, on average), the average Jaccard similarity scores (i.e. the row/column averages of J) are lower, with a fairly even distribution in the range between 0 and 0.13 for each Congressperson. There is a noticeable positive linear correlation between this, and the mean severability value for Congresspeople, as seen in Figure 3. Noticeably, the tail-off in Jaccard similarity is sharp for components with a lower severability, but this is understandable as this would represent Congresspeople with generally low levels of Twitter interaction.

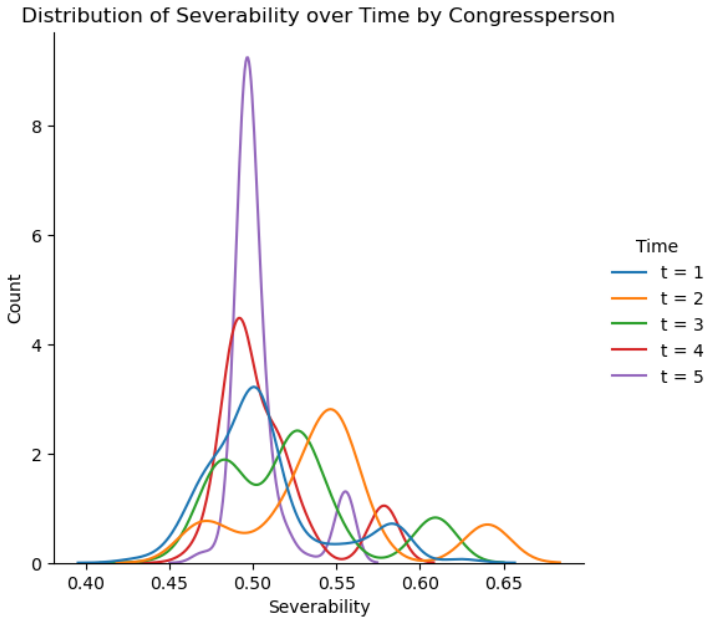

Figure 2 The distribution of severability values at different Markov timesteps, for the optimal severable components generated from certain Congresspeople / seed nodes.

Interestingly, when considering the primary divide between Congresspeople: whether they sit in the Senate or the House of Representatives, it becomes clear that **representatives dominate Twitter interaction volume** compared to senators. The top 33 Congresspeople by average Jaccard similarity are all representatives, mainly due simply to the larger size of their severable components. Perhaps the age divide between the two houses had a role to play — the average age of Members of the House at the beginning of the 117th Congress was 58.4 years; of Senators, 64.3 years[21].

However, when it comes to considering the most 'influential nodes' — which Congresspeople appear the most in others' severable components, **senators are much more prominent**. Although the top eight Congresspeople, with more than occurrences each, are still representatives, there are 8 senators out of the next 19 in the ranking. Intuitively, this makes sense: senators maybe do not generate large and far-reaching severable components but due to the Senate's status as the upper house of US legislature, they are probably interacted with more by other Congresspeople.

Overall, our severability analysis of the Twitter interaction graph revealed distinct patterns of influence, demonstrating severability's power in decoding local clusters within a larger network. It is clear that this is an incredibly relevant and wide-reaching application of the theory behind random walks on graphs.

5 Conclusion

We highlighted *foundational stochastic properties* of random walks, using *reversibility* and *harmonic functions* to describe how information diffuses across networks over time. We expanded on this with generalised properties for random walks on arbitrary graphs. By presenting these constructions, we aim to build physical intuition for the dynamics of random walks. The existence and convergence of a *stationary distribution* underline the modelling appeal of random walks. The *spectral gap*, alongside conductance as a closely related connectivity measure, lets

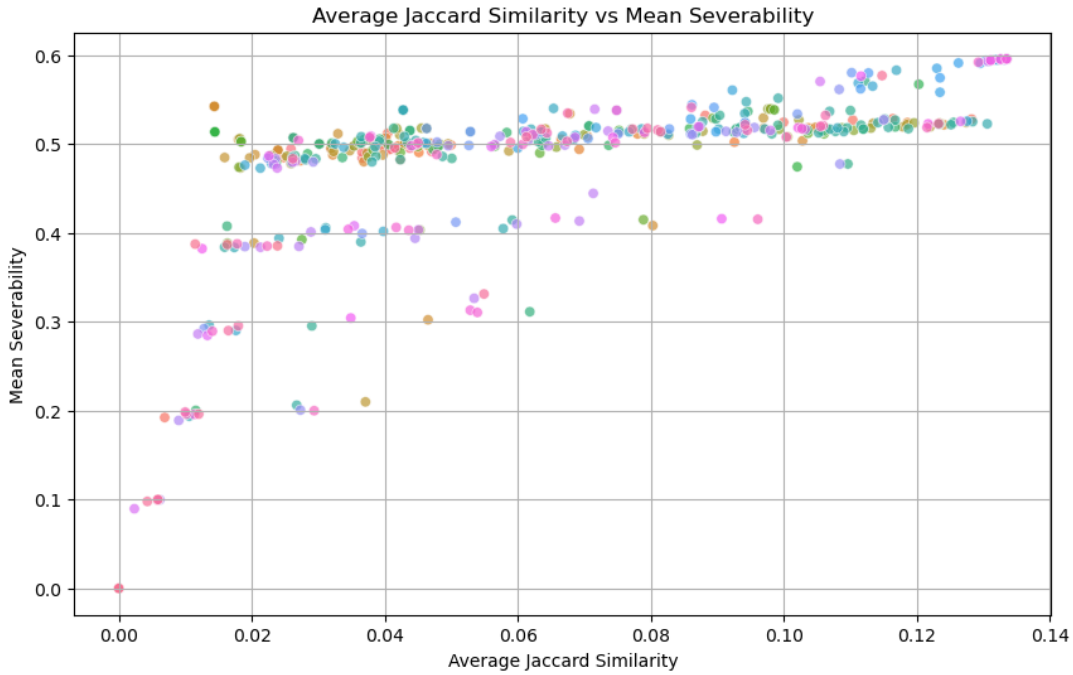

Figure 3 The mean Jaccard similarity scores for Congresspeople against their mean severability value, averaged out over Markov time $t = 1$ to 5.

us analyse the convergence rate. These results form a backbone of spectral properties that help us predict and identify behaviours for applied methods.

PageRank is a fundamental global centrality inspired by random walks, and its versatile teleportation distribution paves the way to the ubiquitous Personalised PageRank. The local PageRank approximation via push algorithm explores the computational and heuristic side of random walks, providing an insight into how large systems like search engines apply model reduction using a biased walk and sweep cuts to find a low-conductance community of relevant data. **Severability** extends this as an emerging local community detection method and offers flexibility and efficiency by allowing overlapping mesoscale components. Despite severability being a recent introduction, we have seen its effectiveness compared to other measures for LGDE, and the insights gained from analysing the Twitter interaction graph.

Looking ahead, we see several directions: further optimisation of local modularity measures; integrating walk-based metrics into machine learning pipelines; and extending these ideas to dynamic or multi-layer networks. Specifically, for severability, it remains to be seen if the measure transforms accurately to directed graphs. Random walks are not just a theoretical curiosity — they offer a unifying framework for designing, interpreting, and improving algorithms across disciplines. The challenge now is to harness their full potential in the data-rich, structure-heavy problems that define modern applied mathematics.

Acknowledgments

Many thanks to our supervisor, Professor Barahona, for his support throughout, and to Juni Schindler and Tim Liu for their help with various applications.

A Appendix — Computing Severability

To convert Fink *et al.*'s directed Twitter interaction graph into an undirected form: take an edge (u, v). If the corresponding edge (v, u) exists, then $W'((u, v)) = W'((v, u)) = W((u, v)) + W((v, u))$, where W represents the initial weight and W' the new weight of an edge. Otherwise, simply add a new edge (v, u) with the same weight as (u, v).

Although severability is only strictly defined for undirected graphs, the computations of retention, mixing, etc. can also be performed on a directed graph, i.e. one where the adjacency matrix is not symmetric. We briefly experimented with a similar analysis on such an adjacency matrix for our network, as well.

Our code for computing the severable components of the Twitter interaction graph is stored in a GitHub repository, specifically in the `sev_gen_undirected.ipynb` file. We create the graph from our provided 'edgelist' using the `networkx` library, generate the adjacency/transition matrices, and then call the `severability.node_component` function, taken from the library that supported Yu *et al.*'s paper[16], with every node as a seed. Then, we perform the analysis described in section 4.3.2. Further details on that analysis, and specific values for individual Congresspeople, can be found in the `sev_analysis_undirected.ipynb` file.

References

[1] D. Weld. Chapter 3: Problem solving using search, 2012. CSE 473: Introduction to Artificial Intelligence.

[2] R. Burioni and D. Cassi. Random walks on graphs: ideas, techniques and results. *Journal of Physics A: Mathematical and General*, 38:R46–R53, February 2005.

[3] G. Grimmett. Probability on graphs, April 2010.

[4] L. Lovász. Random walks on graphs: A survey, 1993.

[5] E. Seneta. *Non-negative matrices and Markov chains*. Springer Science & Business Media, 2006.

[6] L. László, L. Lov, and O. Erdos. Random walks on graphs: A survey. In *Paul Erdös is Eighty (Volume 2)*, pages 1–46, 01 1996.

[7] J. R. Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

[8] G. L. Jones. On the markov chain central limit theorem. *Probability Surveys*, 1(299-320), Jan 2004.

[9] J. Cheeger. A lower bound for the smallest eigenvalue of the laplacian. *Problems in analysis*, 625(195-199):110, 1970.

[10] F. R. Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[11] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[12] D. F. Gleich. Pagerank beyond the web. *SIAM Review*, 57(3):321–363, 2015.

[13] F. Arrigo, D. J. Higham, and V. Noferini. Non-backtracking pagerank. *Journal of Scientific Computing*, 80(3):1419–1437, 2019.

[14] N. Masuda, M. A. Porter, and R. Lambiotte. Random walks and diffusion on networks. *Physics Reports*, 716–717:1–58, Nov. 2017.

[15] R. Andersen, F. Chung, and K. Lang. Local graph partitioning using pagerank vectors. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 475–486, 2006.

[16] Y. W. Yu, J.-C. Delvenne, S. N. Yaliraki, and M. Barahona. Severability of mesoscale components and local time scales in dynamical networks. 2020. arXiv preprint arXiv:2006.02972.

[17] H. Lyu, Y. H. Kureh, J. Vendrow, and M. A. Porter. Learning low-rank latent mesoscale structures in networks. *Nature Communications*, 15(1):224, Jan 2024.

[18] J. Schindler, S. Jha, X. Zhang, K. Buehling, A. Heft, and M. Barahona. Lgde: Local graph-based dictionary expansion. *Computational Linguistics*, pages 1–32, 04 2025.

[19] T. Berry and T. Sauer. Consistent manifold representation for topological data analysis. *Foundations of Data Science*, 1(1):1–38, 2019.

[20] C. G. Fink, N. Omodt, S. Zinnecker, and G. Sprint. A congressional twitter network dataset quantifying pairwise probability of influence. *Data in Brief*, 2023.

[21] J. E. Manning. Membership of the 117th congress: A profile, Dec 2022. Available from https://crsreports.congress.gov/product/pdf/R/R46705.