Optimal Transport for structured data with application on graphs

Titouan Vayer [1] **Laetitia Chapel** [1] **Rémi Flamary** [2] **Romain Tavenard** [3] **Nicolas Courty** [1]

Abstract

This work considers the problem of computing distances between structured objects such as undirected graphs, seen as probability distributions in a specific metric space. We consider **a new transportation distance** (*i.e.* that minimizes a total cost of transporting probability masses) that unveils the geometric nature of the structured objects space. Unlike Wasserstein or Gromov-Wasserstein metrics that focus solely and respectively on features (by considering a metric in the feature space) or structure (by seeing structure as a metric space), our new distance exploits jointly both information, and is consequently called Fused Gromov-Wasserstein (FGW). After discussing its properties and computational aspects, we show results on a graph classification task, where our method outperforms both graph kernels and deep graph convolutional networks. Exploiting further on the metric properties of FGW, interesting geometric objects such as Fréchet means or barycenters of graphs are illustrated and discussed in a clustering context.

1. Introduction

There is a longstanding line of research on learning from structured data, *i.e.* objects that are a combination of a feature and structural information (see for example (Bakir et al., 2007; Battaglia et al., 2018)). As immediate instances, graph data are usually ensembles of nodes with attributes (typically $\mathbb{R}^d$ vectors) linked by some specific relation. Notable examples are found in chemical compounds or molecules modeling (Kriege et al., 2016), brain connectivity (Ktena et al., 2017), or social networks (Yanardag & Vishwanathan, 2015). This generic family of objects also encompasses time series (Cuturi & Blondel, 2017), trees (Day, 1985) or even images (Bach & Harchaoui, 2007).

Being able to leverage on both feature and structural information in a learning task is a tedious task, that requires the association in some ways of those two pieces of information in order to capture the similarity between the structured data. Several kernels have been designed to perform this task (Shervashidze et al., 2011; Vishwanathan et al., 2010). As a good representative of those methods, the Weisfeiler-Lehman kernel (Vishwanathan et al., 2010) captures in each node a notion of vicinity by aggregating, in the sense of the topology of the graph, the surrounding features. Recent advances in graph convolutional networks (Bronstein et al., 2017; Kipf & Welling, 2016; Defferrard et al., 2016) allows learning end-to-end the best combination of features by relying on parametric convolutions on the graph, *i.e.* learnable linear combinations of features. In the end, and in order to compare two graphs that might have different number of nodes and connections, those two categories of methods build a new representation for every graph that shares the same space, and that is amenable to classification.

A transportation distance between structured data. Contrasting with those previous methods, we suggest in this paper to see graphs as probability distributions, embedded in a specific metric space. We propose to define a specific notion of distance between those probability distributions, that can be used in most of the classical machine learning approaches. Beyond its mathematical properties, disposing of a distance between structured data, provided it is meaningful, is desirable in many ways: *i)* it can then be plugged into distance-based machine learning algorithms such as k-nn or t-SNE *ii)* its quality is not dependent on the learning set size, and *iii)* it allows considering interesting quantities such as geodesic interpolation or barycenters. To the best of our knowledge, this is one of the first attempts to define such a distance on structured data.

Yet, defining this distance is not a trivial task. While features can always be compared using a standard metric, such as ℓ_2, comparing structures requires a notion of similarity which can be found *via* the notion of *isometry*, since the graph nodes are not ordered (we define later on which cases two graphs are considered identical). We use the notion of transportation distance to compare two graphs represented as probability distributions. Optimal transport (OT) have

[1]Univ. Bretagne-Sud, CNRS, IRISA, F-56000 Vannes [2]Univ. Côte d'Azur, CNRS, OCA Lagrange, F-06000 Nice [3]Univ. Rennes, CNRS, LETG, F-35000 Rennes. Correspondence to: Titouan Vayer <titouan.vayer@irisa.fr>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

inspired a number of recent breakthroughs in machine learning (*e.g.* (Huang et al., 2016; Courty et al., 2017; Arjovsky et al., 2017)) because of its capacity to compare empirical distributions, and also the recent advances in solving the underlying problem (Peyré & Cuturi, 2018). Yet, the natural formulation of OT cannot leverage the structural information of objects since it only relies on a cost function that compares their feature representations.

However, some modifications over OT formulation have been proposed in order to compare structural information of objects. Following the pioneering work by Mémoli (Memoli, 2011), Peyré *et al.* (Peyré et al., 2016) propose a way of comparing two distance matrices that can be seen as representations of some objects' structures. They use an OT metric called Gromov-Wasserstein distance capable of comparing two distributions even if they do not lie in the same ground space and apply it to compute barycenter of molecular shapes. Even though this approach has wide applications, it only encodes the intrinsic structural information in the transportation problem. To the best of our knowledge, the problem of including both structural and feature information in a unified OT formulation remains largely under-addressed.

OT distances that include both features and structures. Recent approaches tend to incorporate some structure information as a regularization of the OT problem. For example in (Alvarez-Melis et al., 2018) and (Courty et al., 2017), authors constrain transport maps to favor some assignments in certain groups. These approaches require a known and simple structure such as class clusters to work but do not generalize well to more general structural information. In their work (Thorpe et al., 2017), propose an OT distance that combines both a Lagrangian formulation of a signal and its temporal structural information. They define a metric, called Transportation L^p distance, that can be seen as a distance over the coupled space of time and feature. They apply it for signal analysis and show that combining both structure and feature tends to better capture the signal information. Yet, for their approach to work, the structure and feature information should lie in the same ambiant space, which is not a valid assumption for more general problems such as similarity between graphs. In (Nikolentzos et al., 2017), authors propose a graph similarity measure for discrete labeled graph with OT. Using the eigenvector decomposition of the adjency matrix, which captures graph connectivities, nodes of a graph are first embedded in a new space, then a ground metric based on the distance in both this embedding and the labels is used to compute a Wasserstein distance serving as a graph similarity measure.

Contributions. After defining structured data as probability measures (Section 2), we propose a new framework capable of taking into account both structure and feature information into the optimal transport problem. The framework can compare any usual structured machine learning data even if the feature and structure information dwell in spaces of different dimensions, allowing the comparison of undirected labeled graphs. The framework is based on a distance that embeds a trade-off parameter which allows balancing the importance of the features and the structure. We propose numerical algorithms for computing this distance (Section 3), and we evaluate it (Section 4) on both synthetic and real-world graph datasets. We also illustrate the notion of graph barycenters in a clustering problem.

Notations. The simplex histogram with n bins will be denoted as $\Sigma_n = \{h \in (\mathbb{R}_+^*)^n, \sum_{i=1}^n h_i = 1, \}$. We note $\otimes$ the tensor-matrix multiplication, *i.e.* for a tensor $L = (L_{i,j,k,l})$, $L \otimes B$ is the matrix $\left(\sum_{k,l} L_{i,j,k,l} B_{k,l}\right)_{i,j}$. $\langle . \rangle$ is the matrix scalar product associated with the Frobenius norm. For $x \in \Omega$, δ_x denotes the Dirac measure in x.

2. Structured data as probability measures

In this paper, we focus on comparing structured data which combine a feature **and** a structure information. More formally, we consider undirected labeled graphs as tuples of the form $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \ell_f, \ell_s)$ where $(\mathcal{V}, \mathcal{E})$ are the set of vertices and edges of the graph. $\ell_f : \mathcal{V} \to \Omega_f$ is a labelling function which associates each vertex $v_i \in \mathcal{V}$ with a feature $a_i \stackrel{\text{def}}{=} \ell_f(v_i)$ in some feature metric space (Ω_f, d). We will denote by *feature information* the set of all the features $(a_i)_i$ of the graph. Similarly, $\ell_s : \mathcal{V} \to \Omega_s$ maps a vertex v_i from the graph to its structure representation $x_i \stackrel{\text{def}}{=} \ell_s(v_i)$ in some structure space (Ω_s, C) specific to each graph. $C : \Omega_s \times \Omega_s \to \mathbb{R}_+$ is a symmetric application which aims at measuring the similarity between the nodes in the graph. Unlike the feature space however, Ω_s is implicit and in practice, knowing the similarity measure C will be sufficient. With a slight abuse of notation, C will be used in the following to denote both the structure similarity measure and the matrix that encodes this similarity between pairs of nodes in the graph $(C(i,k) = C(x_i, x_k))_{i,k}$. Depending on the context, C can either encode the neighborhood information of the nodes, the edge information of the graph or more generally it can model a distance between the nodes such as the shortest path distance or the harmonic distance (Verma & Zhang, 2017). When C is a metric, such as the shortest-path distance, we naturally endow the structure with the metric space (Ω_s, C). We will denote by *structure information* the set of all the structure embeddings $(x_i)_i$ of the graph.

We propose to enrich the previously described graph with a histogram which serves the purpose of signaling the relative importance of the vertices in the graph. To do so, if we

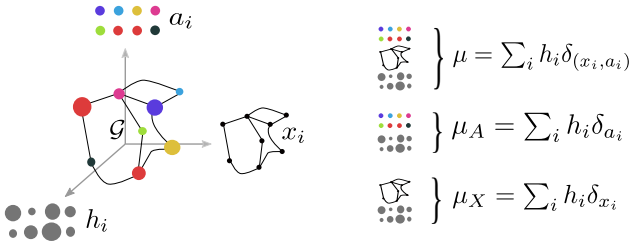

Figure 1. (Left) Labeled graph with $(a_i)_i$ its feature information, $(x_i)_i$ its structure information and histogram $(h_i)_i$ that measures the relative importance of the vertices. (Right) Associated structured data which is entirely described by a fully supported probability measure μ over the product space of feature and structure, with marginals μ_X and μ_A on the structure and the features respectively.

assume that the graph has n vertices, we equip those vertices with weights $(h_i)_i \in \Sigma_n$. Through this procedure, we derive the notion of *structured data* as a tuple $\mathcal{S} = (\mathcal{G}, h_\mathcal{G})$ where $\mathcal{G}$ is a graph as described previously and $h_\mathcal{G}$ is a function that associates a weight to each vertex. This definition allows the graph to be represented by a fully supported probability measure over the product space feature/structure $\mu = \sum_{i=1}^n h_i \delta_{(x_i,a_i)}$ which describes the entire structured data (see Fig. 1). When all the weights are equal (*i.e.* $h_i = \frac{1}{n}$), so all vertices have the same relative importance, the structured data holds the exact same information as its graph. However, weights can be used to encode some *a priori* information. For instance on segmented images, one can construct a graph using the spatial neighborhood of the segmented zones, the features can be taken as the average color in the zone, and the weights as the ratio of image pixels in the zone.

3. Fused Gromov-Wasserstein approach for structured data

We aim at defining a distance between two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, described respectively by their probability measure $\mu = \sum_{i=1}^n h_i \delta_{(x_i,a_i)}$ and $\nu = \sum_{i=1}^m g_j \delta_{(y_j,b_j)}$, where $h \in \Sigma_n$ and $g \in \Sigma_m$ are histograms. Without loss of generality we suppose $(x_i, a_i) \neq (x_j, a_j)$ for $i \neq j$ (same for y_j and b_j).

We introduce $\Pi(h, g)$ the set of all admissible couplings between h and g, *i.e.* the set :

$$\Pi(h,g) = \{\pi \in \mathbb{R}_+^{n \times m} \text{ s.t. } \sum_{i=1}^n \pi_{i,j} = h_j \ , \sum_{j=1}^m \pi_{i,j} = g_i\},$$

where $\pi_{i,j}$ represents the amount of mass shifted from the bin h_i to g_j for a coupling π. To that extent, the matrix π describes a probabilistic matching of the nodes of the two graphs.

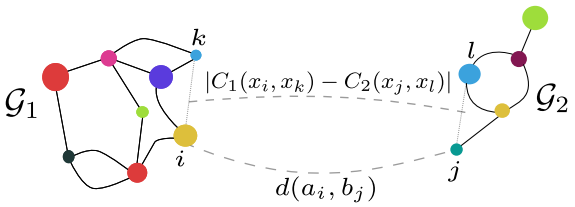

Figure 2. FGW loss E_q for a coupling π depends on both a similarity between each feature of each node of each graph $(d(a_i, b_j))_{i,j}$ and between all intra-graph structure similarities $(|C_1(x_i, x_k) - C_2(x_j, x_l)|)_{i,j,k,l}$.

$M_{AB} = (d(a_i, b_j))_{i,j}$ is a $n \times m$ matrix standing for the distance between the features. The structure matrices are denoted C_1 and C_2, and μ_X and μ_A (resp. ν_Y and ν_B) are representative of the marginals of μ (resp. ν) *w.r.t.* the structure and feature respectively (see Fig. 1). We also define the similarity between the structures by measuring the similarity between all pairwise distances within each graph thanks to the 4-dimensional tensor $L(C_1, C_2)$:

$$L_{i,j,k,l}(C_1, C_2) = |C_1(i,k) - C_2(j,l)|.$$

3.1. FGW distance

We define a novel Optimal Transport discrepancy called the Fused Gromov-Wasserstein distance. It is defined for a trade-off parameter $\alpha \in [0, 1]$ as

$$FGW_{q,\alpha}(\mu, \nu) = \min_{\pi \in \Pi(h,g)} E_q(M_{AB}, C_1, C_2, \pi) \qquad (1)$$

where

$$\begin{aligned} E_q(M_{AB}, C_1, C_2, \pi) &= \left\langle (1-\alpha) M_{AB}^q + \alpha L(C_1, C_2)^q \otimes \pi, \pi \right\rangle \\ &= \sum_{i,j,k,l} (1-\alpha) d(a_i, b_j)^q + \alpha |C_1(i,k) - C_2(j,l)|^q \pi_{i,j} \pi_{k,l} \end{aligned}$$

The FGW distance looks for the coupling π between the vertices of the graph that minimizes the cost E_q which is a linear combinaison of a cost $d(a_i, b_j)$ of transporting one feature a_i to a feature b_j and a cost $|C_1(i,k) - C_2(j,l)|$ of transporting pairs of nodes in each structure (see Fig. 2). As such, the optimal coupling tends to associate pairs of feature and structure points with similar distances within each structure pair and with similar features. As an important feature of FGW, by relying on a sum of (inter- and intra-)vertex-to-vertex distances, it can handle structured data with continuous attributed or discrete labeled nodes (thanks to the definition of d) and can also be computed even if the graphs have different number of nodes.

This new distance is called the FGW distance as it acts as a generalization of the Wasserstein (Villani, 2008) and Gromov-Wasserstein (Memoli, 2011; Solomon et al., 2016) distances as stated in the following theorem:

Theorem 3.1. *Interpolation properties.*

As α tends to zero, the FGW distance recovers the Wasserstein distance between the features $W_q(\mu_A, \nu_B)^q$

$$\lim_{\alpha \to 0} FGW_{q,\alpha}(\mu, \nu) = W_q(\mu_A, \nu_B)^q = \min_{\pi \in \Pi(h,g)} \langle \pi, M_{AB}^q \rangle$$

and as α tends to one, we recover the Gromov-Wasserstein distance $GW_q(\mu_X, \nu_Y)^q$ between the structures:

$$\lim_{\alpha \to 1} FGW_{q,\alpha}(\mu, \nu) = GW_q(\mu_X, \nu_Y)^q = \min_{\pi \in \Pi(h,g)} \langle L(C_1, C_2)^q \otimes \pi, \pi \rangle$$

Proof of this theorem can be found in the supplementary material.

Similarly to the Wasserstein and Gromov-Wasserstein distances, FGW enjoys metric properties over the space of structured data as stated in the following theorem:

Theorem 3.2. *FGW defines a metric for $q = 1$ and a semi-metric for $q > 1$.*

If $q = 1$, and if C_1, C_2 are distance matrices then FGW defines a metric over the space of structured data quotiented by the measure preserving isometries that are also feature preserving. More precisely, FGW satisfies the triangle inequality and is nul iff $n = m$ and there exists a bijection $\sigma : \{1, .., n\} \to \{1, .., n\}$ such that :

$$\forall i \in \{1, .., n\},\ h_i = g_{\sigma(i)} \tag{2}$$

$$\forall i \in \{1, .., n\},\ a_i = b_{\sigma(i)} \tag{3}$$

$$\forall i, k \in \{1, .., n\}^2,\ C_1(i, k) = C_2(\sigma(i), \sigma(k)) \tag{4}$$

If $q > 1$, the triangle inequality is relaxed by a factor 2^{q-1} such that FGW defines a semi-metric.

All proofs can be found in the supplementary material. The resulting application σ preserves the weight of each node (eq. (2)), the features (eq. (3)) and the and the pairwise structure relation between the nodes (eq. (4)). For example, comparing two graphs with uniform weights for the vertices and with shortest path structure matrices, the FGW distance vanishes *iff* the graphs have the same number of vertices and *iff* there exists a one-to-one mapping between the vertices of the graphs which respects both the shortest paths and the features. More informally, it means that graphs have vertices with the same labels connected by the same edges.

The metric FGW is fully unsupervised and can be used in a wide set of applications such as k-nearest-neighbors, distance-substitution kernels, pseudo-Euclidean embeddings, or representative-set methods. Arguably, such a distance also allows for a fine interpretation of the similarity (through the optimal mapping π), contrary to end-to-end learning machines such as neural networks.

3.2. Fused Gromov-Wasserstein barycenter

OT barycenters have many desirable properties and applications (Agueh & Carlier, 2011; Peyré et al., 2016), yet no formulation can leverage both structural and feature information in the barycenter computation. In this section, we consider the FGW distance to define a barycenter of a set of structured data as a Fréchet mean.

We look for the structured data μ that minimizes the sum of (weighted) FGW distances within a given set of structured data $(\mu_k)_k$ associated with structure matrices $(C_k)_k$, features $(B_k)_k$ and base histograms $(h_k)_k$. For simplicity, we assume that the histogram h associated to the barycenter is known and fixed; in other words, we set the number of vertices N and the weight associated to each of them.

In this context, for a fixed $N \in \mathbb{N}$ and $(\lambda_k)_k$ such that $\sum_k \lambda_k = 1$, we aim to find the set of features $A = (a_i)_i$ and the structure matrix C of the barycenter that minimize the following equation:

$$\min_{\mu} \sum_k \lambda_k FGW_{q,\alpha}(\mu, \mu_k) \tag{5}$$
$$= \min_{C \in \mathbb{R}^{N \times N},\ A \in \mathbb{R}^{N \times n}, (\pi_k)_k} \sum_k \lambda_k E_q(M_{AB_k}, C, C_k, \pi_k)$$

Note that this problem is jointly convex *w.r.t.* C and A but not *w.r.t.* π_k. We discuss the proposed algorithm to solve this problem in the next section. Interestingly enough, one can derive several variants of this problem, where the features or the structure matrices of the barycenter can be fixed. Solving the related simpler optimization problem extends straightforwardly. We give examples of such barycenters both in the experimental section where we solve a graph based k-means problem.

3.3. Optimization and algorithmic solution

In this section we discuss the numerical optimization problem for computing the FGW distance between discrete distributions.

Solving the Quadratic Optimization problem. Equation 1 is clearly a quadratic problem *w.r.t.* π. Note that despite the apparent $\mathcal{O}(m^2n^2)$ complexity of computing the tensor product, one can simplify the sum to complexity $\mathcal{O}(mn^2 + m^2n)$ (Peyré et al., 2016) when considering $q = 2$. In this case, the FGW computation problem can be re-written as finding π^* such that:

$$\pi^* = \arg\min_{\pi \in \Pi(h,g)} \quad \text{vec}(\pi)^T Q(\alpha) \text{vec}(\pi) + \text{vec}(D(\alpha))^T \text{vec}(\pi) \tag{6}$$

where $Q = -2\alpha C_2 \otimes_K C_1$ and $D(\alpha) = (1 - \alpha) M_{AB}$. $\otimes_K$ denotes the Kronecker product of two matrices, vec the column-stacking operator. With such form, the resulting optimal map can be seen as a quadratic regularized map

Algorithm 1 Conditional Gradient (CG) for FGW

$\pi^{(0)} \leftarrow \mu_X \mu_Y^\top$
for $i = 1, \dots,$ **do**
  $G \leftarrow$ Gradient from Eq. (7) *w.r.t.* $\pi^{(i-1)}$
  $\tilde{\pi}^{(i)} \leftarrow$ Solve OT with ground loss G
  $\tau^{(i)} \leftarrow$ Line-search for loss (1) with $\tau \in (0, 1)$ using Alg. 2
  $\pi^{(i)} \leftarrow (1 - \tau^{(i)})\pi^{(i-1)} + \tau^{(i)}\tilde{\pi}^{(i)}$
end for

Algorithm 2 Line-search for CG ($q = 2$)

c_{C_1,C_2} from Eq. (6) in (Peyré et al., 2016)
$a = -2\alpha\langle C_1\tilde{\pi}^{(i)}C_2, \tilde{\pi}^{(i)}\rangle$
$b = \langle(1-\alpha)M_{AB} + \alpha c_{C_1,C_2}, \tilde{\pi}^{(i)}\rangle$
$\quad -2\alpha\big(\langle C_1\tilde{\pi}^{(i)}C_2, \pi^{(i-1)}\rangle + \langle C_1\pi^{(i-1)}C_2, \tilde{\pi}^{(i)}\rangle\big)$
$c = E_2(M_{AB}, C_1, C_2, \pi^{(i-1)})$
if $a > 0$ **then**
  $\tau^{(i)} \leftarrow \min(1, \max(0, \frac{-b}{2a}))$
else
  $\tau^{(i)} \leftarrow 1$ if $a + b < 0$ else $\tau^{(i)} \leftarrow 0$
end if

from initial Wasserstein (Ferradans et al., 2014; Flamary et al., 2014). However, unlike these approaches, we have a quadratic but provably non convex term. The gradient G that arises from Eq. (1) can be expressed with the following partial derivative *w.r.t.* π:

$$G = (1 - \alpha)M_{AB}^q + 2\alpha L(C_1, C_2)^q \otimes \pi \qquad (7)$$

that can be computed with $\mathcal{O}(mn^2 + m^2n)$ operations when $q = 2$.

Solving a large scale QP with a classical solver can be computationally expensive. In (Ferradans et al., 2014), authors propose a solver for a graph regularized optimal transport problem whose resulting optimization problem is also a QP. We can then directly use their conditional gradient defined in Alg. 1 to solve our optimization problem. It only needs at each iteration to compute the gradient in Eq. (7) and to solve a classical OT problem for instance with a network flow algorithm. The line-search part is a constrained minimization of a second degree polynomial function which is adapted to the non convex loss in Alg. 2. While the problem is non convex, conditional gradient is known to converge to a local stationary point (Lacoste-Julien, 2016).

Solving the barycenter problem with Block Coordinate Descent (BCD). We propose to minimize eq. (5) using a BCD algorithm, *i.e.* iteratively minimizing with respect to the couplings π_k, to the metric C and the feature vector A. The minimization of this problem *w.r.t.* $(\pi_k)_k$ is equivalent to compute S independent Fused Gromov-Wasserstein distances as discussed above. We suppose that the feature space is $\Omega_f = (\mathbb{R}^d, \ell_2^2)$ and we consider $q = 2$. Minimization *w.r.t.* C in this case has a closed form (see Prop. 4 in (Peyré et al., 2016)) :

$$C \leftarrow \frac{1}{hh^T}\sum_k \lambda_k \pi_k^T C_k \pi_k$$

where h is the histogram of the barycenter as discussed in section 3.2. Minimization *w.r.t.* A can be computed with (eq. (8) in (Cuturi & Doucet, 2014)):

$$A \leftarrow \sum_k \lambda_k B_k \pi_k^T \mathrm{diag}(\frac{1}{h})$$

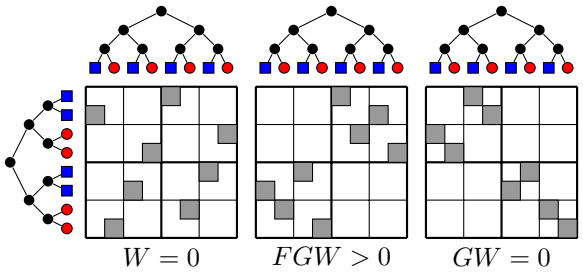

Figure 3. Example of FGW, GW and W on synthetic trees. Dark grey color represents a non null $\pi_{i,j}$ value between two nodes i and j. (Left) the W distance between the features with $\alpha = 0$, (Middle) FGW (Right) the GW between the structures $\alpha = 1$.

4. Experimental results

We now illustrate the behaviour of FGW on synthetic and real datasets. The algorithms presented in the previous section have been implemented using the Python Optimal Transport toolbox (Flamary & Courty, 2017) and will be released upon publication.

4.1. Illustration of FGW on trees

We construct two trees as illustrated in Figure 3, where the 1D node features are shown with colors (in red, features belong to $[0, 1]$ and in blue in $[9, 10]$). The structure similarity matrices C_1 and C_2 are the shortest path between nodes. Figure 3 illustrates the behavior of the FGW distance when the trade-off parameter α changes. The left part recovers the Wasserstein distance ($\alpha = 0$): red nodes are coupled to red ones and the blue nodes to the blue ones. For a alpha close to 1 (right), we recover the Gromov-Wasserstein distance: all couples of points are coupled to another couple of points, without taking into account the features. Both approaches fail in discriminating the two trees. Finally, for an intermediate α in FGW (center), the bottom and first level structure is preserved as well as the feature matching (red on red and blue on blue), resulting on a positive distance.

4.2. Graph-structured data classification

Datasets We consider 12 widely used benchmark datasets divided into 3 groups. BZR, COX2 (Sutherland et al., 2003), PROTEINS, ENZYMES (Borgwardt & Kriegel, 2005), CUNEIFORM (Kriege et al., 2018) and SYNTHETIC (Feragen et al., 2013) are vector attributed graphs. MUTAG (Debnath et al., 1991), PTC-MR (Kriege et al., 2016) and NCI1 (Wale et al., 2008) contain graphs with discrete attributes derived from small molecules. IMDB-B, IMDB-M (Yanardag & Vishwanathan, 2015) contain unlabeled graphs derived from social networks. All datas are available in (Kersting et al., 2016).

Experimental setup Regarding the feature distance matrix M_{AB} between node features, when dealing with real valued vector attributed graphs, we consider the ℓ_2 distance between the labels of the vertices. In the case of graphs with discrete attributes, we consider two settings: in the first one, we keep the original labels (denoted as RAW); we also consider a Weisfeiler-Lehman labeling (denoted as WL) by concatenating the labels of the neighbors. A vector of size H is created by repeating this procedure H times (Vishwanathan et al., 2010; Kriege et al., 2016). In both cases, we compute the feature distance matrix by using $d(a_i, b_j) = \sum_{k=0}^{H} \delta(\tau(a_i^k), \tau(b_j^k))$ where $\delta(x, y) = 1$ if $x \neq y$ else $\delta(x, y) = 0$ and $\tau(a_i^k)$ denotes the concatenated label at iteration k (for $k = 0$ original labels are used). Regarding the structure distances C, they are computed by considering a shortest path distance between the vertices.

For the classification task, we run a SVM using the indefinite kernel matrix $e^{-\gamma FGW}$ which is seen as a noisy observation of the true positive semidefinite kernel (Luss & d'Aspremont, 2007). We compare classification accuracies with the following state-of-the-art graph kernel methods: (SPK) denotes the shortest path kernel (Borgwardt & Kriegel, 2005), (RWK) the random walk kernel (Gärtner et al., 2003), (WLK) the Weisfeler Lehman kernel (Vishwanathan et al., 2010), (GK) the graphlet count kernel (Shervashidze et al., 2009). For real valued vector attributes, we consider the HOPPER kernel (HOPPERK) (Feragen et al., 2013) and the propagation kernel (PROPAK) (Neumann et al., 2016). We build upon the GraKel library (Siglidis et al., 2018) to construct the kernels and C-SVM to perform the classification. We also compare FGW with the PATCHY-SAN framework for CNN on graphs (Niepert et al., 2016)(PSCN) building on our own implementation of the method.

To provide compare between the methods, most papers about graph classification usually perform a nested cross validation (using 9 folds for training, 1 for testing, and reporting the average accuracy of this experiment repeated 10 times) and report accuracies of the other methods taken from the original papers. However, these comparisons are not fair because of the high variance on most datasets *w.r.t.* the folds chosen for training and testing. This is why, in our experiments, the nested cross validation is performed on the same folds for training and testing for *all* methods. In the result tables 1,2 and 3 we add a (*) when the best score does not yield to a significative improvement (based on a Wilcoxon signed rank test on the test scores) compared to the second best one. Note that, because of their small sizes, we repeat the experiments 50 times for MUTAG and PTC-MR datasets. For all methods using SVM, we cross validate the parameter $C \in \{10^{-7}, 10^{-6}, ..., 10^{7}\}$. The range of the WL parameter H is $\{0, 1..., 10\}$, and we also compute this kernel with H fixed at $2, 4$. The decay factor λ for RWK $\{10^{-6}, 10^{-5}..., 10^{-2}\}$, for the GK kernel we set the graphlet size $\kappa = 3$ and cross validate the precision level ϵ and the confidence δ as in the original paper (Shervashidze et al., 2009). The t_{max} parameter for PROPAK is chosen within $\{1, 3, 5, 8, 10, 15, 20\}$. For PSCN, we choose the normalized betweenness centrality as labeling procedure and cross validate the batch size in $\{10, 15, ..., 35\}$ and number of epochs in $\{10, 20, ..., 100\}$. Finally for FGW, γ is cross validated within $\{2^{-10}, 2^{-9}, ..., 2^{10}\}$ and α is cross validated *via* a logspace search in $[0, 0.5]$ and symmetrically $[0.5, 1]$ (15 values are drawn).

Results and discussion

Vector attributed graphs. The average accuracies reported in Table 1 show that FGW is a clear state-of-the-art method and performs best on 4 out of 6 datasets with performances in the error bars of the best methods on the other two datasets. Results for CUNEIFORM are significantly below those from the original paper (Kriege et al., 2018) which can be explained by the fact that the method in this paper uses a graph convolutional approach specially designed for this dataset and that experiment settings are different. In comparison, the other competitive methods are less consistent as they exhibit some good performances on some datasets only.

Discrete labeled graphs. We first note in Table 2 that FGW using WL attributes outperforms all competitive methods, including FGW with raw features. Indeed, the WL attributes allow encoding more finely the neighborood of the vertices by stacking their attributes, whereas FGW with raw features only consider the shortest path distance between vertices, not their sequence of labels. This result calls for using meaningful feature and/or structure matrices in the FGW definition, that can be dataset-dependant, in order to enhance the performances. We also note that FGW with WL attributes outperforms the WL kernel method, highlighting the benefit of an optimal transport-based distance over a kernel-based similarity. Surprisingly results of PSCN are significantly lower than those from the original paper. We believe that it comes from the difference between the

Table 1. Average classification accuracy on the graph datasets with vector attributes.

VECTOR ATTRIBUTES	BZR	COX2	CUNEIFORM	ENZYMES	PROTEIN	SYNTHETIC
FGW SP	**85.12±4.15***	77.23±4.86	**76.67±7.04**	71.00±6.76	**74.55±2.74**	**100.00±0.00**
HOPPERK	84.15±5.26	**79.57±3.46**	32.59±8.73	45.33±4.00	71.96±3.22	90.67±4.67
PROPAK	79.51±5.02	77.66±3.95	12.59±6.67	**71.67±5.63***	61.34±4.38	64.67±6.70
PSCN K=10	80.00±4.47	71.70±3.57	25.19±7.73	26.67±4.77	67.95±11.28	**100.00±0.00**
PSCN K=5	82.20±4.23	71.91±3.40	24.81±7.23	27.33±4.16	71.79±3.39	**100.00±0.00**

Table 2. Average classification accuracy on the graph datasets with discrete attributes.

DISCRETE ATTR.	MUTAG	NCI1	PTC-MR
FGW RAW SP	83.26±10.30	72.82±1.46	55.71±6.74
FGW WL H=2 SP	86.42±7.81	85.82±1.16	63.20±7.68
FGW WL H=4 SP	**88.42±5.67**	**86.42±1.63**	**65.31±7.90**
GK K=3	82.42±8.40	60.78±2.48	56.46±8.03
RWK	79.47±8.17	58.63±2.44	55.09±7.34
SPK	82.95±8.19	74.26±1.53	60.05±7.39
WLK	86.21±8.48	85.77±1.07	62.86±7.23
WLK H=2	86.21±8.15	81.85±2.28	61.60±8.14
WLK H=4	83.68±9.13	85.13±1.61	62.17±7.80
PSCN K=10	83.47±10.26	70.65±2.58	58.34±7.71
PSCN K=5	83.05±10.80	69.85±1.79	55.37±8.28

Table 3. Average classification accuracy on the graph datasets with no attributes.

WITHOUT ATTRIBUTE	IMDB-B	IMDB-M
GW SP	**63.80±3.49**	**48.00±3.22**
GK K=3	56.00±3.61	41.13±4.68
SPK	55.80±2.93	38.93±5.12

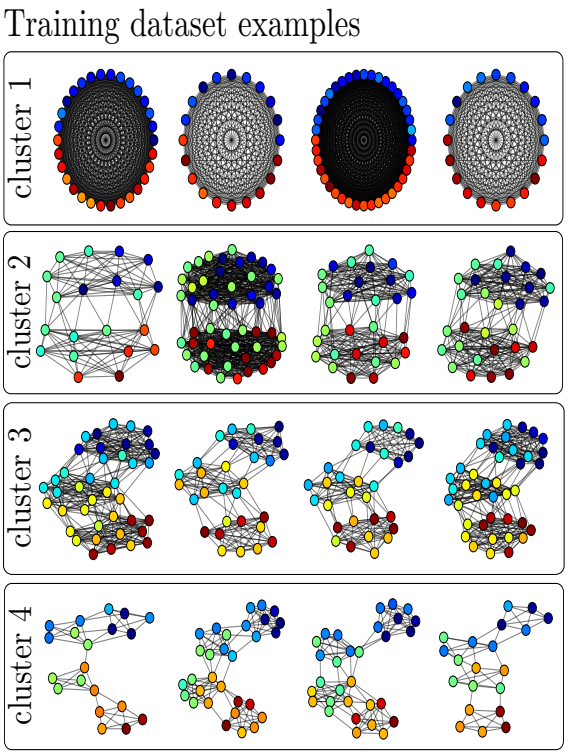

Figure 4. Examples from the clustering dataset, color indicates the labels.

folds assignment for training and testing, which suggests that PSCN is difficult to tune.

Non-attributed graphs. The particular case of the GW distance for graph classification is also illustrated on social datasets, that contain no labels on the vertices. Accuracies reported in Table 3 show that it greatly outperforms SPK and GK graph kernel methods. This is, to the best of our knowledge, the first application of GW for social graph classification.

Comparison between FGW, W and GW During the validation step, the optimal value of α was consistently selected inside the $]0, 1[$ interval, excluding 0 and 1, suggesting that both structure and feature pieces of information are necessary (details are given in the supplementary material).

4.3. Unsupervised learning: graphs clustering

In the last experiment, we evaluate the ability of FGW to perform a clustering of multiple graphs and to retrieve meaningful barycenters of such clusters. To do so, we generate a dataset of 4 groups of community graphs. Each graph follows a simple Stochastic Block Model (Wang & Wong, 1987; Nowicki & Snijders, 2001) and the groups are defined *w.r.t.* the number of communities inside each graph and the distribution of their labels. The dataset is composed of 40 graphs (10 graphs per group) and the number of nodes of each graph is drawn randomly from $\{20, 30, ..., 50\}$ as illustrated in Fig. 4. We perform a k-means clustering using the FGW barycenter defined in eq. (5) as the centroid of

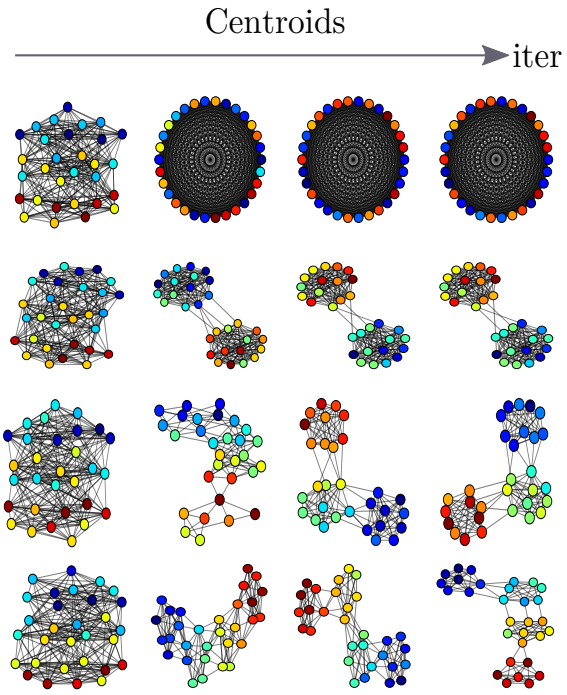

Figure 5. Evolution of the centroids of each cluster in the k-means clustering, from (Left) the random initialization (Right) until convergence to the final centroid.

the groups and the FGW distance for the cluster assignment. We fix the number of nodes of each centroid to 30. We perform a thresholding on the pairwise similarity matrix C of the centroid at the end in order to obtain an adjacency matrix for visualization purposes. The threshold value is empirically chosen so as to minimize the distance induced by the frobenius norm between the original matrix C and the shortest path matrix obtained from the adjency matrix. The evolution of the barycenters along the iterations is reported in Figure 5. We can see that these centroids recover community structures and feature distributions that are representative of their cluster content. On this example, note that the clustering recovers perfectly the known groups in the dataset. To the best of our knowledge, there exists no other method able to perform a clustering of graphs and to retrieve the average graph in each cluster without having to solve a pre-image problem.

5. Discussion and conclusion

Countless problems in machine learning involve structured data, usually stressed in light of the graph formalism. We consider here labeled graphs enriched by an histogram, which naturally leads to represent structured data as probability measures in the joint space of their features and structures. Widely known for their ability to meaningfully compare probability measures, transportation distances are generalized in this paper so as to be suited in the context of structured data, motivating the so-called Fused Gromov-Wasserstein distance. We theoretically prove that it defines indeed a distance on structured data, and consequently on graphs of arbitrary sizes. FGW provides a natural framework for analysis of labeled graphs as we demonstrate on classification, where it reaches and surpasses most of the time the state-of-the-art performances, and in graph-based k-means where we develop a novel approach to represent the clusters centroids using a barycentric formulation of FGW. We believe that this metric can have a significant impact on challenging graph signal analysis problems.

While we considered a unique measure of distance between nodes in the graph structure (shortest path), other choices could be made with respect to the problem at hand, or eventually learned in an end-to-end manner. The same applies to the distance between features. We also envision a potential use of this distance in deep learning applications where a distance between graph is needed (such as graph auto-encoders). Another line of work will also try to lower the computational complexity of the underlying optimization problem to ensure better scalability to very large graphs.

Acknowledgements

This work benefited from the support from OATMIL ANR-17-CE23-0012 project of the French National Research Agency (ANR). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

References

Agueh, M. and Carlier, G. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2): 904–924, 2011.

Alvarez-Melis, D., Jaakkola, T. S., and Jegelka, S. Structured Optimal Transport. In *AISTATS*, 2018.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 214–223, Sydney, Australia, 06–11 Aug 2017.

Bach, F. and Harchaoui, Z. Image classification with segmentation graph kernels. In *CVPR*, volume 00, pp. 1–8, 06 2007.

Bakir, G. H., Hofmann, T., Schölkopf, B., Smola, A. J., Taskar, B., and Vishwanathan, S. V. N. *Predicting Structured Data (Neural Information Processing)*. The MIT Press, 2007. ISBN 0262026171.

Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G., Vaswani, A., Allen, K., Nash, C., Langston, V., Dyer, C., Heess, N., Wierstra, D., Kohli, P., Botvinick, M., Vinyals, O., Li, Y., and Pascanu, R. Relational inductive biases, deep learning, and graph networks. *ArXiv e-prints*, June 2018.

Borgwardt, K. M. and Kriegel, H.-P. Shortest-path kernels on graphs. In *ICDM*, ICDM '05, pp. 74–81. IEEE Computer Society, 2005.

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4): 18–42, 2017.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEETPAMI*, 39(9):1853–1865, 2017.

Cuturi, M. and Blondel, M. Soft-DTW: a differentiable loss function for time-series. In *Proceedings of the ICML*, volume 70, pp. 894–903, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

Cuturi, M. and Doucet, A. Fast computation of wasserstein barycenters. In Xing, E. P. and Jebara, T. (eds.), *ICML*, volume 32 of *Proceedings of Machine Learning Research*, pp. 685–693, Bejing, China, 22–24 Jun 2014.

Day, W. H. Optimal algorithms for comparing trees with labeled leaves. *Journal of classification*, 2(1):7–28, 1985.

Debnath, A. K., Lopez de Compadre, R. L., Debnath, G., Shusterman, A. J., and Hansch, C. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of Medicinal Chemistry*, 34 (2):786–797, 1991.

Defferrard, M., Bresson, X., and Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In *NIPS*, pp. 3844–3852. 2016.

Feragen, A., Kasenburg, N., Petersen, J., de Bruijne, M., and Borgwardt, K. Scalable kernels for graphs with continuous attributes. In *Advances in Neural Information Processing Systems 26*, pp. 216–224. 2013.

Ferradans, S., Papadakis, N., Peyré, G., and Aujol, J.-F. Regularized discrete optimal transport. *SIAM Journal on Imaging Sciences*, 7(3):1853–1882, 2014.

Flamary, R. and Courty, N. Pot python optimal transport library. 2017.

Flamary, R., Courty, N., Tuia, D., and Rakotomamonjy, A. Optimal transport with laplacian regularization: Applications to domain adaptation and shape matching. 2014.

Gärtner, T., Flach, P., and Wrobel, S. On graph kernels: Hardness results and efficient alternatives. In *COLT*, pp. 129–143, 2003.

Huang, G., Guo, C., Kusner, M., Sun, Y., Sha, F., and Weinberger, K. Supervised word mover's distance. In *NIPS*, pp. 4862–4870, 2016.

Kersting, K., Kriege, N. M., Morris, C., Mutzel, P., and Neumann, M. Benchmark data sets for graph kernels, 2016.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016.

Kriege, N., Fey, M., Fisseler, D., Mutzel, P., and Weichert, F. Recognizing cuneiform signs using graph based methods. In *International Workshop on Cost-Sensitive Learning (COST)*, 2018.

Kriege, N. M., Giscard, P., and Wilson, R. C. On valid optimal assignment kernels and applications to graph classification. *CoRR*, abs/1606.01141, 2016.

Ktena, S. I., Parisot, S., Ferrante, E., Rajchl, M., Lee, M., Glocker, B., and Rueckert, D. Distance metric learning using graph convolutional networks: Application to functional brain networks. In *MICCAI*, pp. 469–477, 2017.

Lacoste-Julien, S. Convergence rate of frank-wolfe for non-convex objectives. *arXiv preprint arXiv:1607.00345*, 2016.

Luss, R. and d'Aspremont, A. Support vector machine classification with indefinite kernels. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pp. 953–960, 2007. ISBN 978-1-60560-352-0.

Memoli, F. Gromov wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, pp. 1–71, 2011.

Neumann, M., Garnett, R., Bauckhage, C., and Kersting, K. Propagation kernels: efficient graph kernels from propagated information. *Machine Learning*, 102(2):209–245, Feb 2016.

Niepert, M., Ahmed, M., and Kutzkov, K. Learning convolutional neural networks for graphs. In *ICML*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2014–2023, New York, New York, USA, 2016. PMLR.

Nikolentzos, G., Meladianos, P., and Vazirgiannis, M. Matching node embeddings for graph similarity. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pp. 2429–2435, 2017.

Nowicki, K. and Snijders, T. A. B. Estimation and prediction for stochastic blockstructures. *Journal of the American statistical association*, 96(455):1077–1087, 2001.

Peyré, G. and Cuturi, M. Computational optimal transport. *arXiv preprint arXiv:1803.00567*, 2018.

Peyré, G., Cuturi, M., and Solomon, J. Gromov-wasserstein averaging of kernel and distance matrices. In *ICML*, pp. 2664–2672, 2016.

Shervashidze, N., Vishwanathan, S. V. N., Petri, T. H., Mehlhorn, K., and et al. Efficient graphlet kernels for large graph comparison, 2009.

Shervashidze, N., Schweitzer, P., van Leeuwen, E. J., Mehlhorn, K., and Borgwardt, K. M. Weisfeiler-lehman graph kernels. *J. Mach. Learn. Res.*, 12:2539–2561, November 2011.

Siglidis, G., Nikolentzos, G., Limnios, S., Giatsidis, C., Skianis, K., and Vazirgianis, M. GraKeL: A Graph Kernel Library in Python. *arXiv e-prints*, June 2018.

Solomon, J., Peyré, G., Kim, V. G., and Sra, S. Entropic metric alignment for correspondence problems. *ACM Trans. Graph.*, 35(4):72:1–72:13, July 2016. ISSN 0730-0301. doi: 10.1145/2897824.2925903. URL http://doi.acm.org/10.1145/2897824.2925903.

Sutherland, J. J., O'brien, L. A., and Weaver, D. F. Spline-fitting with a genetic algorithm: A method for developing classification structure- activity relationships. *Journal of chemical information and computer sciences*, 43(6): 1906–1915, 2003.

Thorpe, M., Park, S., Kolouri, S., Rohde, G. K., and Slepčev, D. A transportation l^p distance for signal analysis. *Journal of Mathematical Imaging and Vision*, 59(2):187–210, Oct 2017.

Verma, S. and Zhang, Z.-L. Hunt for the unique, stable, sparse and fast feature learning on graphs. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *NIPS*, pp. 88–98. Curran Associates, Inc., 2017.

Villani, C. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer, 2009 edition, 2008.

Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., and Borgwardt, K. M. Graph kernels. *J. Mach. Learn. Res.*, 11:1201–1242, August 2010.

Wale, N., Watson, I. A., and Karypis, G. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 14 (3):347–375, Mar 2008.

Wang, Y. J. and Wong, G. Y. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

Yanardag, P. and Vishwanathan, S. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1365–1374, 2015.