Graph Classification Gaussian Processes via Spectral Features

Felix L. Opolka[1] **Yin-Cong Zhi**[2] **Pietro Liò**[1] **Xiaowen Dong**[2]

[1]Department of Computer Science and Technology, University of Cambridge, Cambridge, UK
[2]Department of Engineering Science, University of Oxford, Oxford, UK

Abstract

Graph classification aims to categorise graphs based on their structure and node attributes. In this work, we propose to tackle this task using tools from graph signal processing by deriving spectral features, which we then use to design two variants of Gaussian process models for graph classification. The first variant uses spectral features based on the distribution of energy of a node feature signal over the spectrum of the graph. We show that even such a simple approach, having no learned parameters, can yield competitive performance compared to strong neural network and graph kernel baselines. A second, more sophisticated variant is designed to capture multi-scale and localised patterns in the graph by learning spectral graph wavelet filters, obtaining improved performance on synthetic and real-world data sets. Finally, we show that both models produce well calibrated uncertainty estimates, enabling reliable decision making based on the model predictions.

1 INTRODUCTION

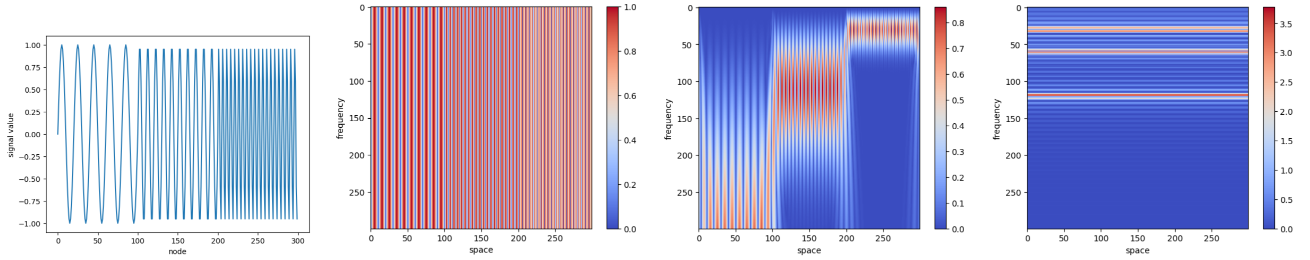
Data that are collected in a network environment, hence supported by a graph structure, have become pervasive in modern data analysis and processing tasks. This poses the new task of *graph classification*, which, similar to image classification, aims at classifying graph-structured data into different classes. For example, representing protein structures as graphs, one may wish to classify whether they are toxic or not; modelling information propagation cascades on social media platforms as graphs, one may wish to detect whether the originating post of each cascade corresponds to fake news or not; considering urban transportation networks as graphs, one may wish to identify whether a particular area is likely to lead to traffic congestion.

Such graph-level classification problems are non-trivial generalisation of classical classification problems: graphs are irregular structures and traditional techniques defined in the Euclidean domains, such as the Fourier transform, cannot be applied directly; data collected on the nodes (or edges) of the graph are often continuous measurements, therefore traditional tools in graph analysis need to be adapted; finally, real-world graphs often come with extremely large size, making scalability of the algorithms a challenge.

Recent efforts to tackle the problem of graph classification mainly fall into two categories. First, graph kernels [Nikolentzos et al., 2021, Kriege et al., 2020, Borgwardt et al., 2020], as traditional ways of comparing graphs or compute distance between them, have been adapted for graph classification. However, it is still a challenge for these methods to handle multi-dimensional and continuous node features as well as graphs of different sizes. Second, graph neural networks [Bruna et al., 2014, Defferrard et al., 2016, Kipf and Welling, 2017], generalisations of neural networks to deal with graph-structured data, can also be utilised for graph classification. In these frameworks, a read-out function is often deployed after the neural network layers to summarise node representations into a single graph representation, for example using summation, averaging, or pooling [Dai et al., 2016, Duvenaud et al., 2015, Gilmer et al., 2017, Ying et al., 2018]. This addressed the issue of comparing graphs of different sizes; however, these architectures are often trained with large amount of data, and the predictions are not easily interpretable.

We address the above limitations in this work, and our main contributions are as follows. First, inspired by the image segmentation literature [Porter and Canagarajah, 1996] as well as recent development in the field of graph signal processing [Shuman et al., 2013, Ortega et al., 2018, Dong et al., 2020, Ortega, 2022], we propose to consider multi-dimensional and continuous node features as graph signals, and compute spectral features using the graph Fourier transform, i.e., energy distribution of the Fourier coefficients in different frequency bands. We then utilise these features in a

Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023), PMLR 216:1575–1585.



(a) Graph signal $\mathbf{x}$ consisting of three concatenated sine-waves of different frequency.

(b) Signal under view **fully localised in space** (original signal $\mathbf{x}$)

(c) Signal under view **partially localised both in space and frequency** (Wavelet transform of $\mathbf{x}$)

(d) Signal under view **fully localised in frequency** (Fourier transform of $\mathbf{x}$)

Figure 1: Visualisation of the different space-frequency resolutions of the spatial view of a signal (b), its wavelet transform (c), and its Fourier transform (d) for the example of concatenated sine-waves (a) on a path graph. A path graph forms a line of nodes where each node, except for the two end nodes, has exactly two neighbours.

Gaussian process framework, which has the advantage of not requiring a validation set and being more interpretable. We show that this simple method already achieves surprisingly competitive performance compared to baselines relying on graph kernels and graph neural networks. Second, we further derive a second method based on the spectral graph wavelets [Hammond et al., 2011], which possesses the additional benefits of capturing multi-scale and localisation information and leads to further improvement in classification performance. Finally, the proposed Gaussian process based methods allow us to quantify uncertainty information in the graph classification results, which to our knowledge has not been considered so far in the literature.

2 PRELIMINARIES

The models proposed here will exploit spectral features of graphs within the framework of learning with Gaussian processes. In the following, we will therefore give a brief introduction to the techniques from graph signal processing and Gaussian process inference that we will use in our methodology.

2.1 GRAPH SIGNAL PROCESSING

Graph signal processing (see Ortega [2022] for a general introduction) offers a range of tools for analysing the spectral properties of graph signals. One of its key tools is the generalisation of the Fourier transform to the graph domain, as described by Shuman et al. [2013]. It breaks down a signal into components of different frequency, giving rise to a complementary view of the signal on the frequency domain as opposed to the spatial domain. Moreover, the Fourier transform is useful for analysing learned transforms of signals, which form the basis for machine learning. Spectral graph theory [Chung, 1997] has generalised the Fourier transform for signals on the Euclidean domain to signals on the more general graph domain. In general, the Fourier transform of a signal on any domain is defined as the decomposition of that signal into the basis functions of the Laplace operator. On the graph domain, the Laplace operator is given by the symmetric and positive-semidefinite Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix of the graph with N nodes and $\mathbf{D} \in \mathbb{R}^{N \times N}$ is the diagonal degree matrix of the graph such that $D_{ii} = \sum_{j=1}^{N} A_{ij}$. Using the eigendecomposition of the graph Laplacian $\mathbf{L} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$, we can define the Fourier transform of a graph signal $\mathbf{x} \in \mathbb{R}^N$ as $\hat{\mathbf{x}} = \mathbf{U}^\top \mathbf{x}$.

The original signal $\mathbf{x}$ and its Fourier transform $\hat{\mathbf{x}}$ form the two extremes of a trade-off relationship between resolution in space and resolution in frequency: $\mathbf{x}$ is fully localised in space but not localised in frequency, whereas $\hat{\mathbf{x}}$ is fully localised in frequency but not localised in space. The wavelet transform is a convenient technique for analysing signals with resolution in both space and frequency. It decomposes a signal into a set of basis functions obtained by scaling and shifting a so-called *mother wavelet*. Hammond et al. [2011] derive a graph signal wavelet transform for signal $\mathbf{x}$ using mother wavelet $b(\lambda)$ as

$$\mathbf{w} = \mathbf{U} b(\beta \boldsymbol{\Lambda}) \mathbf{U}^\top \mathbf{x}, \tag{1}$$

where β is the scale parameter. Adjusting this scale parameter allows examining the signal at varying frequency ranges, which correspond to different localisation behaviour in the spatial node domain.

The relationship between the original spatial signal, Fourier transform, and wavelet transform is best visualised for a signal on a path graph, i.e. a graph that simply forms a line of nodes, as it resembles an interval on the real line. We plot a signal on such a graph consisting of three sine-waves of different frequencies in Figure 1a along with the spectrograms of the signal in the spatial domain (1b), in the frequency domain (1d), and for the wavelet transform (1c), which demonstrate how the wavelet transform trades off between spatial and spectral resolution.

To analyse a signal at multiple scales simultaneously, a low-pass filter $h(\alpha\lambda)$ with parameter α and multiple scaled versions of the mother wavelet can be combined into a more complex wavelet filter function $g_\theta(\lambda) = h(\alpha\lambda) + \sum_{l=1}^{L} b(\beta_l\lambda)$ with the set of scale parameters $\theta = \{\alpha, \beta_1, \beta_2, \ldots\}$.

2.2 GAUSSIAN PROCESSES

Gaussian processes (see Rasmussen and Williams [2005] for a general introduction) are stochastic processes that can be considered multivariate normal distributions extended to infinitely many dimensions. Their properties make them a convenient choice for prior distributions in Bayesian machine learning models. A GP prior on a latent function $f : \mathbb{R}^D \to \mathbb{R}$ is given by

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k_\theta(\mathbf{x}, \mathbf{x}')\right), \tag{2}$$

where $m(\mathbf{x})$ is the mean function of the process and often set to 0 and $k_\theta(\mathbf{x}, \mathbf{x}')$ is its kernel function with a set of kernel hyperparameters θ. Given input data $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N \times D}$ and labels $\mathbf{y} = [y_1, y_2, \ldots, y_N]^\top \in \mathbb{R}$, the GP model can be used for performing probabilistic inference. If the model specifies a Gaussian likelihood, the posterior distribution $p(\mathbf{f}|\mathbf{y})$ is analytically tractable and also follows a Gaussian process distribution. Furthermore, the marginal likelihood $p(\mathbf{y})$ is tractable and can be maximised to optimise the kernel hyperparameters θ.

Inference in a GP model becomes prohibitive for data sets of large size N, as computing the posterior distribution requires inverting an $N \times N$ covariance matrix. *Sparse Gaussian processes* alleviate this computational burden by constructing a smaller pseudo-data set of M so-called *inducing points*, where $M \ll N$. The inducing points are chosen such that the GP posterior they induce is similar to the actual posterior using all N data points. Titsias [2009] proposes a way of learning the inducing inputs as variational parameters by optimising a lower bound to the marginal likelihood.

In case the model specifies a non-Gaussian likelihood such as a Categorical likelihood—as would be the case for classification tasks—the posterior distribution is no longer analytically tractable and needs to be approximated. This can be achieved using a variational approximation to the posterior distribution where the variational family is chosen to be a Gaussian process. Hensman et al. [2015] show how a variational approximation for non-Gaussian likelihoods can be designed for sparse GPs. The resulting model has the benefit that the lower bound to the marginal likelihood, referred to as the *Evidence Lower Bound (ELBO)*, has a data term that factorises over data points and can therefore be maximised using stochastic gradient descent [Hensman et al., 2013].

3 SPECTRAL FEATURE LEARNING FOR GRAPH CLASSIFICATION

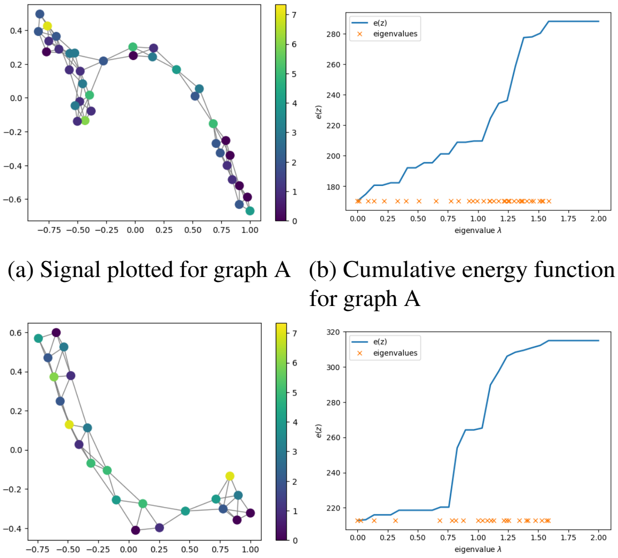
In the following, we present two GP models for graph classification. The first model is focused on simplicity, while the second model trades off simplicity against higher expressive power. The former is based on the graph Fourier transform of the node feature signal and is therefore referred to as **FT-GP**, whereas the latter employs the spectral graph wavelet transform and is therefore referred to as **WT-GP**. The key idea in common for both approaches is that alternatives to the spatial view of the node feature graph signals, as provided by the Fourier and wavelet transform, are better starting points for designing graph kernels. Consequently, we hypothesise that even simple transformations and representations of the signals under these views, mainly focusing on how the energy of these signals are distributed in different parts of the spectrum and/or space, are sufficiently expressive for distinguishing attributed graphs.

Both models are used for typical graph classification tasks where we assume to be given a set of G training set graphs $\mathcal{T} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_G\}$ of varying sizes $N_1, N_2, \ldots, N_G$, alongside their corresponding class labels $y_1, y_2, \ldots y_G$. For each graph i, we are given an adjacency matrix $\mathbf{A}^{(i)} \in \mathbb{R}^{N_i \times N_i}$ and D-dimensional node features $\mathbf{X}^{(i)} \in \mathbb{R}^{N_i \times D}$. For the following exposition, we initially assume one-dimensional node features, i.e. $D = 1$, and describe how to address the case of general D later on. The goal of graph classification is to learn a mapping from the adjacency matrix and node features of a graph to its class label such that it generalises to graphs outside the training set.

3.1 SPECTRAL FEATURES FOR GRAPH-LEVEL PREDICTION

For each graph $\mathcal{G}^{(i)}$ with adjacency matrix $\mathbf{A}^{(i)} \in \mathbb{R}^{N_i \times N_i}$ we can compute the symmetrically normalised graph Laplacian matrix $\mathbf{L}^{(i)} = \mathbf{I} - (\mathbf{D}^{(i)})^{-1/2}\mathbf{A}^{(i)}(\mathbf{D}^{(i)})^{-1/2}$. Its eigendecomposition is $\mathbf{L}^{(i)} = \mathbf{U}^{(i)}\mathbf{\Lambda}^{(i)}\mathbf{U}^{(i)\top}$, where $\mathbf{U}^{(i)} \in \mathbb{R}^{N_i \times N_i}$ denotes the eigenvector matrix and $\mathbf{\Lambda}^{(i)} = \mathrm{diag}(\lambda_1^{(i)}, \ldots, \lambda_{N_i}^{(i)}) \in \mathbb{R}^{N_i \times N_i}$ denotes the diagonal eigenvalue matrix. The spectrum of the symmetrically normalised graph Laplacian is in $[0, 2]$, i.e. $0 \leq \lambda_j^{(i)} \leq 2 \forall j = 1, \ldots, N_i$ [Shuman et al., 2013]. Notably, this holds regardless of the size N_i of the graph.

Using the definition of the graph Fourier transform as presented in Section 2.1, we can compute the Fourier coefficients of the node feature signal for graph $\mathcal{G}^{(i)}$ as $\hat{\mathbf{x}}^{(i)} = \mathbf{U}^{(i)\top}\mathbf{x}^{(i)}$. Based on the eigenvalues of the graph and the Fourier coefficients of the node feature signal, we



(a) Signal plotted for graph A (b) Cumulative energy function for graph A

(c) Signal plotted for graph B (d) Cumulative energy function for graph B

Figure 2: Comparison of two graphs from the ENZYME data set Borgwardt et al. [2005] in terms of cumulative energy function. The same node feature is plotted for both graphs in (a) and (c) where the node colour indicates the node feature value. Figures (b) and (d) show the corresponding cumulative energy functions, revealing the distinct energy profiles of both graphs: in particular, graph A has a larger low-frequency component compared to graph B.

can define a *cumulative energy function* for graph i as

$$e^{(i)}(z) = \sum_{j=1}^{N^{(i)}} \hat{x}_j^{(i)2} \mathbb{1}_{\{\lambda_j^{(i)} \leq z\}}. \tag{3}$$

This function $e^{(i)}(z)$ represents the energy of the node feature signal that is contained in the spectrum up to a frequency z and it thereby encodes both graph structure and node feature information. We hypothesise that the energy profile formulated in this way is expressive enough to distinguish attributed graphs of different classes. Figure 2 visualises the cumulative energy function for two sample graphs from the ENZYMES data set Borgwardt et al. [2005]. We find that the two cumulative energy functions differ clearly, both as a result of different eigenvalue locations and distinct Fourier coefficients.

We can derive a feature vector $\mathbf{e}^{(i)} \in \mathbb{R}^M$ for graph i from $e^{(i)}(z)$ by evaluating $e^{(i)}(z)$ at a sequence of M evaluation points $[h_1, \ldots, h_M]$, therefore

$$e_m^{(i)} = e^{(i)}(h_m) = \sum_{j=1}^{N^{(i)}} \hat{x}_j^{(i)2} \mathbb{1}_{\{\lambda_j^{(i)} \leq h_m\}}. \tag{4}$$

In practice, we will often choose $[h_1, \ldots, h_M]$ to be linearly spaced on the $[0, 2]$ interval of the spectrum. Crucially, this guarantees that all graph representations $\mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(G)}$ are of size M regardless of individual graph sizes.

The above derivation assumes one-dimensional node features ($D = 1$). The graph representations can be generalised to multi-dimensional node features ($D > 1$) by defining a cumulative energy function for each feature dimension and concatenating their discretisations, resulting in graph representations $\mathbf{e}^{(i)}$ of size $M \times D$.

The resulting graph representations can now serve as input into a base kernel of choice and we use the radial basis function (RBF) kernel $k(\mathbf{x}, \mathbf{x}') = \exp(-\frac{l}{2}\|\mathbf{x} - \mathbf{x}'\|_2)$ with lengthscale l.

3.2 SPECTRAL WAVELET FEATURES FOR GRAPH-LEVEL PREDICTION

A key limitation of the FT-GP approach stems from the Fourier transform providing full resolution in frequency but no resolution in space. As a result, we expect the model to under-perform in correctly classifying graphs whose class is determined by localised patterns (cf. Figure 1). This is evaluated in more detail in experiments with synthetic data in Section 5.1. We can partially alleviate this limitation by employing the wavelet transform to derive kernels, allowing us to better trade off between localisation in space and frequency in a flexible manner.

We design wavelet filters consisting of a single low-pass filter and multiple band-pass filters to obtain filtered signals. A particular wavelet filter offers a distinct view of the attributed graph at hand, hence using K of those wavelet filters allows us to obtain a more diverse view of the graph. Moreover, we can seize the capability of GPs to optimise hyper-parameters to find wavelet scales that better distinguish the classes of graphs of the particular data set at hand. Figure 3 plots the wavelet transformed signal for a particular graph and two filters of different scales, to showcase the distinct view of the attributed graph each filter offers.

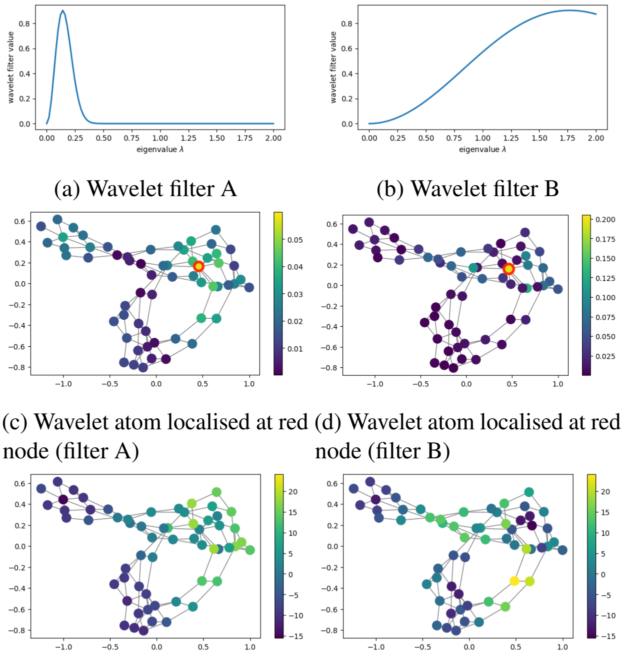
We define each of the K wavelet filter functions as a sum of atomic filters including a single low-pass filter and L band-pass filters

$$g_\theta(\lambda) = h(\alpha\lambda) + \sum_{l=1}^{L} b(\beta_l \lambda), \tag{5}$$

where $h(\alpha\lambda)$ refers to the low-pass filter function with scale α, $b(\beta_l\lambda)$ refers to the l-th band-pass filter with scale β_l, and $\theta = \{\alpha, \beta_1, \beta_2, \ldots\}$ is the set of scale parameters. Reminiscent of the signal energy computed for the FT-GP model proposed earlier, we compute the magnitude of signal filtered with filter k

$$w_k^{(i)} = \|\mathbf{U}^{(i)} g_{\theta_k}(\mathbf{\Lambda}^{(i)}) \mathbf{U}^{(i)\top} \mathbf{x}^{(i)}\|_2 \tag{6}$$

to express how much energy of the signal is captured by the kth filter. The final feature vector $\mathbf{w}^{(i)} \in \mathbb{R}^{K \cdot D}$ for graph



(a) Wavelet filter A

(b) Wavelet filter B

(c) Wavelet atom localised at red node (filter A)

(d) Wavelet atom localised at red node (filter B)

(e) Signal filtered with filter A

(f) Signal filtered with filter B

Figure 3: Comparison of two different filters for the same graph and node feature (as a graph signal) from the ENZYME data set. The band-pass filter of large wavelet scale (a) highlights low frequencies, captures a larger node neighbourhood (c), and produces a smooth signal (e). The band-pass filter of small wavelet scale (b) highlights high frequencies, captures a smaller node neighbourhood (d), and produces a less smooth signal (f).

i stacks the filter coefficients for each of the K wavelet filters and each of the D node feature signals. As before, the resulting signal representations are fed into a radial basis function kernel to obtain the covariance between graphs. The kernel hyper-parameters are optimised alongside the wavelet scale parameters θ in a data-driven way employing type II maximum likelihood estimation.

3.3 RELATIONSHIP TO GRAPH NEURAL NETWORKS

The GP models presented here operate in two stages, one focused on transformations of the node feature signal and the second on deriving a graph-level summary. In this regard, they share similarities with graph neural networks for graph classification, making it worthwhile to compare the two approaches more thoroughly.

Both GP methods, in the first step, translate the node feature signal from the spatial domain to a domain that is either fully localised in frequency or partially localised in both space and frequency. The resulting coefficients thereby combine information from all nodes across the graph, overcoming the spatial locality of each node. In case of WT-GP, the wavelet transform can be considered an aggregation operation of node features in a neighbourhood around each node whose shape is determined by the particular wavelet filter in use [Opolka et al., 2022]. This aggregation of information across graphs is reminiscent of the aggregation operation inherent to graph convolutional networks (GCN) [Kipf and Welling, 2017]. In fact, Wu et al. [2019] describe how the GCN aggregation operation acts as a low-pass filter on the node feature graph signal.

In a second step, the GP methods aggregate coefficients across the whole graph via binning in case of FT-GP and summation in case of WT-GP. Neural network methods apply similar so-called "read out" operations to summarise node representations into a single graph representation via simple summation or averaging [Dai et al., 2016, Duvenaud et al., 2015, Gilmer et al., 2017], as well as more complicated pooling operations [Ying et al., 2018]. In contrast to GCN-based approaches however, the aggregation approach presented here has a straightforward spectral interpretation and allows aggregating information in a way that goes beyond averaging low-pass filtered values, making it possible to capture multi-scale information in the graphs. Furthermore, the GP framework provides the flexibility of a non-parametric approach and outputs uncertainty estimates for predictions, which are examined in more detail in Section 5.4.

3.4 SCALABILITY

In terms of scalability of the GP models, the quantities of concern are the number of graphs in a data set and the size of those graphs. Within the computation of the kernel of either proposed approach, the computational complexity is dominated by the necessary eigendecomposition of the graph Laplacian, which is in $\mathcal{O}(N_i^3)$, where N_i is the size of a graph i. Graphs in graph classification tasks are typically small (cf. Table 2), therefore the number of nodes is not usually an obstacle for scalability. For larger graphs, one can reduce the computational complexity by resorting to the approximate Fourier transform of a graph signal [Le Magoarou et al., 2018].

The total number of graphs in the data set may pose another limit to the scalability of the method when computing the GP posterior. Sparse GPs as discussed in Section 2.2 alleviate this issue by enabling stochastic optimisation in mini-batches when the data set size requires it.

4 RELATED WORK

Applications of Spectral and Wavelet Energy: Our work is related to building classifiers in the graph spectral domain, and wavelets are an extension to such approaches with the benefits of multi-scale properties and better localisation. Solving problems on graphs by making use of the energy distribution in the spectral domain has been demonstrated effectively for the task of image segmentation in classical signal processing [Porter and Canagarajah, 1996], and for geographical mobility prediction in graph signal processing [Dong et al., 2013]. Both suggest that the energy distribution clearly contains a significant level of information which we aim to utilise in this work for graph classification.

Graph Kernels: One of the first studies into kernel functions acting on the graph level is the graph kernels summarised in the surveys of Nikolentzos et al. [2021], Kriege et al. [2020], Borgwardt et al. [2020]. Graph kernels generally have multiple definitions, of which a number of choices have in common the existence of a double sum of a base kernel. To compute the kernel between two graphs, a standard base kernel is chosen and takes as inputs a node feature from each graph, and the double summation then covers all possible pairs of nodes between the two graphs. Similar definition can be applied to graphs that contain edge features. Such designs are limited as any edge connections are ignored in the double sum, and the kernel therefore boils down to a comparison of cross-products of the node features. Another common design is the computation of a product graph, where a new graph is constructed based on the two input graphs. When we simulate a random walk on the product graph, we compute the number of matching walks on the two individual graphs. Random walk kernels do take into account the graph structure, but is limited by scalability as the computation of the kernel over a product graph leads to a $\mathcal{O}(N_i^3 N_j^3)$ complexity.

Graph kernels has also been developed from the Weisfeiler-Lehman (WL) test for graph isomorphism [Huang and Villar, 2021], which also acts as a graph similarity. The WL test produces a series of node "colouring" from a neighbourhood gathering and relabelling procedure. In the WL kernel [Shervashidze et al., 2011], the "colourings" are passed through a hashing function (or histogram mapping) to form inputs to a base kernel. As an alternative, the Wasserstein distance between two sets of labels can be used instead of the hashing function as shown in Togninalli et al. [2019]. Compared to our kernel design, WL-based kernels generally work with the neighbourhoods in the spatial domain, and so they ignore any information in the spectral domain. Additionally, if the graph contains high dimensional node features, the WL does not have an efficient way to encode them into the embeddings for the WL algorithm.

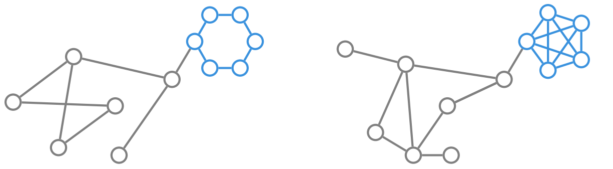
The use of Fourier and wavelets basis are Laplacian-based models. Wavelets in particular, allow for aggregation over a continuous neighbourhood, giving the model the ability to operate on a multi-scale level. Though multi-scale Laplacian kernels exist such as Kondor and Pan [2016], they only operate in the spatial domain by computing kernels between sub-graphs of various neighbourhood sizes. On the other hand, the work of Pineau [2019] does make use of the Laplacian spectral information, but like the previous kernel, it ignores the node features that may come with the graphs, while we focus on the spectral information of the node features.

Graph Neural Network Models: Lastly, graph neural networks (GNNs) have also been applied as a test for graph isomorphism, and as a result can be applied to graph classification. The message passing step of a GNN layer (examples including Duvenaud et al. [2015], Li et al. [2012], Murphy et al. [2019]) is comparable to the neighbourhood gathering step in the WL algorithm. Analysis of certain GNN models showed that they are at best as powerful as the WL test, and the graph isomophism network (GIN) proposed in Xu et al. [2019] achieves the theoretical guarantee of the WL. Meanwhile, simpler models such as GCN [Kipf and Welling, 2017] and GraphSAGE [Hamilton et al., 2017] have been shown to be unable to distinguish certain types of graphs that GIN can handle.

5 EXPERIMENTS

Our work aims to investigate two core empirical questions with regards to the models presented. Firstly, whether the energy profile as captured by either of the presented kernels is sufficiently expressive to classify real-world graphs. Secondly, whether the wavelet transform approach of WT-GP improves upon the FT-GP model based on the Fourier transform. We begin with the latter question by comparing the two proposed models on synthetic data sets.

In all our experiments, we use the same experimental setup, unless explicitly stated otherwise. The FT-GP discretises the cumulative energy function by evaluating it at 30 evaluation points. The WT-GP uses 10 filters consisting of a single low-pass and three band-pass filters. Each low-pass scale of each filter is uniformly randomly initialised to a value between 4.0 and 6.0. The band-pass scales of each filter are uniformly randomly initialised to a value between 0.1 and 5.0. A non-sparse variational Gaussian process as described in Section 2.2 is trained using the L-BFGS-B optimiser [Liu and Nocedal, 1989, Byrd et al., 1995] until the ELBO convergences. All results are cross-validated using 10-folds under a stratified split where in each fold 80% of the data is used for training and 10% for testing. The remaining 10% are set aside for validation to make the evaluation comparable to results in related work, although neither of the two GP models make use of the validation set. This validation procedure is suggested by Errica et al. [2020] to overcome weaknesses in the evaluation of graph classification methods in previous work.



(a) Class 0

(b) Class 1

Figure 4: Visualisation of example graphs of the two different classes in the synthetic ring-vs-clique data set. All graphs consist of a graph component sampled from an Erdős-Rényi (ER) model, which is connected to either a ring graph component in case of class 0 or a fully connected graph component in case of class 1 (highlighted in blue). Both components of each graph can vary in the number of nodes.

	ring-vs-clique	**sbm**
FT-GP	62.5 ±7.5	58.5 ±15.2
WT-GP	**99.5** ±1.5	**91.0** ±5.4

Table 1: Classification accuracy of the FT-GP model compared to the WT-GP model on the two synthetic binary classification data sets.

Data set	**# Graphs**	**# Classes**	**Avg # Nodes**	**Avg # Edges**	**# Node Attr**
ENZYMES	600	6	32.63	62.14	21
MUTAG	188	2	17.93	19.79	7
NCI1	4,110	2	29.87	32.30	37
IMDB-BIN	1,000	2	19.77	96.53	–
IMDB-MUL	1,500	3	13.00	65.94	–

Table 2: Statistics of the data sets used in the empirical evaluation of the proposed models. We use three data sets with and two data sets without node features.

5.1 SYNTHETIC EXPERIMENTS

We design two synthetic data sets to highlight the differences between the two GP models presented here. The first data set, referred to as **ring-vs-clique**, contains two classes. The classes differ in that graphs in class 0 are guaranteed to contain a subgraph that forms a ring of at least size 5, while graphs in class 1 are guaranteed to contain a subgraph that forms a complete graph (or clique) of at least size 5. We generate the class-balanced data set by sampling 200 graphs using the following procedure. In a first step, a graph of uniformly random size between 10 and 30 nodes is sampled from an Erdős-Rényi model [Erdös and Rényi, 1959]. In the second step, depending on the class label, a ring or clique of uniformly random size between 5 and 10 nodes is constructed and connected to a randomly chosen node in the graph sampled in the first step through a single edge. The two different classes are sketched in Figure 4. The data set is designed to test how well a model can distinguish graphs when class labels are determined only by a relatively small, localised subgraph.

The second data set, referred to as **sbm**, contains graphs drawn from a stochastic block model [Holland et al., 1983], where nodes of the same block are connected with a probability of 80% and nodes of different blocks are connected with 10% probability. Graphs of class 0 are drawn from a model with 2 blocks and graphs of class 1 are drawn from a model with 3 blocks. All graphs have a uniform random size between 10 and 30 nodes.

The results for the binary classification tasks on these data sets for the two GP models are shown in Table 1. We find that on both tasks, the wavelet-based GP model strongly outperforms the baseline FT-GP model. In fact, FT-GP performs only slightly better than random guessing on both data sets whereas WT-GP achieves a near perfect accuracy on ring-vs-clique and a high accuracy on the sbm data set. The results confirm that designing graph kernels based on wavelet filtered node feature signals leads to more expressive kernels compared to a kernel based on Fourier transformed node features.

5.2 REAL-WORLD EXPERIMENTS

To examine whether the proposed GP models based on the energy profiles of attributed graphs are sufficiently expressive for classifying real-world graphs, we conduct a number of experiments on benchmark data sets and compare the model performance to popular baseline methods for graph classification.

The data sets in our empirical evaluation are shown in Table 2 along with an overview of their statistics. The ENZYMES data set [Borgwardt et al., 2005] contains protein graphs of enzymes that require classification into the Enzyme Commission top level enzyme classes. The MUTAG data set [Debnath et al., 1991] consists of molecular graphs and the task is to detect an effect on the Salmonella typhimurium bacterium. Similarly, the NCI1 data set [Wale and Karypis, 2006] holds molecular graphs that need to be classified based on whether they are active against certain types of cancer. Finally, the IMDB-BINARY and IMDB-MULTI data sets [Yanardag and Vishwanathan, 2015] are derived from the Internet Movie Database (`imdb.com`) and consist of graphs of actors and actresses who have co-starred together in a film. A classifier has to predict the genre of each graph belonging to a particular actor or actress. The graphs in the first three data sets come with node attributes while the IMDB data sets are unattributed. We add one-hot encoded node degrees as features to ensure all graphs have node attributes.

We compare the proposed GP models to a number of neural network and graph kernel baselines. The neural network baselines include DGCNN [Zhang et al., 2018], GraphSAGE [Hamilton et al., 2017], DiffPool [Ying et al., 2018], and GIN [Xu et al., 2019]. Among graph kernels we compare to the Shortest Path (SP) kernel [Borgwardt and Kriegel, 2005], the Weisfeiler-Lehman (WL) kernel [Shervashidze et al., 2011], and the Multiscale Laplacian (ML) kernel [Kondor and Pan, 2016].

The classification accuracy of each model on all data sets is shown in Table 3 where the results for baseline methods are obtained from the empirical evaluation by Nikolentzos et al. [2021]. Mirroring the results on the synthetic data set, we find that WT-GP performs at least as well as FT-GP on all data sets except on IMDB-MULTI where both perform roughly similarly. Moreover, both models yield competitive performance compared to the baselines, with WT-GP achieving the highest accuracy on three of the five data sets and FT-GP being among the best two models also on three out of five data sets. We highlight the surprising effectiveness of the Fourier features of FT-GP in comparison to the baselines, especially since no learning is involved in the feature construction.

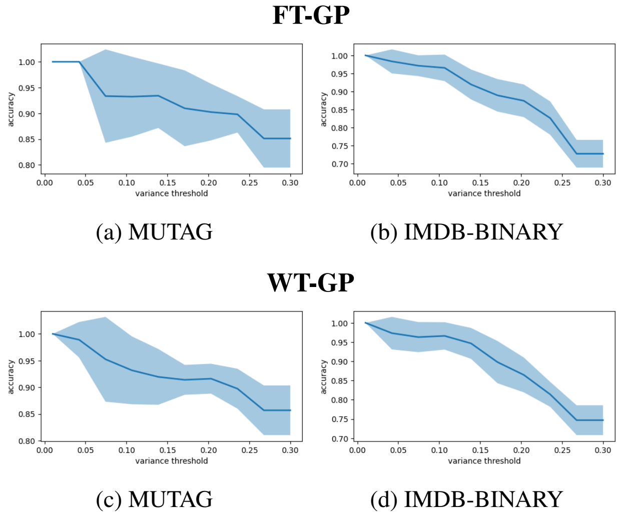
While overall outperforming the neural network baselines, the trend across data sets is comparable between the GP models and the neural networks. In fact, where the kernel methods outperform the GNNs, they also tend to be more comparable to the GP models. This appears to provide evidence for the similarity between the methods proposed here and GNNs, as outlined in Section 3.3.

5.3 ROBUSTNESS ANALYSIS

While the proposed GP models have fewer hyper-parameters than comparable neural network models, a small number of values needs to be selected prior to model fitting. For the FT-GP this is primarily the number of evaluation points M and for WT-GP this is mainly the number of filters K.

We evaluate FT-GP for a range of different numbers of evaluation points M on two data sets, one with and one without node features. The results are shown in Table 4. We find that FT-GP is overall robust to varying the number of evaluation points. We note that larger M linearly increase time and memory complexity for training and inference. Notably, however, larger M do not increase the number of learned parameters of FT-GP (which does not have learned parameters) but merely increase the "resolution" of the approximation of the cumulative energy function and we therefore do not expect FT-GP to begin to over-fit for larger M.

In a similar vein, the performance of WT-GP for varying numbers of filters K is presented in Table 5. The results indicate that WT-GP is robust to different numbers of filters. Similar to M, increasing K linearly increases the time and memory complexity for training and inference. Unlike M, however, increasing K means a larger number of scale parameters need to be estimated using MLE and therefore we do expect the model to over-fit for very large values of K. For the values that are computationally feasible for the given data sets, we have not yet observed over-fitting.



(a) MUTAG (b) IMDB-BINARY

(c) MUTAG (d) IMDB-BINARY

Figure 5: Accuracy of predictions when rejecting predictions that have a variance above a certain threshold. As the threshold becomes lower, i.e. more and more strict (from right to left on x-axis), the prediction accuracy increases.

5.4 QUALITY OF UNCERTAINTY ESTIMATES

One of the core advantages of a Bayesian treatment of graph classification is the availability of uncertainty estimates that can be a crucial part of the real-world application of a model. Uncertainty estimates allow downstream users of the model to weigh the reliability of its predictions and thus more confidently make decisions based on those predictions. The GP models proposed here make uncertainty predictions in the form of the variance of the variational posterior predictive distribution. We can assess the quality of those uncertainty estimates by simulating an experiment where downstream users are allowed to "reject" predictions if their variance is above a certain threshold. We can then plot the accuracy of the predictions that are not rejected. If the uncertainty estimates are well calibrated, we expect the accuracy of the remaining, low-variance (i.e. high certainty) predictions to increase. We plot the results for both our models and two of the data sets in Figure 5. We find that, as expected, the accuracy increases for stricter variance thresholds. As the variance threshold increases, the accuracy rises from the results reported in Table 3 for the complete set of predictions, to 1.0 for only the most confident predictions.

6 CONCLUSIONS

We have proposed two GP models using spectral features to classify graphs. The first approach constructs the energy profile of an attributed graph and compares across features.

	ENZYMES	MUTAG	NCI1	IMDB-BINARY	IMDB-MULTI
DGCNN	38.9 ±5.7	84.0 ±7.1	76.4 ±1.7	69.2 ±3.0	45.6 ±3.4
GraphSAGE	58.2 ±6.0	83.6 ±9.6	76.0 ±1.8	68.8 ±4.5	47.6 ±3.5
DiffPool	59.5 ±5.6	79.8 ±6.7	76.9 ±1.9	68.4 ±3.3	45.6 ±3.4
GIN	**59.6** ±4.5	84.7 ±6.7	**80.0** ±1.4	**71.2** ±3.9	**48.5** ±3.3
SP	timeout	82.4 ±5.5	72.5 ±2.0	58.2 ±4.7	39.2 ±2.3
WL	50.7 ±7.3	**86.7** ±7.3	**85.2** ±2.2	70.7 ±6.8	**51.3** ±4.4
ML	33.2 ±5.8	**87.2** ±7.5	**79.7** ±1.8	69.9 ±4.8	47.7 ±3.2
FT-GP	**60.7** ±4.3	85.7 ±6.2	77.7 ±1.6	**72.7** ±3.9	**48.8** ±2.8
WT-GP	**63.8** ±5.3	**87.3** ±4.8	78.1 ±2.1	**74.6** ±4.1	48.4 ±2.9

Table 3: Comparison of the proposed GP models with common neural network and graph kernel baselines in terms of classification accuracy. Colours indicate the best, second-, and third-best result.

M	ENZYMES	IMDB-BINARY
20	61.2 ±5.1	73.1 ±4.0
25	60.5 ±4.5	73.5 ±3.7
30	60.7 ±4.3	72.7 ±3.9
35	60.3 ±4.9	73.6 ±3.5
40	61.0 ±4.9	73.0 ±3.6
45	61.0 ±4.9	73.0 ±4.0
50	61.0 ±4.7	72.9 ±4.2

Table 4: Performance of FT-GP on a data set with node features (ENZYMES) and a data set without node features (IMDB-BINARY) for varying number of evaluation points M.

K	ENZYMES	IMDB-BINARY
5	64.3 ±5.0	74.6 ±3.8
10	63.8 ±5.3	74.6 ±4.1
15	64.8 ±5.4	74.3 ±3.0
20	65.5 ±5.5	74.2 ±4.1

Table 5: Performance of WT-GP on a data set with node features (ENZYMES) and a data set without node features (IMDB-BINARY) for varying number of filters K.

We find that even though the model requires no learning to obtain these spectral features, it performs competitively to graph neural network and kernel baselines. The second approach learns more complex wavelet filters and compares graphs based on the corresponding filtered node features. It outperforms the first approach both on real-world and synthetic data sets. Our work indicates that spectral features constitute a powerful basis for graph classification both on their own and even more so when combined with the learning capabilities of GPs.

Acknowledgements

FLO acknowledges funding from the Huawei Hisilicon Studentship at the Department of Computer Science and Technology of the University of Cambridge. X.D. acknowledges support from the Oxford-Man Institute of Quantitative Finance and the EPSRC (EP/T023333/1).

References

Karsten Borgwardt, Elisabetta Ghisu, Felipe Llinares-López, Leslie O'Bray, Bastian Rieck, et al. Graph kernels: State-of-the-art and future challenges. *Foundations and Trends® in Machine Learning*, 13(5-6), 2020.

Karsten M. Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In *Fifth IEEE International Conference on Data Mining*, 2005.

Karsten M. Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V. N. Vishwanathan, Alex J. Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 21, 2005.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *2nd International Conference on Learning Representations*, 2014.

Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16 (5), 1995.

Fan R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

Hanjun Dai, Bo Dai, and Le Song. Discriminative embeddings of latent variable models for structured data. In *Proceedings of the 33rd International Conference on Machine Learning*, 2016.

Asim Kumar Debnath, Rosa L. Lopez de Compadre, Gargi Debnath, Alan J. Shusterman, and Corwin Hansch. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of Medicinal Chemistry*, 34(2), 1991.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, 2016.

Xiaowen Dong, Antonio Ortega, Pascal Frossard, and Pierre Vandergheynst. Inference of mobility patterns via spectral graph wavelets. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

Xiaowen Dong, Dorina Thanou, Laura Toni, Michael M. Bronstein, and Pascal Frossard. Graph signal processing for machine learning: A review and new perspectives. *IEEE Signal Processing Magazine*, 37(6), 2020.

David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*, 2015.

Paul Erdös and Alfréd Rényi. On random graphs I. *Publicationes Mathematicae Debrecen*, 1959.

Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. In *8th International Conference on Learning Representations*, 2020.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, 2017.

David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2), 2011.

James Hensman, Nicolò Fusi, and Neil D. Lawrence. Gaussian processes for big data. In *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence*, 2013.

James Hensman, Alexander G. de G. Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, 2015.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2), 1983.

Ningyuan Teresa Huang and Soledad Villar. A short tutorial on the weisfeiler-lehman test and its variants. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations*, 2017.

Risi Kondor and Horace Pan. The multiscale laplacian graph kernel. In *Advances in neural information processing systems*, 2016.

Nils M Kriege, Fredrik D Johansson, and Christopher Morris. A survey on graph kernels. *Applied Network Science*, 5(1), 2020.

Luc Le Magoarou, Rémi Gribonval, and Nicolas Tremblay. Approximate fast graph fourier transforms via multilayer sparse approximations. *IEEE Transactions on Signal and Information Processing over Networks*, 4(2), 2018.

Bin Li, Xingquan Zhu, Lianhua Chi, and Chengqi Zhang. Nested subtree hash kernels for large-scale graph classification over streams. In *2012 IEEE 12th International Conference on Data Mining*, 2012.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(3, (Ser. B)), 1989.

Ryan Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Relational pooling for graph representations. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Giannis Nikolentzos, Giannis Siglidis, and Michalis Vazirgiannis. Graph kernels: A survey. *Journal of Artificial Intelligence Research*, 72, 2021.

Felix Opolka, Yin-Cong Zhi, Pietro Liò, and Xiaowen Dong. Adaptive gaussian processes on graphs via spectral graph wavelets. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, 2022.

Antonio Ortega. *Introduction to Graph Signal Processing*. Cambridge University Press, 2022.

Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106(5), 2018.

Edouard Pineau. Using laplacian spectrum as graph feature representation. *arXiv preprint arXiv:1912.00735*, 2019.

Robert Porter and Nishan Canagarajah. A robust automatic clustering scheme for image segmentation using wavelets. *IEEE transactions on image processing*, 5(4), 1996.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

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

David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3), 2013.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, 2009.

Matteo Togninalli, Elisabetta Ghisu, Felipe Llinares-López, Bastian Rieck, and Karsten Borgwardt. Wasserstein weisfeiler-lehman graph kernels. *Advances in Neural Information Processing Systems*, 32, 2019.

Nikil Wale and George Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. In *Sixth International Conference on Data Mining*, 2006.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *7th International Conference on Learning Representations*, 2019.

Pinar Yanardag and S.V.N. Vishwanathan. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015.

Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L. Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems*, 2018.

Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018.