Interferometric Graph Transform for Community Labeling

Nathan Grinsztajn, Louis Leconte, Philippe Preux, Edouard Oyallon

Interferometric Graph Transform for Community Labeling

Nathan Grinsztajn*
Inria, Univ. Lille, CNRS
Lille, France
nathan.grinsztajn@inria.fr

Louis Leconte*
LIP6, Sorbonne University
CMAP, Ecole Polytechnique, France
louis.leconte@ens-paris-saclay.fr

Philippe Preux
Inria, Univ. Lille, CNRS
Lille, France

Edouard Oyallon
CNRS, LIP6, Sorbonne University
Paris, France

Abstract

We present a new approach for learning unsupervised node representations in community graphs. We significantly extend the Interferometric Graph Transform (IGT) to community labeling: this non-linear operator iteratively extracts features that take advantage of the graph topology through demodulation operations. An unsupervised feature extraction step cascades modulus non-linearity with linear operators that aim at building relevant invariants for community labeling. Via a simplified model, we show that the IGT concentrates around the E-IGT: those two representations are related through some ergodicity properties. Experiments on community labeling tasks show that this unsupervised representation achieves performances at the level of the state of the art on the standard and challenging datasets Cora, Citeseer, Pubmed and WikiCS.

1 Introduction

Graph Convolutional Networks (GCNs) [25] are now the state of the art for solving many supervised (using labeled nodes) and semi-supervised (using unlabeled nodes during training) graph tasks, such as nodes or community labeling. They consist in a cascade of layers that progressively average node representations, while maintaining discriminative properties through supervision. In this work, we are mainly interested in the principles that allow such models to outperform other baselines: we propose a specific class of GCNs, which is unsupervised, interpretable, with several theoretical guarantees while obtaining good accuracies on standard datasets.

One of the reasons why GCNs lack interpretability is because no training objective is assigned to a specific layer except the final one: end-to-end training makes their analysis difficult [34]. They also tend to oversmooth graph representations [47], because applying successively an averaging operator leads to smoother representations. Also, the reason of their success is in general unclear [27]. In this work, we propose to introduce a novel architecture which, by design, will address those issues. Our model can be interpreted through the lens of Stochastic Block Models (SBMs) [19] which are standard, yet are not originally designed to analyze graph attributes through representation learning.

For example, several works [23, 1] prove that a Laplacian matrix concentrates around a low-rank expected Laplacian matrix, via simplified models like a SBM [10]. In the context of community detection, it is natural to assume that the intra-class, inter-class connectivity and feature distributions

*Equal contributions.

Preprint. Under review.

of a random graph are ruled by an SBM. To our knowledge, this work is the first to make a clear connection with those unsupervised models and the self-supervised deep GCNs which solve datasets like Cora, Citeseer, Pubmed, or WikiCS.

Our model is driven by ideas from the Graph Signal Processing [18] community and based on the Interferometric Graph Transform [35], a class of models mainly inspired by the (Euclidean) Scattering Transform [30]. The IGT aims at learning unsupervised (not using node labels at the representation learning stage), self-supervised representations that correspond to a cascade of isometric layer and modulus non-linearity, whose goal is to obtain a form of demodulation [36] that will lead to smoother but discriminative representation, in the particular case of community labeling. Smooth means here, by analogy with Signal Processing [29], that the signal is in the low-frequency domain, which corresponds to a quite lower dimensional space if the spectral decay is fast enough: this is for instance the case with a standard Laplacian [16] or a low-rank SBM adjacency matrix [28]. Here, the degree of invariance of a given representation is thus characterized by the smoothness of the signal.

Our main contribution is to introduce a simplified framework that allows to analyze node labeling tasks based on a non-linear model, via concentration bounds and which is numerically validated. Our other contributions are as follows. First, we introduce a novel graph representation for community labeling, which doesn't involve community labels. It consists in a cascade of linear isometry, band-pass filtering, pointwise absolute value non-linearity. We refer to it as an Interferometric Graph Transform (IGT) (for community labeling), and we show that under standard assumptions on the graph of our interest, a single realization of our representation concentrates around the Expected Interferometric Graph Transform (E-IGT), which can be defined at the node level without incorporating any graph knowledge. We also introduce a novel notion of localized low-pass filter, whose invariance can be adjusted to a specific task. Second, we study the behavior of this representation under an SBM model: with our model and thanks to the structure of the IGT, we are able to demonstrate theoretically that IGT features accumulate around the corresponding E-IGT. We further show that the architecture design of IGTs allows to outperform GCNs in a synthetic setting, which is consistent with our theoretical findings. Finally, we show that this semi-supervised and unsupervised representation is numerically competitive with supervised representations on standard community labeling datasets like Cora, Citeseer, Pubmed and WikiCS.

Our paper is organized as follows. First, we define the IGT in Sec. 3.1 and study its basic properties. Sec. 3.2 defines the E-IGT and bounds its distance from the IGT. Then, we discuss our model in the context of a SBM in Sec. 3.3 and we explain our optimization procedure in Sec. 3.4. Finally, Sec. 4 corresponds to our numerical results. Our source can be found at https://github.com/nathangrinsztajn/igt-community-detection and all proofs of our results can be found in the Appendix.

2 Related Work

We now discuss a very related line of work, namely the IGT [35], which takes source in several conceptual ideas from the Scattering Transform [30]. Both consist in a cascade of unitary transform, absolute value non-linearity and linear averaging, except that the Euclidean structure is neatly exploited via Wavelets Transforms for complex classification tasks in the case of the standard Scattering Transform [5, 37, 2, 36], whereas this structure is implicitly used in the case of IGT. In particular, similarly to a Scattering Transform, an IGT aims at projecting the feature representation in a lower dimensional space (low-frequency space) while being discriminative: the main principle is to employ linear operators, which combined with a modulus non-linearity, leads to a demodulation effect. In our case however, this linear operator is learned. The IGT for community labeling is rather different from standard IGT: first, [35] is not amenable to node labeling because it doesn't preserve node localization, contrary to ours. Second, we do not rely on the Laplacian spectrum explicitely contrary to [12, 35]. Third, the community experiments of [12, 35] are rather the classification of a diffusion process than a node labeling task. This is also similar to the Expected Scattering Transform [31], yet it is applied in a rather different context for reducing data variance, in order to shed lights on standard Deep Neural Networks. Our E-IGT and the Expected-Scattering have a very close architecture, however the linear operators are obtained with rather different criteria (e.g., ours are obtained from a concave procedure rather than convex) and goals (e.g., preserving energy, whereas we try to reduce it). Note however there is no equivalent of the E-IGT for other context that community detection or labeling, which is another major difference with [35]. In addition, our Prop. 3 is new compared to similar results of [31]. Thus while having similar architectures, those works have quite different outcomes and objectives.

Another line of works corresponds to the Graph Scattering Transform [12, 13, 21], which proposes to employ a cascade of Wavelet Transforms that respects the graph structure [18]. Yet, the principles that allow good generalization of those representations are unclear and they have only been tested until now on small datasets. Furthermore, this paper extends all those works by proposing an architecture and theoretical principles which are specific to the task of community labeling. A last related line of work corresponds to the hybrid Scattering-GCNs [33], which combines a GCN with the inner representation of a Scattering Transform on Graphs, yet they employ massive supervision to refine the weights of their architecture, which we do not do.

The architecture of an IGT model for community labeling takes also inspiration from Graph Convolutional Networks (GCNs) [25, 4]. They are a cascade of linear operators and ReLU non-linearities whose each layer is locally averaged along local nodes. Due to this averaging, GCNs exhibit two undesirable properties: first, the oversmoothing phenomenon [27], which makes learning of high-frequencies features difficult; second, the training of deeper GCNs is harder [20] because much information has been discarded by those averaging steps. Other types of Graph Neural Networks succeeded in approximating message-passing methods [9], or have worked on the spatial domain such as Spectral GCNs [6], and Chebynet [11]. In our work, we solely use a well chosen averaging for separating high-frequencies and low-frequencies without using any other extra-structure, which makes our method more generic than those approaches, without using supervision at all.

We further note that theoretical works often address the problem of estimating the expected Laplacian under SBM assumptions [23, 1, 26]. However up to our knowledge, none of those works is applied in a semi-supervised context and they aim at discovering communities rather than estimating communities from a small subset of labels. Moreover, the model remains mostly linear (e.g. based on the spectrum of the adjacency matrix). Here, our representation is non-linear and amenable for learning with a supervised classifier. We also note that several theoretical results have allowed to obtain approximation or stability guarantees for GCNs [41, 5, 22]: our work follows those lines and analyzes a specific type of GCN through the lens of Graph Signal Processing theory [18].

3 Framework

Notations. For a matrix X, we write $\|X\|^2 = \text{Tr}(X^T X) = \sum_{i,j} X_{i,j}^2$ its Frobenius-norm and for an operator L (acting on X), we might consider the related operator norm $\|L\| \triangleq \sup_{\|X\| \leq 1} \|LX\|$. The norm of the concatenation $\{B, C\}$ of two operators B, C is $\|\{B, C\}\|^2 = \|B\|^2 + \|C\|^2$ and this definition can be extended naturally to more than two operators. Note also that we use a different calligraphy between quantities related to the graph (e.g., adjacency matrix $\boldsymbol{\mathcal{A}}$) and operators (e.g., averaging matrix A). We write $A \preccurlyeq B$ if $B - A$ is a symmetric positive matrix. Here, $a_n \sim b_n$ means that $\exists \alpha > 0, \beta > 0 : \alpha |a_n| \leq |b_n| \leq \beta |b_n|$ and $a_n = \mathcal{O}(b_n)$ means $\exists \alpha > 0 : |a_n| \leq \alpha |b_n|$.

3.1 Definition of IGT

Our initial graph data are node features $X \in \mathbb{R}^{n \times P}$ obtained from a graph with n nodes and unormalized adjacency matrix $\boldsymbol{\mathcal{A}}$. We then write $\boldsymbol{\mathcal{A}}_{\text{norm}}$ the normalized adjacency matrix with self-connexion, as introduced by [25]. We note that $\boldsymbol{\mathcal{A}}_{\text{norm}}$ satisfies $0 \preccurlyeq \boldsymbol{\mathcal{A}}_{\text{norm}} \preccurlyeq I$ and has positive entries. In Graph Signal Processing [18], those properties allow to interpret $\boldsymbol{\mathcal{A}}_{\text{norm}}$ as an averaging operator. It means that applying $\boldsymbol{\mathcal{A}}_{\text{norm}}$ to X leads to a linear representation $\boldsymbol{\mathcal{A}}_{\text{norm}} X$ which is smoother than X because $\boldsymbol{\mathcal{A}}_{\text{norm}}$ projects the data in a subspace ruled by the topology (or connectivity) of a given community [12]. The degree of smoothness can be adjusted to a given task simply by considering:

$$A_J \triangleq \boldsymbol{\mathcal{A}}_{\text{norm}}^J \,. \tag{1}$$

This step is analogeous to the rescaling of a low-pass filter in Signal Processing [29], and A_J satisfies:

Lemma 1. *If $0 \preccurlyeq \boldsymbol{\mathcal{A}}_{norm} \preccurlyeq I$ and $\boldsymbol{\mathcal{A}}_{norm}$ has positive entries, then for any $J \in \mathbb{N}$, A_J has positive entry and satisfies also $0 \preccurlyeq A_J \preccurlyeq I$.*

Applying solely A_J leads to a loss of information that we propose to recover via $I - A_J$. This allows to separate low and high-frequencies of the graph in two channels, as expressed by the next lemma:

Lemma 2. *If $0 \preccurlyeq A \preccurlyeq I$, then $\|AX\|^2 + \|(I - A)X\|^2 \leq \|X\|^2$ with equality iff $A^2 = A$.*

Yet, contrary to $A_J X$, $(I - A_J)X$ is not smooth and thus, it might not be amenable for learning because community structures might not be preserved. Furthermore, a linear classifier will not be

sensitive to the linear representation $\{A_J X, (I - A_J)X\}$. Similarly to [35], we propose to apply an absolute value $|.|$ point-wise non-linearity to our representations. Section 3.4 will explain how to estimate isometries $\{W_n\}$, which combined with a modulus, will smooth the signal envelope while preserving signal energy. We now formally describe our architecture and we consider $\{W_n\}$ a collection of isometries, that we progressively apply to an input signal representation $U_0 \triangleq X$ via:

$$U_{n+1} \triangleq |(I - A_J)U_n W_n| \,, \tag{2}$$

and we introduce the IGT representation of order $N \in \mathbb{N}$ with averaging scale $J \in \mathbb{N}$ defined by:

$$S_J^N X \triangleq \{A_J U_0, ..., A_J U_N\} \,. \tag{3}$$

Fig. 1 depicts our architecture. The following explains that S_J^N is non-expansive, thus stable to noise:

Proposition 1. *For $N \in \mathbb{N}$, $S_J^N X$ is 1-Lipschitz leading to:*

$$\|S_J^N X - S_J^N Y\| \leq \|X - Y\| \text{ and, } \|S_J^N X\| \leq \|X\| \,. \tag{4}$$

The next section will describe the E-IGT, which was introduced as the Expected Scattering [31], but in a rather different context: we will show under simplifying assumptions that an IGT for community labeling concentrates around the E-IGT.

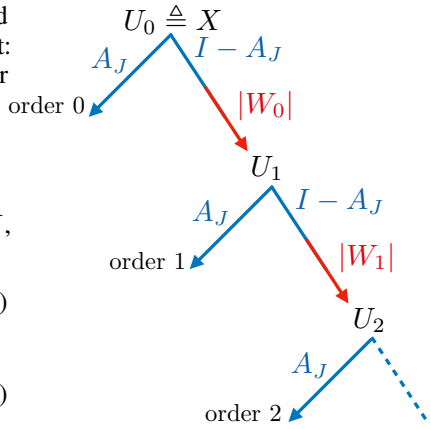

Figure 1: We illustrate our model for $N = 2$. Low and high frequencies are separated (blue) and then the high frequencies are demodulated (red) via an isometry and a non-linear point wise absolute value, and then propagated to the next layer.

3.2 Definition of the Expected-IGT (E-IGT)

Similarly to the previous section, for an input signal $\bar{U}_0 \triangleq X$, we consider the following recursion, introduced in [31]:

$$\bar{U}_{n+1} \triangleq |(\bar{U}_n - \mathbb{E}\bar{U}_n)W_n| \,, \tag{5}$$

which leads to the E-IGT [2] of order N defined by:

$$\bar{S}_N \triangleq \{\mathbb{E}\bar{U}_0, ..., \mathbb{E}\bar{U}_N\} \,. \tag{6}$$

Similarly to Prop. 7, we prove the following stability result:

Proposition 2. *For $N \in \mathbb{N}$, $\bar{S}^N X$ is 1-Lipschitz, meaning that:*

$$\|\bar{S}^N X - \bar{S}^N Y\|^2 \leq \mathbb{E}[\|X - Y\|^2] \,, \tag{7}$$

and furthermore:

$$\|\bar{S}^N X\|^2 \leq \mathbb{E}[\|X\|^2] \,. \tag{8}$$

Proof. Indeed, [31] have proven this for the columns of X. □

Note that this represention is also more amenable to standard supervised classifiers such as SVMs because no operation mixing nodes is involved. Prop. 2 highlights the fact that the E-IGT is non-expansive, and [46] shows that this allows to discriminate the attributes of the distribution of X. However, it is difficult in general to estimate the E-IGT because one does not know the distribution of a given node and it is difficult to estimate it from a single realization as there is a clear curse of dimensionality. However, we will show that S_J^N will be very similar to $\bar{S}^N$ under standard assumptions on communities. We now state the following proposition, which allows to quantify the distance between an IGT and its E-IGT:

Proposition 3. *For any X, N, J, we get:*

$$\|S_J^N X - \bar{S}^N X\| \leq \sqrt{2} \sum_{m=0}^{N} \|(A_J - \mathbb{E})\bar{U}_m\| \,. \tag{9}$$

The proof of this proposition can be found in the Appendix: it fully uses the tree structure of Fig. 1, in order to obtain tighter bounds than [31], as it allows N to be of arbitrary size without diverging. We now bound the distance between the IGT and the E-IGT:

[2] We rename it here because we use rather different principles to obtain the $\{W_0..., W_{N-1}\}$ compared to the original Scattering.

Corollary 1. *For $N \in \mathbb{N}$, we have:*

$$\sup_{\mathbb{E}\|X\|\leq 1} \mathbb{E}[\|S_J^N X - \bar{S}^N X\|] \leq 2^{N+2} \sup_{\mathbb{E}\|X\|\leq 1} \mathbb{E}[\|A_J X - \mathbb{E}X\|]\,. \tag{10}$$

Proof. The next Lemma combined with the norm homogeneity allows to conclude with Prop. 3. □

Lemma 3. *If $\|X\| \leq 1$, then $\|\bar{U}_n\| \leq 2^n$, with $\bar{U}_0 = X$. Also, if $\mathbb{E}[\|X\|] \leq 1$, then $\mathbb{E}[\|\bar{U}_n\|] \leq 2^n$*

Proof. This is true for $n = 0$, and then by induction, since isometry preserves the ℓ^2-norm: $\|\bar{U}_{n+1}\| \leq \|\bar{U}_n\| + \|\mathbb{E}\bar{U}_n\| \leq \|\bar{U}_n\| + \mathbb{E}\|\bar{U}_n\| \leq 2^{n+1}$. The proof is similar for the second part. □

The right term of Eq. 10 measures the ergodicity properties of a given A_J. For instance, in the case of images, a stationary assumption on X implies that $A_J f(X) \approx \mathbb{E}f(X)$ for all measurable f, which is the case for instance for textures [29]. The following proposition shows that in case of exact ergodicity, the two representations have bounded moments of order 2:

Proposition 4. *If $\mathbb{E}[A_J X] = \mathbb{E}X$, and if X has variance $\sigma^2 = \mathbb{E}\|X\|^2 - \|\mathbb{E}X\|^2$, then:*

$$\mathbb{E}[\|S_J^N X - \bar{S}^N X\|^2] \leq 2\sigma^2\,. \tag{11}$$

3.3 Graph model and concentration bounds

In this subsection, we propose to demonstrate novel bounds which improve the upper bound obtained at Prop. 1 by introducing a Stochastic Block Model [19]. We will show that the IGT features of a given community concentrates around the E-IGT feature of this community: IGT features are thus more amenable to be linearly separable. Recall from Sec. 3.1 that $A_1 = \boldsymbol{\mathcal{A}}_{\text{norm}}$, thus we note that for some $m > 0$, via the triangular inequality we get:

$$\|A_1\bar{U}_m - \mathbb{E}\bar{U}_m\| = \|\boldsymbol{\mathcal{A}}_{\text{norm}}\bar{U}_m - \mathbb{E}\bar{U}_m\| \leq \|(\boldsymbol{\mathcal{A}}_{\text{norm}} - \mathbb{E}[\boldsymbol{\mathcal{A}}]_{\text{norm}})\bar{U}_m\| + \|\mathbb{E}[\boldsymbol{\mathcal{A}}]_{\text{norm}}\bar{U}_m - \mathbb{E}[\bar{U}_m]\|\,.$$

Now, the left term can be upper bounded as:

$$\|(\boldsymbol{\mathcal{A}}_{\text{norm}} - \mathbb{E}[\boldsymbol{\mathcal{A}}_{\text{norm}}])\bar{U}_m\| \leq \|\boldsymbol{\mathcal{A}}_{\text{norm}} - \mathbb{E}[\boldsymbol{\mathcal{A}}_{\text{norm}}]\|\|\bar{U}_m\|\,. \tag{12}$$

For the sake of simplicity, we will consider a model with two communities, yet the extension to more communities is straightforward and would simply involve a linear term in the number of communities. We now describe our model. Once the n nodes have been split in two groups of size $n \sim n_1, n \sim n_2$, we assume that each edge between two different nodes is sampled independently with probability p_n (or simply p if not ambiguous) if they belong to the same community and q otherwise. We assume that $q = \tau p$ for some constant $\tau \sim \frac{1}{\sqrt{n}} \ll 1$ and the features belonging to the same community are i.i.d. and σ-sub-Gaussian, and $\|X\| \leq 1$. Those assumptions are not restrictive as they hold in many practical applications (and the second, always holds up to a constant). For a given community $i \in \{1, 2\}$, we write $(\mu_m^i)_{m\leq N}$ its E-IGT. We impose that $p_n \sim \frac{\log(n)}{n}$ in this particular Bernoulli model. Sparse random graphs do not generally concentrate. Yet, according to [23], in the relatively sparse case where $p_n \sim \frac{\log n}{n}$, we get the following spectral concentration bound of the normalized adjacency matrix:

Lemma 4. *Let $\boldsymbol{\mathcal{A}}$ be a symmetric matrix with independent entries $\boldsymbol{\mathcal{A}}_{ij}$ obtained as above. If $n_1 \sim n, n_2 \sim n$, and p is relatively sparse as above, then for all $\nu > 0$, there is a constant C_ν such that, with high probability $\geq 1 - n^{-\nu}$:*

$$\|\boldsymbol{\mathcal{A}}_{norm} - \mathbb{E}[\boldsymbol{\mathcal{A}}]_{norm}\| \leq \frac{C_\nu}{\sqrt{\log n}}\,. \tag{13}$$

Proof. Can be found in [23]. □

Note that in general, $\mathbb{E}[\boldsymbol{\mathcal{A}}]_{\text{norm}} \neq \mathbb{E}[\boldsymbol{\mathcal{A}}_{\text{norm}}]$ and here, because of our model:

$$\mathbb{E}[\boldsymbol{\mathcal{A}}]_{\text{norm}} = \begin{bmatrix} \frac{p}{n_1 p + n_2 q}\mathbf{1}_{n_1\times n_1} & \frac{q}{n_1 p + n_2 q}\mathbf{1}_{n_1\times n_2} \\ \frac{q}{n_1 q + n_2 p}\mathbf{1}_{n_2\times n_1} & \frac{p}{n_1 q + n_2 p}\mathbf{1}_{n_2\times n_2} \end{bmatrix}\,, \tag{14}$$

where $\mathbf{1}_{m\times n}$ is a matrix of ones of size $m \times n$. Now, note also that:

$$\mathbb{E}[\bar{U}_m] = [\mu_m^1 \mathbf{1}_{n_1}^T, \mu_m^2 \mathbf{1}_{n_2}^T], \tag{15}$$

Now, we prove that the IGT will concentrate around the E-IGT, under a Stochastic Block Model and sub-Gaussianity assumptions. We note that a bias term of the order of $\sqrt{n}\tau$ is present, which is consistent with our model assumptions. Note it is also possible to leverage the boundedness assumption yet it will lead to an additional constant term.

Proposition 5. *Under the assumptions above, there exists $C > 1$ s.t. for all $N > 0, \delta > 0$, we have with high probability, larger than $1 - \mathcal{O}(N\delta + n^{-\nu})$:*

$$\|S_1^N X - \bar{S}^N X\| = \mathcal{O}(\sigma \frac{1 + C^N}{1 - C}(\sqrt{\ln \frac{1}{\delta}} + \frac{1}{\sqrt{\log n}})) + \mathcal{O}(\tau\sqrt{n} \sum_{m \leq N} \|\mu_m^2 - \mu_m^1\|). \tag{16}$$

The following proposition allows to estimate the concentration of each IGT order:

Proposition 6. *Assume that each line of $X \in \mathbb{R}^{n\times P}$ is σ-sub-Gaussian. There exists $C > 1, K > 0, C' > 1$ such that $\forall m, \delta > 0$ with probability $1 - 8P\delta$, we have:*

$$\|\mathbb{E}[\boldsymbol{\mathcal{A}}]_{norm}\bar{U}_m - \mathbb{E}[\bar{U}_m]\| \leq K\sigma C^m \sqrt{\ln \frac{1}{\delta}} + C'\sqrt{n}\tau \|\mu_m^2 - \mu_m^1\|. \tag{17}$$

This Lemma shows that a cascade of IGT linear isometries preserves sub-Gaussianity:

Lemma 5. *If each line of X is σ-sub-Gaussian, then each (independent) line of $\bar{U}_m$ is $C^m\sigma$-sub-Gaussian for some universal constant C.*

In order to show the previous Lemma, we need to demonstrate that the modulus of a sub-Gaussian variable is itself sub-Gaussian, which is shown below:

Lemma 6. *There is $C > 0$, s.t. $X \in \mathbb{R}^P$ is σ-sub-Gaussian, then $|(X - \mathbb{E}X)W|$ is $C\sigma$-sub-Gaussian.*

3.4 Optimization procedure

We now describe the optimization procedure of each of our operators $\{W_n\}$, that consists in a greedy layer-wise procedure [3]. Our goal is to specify $|W_n|$ such that it leads to a demodulation effect, as well as to have a fast energy decay. Demodulation means that the envelope of a signal should be smoother, whereas fast decay will allow the use of shallower networks. In practice, it means that at depth n, the energy along the direction of averaging should be maximized, which leads to consider:

$$\max_{W^T W = I} \||A_J|(I - A_J)U_n W|\|. \tag{18}$$

As observed in [35], because the extremal points of the ℓ^2 ball are the norm preserving matrix, this optimization problem is equivalent to:

$$\max_{\|W\|_2 \leq 1} \||A_J|(I - A_J)U_n W|\|. \tag{19}$$

Note that this can be approximatively solved via a projected gradient procedure which projects the operator W on the unit ball for the ℓ^2-norm at each iteration. Furthermore, contrary to [35], we might constrain W to have a rank lower than the ambient space, that we denote by k: increasing k as well as the order N allows to potentially increase the capacity of our model, yet we as discussed in the next section, this wasn't necessary to obtain accuracies at the level of the state of the art.

4 Numerical Experiments

We test our unsupervised IGT features on a synthetic example, and on challenging semi-supervised tasks, in various settings that appeared in the graph community labeling litterature: the **full** [40], **predefined** [25] and **random splits** [25] of Cora, Citeseer, Pubmed, as well as the WikiCS dataset.

4.1 Synthetic example

As GCNs progressivly apply a smoothing operator on subsequent layers, deeper features are less sensitive to intra-community variability. This progressive projection can have a big impact on datasets where discriminative features are close in average, yet have very different distributions over several communities. In order to underline this phenomenon, we propose to study the following synthetic example: following the model and notations of Sec. 3.3, we consider two communities, with an equal number of samples in each and we assume that $P = 1$, $J = 1$, $p = 0.001$ and $q = 0$ and $n = 10000$. Here, we assume the features are centered Gaussians with variance $\sigma_1 = 1$ for the first community and $\sigma_2 = \sigma_1 + \Delta\sigma$ for the second. In other words, $\Delta\sigma$ controls the relative spread of the community features. Our goal is to show numerically that an IGT representation is, by construction, more amenable to distinguish the two communities than a GCN.

As a training set, we randomly sample 20 nodes and use the remaining ones as a validation and test set. For IGT parameters, we pick $J = 2$, $k = 1$ and $N \in \{0, 1, 2\}$. On top of our standardized IGT features, our classification layer consists in a 1-hidden layer MLP of width 128. We train the IGT operators for 50 epochs. We compare our representation with a standard GCN [25] that has two hidden layers and a hidden dimension of 128 for the sake of comparison. Both supervised architectures are trained during 200 epochs using Adam [24] with a learning rate of 0.01 ($\epsilon = 10^{-8}, \beta_1 = 0.9, \beta_2 = 0.999$). We discuss next the accuracies averaged over 5 different seeds on Fig. 2 for various representations and values of $\Delta\sigma$.

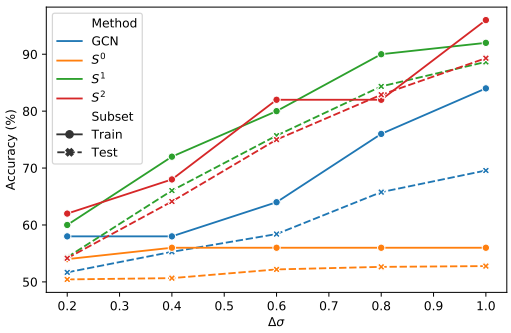

Figure 2: Accuracies of GCN against our method on a synthetic example, for several values of $\Delta\sigma$.

We observe that an order 0 IGT performs poorly for any values of $\Delta\sigma$, which is consistent because the linear smoothing will dilute important informations for node classification. However, non-linear model like IGT (of order larger than 0) or GCN outperforms this linear representation. The IGT outperforms the GCN for all values of $\Delta\sigma$ because as Sec. 3.1 shows, by construction, this representation extracts explicitly the high frequency of the graph, whereas a GCN can only smooth its features and thus will tend to lose in discriminability despite supervision. We note that orders 1 and 2 perform similarly, which is not surprising given the simplistic assumption of this model: all the informative variability is contained in the order 1 IGT and the order 2 is likely to only bring noisy features in this specific case.

4.2 Supervised community detection

First, we describe our experimental protocol on the datasets Cora, CiteSeer and PubMed. Each dataset consists in a set of bag-of-words vectors with citation links between documents. They are made of respectively 5.4k, 4.7k, 44k and 216k edges with features of size respectively 1.4k, 3.7k, 0.5k and 0.3k. For the three first datasets, we test our method in three semi-supervised settings, which consist in three different approaches to split the dataset into train, validation and test sets: at this stage, we would like to highlight we are one of the few methods to try its architecture on those three splits (which we discuss for clarity), which allows to estimate the robustness to various sample complexity. Each accuracy is averaged over 5 runs and we report the standard-deviation in the Appendix. The most standard split is the **predefined split** setting: each training set is provided by [25] and consist in 20 training nodes per class, which represent a fraction 0.052, 0.036, and 0.003 of the data for Cora, CiteSeer and PubMed respectively. 500 and 1000 nodes are respectively used as a validation and test set. Then, we consider the **random split** setting introduced in [25], which is exactly as above except that we randomly extract 5 splits of the data, and we average the accuracies among those splits. Finally, we consider the **full split** setting which was used in [40] and employs 5 random splits of a larger training set: a fraction 0.45, 0.54 and 0.92 of the whole labeled datasets respectively. Note that each of those tasks is transductive yet our method would require minimal adaptation to fit an inductive pipeline. For WikiCS, we followed the only guideline of [32].

Our architectures are designed as follow: an IGT representation only requires 4 hyper-parameters: an adjacency matrix $\mathcal{A}$, an output-size k for each linear isometry, a smoothness parameter J and an IGT order N. Given that the graphs are undirected, $\mathcal{A}$ satisfies the assumption described in Sec. 3.1, yet it would be possible to symmetrize the adjacency matrix of a directed graph. This corresponds to our unsupervised graph representation that will be then fed to a supervised classifier. Sec. 3.3 shows that our IGT representation should concentrate around the E-IGT of their respective community, which means that they should be well separated by a Linear classifier. However, there might be more intra-class variability than the one studied from the lens of our model, thus we decided to use potentially deeper models, e.g., Multi Layer Perceptrons (MLPs) as well as Linear classifiers. We use the same fixed MLP architecture for every dataset: a single hidden layer with 128 features. Our linear model is simply a fully connected layer, and each model is fed to a cross-entropy loss. We note that our MLP is shallow, with few units, and does not involve the graph structure by contrast to semi-supervised GCNs: we thus refer to the combination of IGT and a MLP or a Linear layer as an unsupervised graph representation for node labeling. Note also that a MLP is a scalable classifier in the context of graphs: once the IGT representation is estimated, one can learn the weight of the MLP by splitting the training set in batches, contrary to standard GCNs.

We now describe our training procedure as well as the regularizaton that we incorporated: it was identical for any splits of the data. We optimized our pipeline on Cora and applied it on Citeseer, Pubmed and WikiCS, unless otherwise stated. Each parameter was cross-validated on a validation set, and we report the test accuracy on a test set that was not used until the end. First, we learn each $\{W_m\}_{m \leq N}$ via Adam for 50 epochs and a learning rate of 0.01. Once computed, the IGT features are normalized and are fed to our supervised classifier, that we train again using Adam and a learning rate of 0.01 for at most 200 epochs, with a early stopping procedure and a patience of 30. A dropout ratio which belongs to $\{0, 0.2, 0.4, 0.5, 0.6, 0.8\}$ is potentially incorporated to the one hidden layer of the MLP. On CiteSeer and PubMed, our procedure selected 0.2, on WikiCS 0.8, whereas no-dropout was added on Cora. Furthermore, we incorporated an ℓ^2-regularization with our linear layer which we tried amongst $\{0, 0.001, 0.005, 0.01\}$: we picked 0.005 via cross-validation. We discuss here WikiCS: by cross-validation, we used $J = 1, N = 1, k = 150$ for the linear experiment and $J = 2, N = 1, k = 35$ for the MLP experiment. For the other datasets and every splits, we used $N = 2$ and $k = 10$: we note that less capacity is needed compared to WikiCS, because those datasets are simpler. For the three other datasets, for both the **predefined** and **random splits**, we fix $J = 4$. For the **full split**, we used $J = 1$ for each dataset: we noticed that increasing J degrades the performance, likely because less invariance is required and can be learned from the data, because more samples are available. This makes sense, as the amount of smoothing depends on the variability exhibited by the data. Thanks to the amount of available data, the supervised classifier can estimate the degree of invariance needed for the classification task, which was not possible if using only 20 samples per community.

Tab. 4 reports the semi-supervised accuracy for each dataset, in various settings, and compares standard supervised [14, 25, 40, 8, 7, 42] and unsupervised [38, 43, 15, 39, 17] architectures. Note that each supervised model is trained in an end-to-end manner. The unsupervised models are built differently and we discuss them now briefly: for instance, EP [15], uses a node embedding with a rather different architecture from GCNs. Also, DeepWalk [38] is analogous to a random walk, GraphSage [17] learns an embedding with a local criterion, DGI [43] relies on a mutual information criterion and finally [39] relies on a random field model. Note that each of those models are significantly different from ours and they do not have the same theoretical foundations and properties as ours. As expected, accuracy in the **full** setting is higher than the others. We observe that in general, supervised models outperforms unsupervised models by a large margin except on WikiCS and Citeseer for the **random** and **predefined** splits, for which an IGT obtains better accuracy: it indicates that it has a better inductive bias for this dataset. Note that an IGT obtains competitive accuracies amongst unsupervised representations and this is consistent with the fact that those datasets, discussed above, are likely to satisfy the hypothesis described in Sec. 3.3. In general, a MLP outperforms a linear layer (because it has better approximation properties), except on Citeseer for which the accuracy is similar, which seems to validate that the data of Citeseer follow the model that we introduced in 3.3 on Citeseer, that leads to linear separability.

4.3 Ablation experiments

Table 1: Classification accuracies (in %) for each splits of Cora, Citeseer, Pubmed as well as WikiCS.

Method/Dataset	Cora			Citeseer			Pubmed			WikiCS
	Full	Rand	Pred	Full	Rand	Pred	Full	Rand	Pred	
Supervised										
GAT [42]			83.0			72.5			79.0	77.2
GCN [25]		80.1	81.5		67.9	70.3		**78.9**	79.0	**77.7**
Graph U-Net [14]			**84.4**			**73.2**			**79.6**	
DropEdge [40]	**88.2**			**80.5**			**91.7**			
FastGCN [8]	85.0			77.6			88.0			
OS [7]		**82.3**			**69.7**			77.4		
Unsupervised										
Raw [43, 32]			47.9			49.3			69.1	72.0
DeepWalk [38]			67.2			43.2			65.3	74.4
IGT + MLP (ours)	**87.7**	**78.3**	80.3	**78.4**	67.6	**73.1**	**88.2**	76.2	76.4	**77.2**
IGT + Lin. (ours)	83.3	77.6	77.4	**78.4**	**73.0**	**73.1**	88.1	74.5	73.9	76.7
EP [15]		78.1			71.0			**79.6**		
GraphSage [17]	82.2			71.4			87.1			
DGI [43]			82.3			71.8			76.8	75.4
GMNN [39]			**82.8**			71.5			**81.6**	

Table 2: Linear classification accuracies (in %) for the **predefined split** on Cora's *validation set*, for various values of N, J.

J \ N	0	1	2	3
1	62.4	60.8	62.8	61.4
2	68.6	70.6	72.2	68.6
3	71.4	72.2	**74.6**	72.6
4	72.4	73.2	**74.6**	73.0

In order to understand better the IGT representation, we propose to study the accuracy of an IGT representation on Cora's validation set, as a function of the scale J and the IGT order N. For the sake of simplicity, we consider a linear classifier. Each linear operator is learned with 40 epochs. We picked $k = 10$ and train our basic model for 200 epochs with SGD, the validation accuracies are reported in Tab. 2. As N, J increase, we feed the features to a linear classifier: in general, for $0 \leq N \leq 2$, as N grows the accuracy improves. However, the order 3 IGT decreases the accuracy: this is consistent because it conveys a noise which is amplified by the standardization. As J increases, we smooth our IGT features on more neighbor nodes, which results in better performances for a fixed order N, and is also consistent with the finding of Sec. 3.1.

We performed a second ablation experiment in order to test the inductive bias of our architecture: we considered random $\{W_n\}$ at evaluation time and we obtained respectively on the full split of Cora, Citeseer and Pubmed some accuracy drops of respectively 6.3%, 5.2% and 5.6%. This is relatively smaller drops than DGI [43] which reports for instance some drops of about 10%: our architecture is likely to have a better inductive bias for this task.

5 Conclusion

In this work, we introduced the IGT which is an unsupervised and semi-supervised representation for community labeling. It consists in a cascade of linear isometries and point-wise absolute values. This representation is similar to a semi-supervised GCN, yet it is trained layer-wise, without using labels, and has strong theoretical fundations. Indeed, under a SBM assumption and a large graph hypothesis, we show that an IGT representation can discriminate communities of a graph from a single realization of this graph. It is numerically supported by a synthetic example based on Gaussian features, which shows that an IGT can estimate the community of a given node better than a GCN because it tends to alleviate the over-smoothing phenomenon. This is further supported by our numerical experiments on the standard, challenging datasets Cora, CiteSeer, PubMed and WikiCS: with shallow supervised classifiers, we obtain numerical accuracy which is competitive with semi-supervised approaches.

Future directions could be to either refine our theoretical analysis by weakening our assumptions, or to test our method on inductive tasks. Furthermore, following [35], one can also wonder if this type of approach could be extended to more complex data, in order to obtain stronger theoretical guarantees (e.g., manifold). Finally, future works could also be dedicated to scale our algorithms to very large graphs: this is a challenging task both in terms of memory and computations.

Broader impact. Graph Neural Networks can be used in many domains, like protein prediction, or network analysis to cite only a few, and could become even more prevalent tomorrow. Our work is thus included in a large literature whose societal impact and ethical considerations are to become more and more important. We provide here a new model aiming at learning unsupervised node representation in community graphs graphs. While its most natural application lies in community detection in social science, we hope that the provided theoretical guarantees could be used in the future to provide safer and more readable models toward more various directions.

Acknowledgements

EO, LL, NG would like to acknowledge the CIRM for its one week hospitality which was helpful to this project. EO acknowledges NVIDIA for its GPU donation and this work was granted access to the HPC resources of IDRIS under the allocation 2020-[AD011011216R1] made by GENCI. EO, LL were partly supported by ANR-19-CHIA "SCAI" and ANR-20-CHIA-0022-01 "VISA-DEEP". NG and PP would like to acknowledge the support of the French Ministry of Higher Education and Research, Inria, and the Hauts-de-France region; they also want to thank the Scool research group for providing a great research environment. The authors would like to thank Mathieu Andreux, Alberto Bietti, Edouard Leurent, Nicolas Keriven, Ahmed Mazari, Aladin Virmaux for helpful comments and suggestions.

References

[1] Emmanuel Abbe. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

[2] J. Andén, V. Lostanlen, and S. Mallat. Joint time-frequency scattering for audio classification. In *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2015.

[3] Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon. Greedy layerwise learning can scale to imagenet. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 583–593. PMLR, 2019.

[4] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[5] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[7] Deli Chen, Yankai Lin, Wei Li, Peng Li, Jie Zhou, and Xu Sun. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3438–3445, 2020.

[8] Jie Chen, Tengfei Ma, and Cao Xiao. Fastgcn: Fast learning with graph convolutional networks via importance sampling. In *International Conference on Learning Representations*, 2018.

[9] Zhengdao Chen, Lisha Li, and Joan Bruna. Supervised community detection with line graph neural networks. In *International Conference on Learning Representations*, 2018.

[10] Vincent Cohen-Addad, Adrian Kosowski, Frederik Mallmann-Trenn, and David Saulpic. On the power of louvain in the stochastic block model. *Advances in Neural Information Processing Systems*, 33, 2020.

[11] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375*, 2016.

[12] Fernando Gama, Joan Bruna, and Alejandro Ribeiro. Stability properties of graph neural networks. *IEEE Transactions on Signal Processing*, 68:5680–5695, 2020.

[13] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *International Conference on Machine Learning*, pages 2122–2131. PMLR, 2019.

[14] Hongyang Gao and Shuiwang Ji. Graph u-nets. In *international conference on machine learning*, pages 2083–2092. PMLR, 2019.

[15] Alberto García-Durán and Mathias Niepert. Learning graph representations with embedding propagation. *arXiv preprint arXiv:1710.03059*, 2017.

[16] Nathan Grinsztajn, Philippe Preux, and Edouard Oyallon. Low-rank projections of GCNs laplacian. In *ICLR 2021 Workshop on Geometrical and Topological Representation Learning*, 2021.

[17] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1025–1035, 2017.

[18] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, 2011.

[19] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

[20] Wenbing Huang, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. Tackling over-smoothing for general graph convolutional networks. *arXiv e-prints*, pages arXiv–2008, 2020.

[21] Vassilis N Ioannidis, Siheng Chen, and Georgios B Giannakis. Efficient and stable graph scattering transforms via pruning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[22] Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. *arXiv preprint arXiv:1905.04943*, 2019.

[23] Nicolas Keriven and Samuel Vaiter. Sparse and smooth: improved guarantees for spectral clustering in the dynamic stochastic block model. *arXiv preprint arXiv:2002.02892*, 2020.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[25] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[26] Can M Le, Elizaveta Levina, and Roman Vershynin. Concentration of random graphs and application to community detection. *arXiv preprint arXiv:1801.08724*, 2018.

[27] Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[28] Andreas Loukas and Pierre Vandergheynst. Spectrally approximating large graphs with smaller graphs. In *International Conference on Machine Learning*, pages 3237–3246. PMLR, 2018.

[29] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[30] Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

[31] Stéphane Mallat and Irene Waldspurger. Deep learning by scattering. *arXiv preprint arXiv:1306.5532*, 2013.

[32] Péter Mernyei and Cătălina Cangea. Wiki-cs: A wikipedia-based benchmark for graph neural networks. *arXiv preprint arXiv:2007.02901*, 2020.

[33] Yimeng Min, Frederik Wenkel, and Guy Wolf. Scattering gcn: Overcoming oversmoothness in graph convolutional networks. *arXiv preprint arXiv:2003.08414*, 2020.

[34] Edouard Oyallon. Building a regular decision boundary with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5106–5114, 2017.

[35] Edouard Oyallon. Interferometric graph transform: a deep unsupervised graph representation. In *International Conference on Machine Learning*, pages 7434–7444. PMLR, 2020.

[36] Edouard Oyallon, Eugene Belilovsky, Sergey Zagoruyko, and Michal Valko. Compressing the input for cnns with the first-order scattering transform. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 301–316, 2018.

[37] Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2865–2873, 2015.

[38] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[39] Meng Qu, Yoshua Bengio, and Jian Tang. Gmnn: Graph markov neural networks. In *International conference on machine learning*, pages 5241–5250. PMLR, 2019.

[40] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[41] Luana Ruiz, Zhiyang Wang, and Alejandro Ribeiro. Graph and graphon neural network stability. *arXiv preprint arXiv:2010.12529*, 2020.

[42] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[43] Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. In *ICLR (Poster)*, 2019.

[44] Roman Vershynin. *Concentration of Sums of Independent Random Variables*, page 11–37. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[45] Martin J. Wainwright. *Basic tail and concentration bounds*, page 21–57. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.

[46] Irène Waldspurger. Wavelet transform modulus: phase retrieval and scattering. *Journées équations aux dérivées partielles*, pages 1–10, 2017.

[47] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*, pages 40–48. PMLR, 2016.

6 Appendix

6.1 Proofs

Lemma 2. *If $0 \preccurlyeq A \preccurlyeq I$, then $\|AX\|^2 + \|(I-A)X\|^2 \leq \|X\|^2$ with equality iff $A^2 = A$.*

Proof. We note that for any x, we get:

$$\|Ax\|^2 + \|(I-A)x\|^2 = \|Ax\|^2 + \|x\|^2 + \|Ax\|^2 - 2\langle x, Ax\rangle \tag{20}$$

Yet, $\|Ax\|^2 = \langle x, A^T Ax\rangle \leq \langle x, Ax\rangle$ because $\mathrm{Sp}(A) \subset [0,1]$. Thus,

$$2(\|Ax\|^2 - \langle x, Ax\rangle) + \|x\|^2 \leq \|x\|^2 , \tag{21}$$

with equality $\forall x$ iff $A = A^2$. It is now enough to observe that $\{A, I-A\}$ inherits from those properties. □

The following proposition explains that our representation is non-expansive, and thus stable to noise:

Proposition 7. *For $N \in \mathbb{N}$, $S_J^N X$ is 1-Lipschitz leading to:*

$$\|S_J^N X - S_J^N Y\| \leq \|X - Y\| . \tag{22}$$

and furthermore:

$$\|S_J^N X\| \leq \|X\| . \tag{23}$$

Proof. For two feature matrices X, Y, let us consider U_i and $\tilde{U}_i$ defined from Equation (2), with $U_0 = X$ and $\tilde{U}_0 = Y$. Because $|W_i|$ is a contractive and from Lemma 2,

$$\|U_{i+1} - \tilde{U}_{i+1}\|^2 \leq \|U_i - \tilde{U}_i - A_J(U_i - \tilde{U}_i)\|^2 \tag{24}$$

$$\leq \|U_i - \tilde{U}_i\|^2 - \|A_J(U_i - \tilde{U}_i)\|^2 \tag{25}$$

Hence,

$$\sum_i^N \|A_J(U_i - \tilde{U}_i)\|^2 \leq \|X - Y\|^2 - \|U_n - \tilde{U}_n\|^2 \tag{26}$$

$$\leq \|X - Y\|^2 \tag{27}$$

Taking $X = 0$ leads to the second part as then $SX = 0$. □

This Lemma shows that a cascade of IGT linear isometries preserve sub-Gaussianity:

Lemma 5. *If each line of X is σ-sub-Gaussian, then each (independent) line of $\bar{U}_m$ is $C^m\sigma$-sub-Gaussian for some universal constant C.*

Proof. Apply the Lemma 6 with $W = W_n$ for $n \leq m$ leads to the result. □

In order to show the previous Lemma, we need to demonstrate that the modulus of a sub-Gaussian variable is itself sub-Gaussian, which is shown below:

Lemma 6. *If $X \in \mathbb{R}^P$ is σ-sub-Gaussian, then $|(X - \mathbb{E}X)W|$ is $C\sigma$-sub-Gaussian for some absolute value C.*

Proof. If X is σ-sub-Gaussian, then $X - \mathbb{E}X$ is $C'\sigma$-subGaussian by recentering [44]. We note that as W is unitary, thus $(X - \mathbb{E}X)W$ is also $C'\sigma$-subgaussian. Then, let $u \in \mathbb{R}^p$ an unit vector. We note that:

$$\mathbb{P}(\sum_{i=1}^p u_i |X_i| \geq t) \tag{28}$$

$$\leq \sum_{\epsilon_i \in \{-1,1\}} \mathbb{P}(\{\epsilon_i X_i \geq 0\} \cap \{\sum_i u_i |X_i| \geq t\}) \tag{29}$$

$$= \sum_{\epsilon_i \in \{-1,1\}} \mathbb{P}(\{\epsilon_i X_i \geq 0\} \cap \{\sum_i \epsilon_i u_i X_i \geq t\}) \tag{30}$$

$$\leq 2^p e^{-\frac{t^2}{2C'^2\sigma^2}} = e^{p\ln 2 - \frac{t^2}{2C'^2\sigma^2}} . \tag{31}$$

This leads to the conclusion by sub-Gaussian characterization. □

Proposition 3. *For any X, N, J, we get:*

$$\|S_J^N X - \bar{S}^N X\| \leq \sqrt{2} \sum_{m=0}^{N} \|(A_J - \mathbb{E})\bar{U}_m\| \,. \tag{32}$$

Proof. Here, write $V_J^m = |(X - A_J X)W_m|, V_J^0 X = X$, and define:

$$Y_J^{n,m} X = \{A_J V_J^n ... V_J^{n-m+1} X, A_J V_J^{n-1} ... V_J^{n-m+1} X \tag{33}$$

$$, ..., A_J V_J^{n-m+1} X, A_J X\}, \tag{34}$$

Lemma. *If A_J is a unitary projector and each W_i is unitary, then $Y_J^m X$ is 1-Lipschitz w.r.t. X.*

Proof. We can apply the proposition 2 with the operators $\{W_{n-m+1}, ..., W_n\}$, as this can be interprated as an IGT with different unitary operators. □

Here the idea is to take advantage of the tree structure of the IGT features. Thus when $Y_J^{n,m}$ is computing S_J^n to orders limited in $[n, n-m+1]$, we chain the features with the order $n-m$ to recover $Y_J^{n,m-1}$. To do so, we introduce for $m \geq 1$:

$$\Delta_J^{n,m} X = \{Y_J^m V_J^{n-m} X - Y_J^m \bar{V}^{n-m} X, A_J X - \mathbb{E}X\} \tag{35}$$

$$= \{-Y_J^m \bar{V}^{n-m} X, -\mathbb{E}X\} + Y_J^{m-1} X \,, \tag{36}$$

where $\bar{V}^n X = |(X - \mathbb{E}X)W_n|$, $\bar{V}^0 X = X$ and $\{x, y\}$ stands for a concatenation. This implies that $\Delta_J^{n,m} X$ is a $(m+1)$-uplet (and the symbol $+$ in (36) is thus a couple addition and the convention is that left corresponds to highest order of the couple), and $\Delta_J^{0,0} X = A_J X - \mathbb{E}X = -\mathbb{E}X + S_J^0 X$. The sum over m-uplet with different size is done such that the left elements are summed first. We then notice that:

$$\sum_{m=0}^{N} \Delta_J^{N,m} \bar{V}^{N-m-1} ... \bar{V}_1 X = S_J^N X - \bar{S}^N X \tag{37}$$

because each term of the couple is a telescopic sum (again here, we chain the features with orders in $[n-m-1, 1]$ to obtain the telescopy).
As $Y_J^{n,m}$ is 1-Lipschitz w.r.t. X and since a modulus is non expansive, $\| |(X - A_J X)W_n| - |(X - \mathbb{E}X)W_n| \| \leq \|\mathbb{E}X - A_J X\|$, combining those ingredients we get:

$$\|\Delta_J^{n,m} X\|^2 = \|A_J X - \mathbb{E}X\|^2 + \tag{38}$$

$$\|Y_J^{m-1}|(X - A_J X)W_n| - Y_J^{m-1}|(X - \mathbb{E}X)W_n\|^2 \tag{39}$$

$$\leq 2\|A_J X - \mathbb{E}X\|^2 \,. \tag{40}$$

Then, we further apply the triangular inequality to get the desired result. □

The following proposition shows that in case of exact ergodicity, the IGT and Expected-IGT representations have bounded moments of order 2:

Proposition 4. *Assume that $\mathbb{E}[A_J X] = \mathbb{E}X$, and that X has variance $\sigma^2 = \mathbb{E}\|X\|^2 - \|\mathbb{E}X\|^2$, then:*

$$\mathbb{E}[\|S_J^N X - \bar{S}^N X\|^2] \leq 2\sigma^2 \,. \tag{41}$$

Proof.

$$\mathbb{E}[\|S_J^N X - \bar{S}^N X\|^2] = \mathbb{E}[\|S_J^N X\|^2] + \mathbb{E}[\|\bar{S}^N X\|^2] \tag{42}$$

$$- 2 \sum_{m=0}^{N} \mathbb{E}[\text{Tr}((A_J U_m)^T \mathbb{E}[\bar{U}_m])] \tag{43}$$

$$\leq 2(\mathbb{E}\|X\|^2 - \sum_{m=0}^{N} \mathbb{E}[\text{Tr}((A_J U_m)^T \mathbb{E}[\bar{U}_m])]) \tag{44}$$

The inequality follows from Prop. 2 and Prop. 7. Now, from Lemma 1, $A_J, U_m, \bar{U}_m$ have positive coefficients, thus we get: $2\sum_{m=1}^{N}\mathbb{E}[\text{Tr}((A_J U_m)^T\mathbb{E}[\bar{U}_m])] \geq 0$. The first term allows to conclude as $\bar{U}_0 = U_0 = X$. □

Proposition 6. *Assume that each line of X is σ-sub-Gaussian. There exists $C > 1, K > 0, C' > 0$ such that $\forall m, \delta > 0$ with probability $1 - 8P\delta$, we have:*

$$\|\mathbb{E}[\boldsymbol{\mathcal{A}}]_{norm}\bar{U}_m - \mathbb{E}[\bar{U}_m]\| \tag{45}$$

$$\leq K\sigma C^m\sqrt{\ln\frac{1}{\delta}} + \tau\sqrt{n}C'\|\mu_m^2 - \mu_m^1\|\,. \tag{46}$$

Proof. Here, for the sake of simplicity, X_p corresponds to the p-th row of X. We write μ_m^j the expected-IGT of the node distribution of community j. Here, we have for $t \leq n_1$ (note that the right does not depend on t):

$$[\mathbb{E}[\boldsymbol{\mathcal{A}}]_{\text{norm}}\bar{U}_m]_t - \mathbb{E}[\bar{U}_m]_t \tag{47}$$

$$= \frac{1}{n_1 p + n_2 q}\big(p\sum_{i=1}^{n_1}(\bar{U}_m^i - \mu_m^1) + q\sum_{i=n_1+1}^{n_1+n_2}(\bar{U}_m^i - \mu_m^2)\big) \tag{48}$$

$$+ \frac{n_2 q}{n_1 p + n_2 q}(\mu_m^2 - \mu_m^1)\,. \tag{49}$$

Now, we note that from Lemma 5, $\{\bar{U}_m^i\}_{i\leq n}$ is a family of σC^m-sub-Gaussian independant r.v. From Hoeffding lemma [45], we obtain that for any δ, we have with probability $1 - 4P\delta$:

$$\|\sum_{i=1}^{n_1}(\bar{U}_m^i - \mu_m^1)\| \leq \sqrt{n_1}\sqrt{2}\sigma C^m\sqrt{\ln\frac{1}{\delta}} \quad \text{and}$$

$$\|\sum_{i=n_1+1}^{n_1+n_2}(\bar{U}_m^i - \mu_m^2)\| \leq \sqrt{n_2}\sqrt{2}\sigma C^m\sqrt{\ln\frac{1}{\delta}}\,.$$

As if n is large, by hypothesis $(\frac{p\sqrt{2n_1}+q\sqrt{2n_2}}{n_1 p+n_2 q})\sqrt{n} = \mathcal{O}(1)$. We perform the same for $n_1 < t \leq n_1 + n_2$ We then sum along n and use that $\frac{n_1}{n_2+\tau n_1} + \frac{n_2}{n_1+\tau n_2} = \mathcal{O}(1)$ and $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$. □

6.2 Dataset statistics

Table 3: Dataset Statistics

Datasets	**Nodes**	**Edges**	**Classes**	**Features**	*full* **Train/Val/Test**	*semi* **Train/Val/Test**
Cora	2,708	5,429	7	1,433	1,208/500/1,000	140/500/1,000
Citeseer	3,327	4,732	6	3,703	1,812/500/1,000	120/500/1,000
Pubmed	19,717	44,338	3	500	18,217/500/1,000	60/500/1,000
WikiCS	11,701	216,123	10	300	20 canonical train/valid/test splits	

Table 4: Standard deviations of classification accuracies for each splits of Cora, Citeseer, Pubmed as well as WikiCS.

Method/Dataset	Cora			Citeseer			Pubmed			WikiCS
	Full	Rand	Pred	Full	Rand	Pred	Full	Rand	Pred	
Unsupervised										
IGT + MLP (ours)	0.5	0.8	0.9	0.4	0.8	0.7	0.6	0.5	0.3	0.5
IGT + Lin. (ours)	0.1	0.8	0.2	0.3	0.7	0.5	0.1	0.2	0.1	0.5

6.3 Code and Data availability

All the code is accessible in the folder given in the supplementary materials.

6.4 Training time

We informally noticed that the training of our isometry layers converges quickly. During the supervised training, no multiplication with the adjacency matrix is involved, which can speed up the training compared to GCNs. We further report wall-clock training time in seconds until convergence for our method and for GCNs. For the latter, we used an implementation provided by the authors and trained on the same hardware (with GPU) as our IGT model. For Cora, Citeseer and PubMed respectively, the training time of our IGT layers was 0.45s, 0.57s and 4.88s, whereas the training time of the classification head was 0.25s, 0.24s and 0.94s. By way of comparison, GCN training time was 0.86s, 1.82s, and 1.12s. We would like to highlight that our code works on limited resources and we used a total of 10 GPU hours for developing and benchmarking this project.