ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

An Empirical Study of Over-Parameterized Neural Models based on Graph Random Features

Nicolò Navarin[1], Luca Pasa[1], Luca Oneto[2] and Alessandro Sperduti[1,3] *

1 - University of Padua - Via Trieste 63, 35121, Padua - Italy
2 - University of Genoa - Via Opera Pia 11a, 16145, Genoa - Italy
3 - University of Trento - Via Sommarive, 9 I-38123 Povo - Italy

Abstract. In this paper, we investigate neural models based on graph random features. In particular, we aim to understand when over-parameterization, namely generating more features than the ones necessary to interpolate, may be beneficial for the generalization of the resulting models. Exploiting the algorithmic stability framework and based on empirical evidences from several commonly adopted graph datasets, we will shed some light on this issue.

1 Introduction

When dealing with large-scale problems or aiming for computational efficiency, a commonly adopted approach is to exploit feature sketching (random projections followed by a component-wise non-linearity) in conjunction with a linear classifier. The behavior of linear classifiers on increasing number of random features has been studied from the theoretical point of view, in particular for ridge regression [1] and based on stochastic gradient descent [2], showing, theoretically and empirically, the presence of the double descent and best-overfit phenomena [3–6], namely the ability of these models to improve the generalization performance in over-parameterization, i.e., when having much more parameters than the ones needed to interpolate. To the best of authors' knowledge, no study in literature considers the case of random graph neural features. We speculate that the reason is that the majority of randomized graph networks in literature are based on a recurrent scheme of Reservoir Computing that, while generating expressive features, can result in a high computational complexity with hundreds or thousands of features [7].

Recently, an untrained graph neural model has been proposed [8] that can efficiently generate thousands of non-linear features. The classification (or regression) task is performed by a linear model, e.g. ridge classification, starting from the randomized features. This model allows us to study the behavior of

*This work was partly funded by: the SID/BIRD project *Deep Graph Memory Networks*, Department of Mathematics, University of Padua; the project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU; the PON R&I 2014-2020 project *Smart Waste Treatment* founded by the FSE REAC-EU; the project "iNEST: Interconnected Nord-Est Innovation Ecosystem" funded under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.5 - Call for tender No. 3277 of 30 December 2021 of Italian Ministry of University and Research funded by the European Union – NextGenerationEU, project code: ECS00000043, Concession Decree No. 1058 of June 23, 2022, CUP C43C22000340006.

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

untrained graph neural models when varying the number of generated features. In particular, we aim to understand when it may be convenient from the generalization performance point of view to generate a large number of random graph features (going beyond the interpolation threshold). For this purpose, we will leverage the Algorithmic Stability framework [3] and empirically show its potentiality in giving insights on the generalization ability of over-parameterized neural models based on graph random features. While in this paper we focus on a single randomized neural model for space constraints, it would be possible, in the future, to study other untrained feature extraction methods for graphs.

2 Background and Related Works

In structured data domains the models proposed in the last few years show increasing complexity, leading to novel architectures with a considerably high number of parameters. Unfortunately, this implies a high computational cost, especially in training the models.

Authors of [9] proposed the first model for graph domain that exploits the reservoir computing framework. The proposed model, dubbed GraphESN, is composed of a non-linear reservoir and a feed-forward linear readout. The computation of the global state involves an iterative process (run till convergence). Authors of [10] propose a model, dubbed Multi-resolution Reservoir Graph Neural Network (MRGNN) model, that exploits a Reservoir Convolutional layer for graphs able to simultaneously and directly consider all topological receptive fields up to k-hops. Recently, authors of [11] explored randomized graph convolutions for the task of node classification (differently from this paper in which we consider the more challenging setting of graph classification). The authors propose a single-layer architecture defined as $\mathbf{Z} = \sigma(\mathbf{A}^2\mathbf{X}\mathbf{W})\beta$, where σ is the sigmoid function, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the adjacency matrix of the graph, $W \in \mathbb{R}^{d\times m}$ is the (random) wight matrix for m hidden neurons (that is left untrained), and β are the trained output weights. Simultaneously, in the unstructured domain, researchers have started to questioning themself about a mechanisms, called over-parametrization, that even if studied from a long time [12], have recently unlocked the potentiality of deep network such as the the large language models [13]. In simple words, over-parameterization means that we leverage models with more parameters than the ones necessary to interpolate the data, namely perfectly fit or memorize the available data, to obtain good generalization performance [3]. Recent results showed that, even if counter-intuitive, increasing the number of parameters after the interpolating threshold can increase the generalization ability of the model since it increases its stability [3, 4]. Unfortunately, stability is not always simple to compute to evaluate the generalization ability of a model [3]. We compute stability with the condition of the Gram matrix [3–5] induced by the random graph projection.

3 Graph Random Features and Algorithmic Stability

In this paper, we aim to study if and under which conditions it is convenient to develop overparametrized graph models when using graph random features. We

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

seek an answer in recent research exploiting measures from statistical learning theory, such as the Algorithmic Stability, and exploring their relationship with the observed empirical behaviour of the generalization error.

Let us first summarize a recent approach for generating random features based on graph neural networks [8], that we will exploit in this paper. The randomized architecture is inspired by fully trained graph neural networks, including the non-linearity scheme. Specifically, multiple graph convolution layers are stacked, each one followed by a hyperbolic tangent element-wise non-linear activation function [8]. The authors considered the GCN [14] graph convolution. The hidden node representation computed by the l-th layer is defined as: $\mathbf{H}^{(l)} = tanh(\mathbf{S}\mathbf{H}^{(l-1)}\mathbf{W}^{(l)})$ where $\mathbf{S} = (\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}})$ is the normalized Laplacian adopted by the GCN, $\mathbf{D}$ is the diagonal degree matrix where $d_{ii} = \sum_j a_{ij}$ and, $\mathbf{W}^{(l)}$ are the layer parameters and $\mathbf{H}^0 = \mathbf{X}$. Note that we omit the bias terms for the sake of simplicity. The final node representations are obtained concatenating the representation computed by each graph convolution layer, i.e. $\mathbf{H} = [\mathbf{H}^{(1)}, \ldots, \mathbf{H}^{(\mathbf{L})}]$, where L is the number of layers of the network. Crucially, the weight values in $\mathbf{W}^{(l)}$ are initialized randomly and left untrained and initialized with the Glorot uniform approach [15] with a gain hyperparameter θ to control the effective scaling of $\mathbf{W}^{(l)}$. In the resulting process, a weight matrix of shape $n \times m$ will have entries sampled from a uniform distribution $\mathcal{U}(-a, a)$ where $a = \theta\sqrt{6/n+m}$. To perform graph-level tasks, a *global pooling* layer is exploited to obtain a single representation for the whole graph. The Percentage of Positive Values (PPV) is a non-differentiable pooling mechanism used in randomized networks and defined as: $PPV(\mathbf{z}) = \frac{1}{n}\sum_{i=0}^{n-1} I[z_i > 0]$, where $I[z_i > 0]$ is the indicator function which value is 1 if $z_i > 0$, 0 otherwise. Authors proposed to use as global pooling both the Global Max Pooling and PPV, concatenating the resulting representations. Note that this choice doubles the size of the global graph representation compared to the representations of the single nodes provided in output by the untrained graph convolution. Finally, the authors proposed to use a ridge classifier as a readout for its computational efficiency.

Let us now consider the Algorithmic Stability. From this random graph representation, it is possible to compute an approximation of a specific notion of Algorithmic Stability, the Hypothesis Stability, that, together with the training error, are able to give insights on the generalization error and are fast to compute [3]. Let $\mathbf{h}_g$ be the hidden representation for a graph computed by the model presented before, and $\mathbf{H}$ be the matrix collecting the representations of all training graphs. We can consider this representation as the input of a linear model (the readout). It has been shown that the Hypothesis Stability $\mathcal{A}$ is proportional to the conditioning of the Gramian matrix $\mathbf{H}\mathbf{H}^\intercal$, i.e., $\mathcal{A} \propto Cond(\mathbf{H}\mathbf{H}^\intercal)$ where $Cond$ is a function computing the condition number of a matrix with eigenvalues λ_i, i.e., ${}^{\lambda_{\max}}/_{\lambda_{\min}}$. Thus, we can study the relationship between the approximation of the Hypothesis Stability of such representation and the generalization capabilities of the models trained on such representations. In fact, the smaller the training error and the smaller the stability, the higher the generalization ability of the learned model should be [3].

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

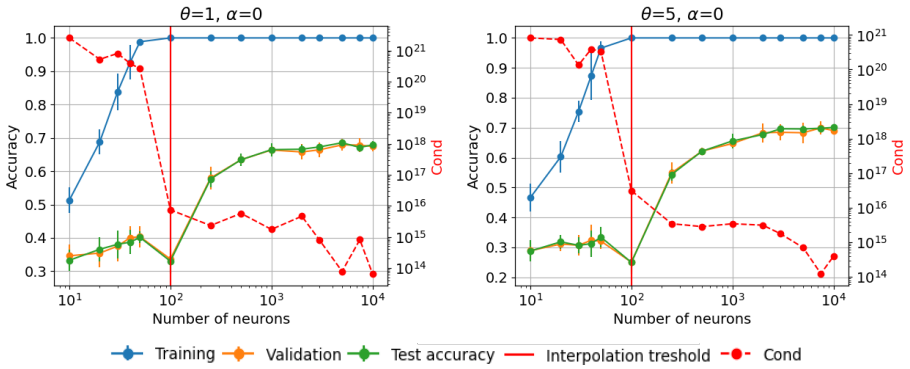

Fig. 1: ENZYMES dataset.

4 Experimental results

In this section, we present some empirical evidences regarding the ability of Algorithmic Stability to explain the good generalization abilities of over-parameterized neural models based on graph random features.

Datasets. Among the different graph classification benchmark datasets available we considered three datasets related to bio-informatics: ENZYMES [16], D&D [16], and NCI1 [17].

Experimental setup. We study the behaviour of the model described in Section 3 varying the number of neurons (parameters) for certain configurations of the hyperparameters. Due to space constraints only a subset, the most informative, of the results are reported. We fixed the number of layers to four. We plot the performance varying the number of neurons for each layer from 10 to $10,000$ ($5,000$ for D&D) per layer. Since we use four layers and concatenate two different readouts, the resulting graph representation is up to size $80,000$ ($40,000$ for D&D). However, since the weights are not trained, we just have to perform the forward phase which is extremely fast even with a high number of features to extract. Then, we trained a ridge classifier characterized by a regularization hyperparameter α taking values in the set $\{0, 10^{-4}, \ldots, 10^{5}\}$. We also considered multiple values of θ, for weight initialization, in the set $\{0.01, 0.1, 1, 3, 5, 10, 30, 50\}$.

Results and Discussion. In this section, we report for different datasets and different hyperparameters configurations that can reach competitive performance, the training, validation, and test accuracies, varying the number of neurons. We also report the Algorithmic Stability estimated via the condition number of the Gram matrix (see Section 3), and the interpolation threshold (i.e., the value of the number of neurons such that the accuracy on the training set is 100% without regularization). Figure 1 reports the results for the ENZIMES dataset for different values of θ. From Figure 1 we can see that there are two different regimes with a phase change. The first regime is the under-parameterized one, in which the actual dimension of the feature space is smaller than the interpolation threshold. In this setting there is a trade-off between accuracy, number

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

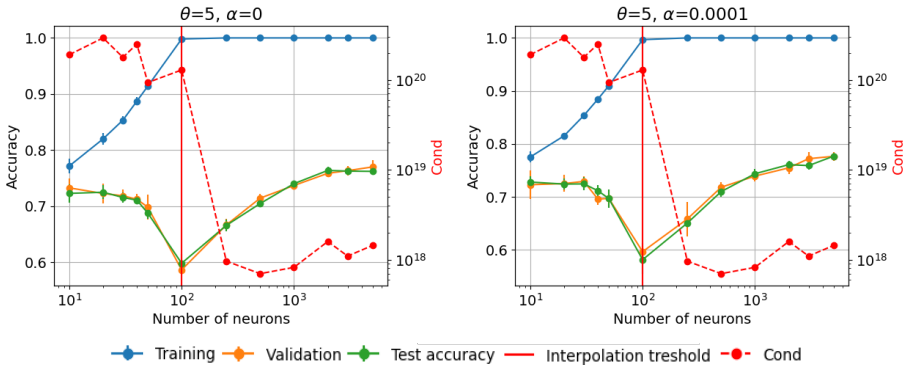

Fig. 2: D&D dataset.

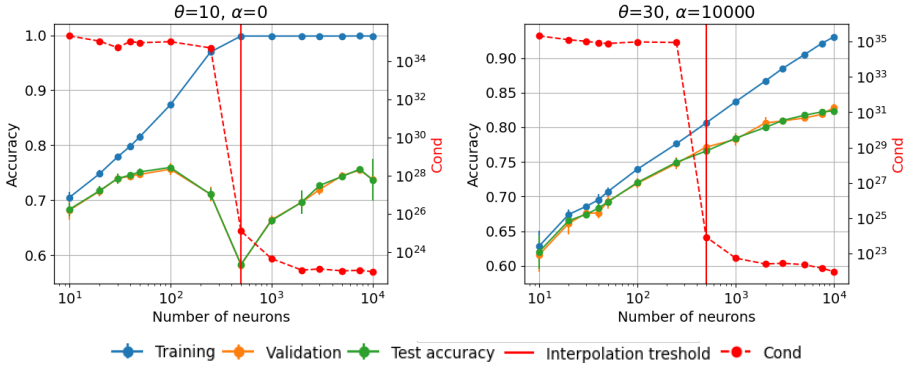

Fig. 3: NCI1 dataset.

of neurons and error, typical of the classical bias-variance trade-off [3]. Note that in this setting the Algorithmic Stability is, relatively, quite high. The second regime is the over-parameterized one, which is the one after the interpolation threshold, that is characterized by two new phenomena. The first one is that the accuracy on the test set starts to increase even if the model is interpolating (double-descent or best-overfit behavior [3]) but in correspondence of the interpolation threshold there is a change of phase in the Algorithmic Stability which suddenly drops around this threshold and then generally continues to decrease after the drop. In other words, Algorithmic Stability is able to tell us that adding more neurons can actually improve generalization instead of hurting it: in fact, in the over-parameterized regimes, accuracies increase while stability decreases which is a clear sign of increasing generalization [3]. Figures 2 and 3 report, similarly to what reported in Figure 1 for the ENZYMES dataset, the results of the D&D and NCI1 datasets. From Figures 2 and 3 it is possible to come up with the same observations derived for the ENZYMES dataset, confirming the empirical evidence that the Algorithmic Stability is able to explain, and suggest, when over-parameterization can be beneficial for the generalization

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

ability of neural models based on graph random features. Note also, that best performances are not always reached with simple empirical risk minimization and sometimes regularization ($\alpha > 0$) is needed but the Algorithmic Stability is always able to provide the necessary insights.

5 Conclusions

In this paper, we investigated the generalization abilities of over-parameterized neural models based on graph random features. In particular, our aim was to understand when over-parameterization, namely generating more features than the ones necessary to interpolate, may be beneficial for the generalization of the resulting models. For this purpose, we rely on the Algorithmic Stability framework that together with empirical evidences from several commonly adopted graph datasets helped us understand why more parameters can improve generalization. Of course, this work is a preliminary but promising step in understating over-parameterized neural models based on graph random features and more theoretical and empirical evidences need to be derived.

References

[1] Z. Chen and H. Schaeffer. Conditioning of Random Feature Matrices: Double Descent and Generalization Error. *arXiv preprint arXiv:2110.11477*, 2021.

[2] F. Liu, J. Suykens, and V. Cevher. On the double descent of random features models trained with sgd. In *Neural Information Processing Systems*, 2022.

[3] L. Oneto, S. Ridella, and D. Anguita. Do we really need a new theory to understand over-parameterization? *Neurocomputing*, 2023.

[4] A. Rangamani, L. Rosasco, and T. Poggio. For interpolating kernel machines, minimizing the norm of the erm solution minimizes stability. *arXiv preprint arXiv:2006.15522*, 2020.

[5] T. Poggio, G. Kur, and A. Banburski. Double descent in the condition number. *arXiv preprint arXiv:1912.06190*, 2019.

[6] B. Ghorbani, S. Mei, T. Misiakiewicz, and A. Montanari. Linearized two-layers neural networks in high dimension. *The Annals of Statistics*, 49(2):1029–1054, 2021.

[7] C. Gallicchio and A. Micheli. Fast and deep graph neural networks. In *AAAI conference on artificial intelligence*, 2020.

[8] N. Navarin, L. Pasa, C. Gallicchio, and A. Sperduti. An untrained neural model for fast and accurate graph classification. In *International Conference on Artificial Neural Networks*, 2023.

[9] C. Gallicchio and A. Micheli. Graph Echo State Networks. In *International Joint Conference on Neural Networks*, 2010.

[10] L. Pasa, N. Navarin, and A. Sperduti. Multiresolution reservoir graph neural network. *IEEE Transaction and Neural Networks Learning System*, 33(6):2642–2653, 2022.

[11] C. Huang, M. Li, F. Cao, H. Fujita, Z. Li, X. Wu, and M. Li. Are Graph Convolutional Networks With Random Weights Feasible? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2751–2768, 2023.

[12] M. Loog, T. Viering, A. Mey, and Others. A brief prehistory of double descent. *Proceedings of the National Academy of Sciences*, 117(20):10625–10626, 2020.

[13] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[14] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[15] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, 2010.

[16] K. M. Borgwardt, C. S. Ong, S. Schönauer, and Others. Protein function prediction via graph kernels. *Bioinformatics*, 21:i47–i56, 2005.

[17] N. Wale, I. A. Watson, and G. Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 2008.