An Aligned Subtree Kernel for Weighted Graphs

Lu Bai BAILU69@HOTMAIL.COM
School of Information, Central University of Finance and Economics, Beijing, China

Luca Rossi L.ROSSI@CS.BHAM.AC.UK
School of Computer Science, University of Birmingham, Birmingham, UK

Zhihong Zhang ZHIHONG@XMU.EDU.CN
Software School, Xiamen University, Xiamen, Fujian, China

Edwin R. Hancock ERH@CS.YORK.AC.UK
Department of Computer Science, University of York, York, UK

Abstract

In this paper, we develop a new entropic matching kernel for weighted graphs by aligning depth-based representations. We demonstrate that this kernel can be seen as an **aligned subtree kernel** that incorporates explicit subtree correspondences, and thus addresses the drawback of neglecting the relative locations between substructures that arises in the R-convolution kernels. Experiments on standard datasets demonstrate that our kernel can easily outperform state-of-the-art graph kernels in terms of classification accuracy.

1. Introduction

Graph kernels are powerful tools for structural analysis in machine learning. The main advantage of using graph kernels is that they provide an implicit embedding of graphs in a high dimensional space where structural information is better preserved. Most of the existing graph kernels are instances of the R-convolution kernels proposed by Haussler (Haussler, 1999). This is a generic way of defining graph kernels by decomposing the input graphs into smaller and simpler substructures and comparing the original graphs in terms of these substructures. The R-convolution kernels can be categorised into the following classes, namely graph kernels based on comparing all pairs of a) walks (Jebara et al., 2004), b) paths (Borgwardt & Kriegel, 2005), c) cycles (Aziz et al., 2013), and d) subgraph or subtree structures (Shervashidze et al., 2009; Kriege & Mutzel, 2012; Bai et al., 2014a;c). One drawback arising in the R-convolution kernels is that they neglect the relative locations of substructures. This is because the R-convolution kernels cannot establish reliable structural correspondences between substructures. Moreover, most R-convolution kernels cannot accommodate weighted graphs, i.e., a graph where each edge is associated with a weight. This is because the distances between pairs of adjacent vertices depend on the edge weight, and it is difficult to identify the isomorphism for weighted (sub)graphs unless the weighted information is discarded. These drawbacks limit the precision of graph kernel measures.

To overcome the shortcomings of existing R-convolution kernels, we propose a novel aligned subtree kernel for weighted graphs by aligning depth-based (DB) representations. In the literature (Crutchfield & Shalizi, 1999; Escolano et al., 2012), DB representations of un-weighted and un-directed graphs are powerful tools for characterising graphs in terms of complexity measures. One approach to computing the DB representation of a graph is based on measuring the entropies on a family of K-layer expansion subgraphs centred on the vertex having a maximum topological distance K to the remaining vertices (Bai & Hancock, 2014). Unfortunately, computing the DB representations for edge-weighted graphs tends to be an elusive task. This is because the distances between different pairs of adjacent vertices depend on the edges weight, making it harder to determine the K-layer expansion subgraphs. To overcome this problem, for each vertex of a weighted graph G_W, we compute its DB representation, i.e., its *entropic representation*. More precisely, we compute the entropic representation on the vertices of the quantum-directed graph G_D that is obtained from G_W through the simulation of a CTQW (Emms et al., 2009). We show that the

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

CTQW can reflect rich characteristics (e.g., the topological and the weighted information) of weighted graphs (see details in Section 2.1). Thus, we show that the entropic representation of the vertices of G_D reflects both the topological and weighted information of G_W. Moreover, we define a new TI method to strengthen the vertex labels in G_W through G_D. We show that each strengthened label of G_W corresponds to a non-backtracking subtree in G_D. Based on the obtained entropic representations for two weighted graphs, we develop a new entropic matching and identify the correspondence between vertices having the same strengthened labels. Finally, we compute the matching-based kernel for weighted graphs by counting the number of matched vertex pairs. We theoretically show the relationship between our kernel and the classical all subgraph kernel. We demonstrate that the new kernel can be seen as an **aligned subtree kernel** that incorporates explicit subtree correspondences. As a result, our kernel not only accommodates both un-weighted and weighted graphs, but also addresses the drawback of neglecting the relative locations between substructures arising in the R-convolution kernels. Experiments on standard graph datasets demonstrate that our kernel can easily outperform state-of-the-art graph kernels in terms of classification accuracy.

2. Preliminary Concepts

2.1. A Quantum-directed Graph through The CTQW

The CTQW represents the quantum counterpart of the continuous-time random walk (CTRW) (Emms et al., 2009). The advantages of using the CTQW are threefold. First, unlike the classical CTRW, the CTQW is reversible, non-ergodic and does not possess a limiting distribution. As a result, the CTQW reduces the tottering problem arising in the CTRW (Kempe, 2003). Second, the CTQW uses qubits rather than bits as the basic representational unit (Nielsen & Chuang, 2000). Thus, the CTQW can capture rich topological information in a graph structure (Aubry et al., 2011; Rossi et al., 2013b; Suau et al., 2013). Third, the CTQW is not dominated by the low frequencies of the Laplacian spectrum, and thus is able to discriminate better among different graph structures.

Let $G_W(V,E)$ be a weighted graph with vertex set V, edge set E, and a weight function $\omega : V \times V \rightarrow \mathbb{R}^+$. If $\omega(u,v) > 0$ ($\omega(u,v) = \omega(v,u)$), we refer to (u,v) as an edge of G_W, and we say that $u \in V$ and $v \in V$ are adjacent. Using the Dirac notation, the basis state corresponding to the CTQW being at a vertex $u \in V$ is defined as $|u\rangle$. A general state of the CTQW at time t is $|\psi_t\rangle = \sum_{u\in V} \alpha_u(t)\,|u\rangle$, where the amplitudes $\alpha_u(t) \in \mathbb{C}$. The probability of the CTQW visiting a vertex $u \in V$ at time t is given by $\Pr(X^t = u) = \alpha_u(t)\alpha_u^*(t)$, where $\alpha_u^*(t)$ is the complex conjugate of $\alpha_u(t)$. Let A be the adjacency matrix of G_W, which satisfies $A(u,v) = w(u,v)$. The degree matrix D is a diagonal matrix whose elements are given by $D(u,u) = d_u = \sum_{v\in V} A(u,v)$, where d_u is the degree of u. We compute the Laplacian matrix as $L = D - A$. The spectral decomposition $L = \Phi^\top \Lambda \Phi$ of the Laplacian matrix L is given by the diagonal matrix $\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_{|V|})$ with the ordered eigenvalues as elements ($\lambda_1 < \lambda_2 < ... < \lambda_{|V|}$) and the matrix $\Phi = (\phi_1|\phi_2|...|\phi_{|V|})$ with the corresponding ordered orthonormal eigenvectors as columns.

Given an initial state $|\psi_0\rangle$, the Schrödinger equation gives us the state of the walk at time t, i.e.,

$$|\psi_t\rangle = \Phi^\top e^{-i\Lambda t}\Phi\,|\psi_0\rangle\,. \tag{1}$$

In this work, to compute the initial state of the CTQW for the weighted graph $G_W(V,E)$, we first transform G_W into an un-weighted graph G by ignoring the edge weighted information, i.e., for each pair of adjacent vertices $v \in V$ and $u \in V$ we have $\omega(u,v) = \omega(u,v) = 1$. We use the rooting of the vertex degree distribution of G as the initial state.

In quantum mechanics, a pure state can be described as a single ket vector. A quantum system, however, can also be in a mixed state, i.e., a statistical ensemble of pure quantum states $|\psi_t\rangle$, each with probability p_t. The density matrix (or density operator) of such a system is defined as $\rho = \sum_t p_t\,|\psi_t\rangle\,\langle\psi_t|$. Let $|\psi_t\rangle$ denote the state corresponding to a CTQW that has evolved from $|\psi_0\rangle$ until a time t. Using Eq.(1), we can define the time-averaged density matrix (i.e., mixed density matrix) ρ_G^T for G as $\rho_G^T = \frac{1}{T}\int_0^T \Phi^\top e^{-i\Lambda t}\Phi\,|\psi_0\rangle\,\langle\psi_0|\,\Phi^\top e^{i\Lambda t}\Phi\,\mathrm{d}t$. Let ϕ_{ra} and ϕ_{cb} denote the (ra)-th and (cb)-th elements of the matrix of eigenvectors Φ of the Laplacian matrix L. When we let $T \rightarrow \infty$, Rossi et al. (Rossi et al., 2013b) have shown that the (r,c)-th element of ρ_G^∞ can be computed as

$$\rho_G^\infty(r,c) = \sum_{\lambda\in\tilde{\Lambda}} \sum_{a\in B_\lambda} \sum_{b\in B_\lambda} \phi_{ra}\phi_{cb}\bar{\psi}_a\bar{\psi}_b, \tag{2}$$

where $\tilde{\Lambda}$ is the set of distinct eigenvalues of the Laplacian matrix L, and B_λ is a basis of the eigenspace associated with λ. The mixed density matrix ρ_G^∞ is a $|V| \times |V|$ matrix. For a weighted graph G_W, the time-averaged probability of the CTQW to visit a vertex $v \in V$ at time $T \rightarrow \infty$ is

$$p_Q(v) = \rho_G^\infty(v,v). \tag{3}$$

Interestingly, despite the non-stationary behaviour of the CTQW, we have that as $T \rightarrow \infty$ the time-averaged probability converges. Moreover, Eq.(1) indicates that the evolution of the CTQW relies on the spectral decomposition of the graph Laplacian, which encapsulates the weight information residing on the edges. As a result, the probability $p_Q(v)$ not only reflects rich interior topology information but also captures the edge weighted information

of G_W. Note that, the CTQW also accommodates un-weighted graphs. For an un-weighted graph, we just need to guarantee that each entry of the adjacency matrix is 1 if the corresponding pairwise vertices are adjacent, and 0 otherwise. See details in the literature (Bai et al., 2014d).

Quantum-directed graph: Let $G_W(V,E)$ be a weighted graph. Its quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$ is established by replacing each edge by a directed edge through a CTQW (for $T=\infty$) and ignoring the weighted information residing on the edge. More specifically, we simulate a CTQW on G_W and we associate with each vertex $v \in V$ the time-averaged probability $p_Q(v)$. The quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$ is defined as

$$\begin{cases} V_D &= V, \\ \overrightarrow{E}_D &= \{\overrightarrow{(u,v)} |\ (u,v) \in E,\ p_Q(u) \geq p_Q(v)\}. \end{cases} \tag{4}$$

The structure of G_D is represented by a $|V_D| \times |V_D|$ adjacency matrix A_D as follows

$$A_D(u_d, v_d) = \begin{cases} 1 & \text{if } (u,v) \in E \text{ and } p_Q(u) \geq p_Q(v), \\ 0 & \text{otherwise.} \end{cases} . \tag{5}$$

where $u_d \in V_D$, $v_d \in V_D$, $u_d = u$ and $v_d = v$. □

From Eq.(4) and Eq.(5), we observe that the directed-quantum graph G_D has the same vertex set of its original weighted graph G_W. Moreover, the directed edges of G_D are established by comparing the probabilities of the CTQW visiting each pair of adjacent vertices in G_W. Since the probability of the CTQW visiting each vertex on G_W depends on the edge weights information of G_W, the quantum-directed graph G_D can not only preserve the main topological information of the weighted graph G_W, but also reflect the edge weighted information of G_W through the transformed directed edges from G_W.

2.2. A Tree-Index Method for Weighted Graphs

In this subsection, we introduce a new non-backtracking TI method for strengthening the vertex label of a weighted graph $G_W(V,E)$ through its quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$ ($V = V_D$). In other words, we compute the strengthened label for a vertex $v_D \in V_D$, and use this as the strengthened label for a vertex $v \in V$ which corresponds to v_D (i.e., $v = v_D$). Each strengthened label of G_W will corresponds to a subtree on G_D. We commence by computing the m-sphere neighbourhood for each vertex of G_D. For each iteration m, the strengthened label for v_D is computed by taking the union of the original label of v_D and the original labels of the m-sphere neighbourhood vertices of v_D. Moreover, We use a Hash function to compress the strengthened label list into a new short label, and use this as the new label of the corresponding vertex v of G_W. The pseudocode of the TI algorithm is shown in Algorithm 1, where the m-sphere neighbourhood of a vertex $v_D \in V_D$ is denoted as $\mathcal{N}(v_D) = \{u_D \in V_D | S(v_D, u_D) = m\}$ and $S(v_D, u_D)$ is the shortest path length between v_D and u_D.

Algorithm 1 Vertex labels strengthening procedure

1: Initialisation.

- Input a weighted graph $G_W(V,E)$, transform G_W into the quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$.
- Set ***m=0***. For each vertex $v_D \in V_D$ ($v_D = v \in V$), assign the original label as the initial label $\mathcal{L}_m(v_D)$.

2: Update the label for each vertex.

- Set ***m=m+1***. For v_D, assign it a new label list as

$$\mathcal{L}_m(v_D) = \cup_{u_D \in \mathcal{N}(v_D)} \{\mathcal{L}_0(u_D), \mathcal{L}_0(v_D)\}, \tag{6}$$

where $\mathcal{L}_{m-1}(v_D)$ is at the end of $\mathcal{L}_0(v_D)$, and $\mathcal{L}_0(u_D)$ is arranged in ascending order.

3: Compress strengthened label lists into new short labels.

- Using the Hash function $\mathbf{H}: \mathcal{L} \to \Sigma$, compress $\mathcal{L}_m(v_D)$ into a new short label for v_D as $\mathcal{L}_m(v_D) = \mathbf{H}(\mathcal{L}_m(v_D))$.
- The strengthened label for $v \in V$ at m iteration is

$$\mathcal{L}_m(v) = \mathcal{L}_m(v_D). \tag{7}$$

4: Repeat steps 2 to 4 until m achieves a pre-specified value.

Note that, in step 4 we use the same function $\mathbf{H}$ for all the graphs. This guarantees that all the identical labels of different graphs are mapped into the same number. Moreover, for the weighted graph $G_W(V,E)$ and its quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$, each strengthened label of a vertex $v \in V$ corresponds to a subtree of height m rooted at the corresponding vertex $v_D \in V_D$ ($v = v_D$) of G_D, and the subtree consists of the m-sphere neighbourhood of v_D and the shortest paths from v_D to the neighbourhood. For a pair of vertices $u \in V$ and $v \in V$, if the strengthened labels $\mathcal{L}_m(u)$ and $\mathcal{L}_m(v)$ are equivalent, the corresponding subtrees rooted at $u_D \in V_D$ ($u = u_D$) and $v_D \in V_D$ ($v = v_D$) are seen as approximately isomorphic. The new TI method is an inexact TI method. In fact, for a vertex on G_D, there may be more than one shortest path from the root to each vertex in its m-sphere neighbourhood. Moreover, the labels of the vertices in the shortest paths are also discarded. However, compared to the existing TI methods in (Bai et al., 2014d; Shervashidze et al., 2011) which cannot avoid or limit the notorious tottering problem, our TI method can completely avoid the tottering problem. This is because the shortest paths from the root to its m-sphere neighbourhood are non-backtracking structures. Note finally that, for iteration m, if $|\mathcal{N}(v_d) = \{u_d \in V_D | S(v_d, u_d) = m\}| = 0$, we say that there is no strengthened label for the vertex v_d and

the corresponding subtree of height m rooted at v_d does not exist. In other words, the strengthened label of $v = v_d$ for the original weighted graph G_W also does not exist.

3. Entropic Matching for Weighted Graphs

3.1. The Entropic Representation for Weighted Graphs

In this subsection, we compute the DB representation, namely the h-layer entropic representation, around a vertex of a weighted graph $G_W(V, E)$. The advantage of using this representation to characterise graphs is that it not only reflects dominant depth complexity information around each vertex of a graph but it also represents the graph or the corresponding vertex in a high dimensional space (Crutchfield & Shalizi, 1999; Bai & Hancock, 2014). Unfortunately, as we have stated, directly computing this representation tends to be elusive, since it is difficult to determine the required expansion subgraphs based on the pairwise distances between vertices of a weighted graph. To address this problem, we transform G_W into a quantum-directed graph $G_D(V_D, E_D)$ through the CTQW. We compute the h-layer entropic representation of G_W on G_D.

We commence by computing the h-layer un-directed DB representation on the un-directed graph $G_U(V_U, E_U)$, which is established by replacing the directed edges of G_D with bidirectional edges. In other words, G_U can also be seen as a transformed graph of G_W where edge weighted information is ignored. The h-layer DB representation has been previously introduced by Bai et al. (Bai et al., 2014b), by generalising the DB complexity trace or representation around the centroid vertex (Bai & Hancock, 2014).

The h-layer un-directed DB representation: For $G_U(V_U, E_U)$ and a vertex $v_U \in V_U$, let a vertex set $N^K_{v_U}$ be defined as $N^K_{v_U} = \{u_U \in V_U \mid S_G(v_U, u_U) \leq K\}$, where S_G is the shortest path matrix of G_U and $S_G(v_U, u_U)$ is the shortest path length between v_U and u_U. For G_U, the K-layer expansion subgraph $\mathcal{G}^K_{v_U}(\mathcal{V}^K_{v_U}; \mathcal{E}^K_{v_U})$ around v_U is

$$\begin{cases} \mathcal{V}^K_{v_U} = \{u_U \in N^K_{v_U}\}; \\ \mathcal{E}^K_{v_U} = \{u_U, w_U \in N^K_{v_U},\ (u_U, w_U) \in E_U\}. \end{cases} \tag{8}$$

For G_U, the h-layer un-directed DB representation around $v_U \in V_U$ is

$$DB^h_{G_U}(v_U) = [H_S(\mathcal{G}^1_{v_U}), \cdots, H_S(\mathcal{G}^K_{v_U}), \cdots, H_S(\mathcal{G}^h_{v_U})]^\top, \tag{9}$$

where $(K \leq h)$. $H_S(\mathcal{G}^K_{v_U})$ is the Shannon entropy of $\mathcal{G}^K_{v_U}$ associated with the steady state of the random walk (Bai & Hancock, 2013) defined as

$$H_S(\mathcal{G}^K_{v_U}) = -\sum_{u_U \in \mathcal{V}^K_{v_U}} P(u_U) \log P(u_U). \tag{10}$$

Here, $P(u_U) = D_{\mathcal{G}^K_{v_U}}(u_U, u_U) \sum_{w_U \in \mathcal{V}^K_{v_U}} D_{\mathcal{G}^K_{v_U}}(w_U, w_U)$ is the probability of the random walk visiting $u_U \in \mathcal{V}^K_{v_U}$, and $D_{\mathcal{G}^K_{v_U}}$ is the diagonal degree matrix of $\mathcal{G}^K_{v_U}$. □

Note that the h-layer un-directed DB representation $DB^h_{G_U}(v_u)$ can reflect the entropy-based information content flow rooted at each vertex on the original weighted graph G_W in terms of the topological information. To capture the weighted information, we also compute the DB representations on the quantum-directed graph G_D.

The h-layer directed DB representation: For $G_W(V, E)$ and its quantum-directed graph $G_D(V_D, \overrightarrow{E}_D)$. We first establish the K-layer directed expansion subgraph of G_D through $G_U(V_U, E_U)$ and the K-layer expansion subgraph $\mathcal{G}^K_{v_U}(\mathcal{V}^K_{v_U}; \mathcal{E}^K_{v_U})$ on G_U. The K-layer directed expansion subgraph $\mathcal{G}^K_{v_D}(\mathcal{V}^K_{v_D}; \overrightarrow{\mathcal{E}}^K_{v_D})$ around $v_D \in V_D$ is

$$\begin{cases} \mathcal{V}^K_{v_D} = \{u_U |\ u_U \in \mathcal{V}^K_{v_U}\}; \\ \overrightarrow{\mathcal{E}}^K_{v_D} = \{\overrightarrow{(u_U, w_U)} \in \overrightarrow{E}_D |\ (u_U, w_U) \in \mathcal{E}^K_{v_U}\}. \end{cases} \tag{11}$$

where $v_D \in V_D$, $v_U \in V_U$, and $v_D = v_U$. In other words, $\mathcal{G}^K_{v_U}$ can be seen as a transformed graph of $\mathcal{G}^K_{v_U}(\mathcal{V}^K_{v_U}; \mathcal{E}^K_{v_U})$ of G_U associated with the directed edges in G_D. For G_D, the h-layer directed DB representation around $v_D \in V_D$ is

$$\overrightarrow{DB}^h_{G_D}(v_D) = [H_M(\mathcal{G}^1_{v_D}), ..., H_M(\mathcal{G}^K_{v_D}), ..., H_M(\mathcal{G}^h_{v_D})]^\top \tag{12}$$

where $H_M(\mathcal{G}^K_{v_D})$ is the mixed entropy of $\mathcal{G}^K_{v_D}$ defined as

$$H_M(\mathcal{G}^K_{v_D}) = H_V(\mathcal{G}_{v_D}) + H_E(\mathcal{G}_{v_D}). \tag{13}$$

Here, $H_V(\mathcal{G}_{v_D})$ is the approximated directed von Neumann entropy of $\mathcal{G}_{v_D}$ (Ye et al., 2013) defined as

$$H_V = 1 - \frac{1}{|\overrightarrow{\mathcal{E}}^K_{v_D}|} - \frac{\sum_{\overrightarrow{(u_D, w_D)} \in \overrightarrow{\mathcal{E}}^K_{v_D}} \left\{ \frac{1}{d_o(u_D) d_i(w_D)} \right\}}{2|\overrightarrow{\mathcal{E}}^K_{v_D}|^2}, \tag{14}$$

where $d_o(u_D)$ is the out-degree of $u_D \in \mathcal{V}^K_{v_D}$ and $d_i(w_D)$ is the in-degree of $w_D \in \mathcal{V}^K_{v_D}$. And $H_E(\mathcal{G}_{v_D})$ is the Shannon entropic signature of $\mathcal{G}_{v_d}$ defined as

$$H_E(\mathcal{G}^K_{v_D}) = -\sum_{u_D \in \mathcal{V}^K_{v_D} \wedge u_D \in V;} P_Q(u_D) \log P_Q(u_D), \tag{15}$$

where $P_Q(u_D)$ is the probability of the CTQW visiting v_D in the original weighted graph $G_W(V, E)$. Note that, the Shannon entropic signature $H_E(\mathcal{G}^K_{v_D})$ is not a strict Shannon entropy measure, since it is not computed by using all the CTQW probabilities of the vertices in $G_W(V, E)$. □

The h-layer directed DB representation $\overrightarrow{DB}^h_{G_D}(v_D)$ not only captures directed information residing on the edges of

the directed-quantum graph G_D, but also encapsulates the probability information of the CTQW visiting the vertices on the original weighted graph G_W. As we have stated, both the probabilities of the CTQW visiting the vertices in G_W and the directed information residing on the edges in G_D can reflect the weighted information on the edges in G_W. The h-layer directed DB representation $DB^h_{G_D}(v_D)$ thus captures the weighted information on G_W.

The h-layer entropic representation: For the weighted graph $G_W(V,E)$ and its quantum-directed graph $G_D(V_D,E_D)$, we compute the h-layer entropic representation $E^h_{G_W}(v)$ around each vertex $v\in V$ by summing the h-layer un-directed DB representation on the un-directed graph $G_U(V_U,E_U)$ transformed from $G_D(V_D,\overrightarrow{E}_D)$ and the h-layer directed DB representation on $G_D(V_D,\overrightarrow{E}_D)$. For $G_W(V,E)$ and $v\in V$, $E^h_{G_W}(v)$ is defined as

$$E^h_{G_W}(v)=DB^h_{G_U}(v_U)+\overrightarrow{DB}^h_{G_D}(v_D), \qquad (16)$$

where $v=v_U$ ($v_U\in V_U$) and $v=v_D$ ($v_D\in V_D$). □

Discussion: Eq.(16) indicates that for the weighted graph $G_W(V,E)$ the h-layer entropic representation around each vertex not only reflects the original topological information of G_W through the un-directed DB representation, but also captures the weighted information of G_W through the directed DB representation. As a result, the h-layer entropic representation provides a convenient way of measuring the information content flow (i.e., the DB information (Escolano et al., 2012)) around a vertex for a weighted graph, and can be seen as a vectorial entropic signature for the vertex. Furthermore, note that, for $G_W(V,E)$ and its quantum-directed graph $G_D(V_D,E_D)$, the strengthened label of a vertex $v\in V$ at the m iteration of the TI method (defined in **Algorithm** 1) corresponds to a subtree of height m rooted at $v_D\in V_D$ ($v=v_D$). Eq.(11) indicates that the K-layer directed expansion subgraph $\mathcal{G}^K_{v_D}(\mathcal{V}^K_{v_D};\overrightarrow{\mathcal{E}}^K_{v_D})$ rooted at v_D from G_D encapsulates the subtree structure, if $K\geq m$. As a result, the h-layer entropic representations also offer us an elegant way of identifying the correspondence information between a pair of subtrees in G_D by aligning the the h-layer entropic representation of G_W, i.e., aligning the subtrees rooted at vertices.

3.2. The Entropic Weighted Graph Matching

We propose a new entropic graph matching method for a pair of weighted attributed graphs by aligning their entropic representations. Our matching method is similar to that previously introduced by Scott et al. in (Scott & Longuet-Higgins, 1991) for point set matching, that computes an affinity matrix in terms of the distances between points (i.e., for a pair of graphs, we commence by computing the vectorial signature for each vertex as the point coordinate of the vertex, and compute the distance measures between the point coordinates of pairwise vertices to establish the affinity matrix). For a pair of weighted graphs $G_{W;p}(V_p,E_p)$ and $G_{W;q}(V_q,E_q)$, we use the h-layer entropic representations $E^h_{G_{W;p}}(v_i)$ and $E^h_{G_{W;q}}(u_j)$ as the point coordinates for the vertices $v_i\in V_p$ and $u_j\in V_q$, respectively. For $G_{W;p}$ and $G_{W;q}$, we compute the Euclidean distance between $E^h_{G_{W;p}}(v_p)$ and $E^h_{G_{W;q}}(u_q)$ as the element $R(i,j)$ of their affinity matrix R, and $R(i,j)$ is

$$R(i,j)=\|E^h_{G_{W;p}}(v_i)-E^h_{G_{W;q}}(u_j)\|_2. \qquad (17)$$

where R is a $|V_p|\times|V_q|$ matrix. $R(i,j)$ represents the distance between the vertex v_i in G_p and the vertex u_j in G_q. Furthermore, for the affinity matrix R, the rows index the vertices of G_p, and the columns index the vertices of G_q.

If $R(i,j)$ is the smallest element simultaneously in row i and in column j, and the strengthened vertex labels $\mathcal{L}_m(v_i)$ and $\mathcal{L}_m(v_j)$ are the same, we say that there is a one-to-one correspondence between v_i and u_j, i.e., v_i and u_j are matched. We record the correspondence using the correspondence matrix $C^{(m)}\in\{0,1\}^{|V_p|\times|V_q|}$ that satisfies

$$C^{(m)}(i,j)=\begin{cases}1 & \text{if } R(i,j) \text{ is the smallest element,}\\ & \text{both in row } i \text{ and in column j,}\\ & \mathcal{L}_m(v_i) \text{ and } \mathcal{L}_m(u_j) \text{ exist,}\\ & \text{and } \mathcal{L}_m(v_i)=\mathcal{L}_m(u_j);\\ 0 & \text{otherwise.}\end{cases} \qquad (18)$$

where the parameter m indicates that the affinity matrix is computed in terms of the strengthened vertex labels from the m-th iteration of **Algorithm 1**. Eq.(18) indicates that if $C^{(m)}(i,j)=1$, the vertices v_i and v_j are matched.

Note that the vertex of a graph may have more than one matched vertex from the other graph. We propose to assign a vertex to at most one matched vertex. One way to achieve this is to update the matrix $C^{(m;h)}$ using the Hungarian algorithm (Munkres, 1957), following the strategy proposed in (Bai et al., 2014b). Unfortunately, the Hungarian algorithm usually requires very expensive computations. An alternative strategy is to randomly select a subset of the matched vertices. In other words, given the correspondence matrix $C^{(m)}$, we select a random entry with value 1, and we set every other element of that row and column of $C^{(m)}$ to 0. Based on our evaluations, this strategy does not influence the effectiveness of the resulting kernel in Section 4, while keeping the computational complexity low.

4. A Graph Kernel from Entropic Matching

4.1. The Entropic Matching Weighted Graph Kernel

Let $G_p(V_p,E_p)$ and $G_q(V_q,E_q)$ be a pair of weighted graphs. Based on the entropic matching in Section 3.2, we commence by computing the correspondence matrix $C^{(m)}$.

The entropic matching kernel $k_{EM}^{(M)}$ is

$$k_{EM}^{(M)}(G_p, G_q) = \sum_{m=0}^{M} \sum_{i=1}^{|V_p|} \sum_{j=1}^{|V_q|} C^{(m)}(i,j), \quad (19)$$

where M is the greatest value of the parameter m (i.e., m varies from 0 to M). Eq.(19) indicates that $k_{EM}^{(M)}(G_p, G_q)$ counts the number of matched vertex pairs between G_p and G_q over the M correspondence matrices $C^{(m)}$. Intuitively, the entropic matching kernel $k_{EM}^{(M)}$ is positive definite. This is because $k_{EM}^{(M)}$ counts pairs of matched vertices over the M correspondence matrices $C^{(m)}$. □

4.2. Relation to the Existing Classical Kernel

The entropic matching kernel can be re-defined in another manner that elucidates its effectiveness, compared to the classical all subgraph (AS) kernel on sample graphs, i.e., the graphs are un-weighted and un-directed. To commence, we review the definition of the AS kernel. Let $G_a(V_a, E_a)$ and $G_b(V_b, E_b)$ be a pair of un-weighted and un-directed graphs. The AS kernel is defined as

$$k_{AS}(G_a, G_b) = \sum_{S_a \sqsubseteq G_a} \sum_{S_b \sqsubseteq G_b} k_I(S_a, S_b), \quad (20)$$

where

$$k_I(S_a, S_b) = \begin{cases} 1 & \text{if } S_a \simeq S_b, \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

As a result, the AS kernel is computed by counting the number of isomorphic subgraph pairs between G_a and G_b.

We re-define the entropic matching kernel for weighted graphs in a manner which is similar to that of the AS kernel between sample graphs. Let a pair of weighted graphs be $G_p(V_p, E_p)$ and $G_q(V_q, E_q)$, and their quantum-directed graphs be $G_{D;p}$ and $G_{D;q}$. Since the entropic matching kernel $k_{EM}^{(M)}$ for weighted graphs is defined by counting the number of matched vertex pairs, that have the same strengthened vertex labels from the m iteration of **Algorithm 1**. Moreover, each strengthened label of a weighted graph corresponds to a subtree on its quantum-directed graph. $k_{EM}^{(M)}(G_p, G_q)$ can thus be re-written as

$$k_{EM}^{(M)}(G_p, G_q) = \sum_{S_p \sqsubseteq G_{D;p}} \sum_{S_q \sqsubseteq G_{D;q}} k_I(S_p, S_q), \quad (22)$$

where

$$k_I(S_p, S_q) = \begin{cases} 1 & \text{if } \mathcal{L}_m(v_p) \text{ corresponds to } S_p, \\ & \mathcal{L}_m(u_q) \text{ corresponds to } S_q, \\ & v_p \text{ and } u_q \text{ are matched, and} \\ & \mathcal{L}_m(v_p) \text{ and } \mathcal{L}_m(u_q) \text{ exist;} \\ 0 & \text{otherwise.} \end{cases} \quad (23)$$

where $v_p \in V_p$ and $v_q \in V_q$. Eq.(22) and Eq.(23) indicate that the entropic kernel $k_{EM}^{(M)}$ can be seen as a kernel that counts the number of isomorphic subtree pairs on quantum-directed graphs. Each pair of isomorphic subtrees are corresponded by a pair of matched or aligned vertices that have the same strengthened labels on the original weighted graphs. In other words, we establish the reliable locational correspondence between the pair of isomorphic subtrees. As a result, the entropic matching kernel is essentially an **aligned subtree kernel** that counts aligned isomorphic subtree pairs on the quantum-directed graphs transformed from the corresponding weighted graphs.

Discussion: Eq.(20) and Eq.(23) indicate that both the kernels k_{AS} and $k_{EM}^{(M)}$ need to identify pairs of isomorphic subgraphs or subtrees. For k_{AS} and $k_{EM}^{(M)}$, each pair of isomorphic subgraphs or subtrees will add an unit value to the kernel function. However, Eq.(20) and Eq.(23) also highlight the following difference between k_{AS} and $k_{EM}^{(M)}$. **a)** For $k_{EM}^{(M)}$, only the subtrees that correspond to a pair of aligned vertices are evaluated with respect to being isomorphic. For k_{AS}, any pair of subgraphs are evaluated for identifying the isomorphism. As a result, the computational efficiency of $k_{EM}^{(M)}$ is faster. **b)** For $k_{EM}^{(M)}$, all pairs of isomorphic subtrees are identified by a corresponding pair of matched vertices. Thus, we ensure locational correspondences between the isomorphic subtrees in the global graphs. By contrast, a pair of subgraphs having no location correspondence may also be seen as isomorphic by k_{AS}.

4.3. Discussion and Related Work

The entropic matching kernel is related to the DB representation defined in (Bai & Hancock, 2014). However, there are two significant differences. First, the DB representation in (Bai & Hancock, 2014) can not be performed on weighted graphs. This is because the distances between pairs of adjacent vertices in a weighted graph are not equivalent. As a result, it is difficult to determine the expansion subgraphs around the centroid vertex. By contrast, the entropic representation can not only capture the topological information for weighted graphs through the h-layer DB representation, but also reflect the edge weighted information through the directed h-layer DB representation on the quantum-directed graph. As a result, the h-layer entropic representation can accommodate weighted graphs. Second, the DB representation from the centroid vertex is a vectorial signature of a graph, i.e., it is an embedding vector for the graph. Embedding a graph into a vector tends to approximate the structural correlations in a low dimensional space, and thus leads to information loss. By contrast, the entropic matching kernel aligning the m-layer entropic representation represents graphs in a high dimensional space and thus better preserves graph structures.

The DB matching kernel developed in (Bai et al., 2014b) is also related to the DB representation in (Bai & Hancock, 2014). Moreover, similarly to our entropic matching kernel, the DB matching kernel can also better preserve graph structures by kernelizing the h-layer DB representation on sample graphs (i.e, un-weighted and un-directed graphs). However, similar to the DB representation in (Bai & Hancock, 2014), the DB kernel also cannot be performed on weighted graphs, because of the similar reason. Moreover, according to the definition in Section 3.1, the required h-layer DB representations for the DB kernel is just the h-layer DB representations for our entropic matching kernel by ignoring the weighted information residing on the edges. By contrast, the required h-layer entropic representations for the entropic matching kernel are computed not only from the h-layer DB representations using the Shannon entropy associated with the steady state random walk, but also from the h-layer directed DB representations using the approximated directed von Neumann entropy thorough the quantum-directed graph and the Shannon entropic signature from the probabilities of the CTQW visiting vertices. Since the CTQW can be performed on both weighted and un-weighted graphs, and capture rich topological information in a graph structure. The h-layer entropic representation for a vertex reflects richer characteristics than the h-layer DB representation. As a result, the entropic matching kernel can capture more information for graphs than the DB matching kernel, even on un-weighted graphs.

Moreover, Bai (Bai, 2014) has demonstrated the relationship between the DB matching kernel and the AS kernel. The DB matching kernel can also identify the locational correspondence between isomorphic substructures (i.e., the h-layer expansion subgraphs), and count the number of isomorphic substructure pairs. Thus, similar to our entropic matching kernel, the DB matching kernel also reflects the locational correspondence between each pair of identified isomorphic substructures. Unfortunately, the identified isomorphic substructure pair number of the DB matching kernel is less than that of our entropic matching kernel. For a pair of sample graphs each having x vertices, the DB matching kernel can only identify x pairs of isomorphic substructures at most. By contrast, the entropic matching kernel can identify Mx pairs of isomorphic substructures (i.e., subtrees) at most. Furthermore, the DB matching kernel ignores the vertex labels. By contrast, the entropic matching kernel accommodates the strengthened vertex labels from the TI method defined in **Algorithm 1**. These again demonstrate that our entropic matching kernel captures more information than the DB matching kernel.

Another kernel related to ours is the subgraph matching kernel (Kriege & Mutzel, 2012). Although this kernel can handle both node and edge attributes, in order to achieve a polynomial computational time, the authors decide to count the number of matchings between subgraphs only up to a fixed size. In contrast, the computation of our kernel remains polynomial even when we let the h-layer expansion subgraph grow to the size of the whole graph. As a consequence, we are able to capture richer structural information. In general, note that any graph kernel what can deal with continuous node attributes (Neumann et al., 2012; Feragen et al., 2013) can be adapted to work on edge weighted graphs by replacing the original graph with its corresponding line graph, effectively moving the weights from the edges to the nodes. However, in the case of dense graphs with n nodes the line graph would have $O(n^2)$ nodes, thus making the computation of the kernel measure harder.

Finally, the idea of using CTQW to measure the similarity between graphs has also been used by Rossi et al. (Rossi et al., 2013a; 2015). Given a pair of graphs, the authors simulate two quantum walks on the union of the graphs and to exploit quantum walk interference effects to measure the similarity between the graphs. However, the union graph is established by roughly connecting all vertex pairs of the graphs and thus lacks vertex correspondence information.

4.4. Computational Analysis

For a pair of graphs each of which has N vertices, computing the entropic matching kernel (i.e., the aligned subtree kernel) requires time complexity $O(N^3 + hMN^2)$. This is because computing the CTQW for each graph relies on the Laplacian spectrum decomposition, and thus requires time complexity $O(N^3)$. Transforming a graph into the quantum-directed graph requires time complexity $O(N^2)$. Performing the TI method and establishing the expansion subgraph on the directed-quantum graph both rely on the shortest path computation, and thus require time complexity $O(N^3)$. For an (un)directed expansion subgraph, computing the directed von Neumann entropy, the Shannon entropy associated with the steady state of the random walk, and the Shannon entropic signature through the probabilities of the CTQW visiting vertices requires time complexity $O(N^2)$, $O(N^2)$ and $O(N)$, respectively. Thus, computing the h-layer entropic representations around the V vertices for a graph requires time complexity $O(hV^3)$. Computing the M affinity matrices requires time complexity $O(hMN^2)$. Identifying the vertex correspondences through the M affinity matrices requires time complexity MN^2. As a result, computing the entropic matching kernel requires time complexity $O(N^3 + hMN^2)$. This indicates that our kernel can be computed in polynomial time.

5. Experimental Results

BAR31, BSPHERE31 and GEOD31: The SHREC 3D Shape database consists of 15 classes and 20 instances per class, for a total of 300 shapes (Biasotti et al., 2003).

Table 1. Classification Accuracy (In % ± Standard Error) Using C-SVM.

Datasets	**ASK**	DBMK	WLSK	SPGK	GCGK	UQJS	JSGK	JTQK
BAR31	**73.10** ± .67	69.40 ± .56	58.53 ± .53	55.73 ± .44	23.40 ± .60	30.80 ± .61	24.10 ± .86	60.56 ± .35
BSPHERE31	**60.30** ± .44	56.43 ± .69	42.10 ± .68	48.20 ± .76	18.80 ± .50	24.80 ± .61	21.76 ± .53	46.93 ± .61
GEOD31	**46.21** ± .69	42.83 ± .50	38.20 ± .68	38.40 ± .65	22.36 ± .55	23.73 ± .66	18.93 ± .50	40.10 ± .46
MUTAG	**87.50** ± .65	85.27 ± .69	82.88 ± .57	83.38 ± .81	82.04 ± .39	83.11 ± .80	82.72 ± .44	85.50 ± .55

Table 2. Runtime of Computing the Kernel Matrix.

Datasets	**ASK**	DBMK	WLSK	SPGK	GCGK	UQJS	JSK	JTQK
BAR31	520”	682”	30”	11”	1”	630”	1”	88”
BSPHERE31	760”	720”	25”	14”	1”	828”	1”	95”
GEOD31	487”	649”	15”	11”	1”	519”	1”	77”
MUTAG	20”	270”	3”	1”	1”	20”	1”	3”

This is an usual benchmark in 3D shape recognition. From the SHREC 3D Shape database, we establish three graph datasets named BAR31, BSPHERE31 and GEOD31 through three different mapping functions. These functions are a) ERG barycentre: distance from the centre of mass/barycentre, b) ERG bsphere: distance from the centre of the sphere that circumscribes the object, and c) ERG integral geodesic: the average of the geodesic distances to the all other points. The number of maximum, minimum and average vertices for the three datasets are a) 220, 41 and 95.42 (for BAR31), b) 227, 43 and 99.83 (for BSPHERE31), and c) 380, 29 and 57.42 (for GEOD31), respectively. For each graph of the datasets, we compute the Euclidean distances between the h-layer DB representations for each pair of adjacent vertices as the edge weight.

MUTAG: The dataset consists of weighted graphs representing 188 chemical compounds. The maximum, minimum and average number of vertices are 28, 10 and 17.93.

Experimental Setup: We evaluate the performance of our entropic matching kernel, i.e., the aligned subtree kernel (ASK), on graph classification problems. We also compare our kernel with several alternative state-of-the-art graph kernels. These graph kernels include 1) the DB matching kernel (DBMK) (Bai, 2014; Bai et al., 2014b), 2) the Weisfeiler-Lehman subtree kernel (WLSK) (Shervashidze et al., 2011), 3) the shortest path graph kernel (SPGK) (Borgwardt & Kriegel, 2005), 4) the graphlet count kernel (Shervashidze et al., 2009) with graphlets of size 4 (GCGK) (Shervashidze et al., 2009), 5) the un-aligned quantum Jensen-Shannon kernel (UQJS) (Bai et al., 2015), 6) the Jensen-Shannon graph kernel (JSGK) (Bai & Hancock, 2013), and 7) the Jensen-Tsallis q-difference kernel (JTQK) (Bai et al., 2014d) associated with $q = 2$. For our ASK kernel $k_{EM}^{(M)}$, we set h to 10 and M to 50. In fact, we observe that the classification accuracy tends to be stable when $h \geq 8$. Moreover, the height of the highest subtrees identified by the TI method in the dataset is about 50. For the WLSK kernel and the JTQK kernel, we set the highest dimension (i.e., the highest height of subtrees) of the Weisfeiler-Lehman isomorphism (for the WLSK kernel) and the tree-index method (for the JTQK kernel) to 10. For the DBMK kernel, we set the highest layer of the required DB representation to 10. For each kernel, we perform 10-fold cross-validation where the classification accuracy is computed using a C-Support Vector Machine (C-SVM). In particular, we make use of the LIBSVM library(Chang & Lin, 2011). For each datasets, we compute the optimal C-SVMs parameters. We report the average classification accuracy (± standard error) and the runtime for each kernel in Table 1 and Table 2, respectively. The runtime is measured under Matlab R2011a running on a 2.5GHz Intel 2-Core processor (i.e., i5-3210m).

Experimental Results: In terms of the classification accuracy, we observe that our ASK kernel can easily outperform all the alternative graph kernels on all the datasets. The reasons for the effectiveness are threefold. First, compared to the WLSK, SPGK, GCGK and JTQK kernels, which also require decomposing graphs into substructures, our kernel considers the relative locations between substructures. This information is neglected by WLSK, SPGK, GCGK and JTQK kernels. Second, compared to the JSGK and QJSK kernels, both of which rely on the similarity measure between global graphs in terms of the classical or quantum JSD, our kernel can identify the correspondence information between both the vertices and the substructures, and thus reflect richer topological characteristics. By contrast, the JSGK and QJSK kernels only reflect global graph similarity information. Third, compared to the DBMK kernel, our kernel can identify more pairs of aligned isomorphic substructures (i.e., subtrees). Finally, only our kernel can capture the edge weighted information. In terms of the runtime, our kernel is not the fastest kernel, but it can still finish the kernel matrix computation in a polynomial time.

6. Conclusions

We have developed a new aligned subtree kernel for weighted graphs. Our kernel not only accommodates both weighted and un-weighted graphs, but also addresses the drawback of neglecting the relative locations between substructures arising in the R-convolution kernels. Experiments demonstrate the effectiveness of our new kernel.

Acknowledgments

L. Bai and Z. Zhang are supported by National Natural Science Foundation of China (61402389 and 61272398). E.R. Hancock is supported by a Royal Society Wolfson Research Merit Award.

References

Aubry, Mathieu, Schlickewei, Ulrich, and Cremers, Daniel. The wave kernel signature: A quantum mechanical approach to shape analysis. In *proceedings of ICCV Workshops*, pp. 1626–1633, 2011.

Aziz, Furqan, Wilson, Richard C, and Hancock, Edwin R. Backtrackless walks on a graph. *IEEE Transactions on Neural Networks and Learning Systems*, 24(6):977–989, 2013.

Bai, Lu. *Information Theoretic Graph Kernels*. University of York, UK, 2014.

Bai, Lu and Hancock, Edwin R. Graph kernels from the jensen-shannon divergence. *Journal of mathematical imaging and vision*, 47(1-2):60–69, 2013.

Bai, Lu and Hancock, Edwin R. Depth-based complexity traces of graphs. *Pattern Recognition*, 47(3):1172–1186, 2014.

Bai, Lu, Bunke, Horst, and Hancock, Edwin R. An attributed graph kernel from the jensen-shannon divergence. In *Proceddings of ICPR*, pp. 88–93, 2014a.

Bai, Lu, Ren, Peng, Bai, Xiao, and Hancock, Edwin R. A graph kernel from the depth-based representation. In *Proceedings of S+SSPR*, pp. 1–11, 2014b.

Bai, Lu, Ren, Peng, and Hancock, Edwin R. A hypergraph kernel from isomorphism tests. In *Proceddings of ICPR*, pp. 3880–3885, 2014c.

Bai, Lu, Rossi, Luca, Bunke, Horst, and Hancock, Edwin R. Attributed graph kernels using the jensen-tsallis q-differences. In *Proceedings of ECML-PKDD*, pp. 99–114, 2014d.

Bai, Lu, Rossi, Luca, Torsello, Andrea, and Hancock, Edwin R. A quantum jensen–shannon graph kernel for unattributed graphs. *Pattern Recognition*, 48(2):344–355, 2015.

Biasotti, Silvia, Marini, Simone, Mortara, Michela, Patanè, Giuseppe, Spagnuolo, Michela, and Falcidieno, Bianca. 3d shape matching through topological structures. In *Proceedings of DGCI*, pp. 194–203, 2003.

Borgwardt, Karsten M and Kriegel, H-P. Shortest-path kernels on graphs. In *Proceedings of ICDM*, pp. 8–pp, 2005.

Chang, Chih-Chung and Lin, Chih-Jen. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011.

Crutchfield, James P and Shalizi, Cosma Rohilla. Thermodynamic depth of causal states: Objective complexity via minimal representations. *Physical review E*, 59(1):275, 1999.

Emms, David, Wilson, Richard C, and Hancock, Edwin R. Graph matching using the interference of continuous-time quantum walks. *Pattern Recognition*, 42(5):985–1002, 2009.

Escolano, Francisco, Hancock, Edwin R, and Lozano, Miguel A. Heat diffusion: Thermodynamic depth complexity of networks. *Physical Review E*, 85(3):036206, 2012.

Feragen, Aasa, Kasenburg, Niklas, Petersen, Jens, de Bruijne, Marleen, and Borgwardt, Karsten. Scalable kernels for graphs with continuous attributes. In *Proceedings of NIPS*, pp. 216–224, 2013.

Haussler, David. Convolution kernels on discrete structures. Technical report, Technical report, Department of Computer Science, University of California at Santa Cruz, 1999.

Jebara, Tony, Kondor, Risi, and Howard, Andrew. Probability product kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

Kempe, Julia. Quantum random walks: an introductory overview. *Contemporary Physics*, 44(4):307–327, 2003.

Kriege, Nils and Mutzel, Petra. Subgraph matching kernels for attributed graphs. In *Proceedings of ICML*, 2012.

Munkres, James. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial & Applied Mathematics*, 5(1):32–38, 1957.

Neumann, Marion, Patricia, Novi, Garnett, Roman, and Kersting, Kristian. Efficient graph kernels by randomization. In *Proceedings of ECML-PKDD*, pp. 378–393. 2012.

Nielsen, Michael A and Chuang, Isaac L. *Quantum computing and quantum information*. Cambridge University Press, Cambridge, 2000.

Rossi, Luca, Torsello, Andrea, and Hancock, Edwin R. Attributed graph similarity from the quantum jensen-shannon divergence. In Hancock, Edwin R. and Pelillo, Marcello (eds.), *SIMBAD*, volume 7953 of *Lecture Notes in Computer Science*, pp. 204–218. Springer, 2013a.

Rossi, Luca, Torsello, Andrea, Hancock, Edwin R, and Wilson, Richard C. Characterizing graph symmetries through quantum jensen-shannon divergence. *Physical Review E*, 88(3):032806, 2013b.

Rossi, Luca, Torsello, Andrea, and Hancock, Edwin R. Measuring graph similarity through continuous-time quantum walks and the quantum jensen-shannon divergence. *Physical Review E*, 91(2):022815, 2015.

Scott, Guy L. and Longuet-Higgins, H. Christopher. An algorithm for associating the features of two images. *The Royal Society of London B: Biological Sciences*, 244 (1309):21–26, 1991.

Shervashidze, Nino, Vishwanathan, S. V. N., Petri, Tobias, Mehlhorn, Kurt, and Borgwardt, Karsten M. Efficient graphlet kernels for large graph comparison. *Journal of Machine Learning Research*, 5:488–495, 2009.

Shervashidze, Nino, Schweitzer, Pascal, Van Leeuwen, Erik Jan, Mehlhorn, Kurt, and Borgwardt, Karsten M. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12:2539–2561, 2011.

Suau, Pablo, Hancock, Edwin R., and Escolano, Francisco. Graph characteristics from the schrödinger operator. In *Proceedings of GbRPR*, pp. 172–181. 2013.

Ye, Cheng, Wilson, Richard C, Comin, César H, Costa, Luciano da F, and Hancock, Edwin R. Entropy and heterogeneity measures for directed graphs. In *Proceedings of SIMBAD*, pp. 219–234, 2013.