Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A quantum Jensen–Shannon graph kernel for unattributed graphs

Lu Bai [a,*], Luca Rossi [b,**], Andrea Torsello [c], Edwin R. Hancock [a,1]

[a] Department of Computer Science, The University of York, York YO10 5DD, UK
[b] School of Computer Science, University of Birmingham, UK
[c] Department of Environmental Science, Informatics, and Statistics, Ca' Foscari University of Venice, Italy

ARTICLE INFO

Article history:
Received 8 September 2013
Received in revised form
20 March 2014
Accepted 21 March 2014

Keywords:
Graph kernels
Continuous-time quantum walk
Quantum state
Quantum Jensen–Shannon divergence

ABSTRACT

In this paper, we use the quantum Jensen–Shannon divergence as a means of measuring the information theoretic dissimilarity of graphs and thus develop a novel graph kernel. In quantum mechanics, the quantum Jensen–Shannon divergence can be used to measure the dissimilarity of quantum systems specified in terms of their density matrices. We commence by computing the density matrix associated with a continuous-time quantum walk over each graph being compared. In particular, we adopt the closed form solution of the density matrix introduced in Rossi et al. (2013) [27,28] to reduce the computational complexity and to avoid the cumbersome task of simulating the quantum walk evolution explicitly. Next, we compare the mixed states represented by the density matrices using the quantum Jensen–Shannon divergence. With the quantum states for a pair of graphs described by their density matrices to hand, the quantum graph kernel between the pair of graphs is defined using the quantum Jensen–Shannon divergence between the graph density matrices. We evaluate the performance of our kernel on several standard graph datasets from both bioinformatics and computer vision. The experimental results demonstrate the effectiveness of the proposed quantum graph kernel.

1. Introduction

Structural representations have been used for over 30 years in pattern recognition due to their representational power. However, the increased descriptiveness comes at the cost of a greater difficulty in applying standard techniques to them, as these usually require data to reside in a vector space. The famous kernel trick [1] allows the focus to be shifted from the vectorial representation of data, which now becomes implicit, to a similarity representation. This allows standard learning techniques to be applied to data for which no obvious vectorial representation exists. For this reason, in recent years pattern recognition has witnessed an increasing interest in structural learning using graph kernels.

1.1. Literature review

1.1.1. Graph kernels

One of the most influential works on structural kernels was the generic R-convolution kernel proposed by Haussler [2]. Here graph kernels are computed by comparing the similarity of each of the decompositions of the two graphs. Depending on how the graphs are decomposed, we obtain different kernels. Generally speaking, most R-convolution kernels count the number of isomorphic substructures in the two graphs. Kashima et al. [3] compute the kernel by decomposing the graph into random walks, while Borgwardt et al. [4] have proposed a kernel based on shortest paths. Here, the similarity is determined by counting the numbers of pairs of shortest paths of the same length in a pair of graphs. Shervashidze et al. [5] have developed a subtree kernel on subtrees of limited size. They compute the number of subtrees shared between two graphs using the Weisfeiler–Lehman graph invariant. Aziz et al. [6] have defined a backtrackless kernel on cycles of limited length. They compute the kernel value by counting the numbers of pairs of cycles of the same length in a pair of graphs. Costa and Grave [7] have defined a so-called neighborhood subgraph pairwise distance kernel by counting the number of pairs of isomorphic neighborhood subgraphs. Recently, Kriege et al. [8] counted the number of isomorphisms between pairs of subgraphs, while Neumann et al. [9] have introduced the concept of propagation kernels to handle partially labeled graphs through the use of continuous-valued vertex attributes.

One drawback of these kernels is that they neglect the locational information for the substructures in a graph. In other words, the similarity does not depend on the relationships between substructures. As a consequence, these kernels cannot establish

* Corresponding author. Tel.: +86 13084142051, +44 7429399030.
** Corresponding author. Tel.: +44 1214144766.
E-mail addresses: bailu69@hotmail.com, lu@cs.york.ac.uk (L. Bai), l.rossi@cs.bham.ac.uk, blextar@gmail.com (L. Rossi), torsello@dsi.unive.it (A. Torsello), erh@cs.york.ac.uk (E.R. Hancock).

[1] He was supported by a Royal Society Wolfson Research Merit Award.

http://dx.doi.org/10.1016/j.patcog.2014.03.028
0031-3203/

reliable structural correspondences between the substructures. This limits the precision of the resulting similarity measure. To overcome this problem, Frohlich et al. [10] introduced alternative optimal assignment kernels. Here each pair of structures is aligned before comparison. However, the introduction of the alignment step results in a kernel that is not positive definite in general [11]. The problem results from the fact that alignments are not in general transitive. In other words, if σ is the vertex-alignment between graph A and graph B, and π is the alignment between graph B and graph C, in general, we cannot guarantee that the alignment between graph A and graph C is $\pi \circ \sigma$. On the other hand, when the alignments are transitive, there is a common simultaneous alignment of all the graphs. Under this alignment, the optimal assignment kernel is simply the sum over all the vertex/edge kernels, and this is positive definite since it is the sum of separate positive definite kernels. While lacking positive definiteness the optimal assignment kernels cannot be guaranteed to represent an implicit embedding into a Hilbert space, they have nonetheless been proven to be very effective in classifying structures. Another example of alignment-based kernels is the edit-distance-based kernels introduced by Neuhaus and Bunke [12]. Here the alignments obtained from graph-edit distance are used to guide random walks on the structures being compared.

An attractive alternative way to measure the similarity of a pair of graphs is to use the mutual information and compute the classical Jensen–Shannon divergence [13]. In information theory, the Jensen–Shannon divergence is a dissimilarity measure between probability distributions. It is symmetric, always well defined and bounded [13]. Bai and Hancock [14] have used the divergence to define a Jensen–Shannon kernel for graphs. Here, the kernel between a pair of graphs is computed using a nonextensive entropy measure in terms of the classical Jensen–Shannon divergence between probability distributions over graphs. For a graph, the elements of the probability distribution are computed from the degree of corresponding vertices. The entropy associated with a probability distribution of an individual graph can thus be directly computed without the need of decomposing the graph into substructures. Hence, unlike the aforementioned existing graph kernels, the Jensen–Shannon kernel between a pair of graphs avoids the computationally burdensome task of determining the similarities between all pairs of substructures. Unfortunately, the required composite entropy for the Jensen–Shannon kernel is computed from a product graph formed by a pair of graphs, and reflects no correspondence information between pairs of vertices. As a result, the Jensen–Shannon graph kernel lacks correspondence information between the probability distributions over the graphs, and thus cannot precisely reflect the similarity between graphs.

There has recently been an increasing interest in quantum computing because of the potential speed-ups over classical algorithms. Examples include Grover's polynomially faster search algorithm [15] and Shor's exponentially faster factorization algorithm [16]. Furthermore, quantum algorithms also offer us a richer structure than their classical counterparts since they use qubits rather than bits as the basic representational unit [32].

1.1.2. Quantum computation

Quantum systems differ from their classical counterparts, since they add the possibility of state entanglement to the classical statistical mixture of classical systems, which results in an exponential increase of the dimensionality of the state-space which is at the basis of the quantum speedup. Pure quantum states are represented as entries in a complex Hilbert space, while potentially mixed quantum states are represented through the density matrix. Mixed states can then be compared by examining their density matrices. One convenient way to do this is to use the quantum Jensen–Shannon divergence, first introduced by Majtey et al. [13,18]. Unlike the classical Jensen–Shannon divergence which is defined as a similarity measure between probability distributions, the quantum Jensen–Shannon divergence is defined as the distance measure between mixed quantum states described by density matrices. Moreover, it can be used to measure both the degree of mixing and the entanglement [13,18].

In this paper, we are interested in computing a kernel between pairs of graphs using the quantum Jensen–Shannon divergence between two mixed states representing the evolution of continuous-time quantum walks on the graphs. The continuous-time quantum walk has been introduced as the natural quantum analogue of the classical random walk by Farhi and Gutmann [19], and has been widely used in the design of novel quantum algorithms. Ambainis et al. [20,21] were among the first to generalize random walks on finite graphs to the quantum world. Furthermore, they have explored the application of quantum walks on graphs to a number of problems [22]. Childs et al. [23,24] have explored the difference between quantum and classical random walks, and then exploited the increased power of quantum walk as a general model of computation.

Similar to the classical random walk on a graph, the state space of the continuous-time quantum walk is the set of vertices of the graph. However, unlike the classical random walk where the state vector is real-valued and the evolution is governed by a doubly stochastic matrix, the state vector of the continuous-time quantum walk is complex-valued and its evolution is governed by a time-varying unitary matrix. The continuous-time quantum walk possesses a number of interesting properties which are not exhibited by the classical random walk. For instance, the continuous-time quantum walk is reversible and non-ergodic, and does not have a limiting distribution. As a result, the continuous-time quantum walk offers us an elegant way to design quantum algorithms on graphs. For further details on quantum computation and quantum algorithms, we refer the reader to the textbook in [32].

There have recently been several attempts to define quantum kernels using the continuous-time quantum walk. For instance, Bai et al. [26] have introduced a novel graph kernel where the similarity between two graphs is defined in terms of the similarity between two quantum walks evolving on the two graphs. The basic idea here is to associate with each graph a mixed quantum state representing the time evolution of a quantum walk. The kernel between the walk is defined as the divergence between the corresponding density operators. However, this quantum divergence measure requires the computation of an additional mixed state where the system has equal probability of being in each of the two original quantum states. Unless this quantum kernel takes into account the correspondences between the vertices of the two graphs, it can be shown that it is not permutation invariant. Rossi et al. [27,28] have attempted to overcome this problem by allowing the two quantum walks to evolve on the union of the two graphs. This exploits the relation between the interferences of the continuous-time quantum walks and the symmetries in the structures being compared. Since both the mixed states are defined on the same structure, this quantum kernel addresses the shortcoming of permutation variance. However, the resulting approach is less intuitive, thus making it particularly hard to prove the positive semidefiniteness of the kernel. Moreover, the union graph is established by roughly connecting all vertex pairs of the two graphs, and thus also lacks vertex correspondence information. As a result, this kernel cannot reflect precise similarity between the two graphs.

1.2. Contributions

The aim of this paper is to overcome the shortcomings of the existing graph kernels by defining a novel quantum kernel. To this

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

end, we develop the work in [26] further. We intend to solve the problems of permutation variance and missing vertex correspondence by introducing an additional alignment step, and thus compute an aligned mixed density matrix. The goal of the alignment is to provide vertex correspondences and a lower bound on the divergence over permutations of the vertices/quantum states. However we approximate the alignment that minimizes the divergence with that which minimizes the Frobenious norm between the density operators. This latter problem is solved using Umeyama's graph matching method [29]. Since the aligned density matrix is computed by taking into account the vertex correspondences between the two graphs, the density matrix reflects the locational correspondences between the quantum walks over the vertices of the two graphs. As a result, our new quantum kernel can not only overcome the shortcomings of missing vertex correspondence and permutation variance that arise with the existing kernels from the classical/quantum Jensen–Shannon divergence, but also overcome the shortcoming of neglecting locational correspondence between substructures that arises in the existing R-convolution kernels. Furthermore, in contrast to what is done in previous assignment-based kernels, we do not align the structures directly. Rather we align the density matrices, which are a special case of a Laplacian family signature [30] well known to provide a more robust representation for correspondence estimation. Finally, we adopt the closed form solution of the density matrix introduced in [27,28] to reduce the computational complexity and to avoid the cumbersome task of explicitly simulating the time evolution of the quantum walk. We evaluate the performance of our kernel on several standard graph datasets from bioinformatics and computer vision. The experimental results demonstrate the effectiveness of the proposed graph kernel, providing both a higher classification accuracy than the unaligned kernel and also achieving a performance comparable or superior to other state-of-the-art graph kernels.

1.3. Paper outline

In Section 2, we introduce the quantum mechanical formalism that will be used to develop the ideas proposed in the paper. We describe how to compute a density matrix for the mixed quantum state of the continuous-time quantum walk on a graph. With the mixed state density matrix to hand, we compute the von Neumann entropy of the continuous-time quantum walk on the graph. In Section 3, we compute a graph kernel using the quantum Jensen–Shannon divergence between the density matrices for a pair of graphs. In Section 4, we provide some experimental evaluations, which demonstrate experimentally the effectiveness of our quantum kernel. In Section 5, we conclude the paper and suggest future research directions.

2. Quantum mechanical background

In this section, we introduce the quantum mechanical formalism to be used in this work. We commence by reviewing the concept of a continuous-time quantum walk on a graph. We then describe how to establish a density matrix for the mixed quantum state of the graph through a quantum walk. With the density matrix of the mixed quantum state to hand, we show how to compute a von Neumann entropy for a graph.

2.1. The continuous-time quantum walk

The continuous-time quantum walk is a natural quantum analogue of the classical random walk [19,31]. Classical random walks can be used to model a diffusion process on a graph, and have proven to be a useful tool in the analysis of graph structure. In general, diffusion is modeled as a Markovian process defined over the vertices of the graph, where the transitions take place on the edges. More formally, let $G=(V,E)$ be an undirected graph, where V is a set of n vertices and $E=(V\times V)$ is a set of edges. The *state vector* for the walk at time t is a probability distribution over V, i.e., a vector $\overrightarrow{p}_t\in\mathbb{R}^n$ whose uth entry gives the probability that the walk is at vertex u at time t. Let A denote the symmetric adjacency matrix of the undirected graph G and let D be the diagonal degree matrix with elements $d_u=\sum_{v=1}^{n}A(u,v)$, where d_u is the degree of the vertex u. Then, the continuous-time random walk on G will evolve according to the equation

$$\overrightarrow{p}_t=e^{-Lt}\overrightarrow{p}_0, \tag{1}$$

where $L=D-A$ is the graph Laplacian.

As in the classical case, the continuous-time quantum walk is defined as a dynamical process over the vertices of the graph. However, in contrast to the classical case where the state vector is constrained to lie in a probability space, in the quantum case the state of the system is defined through a vector of complex *amplitudes* over the vertex set V. The squared norm of the amplitudes sums to unity over the vertices of the graph, with no restriction on their sign or complex phase. This in turn allows both destructive and constructive interferences to take place between the complex amplitudes. Moreover, in the quantum case, the evolution of the walk is governed by a complex valued unitary matrix, whereas the dynamics of the classical random walks is governed by a stochastic matrix. Hence the evolution of the quantum walk is reversible, which implies that quantum walks are non-ergodic and do not possess a limiting distribution. As a result, the behaviour of classical and quantum walks differs significantly. In particular, the quantum walk possesses a number of interesting properties not exhibited in the classical case. One notable consequence of the interference properties of quantum walks is that when a quantum walker backtracks on an edge it does so with the opposite phase. This gives rise to destructive interference and reduces the problem of tottering. In fact the faster hitting time observed in quantum walks on the line is a consequence of the destructive interference of backtracking paths [31].

The Dirac notation, also known as *bra-ket* notation, is a standard notation used for describing quantum states. We call *pure state* a quantum state that can be described using a single *ket* vector $|a\rangle$. In the context of the paper, a ket vector is a complex valued column vector of unit Euclidean length, in an n-dimensional Hilbert space. Its conjugate transpose is a *bra* (row) vector, denoted as $\langle a|$. As a result, the inner product between two states $|a\rangle$ and $|b\rangle$ is written as $\langle a|b\rangle$, while their outer product is $|a\rangle\langle b|$. Using the Dirac notation, we denote the *basis state* corresponding to the walk being at vertex $u\in V$ as $|u\rangle$. In other words, $|u\rangle$ is the vector which is equal to 1 in the position corresponding to the vertex u and is zero everywhere else. A general state of the walk is then expressed as a complex linear combination of these orthonormal basis states, such that the state vector of the walk at time t is defined as

$$|\psi_t\rangle=\sum_{u\in V}\alpha_u(t)|u\rangle, \tag{2}$$

where the amplitude $\alpha_u(t)\in\mathbb{C}$ is subject to the constraints $\sum_{u\in V}\alpha_u(t)\alpha_u^*(t)=1$ for all $u\in V$, $t\in\mathbb{R}^+$, where $\alpha_u^*(t)$ is the complex conjugate of $\alpha_u(t)$.

At each instant of time, the probability of the walk being at a particular vertex of the graph $G(V,E)$ is given by the squared norm of the amplitude associated with $|\psi_t\rangle$. More formally, let X^t be a random variable giving the location of the walk at time t.

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

The probability of the walk being at vertex $u \in V$ at time t is given by

$$\Pr(X^t = u) = \alpha_u(t)\alpha_u^*(t). \tag{3}$$

More generally, a quantum walker represented by $|\psi_t\rangle$ can be observed to be in any state $|\phi\rangle \in \mathbb{C}^n$, and not just in the states $|u\rangle, u \in V$ corresponding to the graph vertices. Given an orthonormal basis $|\phi_1\rangle, \ldots, |\phi_n\rangle$ of $\mathbb{C}^n$, the quantum walker $|\psi_t\rangle$ is observed in state $|\phi_i\rangle$ with probability $\langle\psi_i|\psi_i\rangle$. After being observed on state $|\phi_i\rangle$, the new state of the quantum walker is $|\phi_i\rangle$. Further, given the nature of quantum observation, only orthonormal states can be distinguished by a single observation.

In quantum mechanics, the *Schrodinger equation* is a partial differential equation that describes how quantum systems evolve with time. In the non-relativistic case it assumes the form

$$i\hbar\frac{\partial}{\partial t}|\psi\rangle = \mathcal{H}|\psi\rangle, \tag{4}$$

where the *Hamiltonian* operator $\mathcal{H}$ accounts for the total energy of a system. In the case of a particle moving in an empty space with zero potential energy, this reduces to the Laplacian operator ∇^2. Here, we take the Hamiltonian of the system to be the graph Laplacian matrix L. However, any Hermitian operator encoding the structure of the graph can be chosen as an alternative. The evolution of the walk at time t is then given by the Schrodinger equation

$$\frac{\partial}{\partial t}|\psi_t\rangle = -iL|\psi_t\rangle, \tag{5}$$

where the Laplacian L plays the role of the system Hamiltonian. Given an initial state $|\psi_0\rangle$, the Schrodinger equation has an exponential solution which can be used to determine the state of the walk at a time, t, giving

$$|\psi_t\rangle = e^{-iLt}|\psi_0\rangle, \tag{6}$$

where $U_t = e^{-iLt}$ is a unitary matrix governing the evolution of the walk. To simulate the evolution of the quantum walk, it is customary to rewrite Eq. (6) in terms of the spectral decomposition $L = \Phi^T\Lambda\Phi$ of the Laplacian matrix L, where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{|V|})$ is a diagonal matrix with the ordered eigenvalues as elements $(\lambda_1 < \lambda_2 < \cdots < \lambda_{|V|})$ and $\Phi = (\phi_1|\phi_2|\cdots|\phi_{|V|})$ is a matrix with the corresponding ordered orthonormal eigenvectors as columns. Hence, Eq. (6) can be rewritten as

$$|\psi_t\rangle = \Phi^T e^{-i\Lambda t}\Phi|\psi_0\rangle. \tag{7}$$

In this work, we let $\alpha_u(0)$ be proportional to d_u and using the constraints applied to the amplitude we get

$$\alpha_u(0) = \sqrt{\frac{d_u}{\sum d_u}}. \tag{8}$$

Note that there is no particular indication as how to choose the initial state. However, choosing a uniform distribution over the vertices of G would result in the system remaining stationary, since the initial state vector would be an eigenvector of the Laplacian of G and thus a stationary state of the walk.

2.2. A density matrix from the mixed state

While a pure state can be naturally described using a single ket vector, in general, a quantum system can be in a *mixed state*, i.e., a statistical ensemble of pure quantum states $|\psi_i\rangle$, each with probability p_i. The *density operator* (or *density matrix*) of such a system is defined as

$$\rho = \sum_i p_i|\psi_i\rangle\langle\psi_i|. \tag{9}$$

Density operators are positive unit-trace matrices directly linked with the observables of the (mixed) quantum system. Let O be an observable, i.e., a Hermitian operator acting on the quantum states and providing a measurement. Without loss of generality we have $O = \sum_{i=1}^m v_i P_i$, where P_i is the orthogonal projector onto the i-th observation basis, and v_i is the measurement value when the quantum state is observed to be in this observation basis.

The expectation value of the measurement over a mixed state can be calculated from the density matrix ρ:

$$\langle O\rangle = \mathrm{tr}(\rho O), \tag{10}$$

where tr is the trace operator. Similarly, the observation probability on the i-th observation basis can be expressed in terms of the density matrix ρ as

$$\Pr(X = i) = \mathrm{tr}(\rho P_i). \tag{11}$$

Finally, after the measurement, the corresponding density operator will be

$$\rho' = \sum_{i=1}^m P_i\rho P_i. \tag{12}$$

For a graph $G(V,E)$, let $|\psi_t\rangle$ denote the state corresponding to a continuous-time quantum walk that has evolved from time $t=0$ to time $t=T$. We define the time-averaged density matrix ρ_G^T for $G(V,E)$

$$\rho_G^T = \frac{1}{T}\int_0^T |\psi_t\rangle\langle\psi_t|\,\mathrm{d}t, \tag{13}$$

which can be expressed in terms of the initial state $|\psi_0\rangle$ as

$$\rho_G^T = \frac{1}{T}\int_0^T e^{-iLt}|\psi_0\rangle\langle\psi_0|e^{iLt}\,\mathrm{d}t. \tag{14}$$

In other words, ρ_G^T describes a system which has an equal probability of being in any of the pure states defined by the evolution of the quantum walk from $t=0$ to $t=T$. Using the spectral decomposition of the Laplacian we can rewrite the previous equation as

$$\rho_G^T = \frac{1}{T}\int_0^T \Phi^T e^{-i\Lambda t}\Phi|\psi_0\rangle\langle\psi_0|\Phi^T e^{i\Lambda t}\Phi\,\mathrm{d}t. \tag{15}$$

Let ϕ_{xy} denote the (xy) th element of the matrix of eigenvectors Φ of the Laplacian. The (r,c)th element of ρ_T can be computed as

$$\rho_G^T(r,c) = \frac{1}{T}\int_0^T \left(\sum_{k=1}^n\sum_{l=1}^n \phi_{rk}e^{-i\lambda_k t}\phi_{lk}\psi_{0l}\right)\left(\sum_{x=1}^n\sum_{y=1}^n \psi_{0x}\phi_{xy}e^{i\lambda_y t}\phi_{cy}\right)\mathrm{d}t. \tag{16}$$

Let $\overline{\psi}_k = \sum_l \phi_{lk}\psi_{0l}$ and $\overline{\psi}_y = \sum_x \phi_{xy}\psi_{0y}$, where ψ_{0l} denotes the lth element of $|\psi_0\rangle$. Then

$$\rho_G^T(r,c) = \frac{1}{T}\int_0^T \left(\sum_{k=1}^n \phi_{rk}e^{-i\lambda_k t}\overline{\psi}_k \sum_{y=1}^n \phi_{cy}e^{i\lambda_y t}\overline{\psi}_y\right)\mathrm{d}t, \tag{17}$$

which can be rewritten as

$$\rho_G^T(r,c) = \sum_{k=1}^n\sum_{y=1}^n \phi_{rk}\phi_{cy}\overline{\psi}_k\overline{\psi}_y\frac{1}{T}\int_0^T e^{i(\lambda_y-\lambda_k)t}\,\mathrm{d}t. \tag{18}$$

If we let $T\to\infty$, Eq. (18) further simplifies to

$$\rho_G^\infty(r,c) = \sum_{\lambda\in\tilde{\Lambda}}\sum_{x\in B_\lambda}\sum_{y\in B_\lambda}\phi_{rx}\phi_{cy}\overline{\psi}_x\overline{\psi}_y, \tag{19}$$

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. As a consequence, computing the time-averaged density matrix relies on computing the eigendecomposition of $G(V,E)$, and hence has time complexity $O(|V|^3)$. Note that in the remainder of this paper we will simplify our notation by referring to the time-averaged density matrix as the density matrix associated with a graph.

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

Moreover, we will assume that ρ_G^T is computed for $T \to \infty$, unless otherwise stated, and we will denote it as ρ_G.

2.3. The von Neumann entropy of a graph

In quantum mechanics, the *von Neumann entropy* [32] H_N of a mixture is defined in terms of the trace and logarithm of the density operator ρ

$$H_N = -\mathrm{tr}(\rho \log \rho) = -\sum_i \xi_i \ln \xi_i, \tag{20}$$

where $\xi_1, \ldots, \xi_n$ are the eigenvalues of ρ. Note that if the quantum system is a pure state $|\psi_i\rangle$ with probability $p_i = 1$, then the von Neumann entropy $H_N(\rho) = -\mathrm{tr}(\rho \log \rho)$ is zero. On other hand, a mixed state generally has a non-zero von Neumann entropy associated with its density operator.

Here we propose to associate to each graph the von Neumann entropy of the density matrix computed as in Eq. (19). Consider a graph $G(V,E)$, its von Neumann entropy is

$$H_N(\rho_G) = -\mathrm{tr}(\rho_G \log \rho_G) = -\sum_j^{|V|} \lambda_j^G \log \lambda_j^G, \tag{21}$$

where $\lambda_1^G, \ldots, \lambda_j^G, \ldots, \lambda_{|V|}^G$ are the eigenvalues of ρ_G.

2.4. Quantum Jensen–Shannon divergence

The classical Jensen–Shannon divergence is a non-extensive mutual information measure defined between probability distributions over potentially structured data. It is related to the Shannon entropy of the two distributions [33]. Consider two (discrete) probability distributions $\mathcal{P} = (p_1, \ldots, p_a, \ldots, p_A)$ and $\mathcal{Q} = (q_1, \ldots, q_b, \ldots, q_B)$, then the classical Jensen–Shannon divergence between $\mathcal{P}$ and $\mathcal{Q}$ is defined as

$$D_{JS}(\mathcal{P}, \mathcal{Q}) = H_S\left(\frac{\mathcal{P}+\mathcal{Q}}{2}\right) - \frac{1}{2}H_S(\mathcal{P}) - \frac{1}{2}H_S(\mathcal{Q}), \tag{22}$$

where $H_S(\mathcal{P}) = \sum_{a=1}^{A} p_a \log p_a$ is the Shannon entropy of distribution $\mathcal{P}$. The classical Jensen–Shannon divergence is always well defined, symmetric, negative definite and bounded, i.e., $0 \le D_{JS} \le 1$.

The quantum Jensen–Shannon divergence has recently been developed as a generalization of the classical Jensen–Shannon divergence to quantum states by Lamberti et al. [13]. Given two density operators ρ and σ, the quantum Jensen–Shannon divergence between them is defined as

$$D_{QJS}(\rho, \sigma) = H_N\left(\frac{\rho+\sigma}{2}\right) - \frac{1}{2}H_N(\rho) - \frac{1}{2}H_N(\sigma). \tag{23}$$

The quantum Jensen–Shannon divergence is always well defined, symmetric, negative definite and bounded, i.e., $0 \le D_{QJS} \le 1$ [13].

Additionally, the quantum Jensen–Shannon divergence verifies a number of properties which are required for a good distinguishability measure between quantum states [13,18]. This is a problem of key importance in quantum computation, and it requires the definition of a suitable distance measure. Wootters [34] was the first to introduce the concept of statistical distance between pure quantum states. His work is based on extending a distance over the space of probability distributions to the Hilbert space of pure quantum states. Similarly, the relative entropy [35] can be seen as a generalization of the information theoretic Kullback–Leibler divergence. However, the relative entropy is unbounded, it is not symmetric and it does not satisfy the triangle inequality.

The square root of the QJSD, on the other hand, is bounded, it is a distance and, as proved by Lamberti et al. [13], it satisfies the triangle inequality. This is formally proved for pure states, while for mixed states there is only empirical evidence. Some alternatives to the QJSD are described in the literature, most notably the Bures distance [36]. Both the Bures distance and the QJSD require the full density matrices to be computed. However, the QJSD turns out to be faster to compute. On the one hand, the Bures distance involves taking the square root of matrices, usually computed through matrix diagonalization and which has time complexity $O(n^3)$, where n is the number of vertices in the graph. On the other hand, the QJSD simply requires computing the eigenvalues of the density matrices, which can be done in $O(n^2)$.

3. A quantum Jensen–Shannon graph kernel

In this section, we propose a novel graph kernel based on the quantum Jensen–Shannon divergence between continuous-time quantum walks on different graphs. Suppose that the graphs under consideration are represented by a set $\{G, \ldots, G_a, \ldots, G_b, \ldots, G_N\}$, we consider the continuous-time quantum walks on a pair of graphs $G_a(V_a, E_a)$ and $G_b(V_b, E_b)$, whose mixed state density matrices ρ_a and ρ_b can be computed using Eq. (18) or Eq. (19). With the density matrices ρ_a and ρ_b to hand we compute the quantum Jensen–Shannon divergence $D_{QJS}(\rho_a, \rho_b)$ using Eq. (23). However the result depends on the relative order of the vertices in the two graphs, and we need a way to describe the mixed quantum state given by the walk on both graphs. To this end, we consider two strategies. We refer to these as the *unaligned* density matrix and the *aligned* density matrix. In the first case the density matrices are added maintaining the vertex order of the data. This results in the *unaligned* kernel

$$k_{QJSU}(G_a, G_b) = \exp(-\mu D_{QJS}(\rho_a, \rho_b)), \tag{24}$$

where μ is a decay factor in the interval $0 < \mu \le 1$,

$$D_{QJS}(\rho_a, \rho_b) = H_N\left(\frac{\rho_a+\rho_b}{2}\right) - \frac{1}{2}H_N(\rho_a) - \frac{1}{2}H_N(\rho_b) \tag{25}$$

is the quantum Jensen Shannon divergence between the mixed state density matrices ρ_a and ρ_b of G_a and G_b, and $H_N(.)$ is the von Neumann entropy of a graph density matrix defined in Eq. (21). Here μ is used to ensure that the large value does not tend to dominant the kernel value, and in this work we use $\mu = 1$. The proposed kernel can be viewed as a similarity measure between the quantum states associated with a pair of graphs. Unfortunately, since the unaligned density matrix ignores the vertex correspondence information for a pair of graphs, it is not permutation invariant to vertex order. Instead it relies only on the global properties and long-range interactions between vertices to differentiate between structures.

In the second case the mixing is performed after the density matrices are optimally aligned, thus computing a lower bound of the divergence over the set Σ of state permutations. This results in the *aligned* kernel

$$\begin{aligned} k_{QJSA}(G_a, G_b) &= \max_{Q \in \Sigma} \exp\left(-\mu D_{QJS}\left(\rho_a, Q\rho_b Q^T\right)\right) \\ &= \exp\left(-\mu \min_{Q \in \Sigma} D_{QJS}\left(\rho_a, Q\rho_b Q^T\right)\right). \end{aligned} \tag{26}$$

Here the alignment step adds correspondence information allowing for a more fine-grained discrimination between structures.

In both cases the density matrix of smaller size is padded to the size of the larger one before computing the divergence. Note that this is equivalent to padding the adjacency matrix of the smaller graph to be the same size as the larger one, since both these operations are simply increasing the dimension of the kernel of the density matrix.

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

3.1. Alignment of the density matrix

The definition of the aligned kernel requires the computation of the permutation that minimizes the divergence between the graphs. This is equivalent to minimizing the von Neumann entropy of the mixed state $(\rho_a+\rho_b)/2$. However, this is a computationally hard task, and so we perform a two step approximation to the problem. First, we approximate the solution by minimizing the Frobenious (L_2) distance between matrices rather than the von Neumann entropy. Second we find an approximate solution to the L_2 problem using Umeyama's method [29].

The rationale behind this approximate method is given by the fact that we already have the eigenvectors and eigenvalues of the Hamiltonian to hand, and with these we can efficiently compute the eigendecomposition of the density matrices [37] (which is required in order to apply Umeyama's spectral approach). More precisely, the authors of [37] show that

$$L\rho_a = \left(\sum_{\lambda\in\tilde{\Lambda}} \lambda P_\lambda P_\lambda^T\right)\left(\sum_{\lambda\in\tilde{\Lambda}} P_\lambda \rho_0 P_\lambda^T\right) = \sum_{\lambda\in\tilde{\Lambda}} P_\lambda \lambda \rho_0 P_\lambda^T$$
$$= \left(\sum_{\lambda\in\tilde{\Lambda}} P_\lambda \rho_0 P_\lambda^T\right)\left(\sum_{\lambda\in\tilde{\Lambda}} \lambda P_\lambda P_\lambda^T\right) = \rho_a L, \tag{27}$$

where ρ_0 denotes the density matrix for G at time $t=0$ and $P_\lambda = \sum_{k=1}^{\mu(\lambda)} \phi_{\lambda,k}\phi_{\lambda,k}^T$ is the projection operator onto the subspace spanned by the $\mu(\lambda)$ eigenvectors $\phi_{\lambda,k}$ associated with the eigenvalue λ of the Hamiltonian. In other words, given the spectral decomposition of the Laplacian matrix $L=\Phi\Lambda\Phi^T$, if we express the density matrix in the eigenvector basis given by Φ, then it assumes a block diagonal form, where each block corresponds to an eigenspace of L corresponding to a single eigenvalue. As a consequence, if L has all eigenvalues distinct, then ρ_a will have the same spectrum as L. Otherwise, we need to independently compute the eigendecomposition of each diagonal block, resulting in a complexity $O(\sum_{\lambda\in\tilde{\Lambda}}\mu(\lambda)^2)$, where $\mu(\lambda)$ denotes the multiplicity of the eigenvalue λ.

3.2. Properties of the quantum Jensen–Shannon kernel for graphs

Theorem 3.1. *If the Quantum Jensen–Shannon divergence is negative definite, then the unaligned Quantum Jensen–Shannon Kernel is positive definite.*

Proof. First note that for the computation of Quantum Jensen–Shannon divergence, padding the smaller matrix to the size of the larger one is equivalent to padding both to an even larger size. Thus we can consider the computation of the kernel matrix over N graphs to be performed over density matrices padded to the size of the larger graph. This eliminates the dependence of the density matrix of the first graph on the choice of the second graph.

In [38] the authors show that if $s(G_a,G_b)$ is a conditionally positive semidefinite kernel, i.e., $-s(G_a,G_b)$ is a negative definite kernel, then $k_s=\exp(\lambda s(G_a,G_b))$ is positive semidefinite for any $\lambda>0$.

Hence, if the Quantum Jensen–Shannon divergence $D_{QJS}(G_a,G_b)$ is negative definite, then $-D_{QJS}(G_a,G_b)$ is conditionally positive semidefinite, and the Quantum Jensen–Shannon kernel $k_{QJS}(G_a,G_b)=\exp(-\mu D_{QJS}(G_a,G_b))$ is positive definite. □

Note that this proof is conditioned on the negative definiteness of the Quantum Jensen–Shannon divergence, as is the case for the classical Jensen–Shannon divergence. Currently this is only a conjecture, proved to be true on pure quantum states, and for which there is a strong empirical evidence [39].

In general for the optimal alignment kernel [10,11] we cannot guarantee positive definiteness for the aligned version of our quantum kernel due to the lack of transitivity of the alignments. We do have, however, a similar result when using any (possibly non optimal) set of alignments that satisfy transitivity.

Corollary 3.2. *If the Quantum Jensen–Shannon divergence is negative definite and the density matrices are aligned by a transitive set of permutations, then the unaligned Quantum Jensen–Shannon Kernel is positive definite.*

Proof. If the alignments are transitive, each of the density matrices can be aligned to a single common frame, for example that of the first graph. The aligned kernel on the original unaligned data is equivalent to the unaligned kernel on the new pre-aligned data. □

Another important property for a kernel is its independence on the ordering of the data. There are two ways by which the kernel can depend on this ordering: First, it might depend on the order in which different graphs are presented. Second, it might depend on the order in which the vertices are presented in each graph.

Since the kernel only depends on second order relations, a sufficient condition for the first type of independence is for the kernel to be deterministic and symmetric. If this is so, then a permutation of the graphs will just result in the same permutation to be applied to the rows and columns of the kernel matrix. For the second type of invariance, we need full invariance of the computed kernel values with respect to vertex permutations.

The unaligned kernel trivially satisfies the first condition since the Quantum Jensen–Shannon divergence is both deterministic and symmetric, while it fails to satisfy the second condition, thus resulting in a kernel that is independent on the order of the graphs, but not on the vertex order within each graph. For the aligned kernel, on the other hand, the addition of the alignment step requires a little more analysis.

Theorem 3.3. *The aligned Quantum Jensen–Shannon kernel is independent both to graph and to vertex order.*

Proof. Recall that given two graphs G_a and G_b with mixed state density matrices ρ_a and ρ_b, the aligned kernel is

$$k_{QJSA}(G_a,G_b) = \max_{Q\in\Sigma} \exp\left(-\mu D_{QJS}\left(\rho_a, Q\rho_b Q^T\right)\right). \tag{28}$$

As a result the kernel value is uniquely identified even when there are multiple possible alignments. Due to their optimality these alignments must all result in the same optimal divergence. It is easy to see that the kernel is also symmetric. In fact, since the von Neumann entropy is invariant to similarity transformations, we have

$$H_N\left(\frac{\rho_a+Q\rho_bQ^T}{2}\right) = H_N\left(Q\frac{Q^T\rho_aQ+\rho_b}{2}Q^T\right) = H_N\left(\frac{Q^T\rho_aQ+\rho_b}{2}\right). \tag{29}$$

From the above $D_{QJS}(\rho_a,Q\rho_bQ^T)=D_{QJS}(Q^T\rho_aQ,\rho_b)=D_{QJS}(\rho_b,Q^T\rho_aQ)$. Hence,

$$k_{QJSA}(G_a,G_b) = \max_{Q\in\Sigma}\exp(-\mu D_{QJS}(\rho_a,Q\rho_bQ^T))$$
$$= \max_{Q\in\Sigma}\exp(-\mu D_{QJS}(\rho_b,Q\rho_aQ^T)) = k_{QJSA}(G_b,G_a). \tag{30}$$

As a result, the aligned kernel is deterministic and symmetric, even if the optimizer might not be. Thus the kernel is independent of the order of the graphs.

The independence of vertex ordering again derives directly from the optimality of the value of the divergence. Let G_b' be a vertex-permuted version of G_b, such that its mixing matrix $\rho_b'=T\rho_bT^T$ for a permutation matrix $T\in\Sigma$, and assume that $k_{QJSA}(G_a,G_b')\neq k_{QJSA}(G_a,G_b)$. Without loss of generality we can assume $k_{QJSA}(G_a,G_b')>k_{QJSA}(G_a,G_b)$. Let

$$Q* = \operatorname*{argmax}_{Q\Sigma}\exp(-\mu D_{QJS}(\rho_a,Q\rho_b'Q^T)), \tag{31}$$

then

$$\exp(-\mu D_{QJS}(\rho_a, Q*\rho_b'Q*^T)) = \exp(-\mu D_{QJS}(\rho_a, Q*T\rho_b T^T Q*^T)) > \max_{Q\in\Sigma} \exp(-\mu D_{QJS}(\rho_a, Q\rho_b Q^T)), \quad (32)$$

leading to a contradiction. □

Note that these properties depend on the optimality of the alignment with respect to the Quantum Jensen–Shannon divergence. We approximate the optimal alignment by the minimization of the Frobenious (L_2) norm between the density matrices. This is approximated using Umeyama's algorithm since the resulting matching problem is still in general NP-hard. However, in the proof, the optimality was used as a means to uniquely identify the value of the divergence. Any other uniquely identified value would still give an order invariant kernel, even if this is suboptimal.

Eq. (29) does not depend on the choice of optimal alignment, and so it would still hold under L_2 optimality. The main problem is whether two permutations leading to the same (minimal) Frobenious distance between the density matrices can give different divergence values. If permutations Q_1 and Q_2 differ by the combination of symmetries present in the two graphs, i.e., $Q_1 = SQ_2R$ with $S \in \mathrm{Aut}(G_a)$ and $R \in \mathrm{Aut}(G_b)$, then clearly they will have the same divergence since $\rho_a = S\rho_s S^T$ and $\rho_b = R\rho_b R^T$, but that might not characterize all the L_2 optimal permutations. Nonetheless, we have the following

Theorem 3.4. *If the L_2 optimal permutation is unique modulo isomorphism in the two graphs, then the L_2 approximation is independent of the order of the graphs.*

Proof. Derives directly from the uniqueness of the kernel value for all the optimizers. □

There remains another source of error, since Umeyama's algorithm is only an approximation to the Frobenious minimizer. The impact of this is much wider since the lack of optimality implies an inability to uniquely identify the value of the kernel in an invariant way. However, this is implicit in all alignment-based approaches due to the NP hard nature of the graph matching problem.

3.3. Complexity analysis

For a graph dataset having N graphs, the kernel matrix of the quantum Jensen–Shannon graph kernel can be computed using Algorithm 1.

Algorithm 1. Computing the kernel matrix of the quantum Jensen–Shannon graph kernel on N graphs.

Input: A graph dataset $\mathbf{G} = \{G_1, \ldots, G_a, \ldots, G_b, \ldots, G_N\}$ with N test graphs. **Output:** A $|N| \times |N|$ kernel matrix K_{QJS}.

1. Computing the density matrices of graphs: For each graph $G_a(V_a, E_a)$, compute the density matrix evolving from time 0 to time T using the definition in Section 19
2. Computing the density matrix for each pair of graphs: For each pair of graphs $G_a(V_a, E_a)$ and $G_b(V_b, E_b)$ with their density matrices ρ_a and ρ_b, compute $\frac{\rho_a+\rho_b}{2}$ using the unaligned or the aligned procedure.
3. Computing the kernel matrix for the N graphs: Compute the kernel matrix K_{QJS}. For each pair of graphs $G_a(V_a, E_a)$ and $G_b(V_b, E_b)$, compute the (G_a, G_b) th element of K_{QJS} as $k_{QJS}(G_a, G_b)$ using Eq. (24) or Eq. (26).

For N graphs each of which has n vertices, the time complexity of the quantum Jensen–Shannon graph kernel on all pairs of these graphs is given by the three computational steps of Algorithm 1. These are (a) The computation of the density matrix for each graph. Based on the definition in Section 2.2, this step has time complexity $O(Nn^3)$. (b) The computation of the aligned density matrix and the unaligned density matrix for all pairs of graphs. This has time complexities $O(N^2n^4)$ and $O(N^2n)$. (c) As a result, the time complexities of the quantum kernel using the aligned and unaligned density matrices are $O(Nn^3+N^2n^4)$ and $O(Nn^3+N^2n)$, respectively.

4. Experimental evaluation

In this section, we test our graph kernels on several standard graph based datasets. There are three stages to our experimental evaluation. First, we evaluate the stability of the von Neumann entropies obtained from the density matrices with time t. Second, we empirically compare our new kernel with several alternative state-of-the-art graph kernels. Finally, we evaluate the computational efficiency of our new kernel.

4.1. Graph datasets

We show the performance of our quantum Jensen–Shannon graph kernel on five standard graph based datasets from bioinformatics and computer vision. These datasets include MUTAG, PPIs, PTC(MR), COIL5 and Shock.

Some statistic concerning the datasets are given in Table 1.

MUTAG: The MUTAG dataset consists of graphs representing 188 chemical compounds, and aims to predict whether each compound possesses mutagenicity [40]. Since the vertices and edges of each compound are labeled with a real number, we transform these graphs into unweighted graphs.

PPIs: The PPIs dataset consists of protein–protein interaction networks (PPIs) [41]. The graphs describe the interaction relationships between histidine kinases in different species of bacteria. Histidine kinase is a key protein in the development of signal transduction. If two proteins have direct (physical) or indirect (functional) association, they are connected by an edge. There are 219 PPIs in this dataset and they are collected from five different kinds of bacteria with the following evolution order (from older to more recent): *Aquifex*4 and *thermotoga*4 PPIs from *Aquifex aelicus* and *Thermotoga maritima*, respectively, *Gram-Positive*52 PPIs from *Staphylococcus aureus*, *Cyanobacteria*73 PPIs from *Anabaena variabilis* and *Proteobacteria*40 PPIs from *Acidovorax avenae*. There is an additional class (*Acidobacteria*46 PPIs) which is more controversial in terms of the bacterial evolution since they were discovered. Here we select Proteobacteria40 PPIs and Acidobacteria46 PPIs as the testing graphs.

PTC: The PTC (The Predictive Toxicology Challenge) dataset records the carcinogenicity of several 100 chemical compounds for male rats (MR), female rats (FR), male mice (MM) and female mice (FM) [42]. These graphs are very small (i.e., 20–30 vertices), and sparse (i.e., 25–40 edges. We select the graphs of male rats (MR) for evaluation. There are 344 test graphs in the MR class.

COIL5: We create a dataset referred to as COIL5 from the COIL image database. The COIL database consists of images of 100 3D

Table 1
Information of the graph based datasets.

Datasets	MUTAG	PPIs	PTC	COIL5	Shock
Max # vertices	28	232	109	241	33
Min # vertices	10	3	2	72	4
Ave # vertices	17.93	109.60	25.60	144.90	13.16
# graphs	188	86	344	360	150
# classes	2	2	2	5	10

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

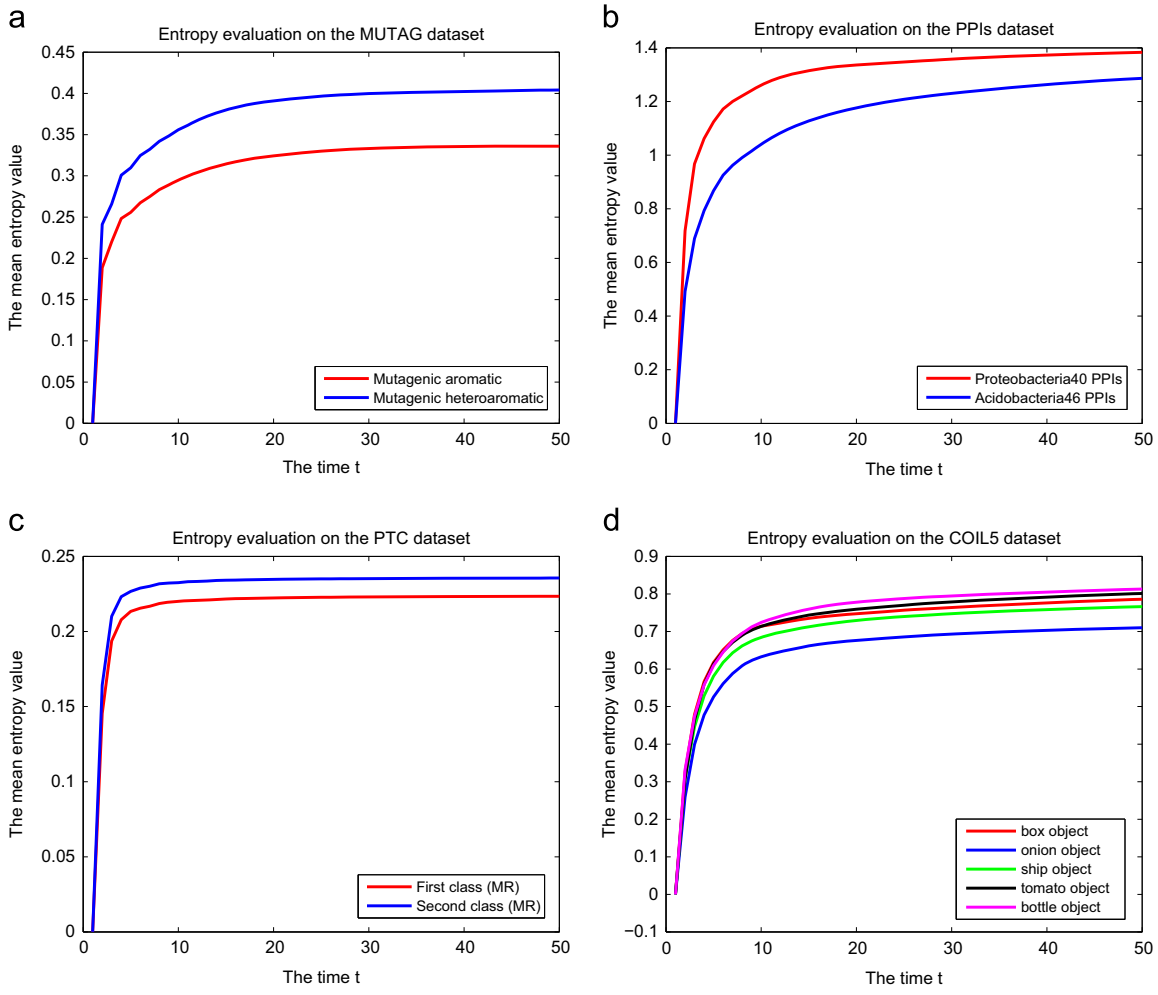

Fig. 1. Evaluations on the von Neumann entropy with time t: (a) on the MUTAG dataset, (b) on the PPIs dataset, (c) on the PTC(MR) dataset and (d) on the COIL5 dataset.

objects. In our experiments, we use the images for the first five objects. For each of these objects we employ 72 images captured from different viewpoints. For each image we first extract corner points using the Harris detector, and then establish Delaunay graphs based on the corner points as vertices. Each vertex is used as the seed of a Voronoi region, which expands radially with a constant speed. The linear collision fronts of the regions delineate the image plane into polygons, and the Delaunay graph is the region adjacency graph for the Voronoi polygons. Associated with each object, there are 72 graphs from the 72 different object views. Since the graphs of an object through different viewpoints are very different, this dataset is hard for graph classification.

Shock: The Shock dataset consists of graphs from the Shock 2D shape database. Each graph is a skeletal-based representation of the differential structure of the boundary of a 2D shape. There are 150 graphs divided into 10 classes. Each class contains 15 graphs.

4.2. Evaluations on the von Neumann entropy with time t

We commence by investigating the von Neumann entropy associated with the graphs as the time t varies. In this experiment, we use the testing graphs in the MUTAG, PPIs, PTC and COIL5 datasets, while we do not perform the evaluation on the Shock dataset. For each graph, we allow a continuous-time quantum walk to evolve with $t=0,1,2,\ldots,50$. As the walk evolves from time $t=0$ to time $t=50$ we compute the corresponding density matrix using Eq. (15). Thus, at each time t we can compute the von Neumann entropy of the corresponding density matrix associated with each graph. The experimental results are shown in Fig. 1. The subfigures of Fig. 1 show the mean von Neumann entropies for the graphs in the MUTAG, PPIs, PTC and COIL5 datasets separately. The x-axis shows the time t from 0 to 50, and the y-axis shows the mean value of the von Neumann entropies for graphs belonging to the same class. Here the different lines represent the entropies for the different classes of graphs. These plots demonstrate that the von Neumann entropy can be used to distinguish the different classes of graphs present in a dataset.

4.3. Experiments on standard graph datasets from bioinformatics

4.3.1. Experimental setup

We now evaluate the performance of our quantum Jensen–Shannon kernel using both the aligned density matrix (QJSA) and the unaligned density matrix (QJSU). Furthermore, we compare our kernel with several alternative state-of-the-art graph kernels. These kernels include (1) the Weisfeiler–Lehman subtree kernel (WL) [5], (2) the shortest path graph kernel (SPGK) [4], (3) the Jensen–Shannon graph kernel associated with the steady state random walk (JSGK) [14], (4) the backtrackless random walk kernel using the Ihara zeta function based cycles (BRWK) [6], and (5) the random-walk graph kernel [3]. For our quantum kernel, we decide to let $t\to\infty$. For the Weisfeiler–Lehman subtree kernel, we set the dimension of the Weisfeiler–Lehman isomorphism as 10. Based on the definition in [5], this means that we compute 10 different Weisfeiler–Lehman subtree kernel matrices (i.e., $k(1),k(2),\ldots,k(10)$) with different subtree heights

Table 2
Accuracy comparisons (in % ± standard errors) on graph datasets abstracted from bioinformatics and computer vision.

Datasets	MUTAG	PPIs	PTC(MR)	COIL5	Shock
QJSU	82.72 ± .44	69.50 ± 1.20	56.70 ± .49	**70.11** ± .61	40.60 ± . 92
QJSA	82.83 ± .50	73.37 ± 1.04	**57.39** ± .46	69.75 ± .58	**42.80** ± .86
WL	82.05 ± .57	**78.50** ± 1.40	56.05 ± .51	33.16 ± 1.01	36.40 ± 1.00
SPGK	**83.38** ± .81	61.12 ± 1.09	56.55 ± .53	69.66 ± .52	37.88 ± .93
JSGK	83.11 ± .80	57.87 ± 1.36	57.29 ± .41	69.13 ± .79	21.73 ± .76
BRWK	77.50 ± .75	53.50 ± 1.47	53.97 ± .31	14.63 ± .21	0.33 ± .37
RWGK	80.77 ± .72	55.00 ± .88	55.91 ± .37	20.80 ± .47	2.26 ± 1.01

Table 3
Runtime comparisons on graph datasets abstracted from bioinformatics and computer vision.

Datasets	MUTAG	PPIs	PTC	COIL5	Shock
QJSU	20″	59″	1′46″	18′20″	14″
QJSA	1′30″	23′25″	16′40″	8h29′	32″
WL	4″	13″	11″	1′5″	3″
SPGK	1″	7″	1″	31″	1″
JSGK	1″	1″	1″	1″	1″
BRWK	2″	14′20″	3″	16′46″	8″
RWGK	46″	1′7″	2′35″	19′40″	23″

h (h=1,2,…,10), respectively. Note that the WL and SPGK kernels are able to accommodate attributed graphs. In our experiments, we use the vertex degree as a vertex label for the WL and SPGK kernels.

For each kernel and dataset, we perform a 10-fold cross-validation using a C-Support Vector Machine (C-SVM) in order to evaluate the classification accuracies of the different kernels. More specifically, we use the C-SVM implementation of LIBSVM [43]. For each class, we use 90% of the samples for training and the remaining 10% for testing. The parameters of the C-SVMs are optimized separately for each dataset. We repeat the evaluation 10 times and we report the average classification accuracies (± standard error) of each kernel in Table 2. Furthermore, we also report the runtime of computing the kernel matrices of each kernel in Table 3, with the runtime measured under Matlab R2011a running on a 2.5 GHz Intel 2-Core processor (i.e., i5-3210m). Note that, for the WL kernel, the classification accuracies are the average accuracies over all the 10 matrices, and the runtime refers to that required for computing all the 10 matrices (see [5] for details).

4.3.2. *Results and discussion*

(a) On the MUTAG dataset, the accuracies for all of the kernels are similar. The SPGK kernel achieves the highest accuracy. Yet, the accuracies of our QJSA and QJSU quantum kernels are competitive with that of the SPGK kernel and outperform those of other

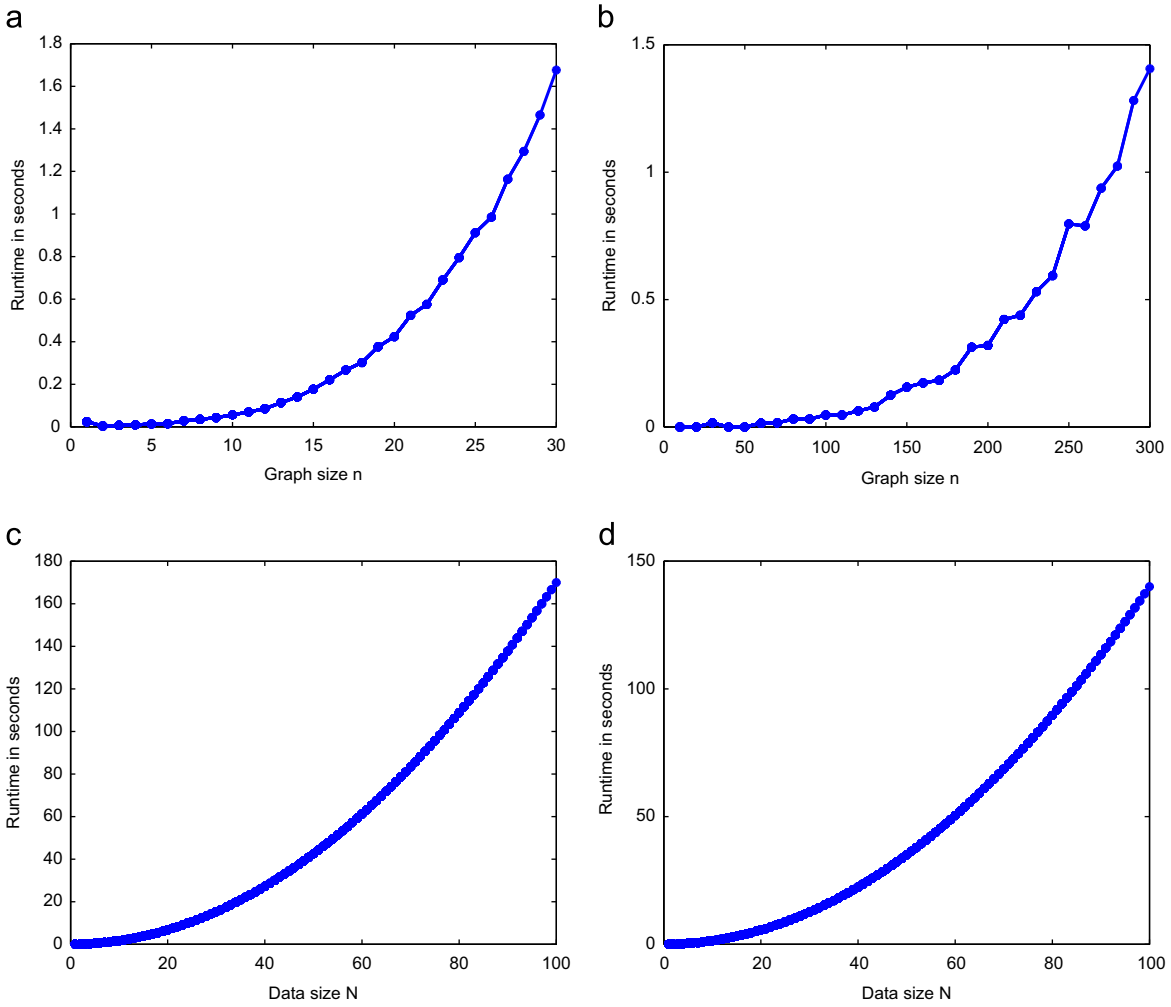

Fig. 2. Runtime evaluation: (a) QJSA with graph size n, (b) QJSU with graph size n, (c) QJSA with data size N and (d) QJSU with data size N.

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028

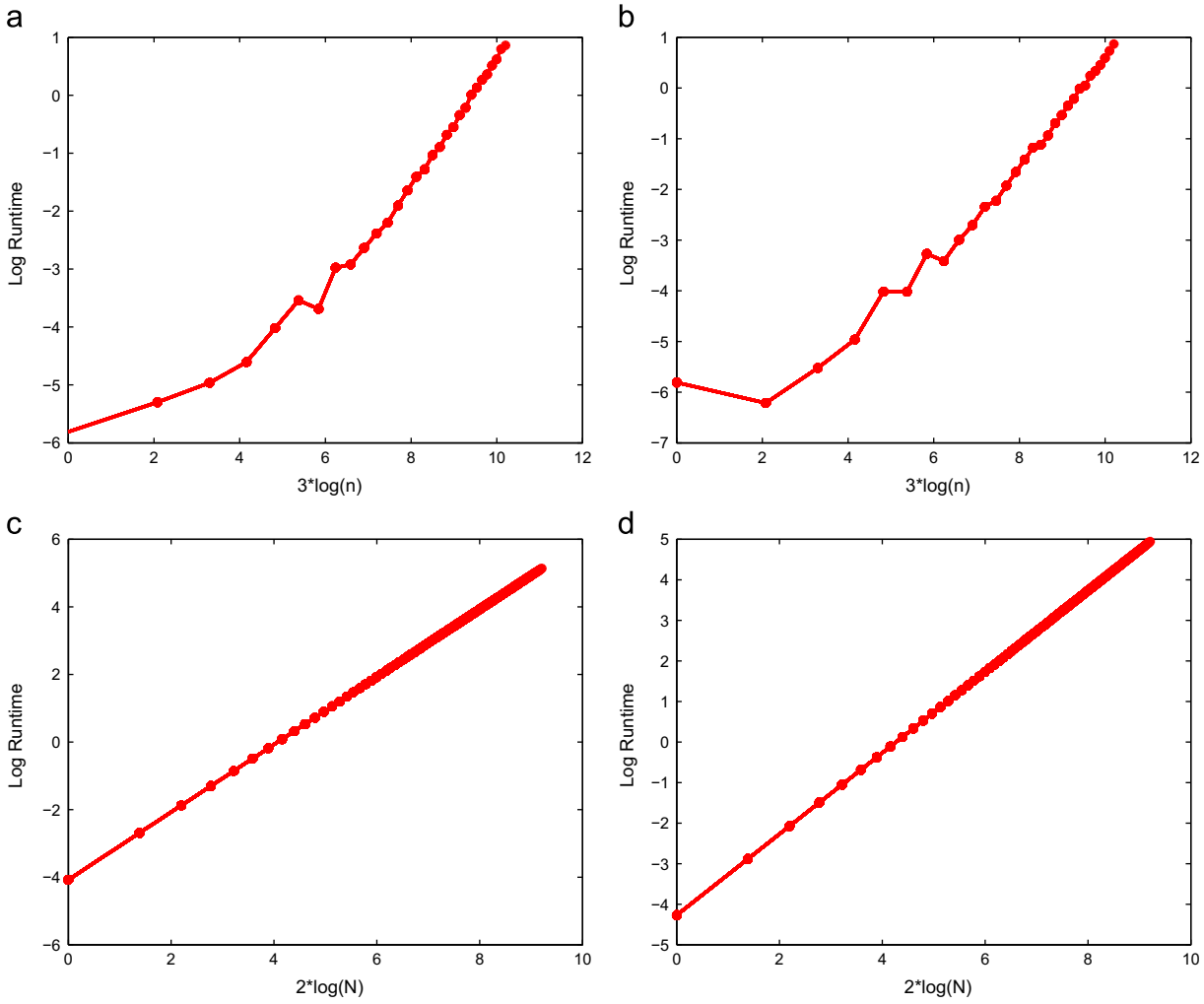

Fig. 3. Log runtime evaluation: (a) log runtime vs 3 log (n) using QJSA kernel, (b) log runtime vs 3 log (n) using QJSU kernel, (c) log runtime vs 2 log (N) using QJSA kernel and (d) log runtime vs 2 log (N) using QJSU kernel.

kernels. (b) On the PPIs dataset, the WL kernel achieves the highest accuracy. The accuracies of our QJSA and QJSU quantum kernels are lower than that of the WL kernel, but outperform those of other kernels. (c) On the PTC dataset, the accuracies for all of the kernels are similar. Our QJSA quantum kernel achieves the highest accuracy. The accuracy of our QJSU quantum kernel is competitive or outperforms that of other kernels. (d) On the COIL5 dataset, the accuracies of all the kernels are similar with the exception of the WL, RWGK and BRWK kernels. Our QJSA quantum kernel achieves the highest accuracy. The accuracy of our QJSU quantum kernel overcomes that of other kernels. (e) On the Shock dataset, our QJSA quantum kernel achieves the highest accuracy. The accuracy of our QJSU quantum kernel overcomes that of other kernels.

Overall, in terms of classification accuracy, our QJSA and QJSU quantum Jensen–Shannon graph kernels outperform or are competitive with the state-of-the-art kernels. Especially, the classification accuracies of our quantum kernel are significantly better than those of the graph kernels using the classical Jensen–Shannon divergence, the classical random walk and the backtrackless random walk. This suggests that our kernel, which makes use of continuous-time quantum walks to probe the graphs structure, is successful in capturing the structural similarities between different graphs. Furthermore, we observe that the performance of our QJSA quantum kernel is better than that of our QJSU quantum kernel. The reason for this is that the aligned density matrix computed through Umeyama's matching method can reflect the precise correspondence information between pairs of vertices in these graphs, while the unaligned density matrix ignores the correspondence information, and it is not permutation invariant to the vertex order.

In terms of the runtime, all the kernels can complete the computation in polynomial time on all the datasets. The computation efficiency of our QJSU is lower than that of the WL, SPGK and JSGK kernels, but it is faster than that of the BRWK and RWGK kernels (i.e., the kernels using the classical random walk and the backtrackless random walk). The computational efficiency of our QJSA kernel is lower than any alternative kernel. The reason for this is that the QJSA kernel requires extra computation for the alignment of the density matrices.

4.4. Computational evaluation

In this subsection, we evaluate the relationship between the computational overheads (i.e., the CPU runtime) of our kernel and the structural complexity or the number of the associated graphs.

4.4.1. Experimental setup

We evaluate the computational efficiency of our quantum kernel on randomly generated graphs with respect to two parameters: (a) the graph size n and (b) the graph dataset size N.

More specifically, we vary $n = \{10, 20, \ldots, 300\}$ and $N = \{1, 2, \ldots, 100\}$, separately.

We first generate 30 pairs of graphs with an increasing number of vertices. We report the runtime for computing the kernel values for all the pairs of graphs. We also generate 100 graph datasets with an increasing number of test graphs. Each test graph has 50 vertices. We report the runtime for computing the kernel matrices of the graphs from each dataset. The CPU runtime is reported in Fig. 2. The experiments are run in Matlab R2011a on a 2.5 GHz Intel 2-Core processor (i.e., i5-3210m).

4.4.2. Experimental results

Fig. 2(a) and (c) shows the results obtained with the QJSA kernel, when varying the parameters n and N, respectively. Fig. 2 (b) and (d) shows those obtained with the QJSU kernel. Furthermore, for the parameters n and N, we also plot the log runtime versus $3\log(n)$ and $2\log(N)$ respectively. Fig. 3(a) and (c) shows those obtained with the QJSA kernel. Fig. 3(b) and (d) shows those obtained with the QJSA kernel. In Fig. 3 there is an approximately linear relationship between the log runtime and the corresponding log parameters (i.e., $3\log(n)$ and $2\log(N)$).

From Fig. 2 and 3 we obtain the following conclusions. When varying the number of vertices n of the graphs, we observe that the runtime for computing the quantum Jensen–Shannon QJSA and QJSU graph kernels scales approximately cubically with n. When varying the graph dataset size N, we observe that the runtime for computing the QJSA and QJSU graph kernels scales quadratically with N. These computational evaluations verify that our quantum Jensen–Shannon graph kernel can be computed in polynomial time.

Furthermore, we observe that the QJSU kernel is a little more efficient than the QJSA kernel. The reason for this is that the QJSA kernel requires extra computations for evaluating the vertex correspondences.

5. Conclusion

In this paper, we have developed a novel graph kernel by using the quantum Jensen–Shannon divergence and the continuous-time quantum walk on a graph. Given a graph, we evolved a continuous-time quantum walk on its structure and we showed how to associate a mixed quantum state with the vertices of the graph. From the density matrix for the mixed state we computed the von Neumann entropy. With the von Neumann entropies to hand, the kernel between a pair of graphs was defined as a function of the quantum Jensen–Shannon divergence between the corresponding density matrices. Experiments on several standard datasets demonstrate the effectiveness of the proposed graph kernel.

Our future work includes extending the quantum graph kernel to hypergraphs. Bai et al. [44] have developed a hypergraph kernel by using the classical Jensen–Shannon divergence, and Ren et al. [45] have explored the use of discrete-time quantum walks on directed line graphs, which can be used as structural representations for hypergraphs. It would thus be interesting to extend these works by using the quantum Jensen–Shannon divergence to compare the quantum walks on the directed line graphs associated with a pair of hypergraphs.

Conflict of interest statement

None declared.

Acknowledgements

Edwin R. Hancock is supported by a Royal Society Wolfson Research Merit Award.

We thank Prof. Karsten Borgwardt and Dr. Nino Shervashidze for providing the Matlab implementation for the various graph kernel methods, and Dr. Geng Li for providing the graph datasets. We also thank Prof. Horst Bunke and Dr. Peng Ren for the constructive discussion and suggestions.

References

[1] B. Scholkopf, A. Smola, Learning with Kernels, MIT Press, 2002.
[2] D. Haussler, Convolution Kernels on Discrete Structures, Technical Report UCS-CRL-99-10, Santa Cruz, CA, USA, 1999.
[3] H. Kashima, K. Tsuda, A. Inokuchi, Marginalized kernels between labeled graphs, in: Proceedings of International Conference on Machine Learning (ICML), 2003, pp. 321–328.
[4] K.M. Borgwardt, H.P. Kriegel, Shortest-path kernels on graphs, in: Proceedings of the IEEE International Conference on Data Mining (ICDM), 2005, pp. 74–81.
[5] N. Shervashidze, P. Schweitzer, E.J. van Leeuwen, K. Mehlhorn, K.M. Borgwardt, Weisfeiler–Lehman graph kernels, J. Mach. Learn. Res. 1 (2010) 1–48.
[6] F. Aziz, R.C. Wilson, E.R. Hancock, Backtrackless walks on a graph, IEEE Trans. Neural Netw. Learn. Syst. 24 (2013) 977–989.
[7] F. Costa, K.D. Grave, Fast neighborhood subgraph pairwise distance kernel, in: Proceedings of International Conference on Machine Learning (ICML), 2010, pp. 255–262.
[8] Kriege Nils, Mutzel Petra, Subgraph matching Kernels for attributed graphs, in: Proceedings of International Conference on Machine Learning (ICML), 2012.
[9] Neumann Marion, Novi Patricia, Roman Garnett, Kristian Kersting, Efficient graph kernels by randomization, Machine Learning and Knowledge Discovery in Databases, Springer, Berlin, Heidelberg (2012) 378–393.
[10] Holger Frohlich, Jorg K. Wegner, Florian Sieker, Andreas Zell, Optimal assignment kernels for attributed molecular graphs, in: International Conference on Machine Learning, 2005, pp. 225–232.
[11] Jean-Philippe Vert, The Optimal Assignment Kernel is Not Positive Definite, arXiv:0801.4061, 2008.
[12] Michel Neuhaus, Horst Bunke, Edit distance-based kernel functions for structural pattern classification, Pattern Recognit. 39 (10) (2006) 1852–1863.
[13] P. Lamberti, A. Majtey, A. Borras, M. Casas, A. Plastino, Metric character of the quantum Jensen–Shannon divergence, Phys. Rev. A 77 (2008) 052311.
[14] L. Bai, E.R. Hancock, Graph kernels from the Jensen–Shannon divergence, J. Math. Imaging Vis. 47 (2013) 60–69.
[15] L.K. Grover, A fast quantum mechanical algorithm for database search, in: Proceedings of the ACM Symposium on the Theory of Computing (STOC), 1996, pp. 212–219.
[16] P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput. 26 (1997) 1484–1509.
[18] A. Majtey, P. Lamberti, D. Prato, Jensen–Shannon divergence as a measure of distinguishability between mixed quantum states, Phys. Rev. A 72 (2005) 052310.
[19] E. Farhi, S. Gutmann, Quantum computation and decision trees, Phys. Rev. A 58 (1998) 915.
[20] D. Aharonov, A. Ambainis, J. Kempe, U. Vazirani, Quantum walks on graphs, in: Proceedings of ACM Theory of Computing (STOC), 2001, pp. 50–59.
[21] A. Ambainis, E. Bach, A. Nayak, A. Vishwanath, J. Watrous, One-dimensional quantum walks, in: Proceedings of ACM Theory of Computing (STOC), 2001, pp. 60–69.
[22] A. Ambainis, Quantum walks and their algorithmic applications, Int. J. Quantum Inf. 1 (2003) 507–518.
[23] A.M. Childs, E. Farhi, S. Gutmann, An example of the difference between quantum and classical random walks, Quantum Inf. Process. 1 (2002) 35–53.
[24] A.M. Childs, Universal computation by quantum walk, Phys. Rev. Lett. 102 (18) (2009) 180501.
[26] L. Bai, E.R. Hancock, A. Torsello, L. Rossi, A quantum Jensen–Shannon graph kernel using the continuous-time quantum walk, in: Proceedings of Graph-Based Representations in Pattern Recognition (GbRPR), 2013, pp. 121–131.
[27] L. Rossi, A. Torsello, E.R. Hancock, A continuous-time quantum walk kernel for unattributed graphs, in: Proceedings of Graph-Based Representations in Pattern Recognition (GbRPR), 2013, pp. 101–110.
[28] L. Rossi, A. Torsello, E.R. Hancock, Attributed graph similarity from the quantum Jensen–Shannon divergence, in: Proceedings of Similarity-Based Pattern Recognition (SIMBAD), 2013, pp. 204–218.
[29] S. Umeyama, An eigendecomposition approach to weighted graph matching problems, IEEE Trans. Pattern Anal. Mach. Intell. 10 (5) (1988) 695–703.
[30] Nan Hu, Leonidas Guibas, Spectral Descriptors for Graph Matching, arXiv:1304.1572, 2013.
[31] J. Kempe, Quantum random walks: an introductory overview, Contemp. Phys. 44 (2003) 307–327.
[32] M. Nielsen, I. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2010.

[33] A.F. Martins, N.A. Smith, E.P. Xing, P.M. Aguiar, M.A. Figueiredo, Nonextensive information theoretic kernels on measures, J. Mach. Learn. Res. 10 (2009) 935–975.
[34] W.K. Wootters, Statistical distance and Hilbert space, Phys. Rev. D 23 (2) (1981) 357.
[35] G. Lindblad, Entropy, information and quantum measurements, Commun. Math. Phys. 33 (4) (1973) 305–322.
[36] D. Bures, An extension of Kakutani's theorem on infinite product measures to the tensor product of semifinite W∗-algebras, Trans. Am. Math. Soc. 135 (1969) 199–212.
[37] L. Rossi, A. Torsello, E.R. Hancock, R. Wilson, Characterising graph symmetries through quantum Jensen–Shannon divergence, Phys. Rev. E 88 (3) (2013) 032806.
[38] R. Konder, J. Lafferty, Diffusion kernels on graphs and other discrete input spaces, in: Proceedings of International Conference on Machine Learning (ICML), 2002, pp. 315–322.
[39] B. Jop, P. Harremos, Properties of classical and quantum Jensen–Shannon divergence, Phys. Rev. A 79 (5) (2009) 052311.
[40] A.K. Debnath, R.L. Lopez de Compadre, G. Debnath, A.J. Shusterman, C. Hansch, Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds, correlation with molecular orbital energies and hydrophobicity, J. Med. Chem. 34 (1991) 786–797.
[41] F. Escolano, E.R. Hancock, M.A. Lozano, Heat diffusion: thermodynamic depth complexity of networks, Phys. Rev. E 85 (2012) 036206.
[42] G. Li, M. Semerci, B. Yener, M.J. Zaki, Effective graph classification based on topological and label attributes, Stat. Anal. Data Mining 5 (2012) 265–283.
[43] C.-C. Chang, C.-J. Lin, LIBSVM: A Library for Support Vector Machines. Software available at ⟨http://www.csie.ntu.edu.tw/cjlin/libsvm⟩, 2011.
[44] L. Bai, E.R. Hancock, P. Ren, A Jensen–Shannon kernel for hypergraphs, in: Proceedings of Structural, Syntactic, and Statistical Pattern Recognition-Joint IAPR International Workshop (SSPR&SPR), 2012, pp. 79–88.
[45] P. Ren, T. Aleksic, D. Emms, R.C. Wilson, E.R. Hancock, Quantum walks, Ihara zeta functions and cospectrality in regular graphs, Quantum Inf. Process. 10 (2011) 405–417.

Lu Bai received both the B.Sc. and M.Sc. degrees from Faculty of Information Technology, Macau University of Science and Technology, Macau SAR, P.R. China. He is currently pursuing the Ph.D. degree in the University of York, York, UK. He is a member of British Machine Vision Association and Society for Pattern Recognition (BMVA). His current research interests include structural pattern recognition, machine learning, soft computing and approximation reasoning, especially in kernel methods and complexity analysis on (hyper)graphs and networks.

Luca Rossi received his B.Sc., M.Sc., and Ph.D. in computer science from Ca' Foscari University of Venice, Italy. He is now a Postdoctoral Research Fellow at the School of Computer Science of the University of Birmingham. His current research interests are in the areas of graph-based pattern recognition, machine learning, network science and computer vision.

Andrea Torsello received his Ph.D. in computer science at the University of York, UK, and is currently an assistant professor at Ca' Foscari University of Venice, Italy. His research interests are in the areas of computer vision and pattern recognition, in particular, the interplay between stochastic and structural approaches and game-theoretic models. Recently, he co-edited a special issue of Pattern Recognition on "Similarity-based pattern recognition." He has published around 40 technical papers in refereed journals and conference proceedings and has been in the program committees of various international conferences and workshops. He was a co-chair of the edition of GbR, a well-established IAPR workshop on Graph-based methods in Pattern Recognition, held in Venice in 2009. Since November 2007 he holds a visiting research position at the Information Technology Faculty of Monash University, Australia.

Edwin R. Hancock received the B.Sc., Ph.D., and D.Sc. degrees from the University of Durham, Durham, UK. He is now a professor of computer vision in the Department of Computer Science, University of York, York, UK. He has published nearly 150 journal articles and 550 conference papers. He was the recipient of a Royal Society Wolfson Research Merit Award in 2009. He has been a member of the editorial board of the IEEE Transactions on Pattern Analysis and Machine Intelligence, Pattern Recognition, Computer Vision and Image Understanding, and Image and Vision Computing. His awards include the Pattern Recognition Society Medal in 1991, outstanding paper awards from the Pattern Recognition Journal in 1997, and the best conference best paper awards from the Computer Analysis of Images and Patterns Conference in 2001, the Asian Conference on Computer Vision in 2002, the International Conference on Pattern Recognition (ICPR) in 2006, British Machine Vision Conference (BMVC) in 2007, and the International Conference on Image Analysis and Processing in 2009. He is a Fellow of the International Association for Pattern Recognition, the Institute of Physics, the Institute of Engineering and Technology, and the British Computer Society. He was appointed as the founding Editor-in-Chief of the Institute of Engineering & Technology Computer Vision Journal in 2006. He was a General Chair for BMVC in 1994 and the Statistical, Syntactical and Structural Pattern Recognition in 2010, Track Chair for ICPR in 2004, and Area Chair at the European Conference on Computer Vision in 2006 and the Computer Vision and Pattern Recognition in 2008. He established the energy minimization methods in the Computer Vision and Pattern Recognition Workshop Series in 1997.

Please cite this article as: L. Bai, et al., A quantum Jensen–Shannon graph kernel for unattributed graphs, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.028