Learning to Predict Graphs with Fused Gromov-Wasserstein Barycenters

Luc Brogat-Motte [1] **Rémi Flamary** [2] **Céline Brouard** [3] **Juho Rousu** [4] **Florence d'Alché-Buc** [1]

Abstract

This paper introduces a novel and generic framework to solve the flagship task of supervised labeled graph prediction by leveraging Optimal Transport tools. We formulate the problem as regression with the Fused Gromov-Wasserstein (FGW) loss and propose a predictive model relying on a FGW barycenter whose weights depend on inputs. First we introduce a non-parametric estimator based on kernel ridge regression for which theoretical results such as consistency and excess risk bound are proved. Next we propose an interpretable parametric model where the barycenter weights are modeled with a neural network and the graphs on which the FGW barycenter is calculated are additionally learned. Numerical experiments show the strength of the method and its ability to interpolate in the labeled graph space on simulated data and on a difficult metabolic identification problem where it can reach very good performance with very little engineering.

1. Introduction

Graphs allow to represent entities and their interactions. They are ubiquitous in real-world: social networks, molecular structures, biological protein-protein networks, recommender systems, are naturally represented as graphs. Nevertheless, graphs structured data can be challenging to process. An important effort has been made to design well-tailored machine learning methods for graphs. For example, many kernels for graphs have been proposed allowing to perform graph classification, graph clustering, graph regression (Kriege et al., 2020). Many deep learning architecture have also been developped (Zhang et al., 2022), including Graph Convolutional Networks (GCNs) that are powerful models for processing graphs.

Most of existing works in machine learning consider graphs as inputs, but predicting a graph as output given an input from an arbitrary input space has received much less attention. In this work, we target the difficult problem of supervised learning of graph-valued functions. In contrasts with node classification (Bhagat et al., 2011), or link prediction (Lü & Zhou, 2011), entire graphs are predicted. Supervised Graph Prediction (SGP) can be considered as an emblematic instance of Structured Prediction (SP) with the difficulty that the output space is of finite but huge cardinality and contains structures of different sizes. In principle, any of the three main approaches to SP, energy-based models, surrogate approaches and end-to-end learning, are eligible. In energy-based models (Tsochantaridis et al., 2005; Chen et al., 2015; Belanger & McCallum, 2016), predictions are obtained by maximizing a score function for input-output pairs over the output space. In surrogate approaches (Cortes et al., 2005; Geurts et al., 2006; Brouard et al., 2016b; Ciliberto et al., 2016), a feature map is used to embed the structured outputs. After minimizing a surrogate loss a decoding procedure is used to map back the surrogate solution. End-to-end learning methods attempt to solve structured prediction by directly learning to generate a structured object (Belanger et al., 2017; Silver et al., 2017) and leverage differentiable and relaxed definition of energy-based methods (see for instance Pillutla et al. (2018); Mensch & Blondel (2018)).

Nevertheless, to our knowledge, among surrogate methods, only Input Output Kernel Regression (IOKR) (Brouard et al., 2016b) that leverages kernel trick in the output space has been successfully applied to SGP while on the side of end-to-end learning, several generative models allow to build and predict graphs but in general in an unsupervised setting. Gómez-Bombarelli et al. (2018) try to obtain a continuous representation of molecules using a variational autoencoding (VAE) of text representations of molecules (SMILES). Kusner et al. (2017) incorporates in the VAE architecture knowledge about the structure of SMILES thanks to its available grammar. Olivecrona et al. (2017); Liu et al. (2017); Li et al. (2018a); You et al. (2018); Shi et al. (2020) propose models that generate graphs using a sequential process generating one node/edge at a time, and train it by maximizing the likelihood.

[1]LTCI, Télécom Paris, Institut Polytechnique de Paris, France [2]Ecole Polytechnique, Institut Polytechnique de Paris, CMAP, UMR 7641, Palaiseau, France [3]Université de Toulouse, INRAE, UR MIAT, France [4]Department of Computer Science, Aalto University, Finland. Correspondence to: Luc Brogat-Motte <luc.motte@telecom-paris.fr>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

In supervised graph prediction, the crucial issue is to learn or leverage appropriate representations of graphs, a problem tightly linked with the choice of a loss function. Typical graph representations usually rely on graph kernels leveraging fingerprint representations, i.e. a bag of motifs approach (Ralaivola et al., 2005), or more involved kernels such the Weisfeiler-Lehman kernel (Shervashidze et al., 2011). In this work, we propose to exploit another kind of graph representation, opening the door to the use of an Optimal Transport loss, and derive an end-to-end learning approach that constrasts to energy-based learning and surrogate methods.

Successful applications of optimal transport (OT) in machine learning are becoming increasingly numerous thanks to the advent of numerical optimal transport (Cuturi, 2013; Altschuler et al., 2017; Peyré et al., 2019). Examples include domain adaptation (Courty et al., 2016), unsupervised learning (Arjovsky et al., 2017), multi-label classification (Frogner et al., 2015), natural language processing (Kusner et al., 2015), fair classification (Gordaliza et al., 2019), supervised representation learning (Flamary et al., 2018). Optimal transport provide meaningful distances between probability distributions, by leveraging the geometry of the underlying metric spaces.
Supervised learning with optimal transport losses has been considered in Frogner et al. (2015); Bonneel et al. (2016); Luise et al. (2018); Mensch et al. (2019) for predicting histograms. But traditional OT loss can be applied only between distributions lying in the same space, preventing their use on structured data such as graphs. Mémoli (2011) proposed the Gromov-Wasserstein distance that can measure similarity between metric measure space and has been used as a distance between graphs in several applications such as computing graph barycenters (Peyré et al., 2016) or for performing graph node embedding (Xu et al., 2019b) and graph partitioning (Xu et al., 2019a). This distance has been extended to the Fused Gromov-Wasserstein distance (FGW) in Vayer et al. (2019; 2020) with applications to attributed graphs classification, barycenter estimation and more recently dictionary learning (Vincent-Cuaz et al., 2021). Those novel divergences that can be used on graphs are a natural fit, first as a loss term in graph prediction but also as a way to model the space of graphs for instance using FGW barycenters.

Contributions. In this paper we present the following novel contributions. First we propose a novel and and general framework in Sec. 3 for graph prediction building on FGW as a loss and FGW barycenter as a way to interpolate in the target space. The framework is studied theoretically in Sec. 4 in the non-parametric case for which we provide consistency and excess risk bounds. Then a parametric version of the model building on deep neural network and learning of the template graphs is proposed in Sec. 5 with a simple stochastic gradient algorithm. Finally we provide some numerical experiments in Sec. 6 on synthetic and real life metabolite prediction datasets.

2. Background on OT for graphs

We begin by introducing how to represent graphs and define distances between graph by leveraging the Fused Gromov-Wasserstein distance.

Notations. 1_n is the all-ones vector with size n. $_x$ denotes the Dirac measure in x for x in a measurable space. Identity matrix in $\mathrm{R}^{N\ N}$ is noted I_N. $L(A)$ the set of bounded linear operator from A to A. $M(A;B)$ the set of measurable functions from A to B.

Graph represented as metric measure spaces. Denote $n_{max}\ 2\ \mathrm{N}$ the maximal number of nodes (vertices) in the graphs we consider in this paper. We define $F\quad \mathrm{R}^d$ a finite feature space of size $jFj < 1$. A labeled graph y of $n\quad n_{max}$ nodes is represented by a triplet $y = (C;F;h)$ where $C = C^T\ 2\ f0;1g^{n\ \ n}$ is the adjacency matrix, and $F = (F_i)_{i=1}^n$ is a n-tuple composed of feature vectors $F_i\ 2\ F\quad \mathrm{R}^d$ labeling each node indexed by i. The space of labeled graphs Y is thus defined as $Y = f(C;F;h)\,j\,n\quad n_{max};C\ 2\ f0;1g^{n\ \ n};C^T = C;F = (F_i)_{i=1}^n\ 2\ F^n;h = \frac{1}{n}1_n g$. Observe that we equipped all graphs with a uniform discrete probability distributions over the nodes $= \sum_{i=1}^n h_i\ _{u_i}$ where $u_i = (v_i;F_i)$ represents the structure v_i (encoded only through $C(i;j);\ 8j$) and the feature information F_j attached to a vertex i (Vayer et al., 2019). These weights indicate the relative importance of the vertices in the graph. In absence of this information, we simply fix uniform weights $h_i = \frac{1}{n}$ for a graph of size n. Now, let us introduce the space of continuous relaxed graphs with **fixed size** n: $Z_n = f(C;F;h)\,j\,C\ 2\ [0;1]^{n\ \ n};C^T = C;F\ 2\ \mathrm{Conv}(F)^n;h = n^{\ 1}1_n g$. $\mathrm{Conv}(F)$ denotes the convex hull of F in $\mathrm{R}^{n\ \ d}$. We call $Z = \bigcup_{(Z_i)_{i=1}^{n_{max}}}$ and want to emphasize that $Y\quad Z$.

Gromov-Wasserstein (GW) distance. The Gromov-Wassertein distance between metric measure space has been introduced by Mémoli (2011) for object matching. The GW distance defines an OT problem to compare these objects, with the key property that it defines a strict metric on the collection of isomorphism classes of metric measure spaces. In this paper, we adopt this angle to address graph representation and graph comparison, opening the door to define a loss for supervised graph prediction. Let $z_1 = (C_1;n_1^{\ 1}1_{n_1})$ and $z_2 = (C_2;n_2^{\ 1}1_{n_2})$ be the representation of two graphs with respectively $n_1\ 2\ \mathrm{N}$ and $n_2\ 2\ \mathrm{N}$ nodes, the Gromov-Wasserstein (GW) distance

between z_1 and z_2, $\mathrm{GW}_2^2(z_1, z_2)$, is defined as follows:

$$\min_{P \in \mathcal{P}(n_1, n_2)} \sum_{i,k=1}^{n_1} \sum_{j,l=1}^{n_2} \left(C_1(i,k) - C_2(j,l)\right)^2 P_{i,j} P_{k,l}, \quad (1)$$

where $\mathcal{P}(n_1, n_2) = \{P \in \mathbb{R}_+^{n_1 \times n_2} \mid P\mathbf{1}_{n_2} = n_1^{-1}\mathbf{1}_{n_1}, P^\top \mathbf{1}_{n_1} = n_2^{-1}\mathbf{1}_{n_2}\}$. GW_2 can be used to compare unlabeled graphs with potentially different numbers of nodes, it is symmetric, positive and satisfies the triangle inequality. Furthermore, it is equal to zero when z_1 and z_2 are isomorphic, namely when there exist a bijection $\sigma : [\![1, n_1]\!] \to [\![1, n_2]\!]$ such that $C_2(\sigma(i), \sigma(j)) = C_1(i,j)$ for all $i, j \in [\![1, n_1]\!]$. GW provides a distance on the unlabeled graph quotiented by the isomorphism, making it a natural metric when comparing graphs.

Fused Gromov-Wasserstein (FGW) distance. The FGW distance has been proposed recently as an extension of GW that can be used to measure the similarity between attributed graphs (Vayer et al., 2020). For a given $\alpha \in [0, 1]$, the FGW distance between two labeled weighted graphs represented as $z_1 = (C_1, F_1, n_1^{-1}\mathbf{1}_{n_1})$ and $z_2 = (C_2, F_2, n_2^{-1}\mathbf{1}_{n_2})$ is defined as follows (Vayer et al., 2020):

$$\mathrm{FGW}_2^2(z_1, z_2) = \min_{P \in \mathcal{P}(n_1, n_2)} \sum_{i,k,j,l} (1-\alpha)\|F_1(i) - F_2(j)\|_{\mathbb{R}^d}^2 P_{i,j} P_{k,l} + \alpha \left(C_1(i,k) - C_2(j,l)\right)^2 P_{i,j} P_{k,l}.$$

The optimal transport plan matches the vertices of the two graphs by minimizing the discrepancy between the labels, while preserving the pairwise similarities between the nodes. Parameter α governs the trade-off between structure and label information. Its choice is typically driven by the application.

3. Graph prediction with Fused Gromov-Wasserstein

Relaxed Supervised Graph Prediction. In this work, we consider labeled graph prediction as a *relaxed* structured output prediction problem. We assume that X is the input space and that the predictions belong to the space Z_n defined in Section 2, for a given value of n, while we observe training data in the finite set Y. We define an asymmetric partially relaxed structured loss function $\Delta : Z_n \times Y \to \mathbb{R}^+$. Given a finite sample $(x_i, y_i)_{i=1}^N$ independently drawn from an unknown distribution ρ on $X \times Y$, we consider the problem of estimating a target function $f^* : X \to Z_n$ with values in the structured objects Z_n that minimizes the expected risk:

$$R^n(f) = \mathbb{E}\left[\Delta(f(X), Y)\right], \quad (2)$$

by an estimate $\hat{f}$ obtained by minimizing the empirical counterpart of the true risk, namely the empirical risk:

$$\hat{R}^n(f) = \sum_{i=1}^N \Delta(f(x_i), y_i), \quad (3)$$

over the hypothesis space $\mathcal{G}^n \subset \mathcal{M}(X, Z_n)$. The goal of this paper is to provide a whole framework to address this family of problems instantiated by $n \in [\![1, n_{max}]\!]$. Note that the complexity of the task depends primarily on n.

FGW as training loss. We propose in this paper to use the FGW distance as the loss. More precisely, we define:

$$\forall (z, y) \in Z_n \times Y, \quad \Delta_{\mathrm{FGW}}(z, y) := \mathrm{FGW}_2^2(z, z_y), \quad (4)$$

where $z_y = (C_y, F_y, n_y^{-1}\mathbf{1}_{n_y}) \in Z_{n_y} \subset Z$ is the representation of $y = (C_y, F_y, n_y^{-1}\mathbf{1}_{n_y}) \in Y$. As FGW is defined for graphs of different sizes, the expression in Eq. (4) is well posed. Accordingly, for all $i = 1, \dots, N$, we denote $z_i \in Z_{n_i}$ the relaxed version of $y_i \in Y$ with number of nodes n_i.

Supervised Graph Prediction with FGW. Having fixed a value for n and following these definitions, the empirical risk minimization problem now writes as follows. Given the training sample $\{(x_i, z_i)_{i=1}^N\}$, we want to find a minimizer over $\mathcal{G}^n \subset \mathcal{M}(X, Z_n)$ of the following problem:

$$\min_{f \in \mathcal{G}^n} \sum_{i=1}^N \mathrm{FGW}_2^2(f(x_i), z_i). \quad (5)$$

Remark 3.1 (Role of the graph sizes for the FGW distance). For the FGW distance, it is worth noting that graphs' sizes act as resolutions, namely levels of prevision in the description of graphs. We denote by $\overset{\mathrm{FGW}}{\subset}$ the subset symbol for the equivalence classes induced by the FGW metric. We approximately have, for $n < n'$, $Z_n \overset{\mathrm{FGW}}{\subset} Z_{n'}$ depending if exact or approximate resampling is possible. For instance, we exactly have, for all $n \in \mathbb{N}^*$, $Z_n \overset{\mathrm{FGW}}{\subset} Z_{2n}$. It means that low-resolution graphs can be represented exactly as high-resolution graphs. Conversely, one can approximate a high-resolution graph with a low-resolution graph. This property is leveraged in the model hereinafter proposed. Note that if one wants to compare two graphs, with equal weights on each node, it is still possible to do padding: add nodes with no neighbours, and with a chosen constant label.

Structured prediction model. To address this structured regression problem, we propose a generic model $f : X \to Z_n$ expressed as a **conditional FGW barycenter** computed

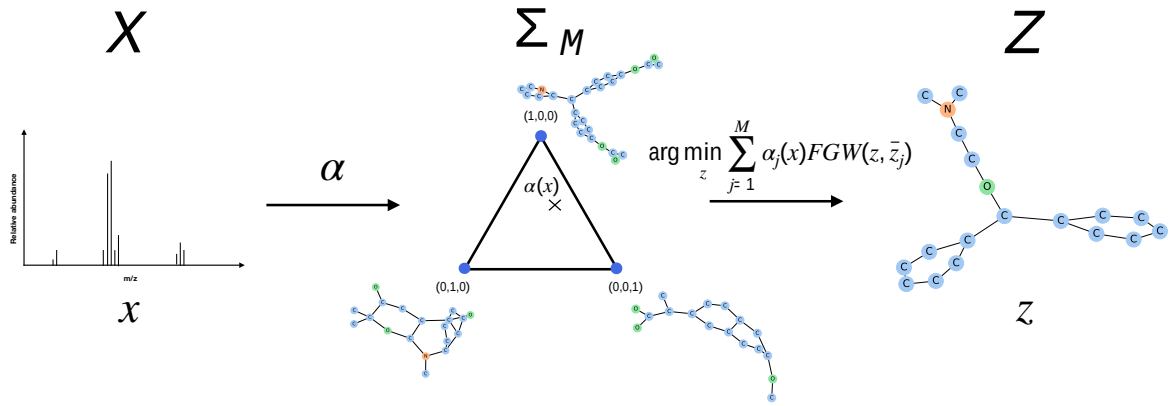

Figure 1. Proposed supervised graph prediction model. The input x (left) is mapped with $\alpha(x)$ onto the simplex (center) where the weights are used for computing the prediction as a FGW barycenter (right).

over M template graphs $z_j \in Z$ (See Figure 1):

$$f(x) = \arg\min_{z \in Z_n} \sum_{j=1}^{M} {}_j(x;W)\mathrm{FGW}_2^2(z;z_j); \quad (6)$$

where the weights ${}_j(x;W) : X \to \mathbb{R}^+$ are functions that can be understood as similarity scores between x and x_j. We include in a single parameter $= (M;(z_j)_{j=1}^M;W)$ all model's parameters.

A key feature of the proposed model f is that it interpolates in the graph space Z by using the Fréchet mean with respect to the FGW distance. Therefore, it inherits the good properties of FGW, especially including the invariance under isomorphism (two isomorphic graphs have equal scores in Eq. (6)). Moreover, in terms of computations, the proposed model leverages the recent advances in computational optimal transport such as Conditional Gradient descent (Vayer et al., 2019) or Mirror descent for (F)GW with entropic regularization (Peyré et al., 2016).

Properties of f . Relying on recent works that studied in a large extent GW and FGW barycenters, we now discuss the shape of the recovered objects (Peyré et al., 2016; Vayer et al., 2020, Eq. 14). The evaluation of f on input x writes as follows: $f(x) = (C(x;\ );F(x;\ );n\ ^1 1_n)$, where the structure and feature barycenters are:

$$C(x;\ ) = n^2 \sum_{j=1}^{M} {}_j(x;W)\ {}_j^T C_j\ {}_j \in [0;1]^{n\ \ n}; \quad (7)$$

$$F(x;\ ) = n \sum_{j=1}^{M} {}_j(x;W)F_j\ {}_j^T \in \mathbb{R}^{n\ \ d}: \quad (8)$$

The $(\ {}_j)_j$ are the optimal transport plans from $(C_j;F_j)_j$ to the barycenter $(C(x;\ );F(x;\ ))$ (Cuturi & Doucet, 2014, Eq. (8)) , and thus depend on . Note that a very appealing property of using FGW barycenter is that the order n (that fixes the prediction space Z_n) of the prediction does not depend on the parameters . This means that a unique trained model can predict several objects with a different resolution n allowing better interpretation at small resolution and finer modeling at higher resolution. This will be illustrated in the experimental section.

In the next sections, we propose two different approaches to learn and define the conditional barycenter. The first one in Section 4 leads to a purely nonparametric estimator with $M = N$ and $z_j = z_j$ and the second one proposed in Section 5 relies on a deep neural network for the weight functions ${}_j$s' while the template graphs $(z_j)_{j=1}^M$ are learned as well.

4. Nonparametric conditional Gromov-Wasserstein barycenter

Non-parametric estimator with kernels. Before addressing the general problem of learning both the template graphs and the weight function , we adopt a nonparametric point of view to address the structured regression problem. Under some conditions we recover a FGW conditional barycenter estimator of the following form:

$$f_W(x) = \arg\min_{z \in Z_n} \sum_{j=1}^{N} {}_j(x;W)\mathrm{FGW}_2^2(z;z_j); \quad (9)$$

where $= W$ is now the single parameter to learn and the template graphs z_j are not estimated but set as all the training samples z_j. Similarly to scalar or vector-valued regression, one can find many different ways to define the weight functions ${}_i$ in the large family of nonparametric estimators (Geurts et al., 2006; Ciliberto et al., 2020). We propose here a kernel approach that leverages kernel ridge regression.

Defining a positive definite kernel on the input space $k : X \times X \to \mathbb{R}$, one can consider the coefficients of kernel ridge estimation as in Brouard et al. (2016b); Ciliberto et al.

(2020) to define the weight function $\alpha : X \to \mathbb{R}^N$:

$$\alpha(x) = (K + \lambda I_N)^{-1} k_x \tag{10}$$

with the Gram matrix $K = (k(x_i, x_j))_{ij} \in \mathbb{R}^{N \times N}$ and the vector $k_x^T = (k(x, x_1), \dots, k(x, x_N))$. Such a model leverages learning in vector-valued Reproducing Kernel Hilbert Spaces and is rooted in the Implicit Loss Embedding (ILE) framework proposed and studied by Ciliberto et al. (2020).

Example 4.1. In the metabolite identification problem (see Section 6), the input takes the form of tandem mass spectra. A typical relevant kernel k for such data is the probability product kernel (PPK) (Heinonen et al., 2012).

4.1. Theoretical justification for the proposed model

The framework SELF (Ciliberto et al., 2016) and its extension ILE (Ciliberto et al., 2020) concerns general regression problems defined by an asymmetric loss $\ell : Z \times Y \to \mathbb{R}$ that can be written using output embeddings, allowing to solve a surrogate regression problem in the output embedding space. We recall the ILE property and the resulting benefits, especially when working in vector-valued Reproducing Kernel Hilbert Space.

Definition 4.2 (ILE)**.** For given spaces Z, Y, a map $\ell : Z \times Y \to \mathbb{R}$ is said to admit an Implicit Loss Embedding (ILE) if there exists a separable Hilbert space U and two measurable bounded maps $\psi : Z \to U$ and $\varphi : Y \to U$, such that for any $z \in Z, y \in Y$: $\ell(z, y) = \langle \psi(z), \varphi(y) \rangle_U$.

Note that this definition highlights an asymmetry between the processing of z and y. A regression problem based on a loss satisfying the ILE condition enjoys interesting properties. The following true risk minimization problem: $\min_f \mathbb{E}[\ell(f(X), Y)] := \mathbb{E}[\langle h(f(X)), \varphi(Y) \rangle_U]$, can be converted into i) a surrogate (intermediate) and simpler least-squares regression problem into the implicit embedding space U, i.e. $\min_{h: X \to U} \mathbb{E}[\|h(X) - \varphi(Y)\|_U^2]$, and ii) a decoding phase: $f(x) := \arg\min_z \langle z, h(x) \rangle_U$, where h is solution of problem i), i.e. $h(x) = \mathbb{E}[\varphi(Y) | x]$. A nice property proven by Ciliberto et al. (2020) is the one of Fisher consistency, f is exactly the minimizer of problem in Eq. (2), justifying the surrogate approaches.

Structured prediction with implicit embedding and kernels. Assuming the loss is ILE, when relying on a i.i.d. training sample $\{(x_i, y_i)_{i=1}^N\}$, one gets $\hat{h}$ an estimator of h by minimizing the corresponding (regularized) empirical risk and then builds $\hat{f}$.

If we choose to search $\hat{h}$ in the vector-valued Reproducing Kernel Hilbert Space H_K associated to the decomposable operator-valued kernel $K : X \times X \to \mathcal{L}(U)$ of the form $K(x, x') = I_U k(x, x')$ where k is the positive definite kernel defined in Section 4 and I_U is the identity operator on the Hilbert space U, then the solution to the problem:

$$\min_{h \in H_K} \sum_{i=1}^N \|h(x_i) - \varphi(y_i)\|_U^2 + \lambda \|h\|_{H_K}^2,$$

for $\lambda > 0$, writes as $\hat{h}(x) = \sum_{i=1}^N \alpha_i(x) \varphi(y_i)$ with $\alpha(x)$ verifying Eq. (10). Then, $\hat{f}(x)$ can be expressed as

$$\arg\min_{z \in Z} \Big\langle h(z), \sum_{i=1}^N \alpha_i(x) \varphi(y_i) \Big\rangle_U = \arg\min_{z \in Z} \sum_{i=1}^N \alpha_i(x) \ell(z, y_i)$$

We show in the following proposition that FGW admits an ILE. This allows us to obtain theoretical guarantees from Ciliberto et al. (2020) for our estimator.

Proposition 4.3. *FGW admits an ILE.*

Proof. Y is a finite space by definition. Z_n is a compact space as $[0, 1]^{n \times n}$ and $\mathrm{Conv}(F)^n$ are compact (F is finite). Moreover, $\forall y \in Y, z \to \mathrm{FGW}(z, y)$ is a continuous map (See Lemma A.1). Therefore, according to Theorem 7 from Ciliberto et al. (2020) $\mathrm{FGW} : Z_n \times Y \to \mathbb{R}$ admits an ILE. □

4.2. Excess-risk bounds

Since FGW is ILE, the proposed estimator enjoys consistency (See Theorem A.2 in Appendix). Moreover, under an additional technical assumption (Assumption A.3 in Appendix), it verifies the following excess-risk-bound.

Theorem 4.4 (Excess-risk bounds)**.** *Let k be a bounded continuous reproducing kernel such that $\kappa^2 := \sup_{x \in X} k(x, x) < +\infty$. Let ρ be a distribution on $X \times Y$. Let $\lambda \in (0, 1]$ and N_0 sufficiently large such that $N_0^{-1/2} \geq \frac{9\kappa^2}{N_0} \log \frac{N_0}{\delta}$. Under Assumption A.3, for any $N \geq N_0$, if f_W is the proposed estimator built from N independent couples $(x_i, y_i)_{i=1}^N$ drawn from ρ. Then, with probability $1 - \delta$*

$$R^n(f_W) - R^n(f^*) \leq c \log(4/\delta) N^{-1/4}, \tag{11}$$

with c a constant independent of N and δ.

Note that $N^{-1/4}$ is the typical rate for structured prediction problems without further assumptions on the problem (Ciliberto et al., 2016; 2020). Theorem 4.4 relies on the attainability assumption A.3. This can be interpreted as the fact that the proposed GW barycentric model defines an hypothesis space which is able to deal with graph prediction problems that are smooth with respect to the FGW metric. This corroborates with the intuition that for such problems FGW interpolation will obtain good prediction results. We illustrate this theoretical insight on a synthetic dataset in the experimental section. Furthermore, both theorems are valid for any $Z_n, n \in \mathbb{N}^*$, that is, they provide guarantees for all regression problems defined in Eq. (2) for all $n \in \mathbb{N}^*$.

5. Neural network-based conditional Gromov-Wasserstein barycenter

In this section, we discuss how to train a neural network model estimator as defined in Equation (6) where the template graphs z_j are learned simultaneously with the weight function . This provides a very generic model that inherits the flexibility of deep neural networks and their ability to learn input data representation.

Parameters of the model. First we recap the different parameters that we want to optimize. First, the weights $(x; W)$ of the barycenter are modeled by a deep neural network with parameters W. Next the templates M graphs z_j are also estimated allowing the model to better adapt to the prediction task. It is important to note that M is also a parameter of the model that will tune the complexity of the model and will need to be validated in practice. Note that this parametric formulation is better suited to large scale datasets since the complexity of the predictor will be fixed by M instead of increasing with the number of training data N as in non-parametric models.

Stochastic optimization of the model. We optimize the parameters of the model using a classical ADAM (Kingma & Ba, 2014) stochastic optimization procedure where the gradients are taken over samples or minibatches of the full empirical distribution.

We now discuss the computation of the stochastic gradient on a training sample $(x_i; y_i)$. First note that the gradient of FGW$(f\ (x_i); y_i)$ *w.r.t.* is actually the gradient of a bi-level optimization problem since f is the solution of a FGW barycenter. The barycenter solutions expressed in Equations (7) and (8) actually depends on the optimal OT plans $(\ _j)_j$ of the barycenter that depends themselves on . But in practice the OT plans $(\ _j)_j$ are solutions of a non-convex and non-smooth quadratic program and are with high probability on a border of the polytope (Maron & Lipman, 2018). This means that we can assume that a small change in will not change their value and a reasonable differential of $(\ _j)_j$ *w.r.t.* is the null vector. This actually corresponds in Pytorch (Paszke et al., 2019) notation to "detach" the OT plan with respect to the input which is done by default in POT toolbox (Flamary et al., 2021). The gradient of the outer FGW loss can be easily computed as the gradient of the loss with the fixed optimal plan $_i$ using the theorem from (Bonnans & Shapiro, 1998). Computing a sub-gradient of the loss FGW$(f\ (x_i); y_i)$ can then be done with the following steps:

1. $(\ _j)_j$ Compute the barycenter $f\ (x_i)$.
2. $_i$ Compute the loss FGW$(f\ (x_i); y_i))$.
3. r Compute the gradient of FGW$(f\ (x_i); y_i))$ with fixed OT plans $(\ _j)_j$ and $_i$.

Note that for the matrices C_j in the templates, the stochastic update is actually a projected gradient step onto the set of matrices with components belonging to $[0; 1]$.

6. Numerical experiments

In this section, we evaluate the proposed method on a synthetic problem and the metabolite identification problem. A Python implementation of the method is available on github[1].

6.1. Synthetic graph prediction problem

Problem and dataset. We consider the following graph prediction problem. Given an input x drawn uniformly in $[1; 6]$, y is drawn using a Stochastic block model with bxc blocks, such that the biggest block smoothly splits into two blocks when x is between two integers (see Figure 2, bottom line). Each node has a label, which is an integer indicating the block the node is belonging to. More precisely, we take randomly from 40 to 45 nodes for each graph (uniformly in $J40; 45K$. There is a probability $0{:}9$ of connection between nodes belonging to the same block, and a probability $0{:}01$ of connection between nodes belonging to different blocks. The probability of connection between nodes belonging to the splitting blocks is $p(x) = 0{:}889(x \quad bxc) + 0{:}01$. When a node belongs to the new appearing block its label is the new block's label with probability $(x \quad bxc)$, and the splitting block's label otherwise. We generate a training set of $N = 50$ couples $(x_i; y_i)_{i=1}^N$. Notice that the considered learning problem is highly difficult as one want to predict a graph from a continuous value in $[1; 6]$.

Experimental setting. We test the parametric version of the proposed method with learning of the templates. We use $M = 10$ templates, with 5 nodes, and initialize them drawing $C_i \ 2\ \mathrm{R}^{5\ \ 5}; F_i \ 2\ \mathrm{R}^{5\ \ 1}$ uniformly in $[0; 1]^{5\ \ 5}$ and $[0; 1]^{5\ \ 1}$. The weights $(x; W) \ 2\ \mathrm{R}^M$ are implemented using a three-layer (100 neurons in each hidden layer) fully connected neural network with ReLU activation functions, and a final softmax layer. We use = 1=2 as FGW's balancing parameter and a prediction size of $n = 40$ during training. During training, we optimize the parameters of the model using the continuous relaxed graph prediction model. Interestingly this prediction provides us with continuous versions of the adjacency matrices so we can generate discrete graphs by randomly sampling each edge with a Bernouilli distribution of parameter given by $C(x; \ )$.

Supervised learning result. The estimated graph prediction model on the synthetic dataset is illustrated in Figure 2. We can see that the learned map is indeed recovering the

[1] https://github.com/lmotte/graph-prediction-with-fused-gromov-wasserstein

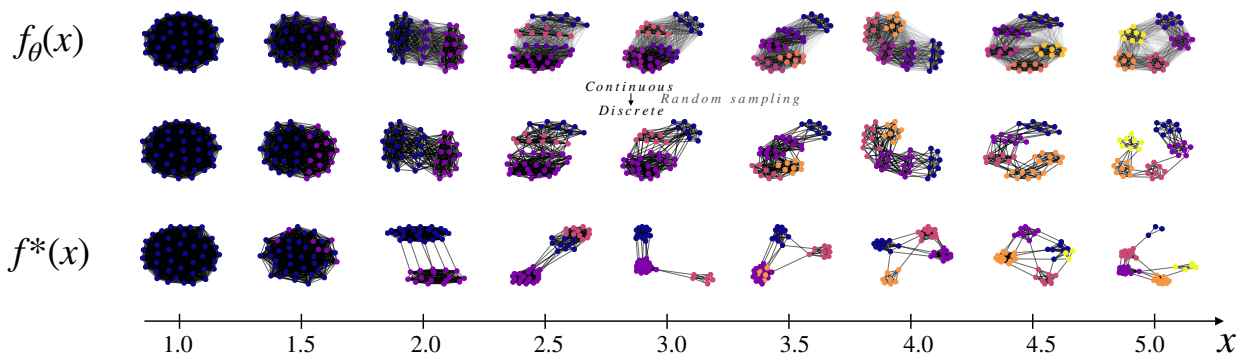

Figure 2. Graph prediction on the synthetic dataset as a function of the 1D input x. (top) estimated continuous prediction $f_\theta(x)$, (middle) discrete realizations following the continuous prediction, (bottom) true graph prediction function $f^\star(x)$.

evolution of the graphs as a function of x. This shows, as suggested by the theoretical results in Section 4, that the FGW metric is a a good data fitting term and that FGW barycenters are a good way to interpolate continuously between discrete objects. This is particularly true on this problem where a small change w.r.t x induces small change in the output of f (x) according to the FGW metric.

Interpretability and flexibility of the proposed model. We now illustrate how interpretable is the estimated model. First we recall that the prediction is actually a Fréchet mean w.r.t the FGW distance, according to the weights $_j(x)$ and the templates $(z_j)_{j=1}^m$. In practice it means that we can plot the template graphs $(z_j)_{j=1}^m$ to check that the learned templates are indeed similar (with less nodes) to training data. But on this synthetic dataset we can also plot the trajectory of the barycenter weights $_j$ on the simplex as a function of x which we did in Figure 3. We can see in the figure that in practice the weights $_j(x)$ are sparse concentrated on the templates on the left of the Figure starting with a graph with one connected cluster and ending with a graph with 5 clusters following the true model $f^?$.

We now illustrate one very interesting property of our model: the ability to predict graphs with a varying number of nodes n for a given input x. An example of the predicted graphs for $x = 5$ is provided in Figure 4. It is interesting to note that even with small templates of 5 nodes, the proposed barycentric graph prediction model is able to predict big graphs while preserving their global structure. This is particularly true for Stochastic Block Models graphs that can by construction be factorized with a small number of clusters. Note that the number of nodes in the templates $(z_j)_{j=1}^m$ can be seen as a regularization parameter. The model is also very flexible in the sens that the FGW barycenter modeling allows for templates with different number of nodes allowing for a coarse to fine modeling of the data.

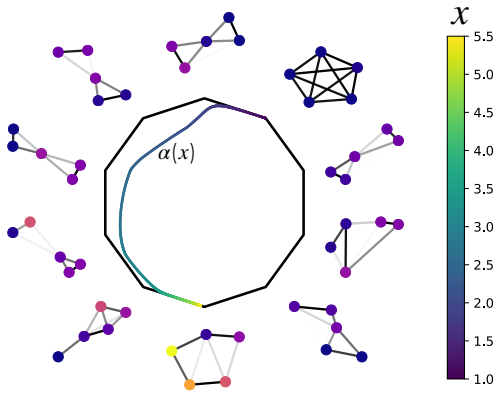

Figure 3. Learned templates $(z_j)_{j=1}^m$ on the synthetic dataset and trajectory of the weights (x) on the simplex as a function of x.

6.2. Metabolite identification problem

Problem and dataset. An important problem in metabolomics is to identify the small molecules, called metabolites, that are present in a biological sample. Mass spectrometry is a widespread method to extract distinctive features from a biological sample in the form of a tandem mass (MS/MS) spectrum. The goal of this problem is to predict the molecular structure of a metabolite given its tandem mass spectrum. Labeled data are expensive to obtain, and despite the problem complexity not many labeled data are available in datasets. Here we consider a set of 4138 labeled data, that have been extracted and processed in Dührkop et al. (2015), from the GNPS public spectral library (Wang et al., 2016). Datasets and code for reproducing the metabolite identification experiments are available on github[2].

Experimental setting. We test the nonparametric version of the proposed method, using a probability product kernel on the mass spectra, as it has been shown to be a good choice

[2] `lmotte/metabolite-identification-with-fused-gromov-wasserstein`

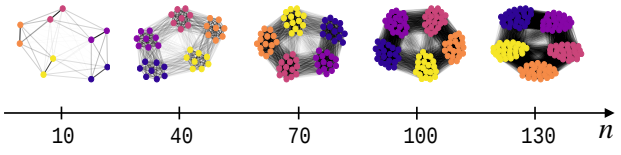

Figure 4. Predicted graphs with the estimated model $f_\theta(x)$ with a varying number of nodes n for $x = 5$.

on this problem (Brouard et al., 2016a). We use $= 0.5$ as FGW balancing parameter. We split the dataset into a training set of size $N = 3000$ and a test set of size $N_{te} = 1138$. On this problem, structured prediction approaches that have been proposed fall back on the availability of a known candidate set of output graphs for each input spectrum (Brouard et al., 2016a). This means that in practice for prediction on new data, we will not solve the FGW barycenter in (6) but search among the possible candidates in Y the one minimizing the barycenter loss.

In a first experiment, we evaluate the performance of FGW as a graph metric. To this end we compare the performance of various graph metrics $D : Y \quad Y \to \mathbb{R}^+$ used in the model: $\arg\min_{y \in Y} \sum_{j=1}^{N} \_{j}(x; W) D(y; y_j)$. We consider the metric induced by the standard Weisfeiler–Lehman (WL) graph kernel that consists in embedding graphs as a bag of neighbourhood configurations (Shervashidze et al., 2011). The FGW one-hot distance corresponds to the FGW distance and using a one-hot encoding of the atoms. The FGW fine distance corresponds to the one-hot distance concatenated with additional atom features: number of attached hydrogens, number of heavy neighbours, formal charge, is in a ring, is in an aromatic ring. Additional features are normalize by their maximum values in the molecule at hand. The FGW diffuse distance corresponds to the FGW distance and using a one-hot encoding of the atoms which has been diffused, namely: $F_{\text{diff}} = e^{\quad \text{Lap}(C)} F$, where > 0, $\text{Lap}(C)$ denotes the normalized Laplacian of C as proposed in Barbe et al. (2020). Fingerprints are molecule representations, well engineered by experts, that are binary vectors. Each value of the fingerprint indicates the presence or absence of a certain molecular property (generally a molecular substructure). Several machine learning approaches using fingerprints as output representations have obtained very good performances for metabolite identification (Dührkop et al., 2015; Brouard et al., 2016a; Nguyen et al., 2018) or other tasks, such as metabolite structural annotation (Hoffmann et al., 2021). In the last two Casmi challenges (Schymanski et al., 2017), such approaches have obtained the best performances for the best automatic structural identification category. Here we consider the metrics induced by linear and Gaussian kernels between fingerprints of length $d = 2765$. Notice that, in this case, the structured prediction method corresponds to IOKR-Ridge proposed in Brouard et al. (2016b). For the FGW metrics, we compute them using the 5 greatest weights $\_{i}(x)$. We evaluate the results in terms of Top-k accuracy: percentage of true output among the k outputs given by the k greatest scores in the model. The two hyperparameters (ridge regularization parameter and the output metric's parameter) are selected using a validation set (1/5 of the training set) and Top-1 accuracy.

	TOP-1	TOP-10	TOP-20
WL KERNEL	9.8%	29.1%	37.4%
LINEAR FINGERPRINT	28.6%	54.5%	59.9%
GAUSSIAN FINGEPRINT	41.0%	62.0%	67.8%
FGW ONE-HOT	12.7%	37.3%	44.2%
FGW FINE	18.1%	46.3%	53.7%
FGW DIFFUSE	27.8%	52.8%	59.6%

Table 1. Top-k accuracies for various graph metrics on the metabolite identification dataset.

Graph metrics comparison. The results given in Table 1 shows that Gaussian fingerprints is the best performing metric on this dataset when a candidate set is available. We see that the FGW greatly benefits from the improved fine and diffuse metrics showing the adaptation potential of the FGW metric to the graph space at hand reaching competitive performance against fingerprints with linear kernel and beating WL kernels. The method proposed in this work is the first generic approach that obtained good Top-k accuracies without using expert-derived molecular graph representations.

Predicting novel molecules. Being able to interpolate novel graphs without using predefined finite candidate sets is a great advantage of the proposed method. Such computation is in general intractable (e.g. with WL and fingerprint metrics). In this experiment, we evaluate the performance of the estimator when computing the barycenter over Z_n, and not over the candidate sets. For a given test input x, let us define $d_0(x)$ the FGW (one-hot) distance of the training molecule with the greatest $\_{j}(x)$ to the true molecule. $d_0(x)$ measures the level of interpolation difficulty: very small d_0 means that the true molecule is close to a training molecule and no interpolation is required. We compute, over 1000 test data, the mean $d_0(x)$ and the mean FGW (one-hot) distance between the predicted barycenter (using the 10 largest $\_{j}(x)$) and the true test molecule. In Figure 5, we plot the two mean distances, with respect to a filtering threshold d_{min} such that only the test point with $d_0(x) > d_{min}$ are used when computing these means. We can see that the FGW interpolation allows to become closer to the true output than only predicting the output with the greatest weight $\_{j}(x)$, even more when interpolation is required ($d_0(x)$ big). This validates the choice of FGW as a way to interpolate

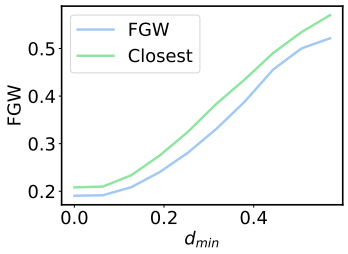

Figure 5. No candidate set setting. In average, the FGW barycenter (blue) using the 10 molecules with the greatest weights $_j(x)$ is closer to the true molecule, than the molecule with the greatest weight $_j(x)$: closest template prediction (green).

between real-world graphs.

Comparison with a flow-based deep graph generation method. As mentioned previously, to the best of our knowledge, there is no generic method for graph prediction able to deal with any graph space at hand. The only existing methods, that do not require expert-derived graph representations available for a specific graph space, are unsupervised deep graph generation methods (Li et al., 2018b; Liao et al., 2019; Zang & Wang, 2020; Mercado et al., 2021). We propose to compare our approach by designing a new generic graph prediction method. We use the deep generative graph representations from MoFlow (Zang & Wang, 2020) learned from 249.455 molecules and which obtained state-of-the-art results in (unsupervised) molecular graph generation. The latent representations are learned via kernel ridge regression, then we predict the candidate with the closest latent representation to the estimated one. Note that because the pre-trained model's architecture can not handle all atoms present in the metabolite dataset, we removed from the dataset the molecules with not handled atoms. Moreover, we compute the test predictions using the test spectra with less than 300 candidates for faster computation: 286 test points. The results are given in Table 2. We observe that FGW diffuse exhibits far better performance than the MoFlow approach.

	TOP-1	TOP-10	TOP-20
GAUSSIAN FINGERPRINT	46.2%	77.8%	84.9%
FGW DIFFUSE	40.3%	69.7%	78.3%
MOFLOW REPRESENTAT.	20.0%	58.2%	68.4%

Table 2. Top-k accuracies obtained using deep molecular graph representations in comparison to the proposed FGW metric, and expert-derived fingerprint representations.

7. Conclusion

We proposed in this work a novel framework for graph prediction using optimal transport barycenters to interpolate continuously in the output space. We discussed both a non-parametric estimator with theoretical guarantees and a parametric one based on neural network models that can be estimated with stochastic gradient methods. The method was illustrated on synthetic and real life data showing the interest of the continuous relaxation especially when targets are not available.
Future works include estimation of the target number of nodes $n(x)$ and supervised learning of complementary feature on the templates that can guide the FGW barycenters.

Acknowledgements

The first and last authors are funded by the French National Research Agency (ANR) through ANR-18-CE23-0014 APi (Apprivoiser la Pré-image) and the Télécom Paris Research Chair DSAIDIS. This work was also partially funded through the projects OATMIL ANR-17-CE23-0012, 3IA Côte d'Azur Investments ANR-19-P3IA-0002 of the French National Research Agency (ANR) and was produced within the framework of Energy4Climate Interdisciplinary Center (E4C) of IP Paris and Ecole des Ponts ParisTech. It was supported by 3rd Programme d'Investissements d'Avenir ANR-18-EUR-0006-02. This action benefited from the support of the Chair "Challenging Technology for Responsible Energy" led by l'X – Ecole polytechnique and the Fondation de l'Ecole polytechnique, sponsored by TOTAL. This research was partially funded by Academy of Finland grant 334790 (MAGITICS).

References

Altschuler, J., Weed, J., and Rigollet, P. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. *arXiv preprint arXiv:1705.09634*, 2017.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Barbe, A., Sebban, M., Gonçalves, P., Borgnat, P., and Gribonval, R. Graph diffusion wasserstein distances. In *ECML PKDD 2020-European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 1–16, 2020.

Belanger, D. and McCallum, A. Structured prediction energy networks. In *International Conference on Machine Learning*, pp. 983–992. PMLR, 2016.

Belanger, D., Yang, B., and McCallum, A. End-to-end learning for structured prediction energy networks. In *International Conference on Machine Learning*, pp. 429–439. PMLR, 2017.

Bhagat, S., Cormode, G., and Muthukrishnan, S. Node classification in social networks. In *Social network data analytics*, pp. 115–148. Springer, 2011.

Bonnans, J. F. and Shapiro, A. Optimization problems with perturbations: A guided tour. *SIAM review*, 40(2):228–264, 1998.

Bonneel, N., Peyré, G., and Cuturi, M. Wasserstein barycentric coordinates: histogram regression using optimal transport. *ACM Trans. Graph.*, 35(4):71–1, 2016.

Brouard, C., Shen, H., Dührkop, K., d'Alché Buc, F., Böcker, S., and Rousu, J. Fast metabolite identification with input output kernel regression. *Bioinformatics*, 32 (12):i28–i36, 2016a.

Brouard, C., Szafranski, M., and d'Alché Buc, F. Input output kernel regression: Supervised and semi-supervised structured output prediction with operator-valued kernels. *Journal of Machine Learning Research*, 17:np, 2016b.

Caponnetto, A. and De Vito, E. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

Chen, L.-C., Schwing, A., Yuille, A., and Urtasun, R. Learning deep structured models. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1785–1794, Lille, France, 07–09 Jul 2015. PMLR.

Ciliberto, C., Rosasco, L., and Rudi, A. A consistent regularization approach for structured prediction. *Advances in neural information processing systems*, 29:4412–4420, 2016.

Ciliberto, C., Rosasco, L., and Rudi, A. A general framework for consistent structured prediction with implicit loss embeddings. *J. Mach. Learn. Res.*, 21(98):1–67, 2020.

Cortes, C., Mohri, M., and Weston, J. A general regression technique for learning transductions. In Raedt, L. D. and Wrobel, S. (eds.), *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pp. 153–160. ACM, 2005.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26:2292–2300, 2013.

Cuturi, M. and Doucet, A. Fast computation of wasserstein barycenters. In *International conference on machine learning*, pp. 685–693. PMLR, 2014.

Dührkop, K., Shen, H., Meusel, M., Rousu, J., and Böcker, S. Searching molecular structure databases with tandem mass spectra using csi: Fingerid. *Proceedings of the National Academy of Sciences*, 112(41):12580–12585, 2015.

Flamary, R., Cuturi, M., Courty, N., and Rakotomamonjy, A. Wasserstein discriminant analysis. *Machine Learning*, 107(12):1923–1945, 2018.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., et al. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

Frogner, C., Zhang, C., Mobahi, H., Araya-Polo, M., and Poggio, T. Learning with a wasserstein loss. *arXiv preprint arXiv:1506.05439*, 2015.

Geurts, P., Wehenkel, L., and d'Alché Buc, F. Kernelizing the output of tree-based methods. In *Proceedings of the 23rd international conference on Machine learning*, pp. 345–352, 2006.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Gordaliza, P., Del Barrio, E., Fabrice, G., and Loubes, J.-M. Obtaining fairness using optimal transport theory. In *International Conference on Machine Learning*, pp. 2357–2365. PMLR, 2019.

Heinonen, M., Shen, H., Zamboni, N., and Rousu, J. Metabolite identification and molecular fingerprint prediction through machine learning. *Bioinformatics*, 28 (18):2333–2341, 07 2012. ISSN 1367-4803. doi: 10.1093/bioinformatics/bts437. URL https://doi.org/10.1093/bioinformatics/bts437.

Hoffmann, M. A., Nothias, L.-F., Ludwig, M., Fleischauer, M., Gentry, E. C., Witting, M., Dorrestein, P. C., Dührkop, K., and Böcker, S. High-confidence structural annotation of metabolites absent from spectral libraries. *Nature Biotechnology*, pp. 1–11, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kriege, N. M., Johansson, F. D., and Morris, C. A survey on graph kernels. *Applied Network Science*, 5(1):1–42, 2020.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. From word embeddings to document distances. In *International conference on machine learning*, pp. 957–966. PMLR, 2015.

Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. Grammar variational autoencoder. In *International Conference on Machine Learning*, pp. 1945–1954. PMLR, 2017.

Li, Y., Vinyals, O., Dyer, C., Pascanu, R., and Battaglia, P. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018a.

Li, Y., Zhang, L., and Liu, Z. Multi-objective de novo drug design with conditional graph generative model. *Journal of cheminformatics*, 10(1):1–24, 2018b.

Liao, R., Li, Y., Song, Y., Wang, S., Hamilton, W., Duvenaud, D. K., Urtasun, R., and Zemel, R. Efficient graph generation with graph recurrent attention networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Liu, B., Ramsundar, B., Kawthekar, P., Shi, J., Gomes, J., Luu Nguyen, Q., Ho, S., Sloane, J., Wender, P., and Pande, V. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS central science*, 3 (10):1103–1113, 2017.

Lü, L. and Zhou, T. Link prediction in complex networks: A survey. *Physica A: statistical mechanics and its applications*, 390(6):1150–1170, 2011.

Luise, G., Rudi, A., Pontil, M., and Ciliberto, C. Differential Properties of Sinkhorn Approximation for Learning with Wasserstein Distance. In *NIPS 2018 - Advances in Neural Information Processing Systems*, pp. 5864–5874, Montreal, Canada, December 2018. 26 pages, 4 figures.

Maron, H. and Lipman, Y. (probably) concave graph matching. *arXiv preprint arXiv:1807.09722*, 2018.

Mensch, A. and Blondel, M. Differentiable dynamic programming for structured prediction and attention. In *International Conference on Machine Learning*, pp. 3462–3471. PMLR, 2018.

Mensch, A., Blondel, M., and Peyré, G. Geometric losses for distributional learning. In *International Conference on Machine Learning*, pp. 4516–4525. PMLR, 2019.

Mercado, R., Rastemo, T., Lindelöf, E., Klambauer, G., Engkvist, O., Chen, H., and Bjerrum, E. J. Graph networks for molecular design. *Machine Learning: Science and Technology*, 2(2):025023, 2021.

Mémoli, F. The gromov–wasserstein distance and the metric approach to oject matching. *Found Comput Math*, 11(4): 417–487, 2011.

Nguyen, D. H., Nguyen, C. H., and Mamitsuka, H. SIMPLE: Sparse Interaction Model over Peaks of moLEcules for fast, interpretable metabolite identification from tandem mass spectra. *Bioinformatics*, 34(13):i323–i332, 2018.

Olivecrona, M., Blaschke, T., Engkvist, O., and Chen, H. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):1–14, 2017.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

Peyré, G., Cuturi, M., and Solomon, J. Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*, pp. 2664–2672. PMLR, 2016.

Peyré, G., Cuturi, M., et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

Pillutla, V. K., Roulet, V., Kakade, S. M., and Harchaoui, Z. A smoother way to train structured prediction models. *Advances in Neural Information Processing Systems 31*, 2018.

Ralaivola, L., Swamidass, S. J., Saigo, H., and Baldi, P. Graph kernels for chemical informatics. *Neural Networks*, 18(8):1093–1110, 2005.

Schymanski, E. L., Ruttkies, C., Krauss, M., Brouard, C., Kind, T., Dührkop, K., Allen, F., Vaniya, A., Verdegem, D., Böcker, S., et al. Critical assessment of small molecule identification 2016: automated methods. *Journal of cheminformatics*, 9(1):1–21, 2017.

Shervashidze, N., Schweitzer, P., Van Leeuwen, E. J., Mehlhorn, K., and Borgwardt, K. M. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9), 2011.

Shi, C., Xu, M., Zhu, Z., Zhang, W., Zhang, M., and Tang, J. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*, 2020.

Silver, D., Hasselt, H., Hessel, M., Schaul, T., Guez, A., Harley, T., Dulac-Arnold, G., Reichert, D., Rabinowitz,

N., Barreto, A., et al. The predictron: End-to-end learning and planning. In *International Conference on Machine Learning*, pp. 3191–3199. PMLR, 2017.

Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y., and Singer, Y. Large margin methods for structured and interdependent output variables. *Journal of machine learning research*, 6(9), 2005.

Vayer, T., Chapel, L., Flamary, R., Tavenard, R., and Courty, N. Optimal transport for structured data with application on graphs. In *International Conference on Machine Learning (ICML)*, 2019.

Vayer, T., Chapel, L., Flamary, R., Tavenard, R., and Courty, N. Fused gromov-wasserstein distance for structured objects. *Algorithms*, 13(9):212, 2020.

Vincent-Cuaz, C., Vayer, T., Flamary, R., Corneli, M., and Courty, N. Online graph dictionary learning. In *International Conference on Machine Learning (ICML)*, 2021.

Wang, M., Carver, J. J., Phelan, V. V., Sanchez, L. M., Garg, N., Peng, Y., Nguyen, D. D., Watrous, J., Kapono, C. A., Luzzatto-Knaan, T., et al. Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nature biotechnology*, 34(8):828–837, 2016.

Xu, H., Luo, D., and Carin, L. Scalable gromov-wasserstein learning for graph partitioning and matching. *Advances in neural information processing systems*, 32:3052–3062, 2019a.

Xu, H., Luo, D., Zha, H., and Duke, L. C. Gromov-wasserstein learning for graph matching and node embedding. In *International conference on machine learning*, pp. 6932–6941. PMLR, 2019b.

You, J., Liu, B., Ying, R., Pande, V., and Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. *arXiv preprint arXiv:1806.02473*, 2018.

Zang, C. and Wang, F. Moflow: an invertible flow model for generating molecular graphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 617–626, 2020.

Zhang, Z., Cui, P., and Zhu, W. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 34:249–270, 2022.

A. Theory

A.1. Proof of FGW continuity

We prove the continuity of $\mathrm{FGW}(\cdot; y) : Z_n \to \mathbb{R}$ for any $y \in Y$. Such result is crucial to prove the ILE property of $\mathrm{FGW} : Z_n \times Y \to \mathbb{R}$.

Lemma A.1 (FGW continuity)**.** *Let $y = (C_2; F_2)$ with $C_2 \in \mathbb{R}^{n_2 \times n_2}; F_2 \in \mathbb{R}^{n_2 \times d}$, $n_2; d \in \mathbb{N}$. The map $FGW(\cdot; y) : Z_n \to \mathbb{R}$ is continuous.*

Proof. Recall that for any $z = (C; F) \in Z_n$:

$$\mathrm{FGW}_2^2(z; y) = \min_{\in P_{n1;n2}} \sum_{i;k;j;l} (1 -)kF(i) - F_2(j)k^2_{\mathbb{R}^d} + (C(i;k) - C_2(j;l))^2 \; _{i;j} \; _{k;l}: \tag{12}$$

Using the inequality $|\min f(\cdot) - \min g(\cdot)| \leq \sup |f(\cdot) - g(\cdot)|$ for any $f; g : P_{n1;n2} \to \mathbb{R}$, we have for any $dz = (dC; dF) \in Z_n$

$$|\mathrm{FGW}_2^2(z + dz; y) - \mathrm{FGW}_2^2(z; y)| \leq \sup_{\in P_{n1;n2}} \Big| \sum_{i;k;j;l} (1 -) \left(\hbar dF(i)|F_2(j)|_{\mathbb{R}^d} + o(\|dF(i)\|_{\mathbb{R}^d})\right) \tag{13}$$

$$+ (dC(i;k)C_2(j;l) + o(dC(i;k))) \; _{i;j} \; _{k;l}\Big|$$

$$\leq n n_2 (1 -) \left(\|dF\|_{\mathbb{R}^{n \times d}} \|F_2\|_{\mathbb{R}^{n \times d}} + o(\|dF\|_{\mathbb{R}^{n \times d}})\right) \tag{14}$$

$$+ (\|dC\|_{\mathbb{R}^{n \times n}} \|dC_2\|_{\mathbb{R}^{n_2 \times n_2}} + o(\|dC\|_{\mathbb{R}^{n \times n}}))$$

$$= O(\|dz\|_{\mathbb{R}^{n \times n} \times \mathbb{R}^{n \times d}}) \underset{dz \to 0}{\to} 0 \tag{15}$$

where from (13) to (14) we have used the Cauchy–Schwarz inequality, and the fact that $\forall (i; j) \in [\![1; n]\!] \times [\![1; n_2]\!]; \; _{ij} \leq 1$. We conclude that $z \to \mathrm{FGW}_2^2(z; y)$ is a continuous on $\mathbb{R}^{n \times n} \times \mathbb{R}^{n \times d}$, hence on Z_n.

□

A.2. Universal consistency theorem

We restate the universal consistency theorem from Ciliberto et al. (2020) that is verified by our estimator because of the proved ILE property.

Theorem A.2 (Universal Consistency)**.** *Let k be a bounded universal reproducing kernel. For any $N \in \mathbb{N}$ and any distribution ρ on $X \times Y$ let f_W be the proposed estimator built from N independent couples $(x_i; y_i)_{i=1}^N$ drawn from ρ. Then, if $= N^{-1=2}$,*

$$\lim_{N \to +\infty} R^n(f_W) = R^n(f^*) \quad \textit{with probability } 1: \tag{16}$$

A.3. Attainability assumption

The following assumption is required to obtain finite sample bounds. It is a standard assumption in learning theory (Caponnetto & De Vito, 2007). It corresponds to assume that the solution h^* of the surrogate problem indeed belongs to the considered hypothesis space, namely the reproducing kernel Hilbert space induced by the chosen operator-valued kernel $K(x; x') = k(x; x')I_U$.

Assumption A.3 (attainable case)**.** We assume that there exists a linear operator $H : H_x \to U$ with $\|H\|_{\mathrm{HS}} < +\infty$ such that

$$\mathbb{E}_{Y|x}[\ell(Y)] = Hk(x; \cdot) \tag{17}$$

with H_x the reproducing kernel Hilbert space associated to the kernel $k(x; x')$.

B. Neural network model and training algorithm

Choice of the templates. As always in deep learning, parameter initialization is an important aspect and we discuss now how to initialize the templates z_j. In practice they can be initialized at random with matrices C_j drawn uniformly in $[0; 1]$

or chosen at random from training samples as suggested by the non-parametric model. One interesting aspect is that the number of nodes do not need to be the same for all templates. This means that one can have both templates with few nodes and templates with a larger number of nodes allowing for a coarse to-fine modeling of the graphs.

Pseudocode. We give the pseudocode for the proposed neural network training algorithm. This algorithm has been implemented in Python using the POT library: Python Optimal Transport (Flamary et al., 2021), and Pytorch library (Paszke et al., 2019).

Algorithm 1 Neural network-based model training - One stochastic gradient descent step

Input: x ! (x) neural network's parameters W. Templates $(z_j)_{j=1}^M$. Dictionary learning (True or False).
1. If Dictionary learning is True: $= (W; (z_j)_{j=1}^M)$. Otherwise: $= W$.
2. $(_j)_{j=1}^M$ Compute the barycenter $f\ (x_i)$.
3. $_i$ Compute the losses FGW$(f\ (x_i); y_i))$.
4. r Compute the gradient of FGW$(f\ (x_i); y_i))$ with fixed OT plans $(_j)_j$ and $_i$.

Return: Updated neural network's parameters W, updated templates $(z_j)_{j=1}^M$.

Python implementation on github. The code is available on github at `https://github.com/lmotte/graph-prediction-with-fused-gromov-wasserstein`.

C. Justification of the algorithms

Reminder on ILE and surrogate problem:
Recall that $\hat{h}$ is solving a least-squares problem, that is estimate $h\ (x) = \mathsf{E}_{zjx}[\ (z)]$. Moreover, we can write $f\ (x) = \arg\min_{\hat{z}} \mathsf{E}_{zjx}[\ (\hat{z}; z)]$. Now, we can provide intuition in the following derivations about the construction of $\hat{f}$ exploiting the linearity of expectation.

$$\hat{f}(x) = \arg\min_{\hat{z}} h\ (\hat{z}); \hat{h}(x) i_H$$
$$\arg\min_{\hat{z}} h\ (\hat{z}); h\ (x) i_H :$$

Moreover, we have:

$$h\ (\hat{z}); h\ (x) i_H = \mathsf{E}_{zjx}[h\ (\hat{z}); \ (z) i_H]$$
$$= \mathsf{E}_{zjx}[\ (\hat{z}; z)]$$

and thus, taking the "arg min" gives:

$$\hat{f}(x) \quad f\ (x):$$

D. Discussion about keeping only the greatest weights $\alpha_i(x)$ in the barycenter computation

In the metabolite identification experiments we computed the barycenter only using the 5 greatest ones. In the following experiments, we show that, beyond the considerable computational interest, this approximation is also statistically beneficial on this dataset. We compute the test Top-k accuracies by changing the number of kept $_i$. From Figure 6, it seems that the best number of kept $_i(x)$ seems to be around 10.

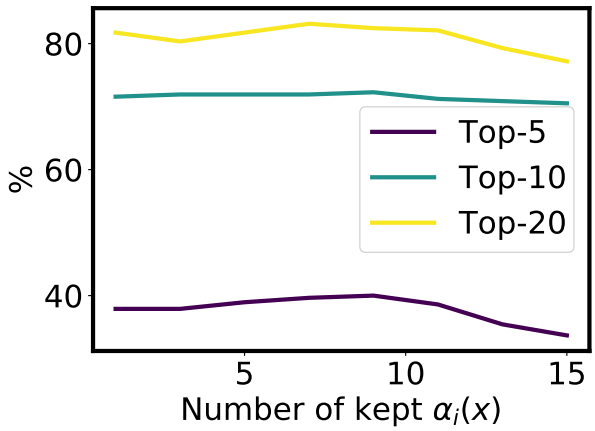

Figure 6. Top-k accuracies of $f_\theta(x)$ using a varying number of kept $\alpha_i(x)$.