LEARNING EXPANDING GRAPHS FOR SIGNAL INTERPOLATION

Bishwadeep Das and Elvin Isufi

ABSTRACT

Performing signal processing over graphs requires knowledge of the underlying fixed topology. However, graphs often grow in size with new nodes appearing over time, whose connectivity is typically unknown; hence, making more challenging the downstream tasks in applications like cold start recommendation. We address such a challenge for signal interpolation at the incoming nodes blind to the topological connectivity of the specific node. Specifically, we propose a stochastic attachment model for incoming nodes parameterized by the attachment probabilities and edge weights. We estimate these parameters in a data-driven fashion by relying only on the attachment behaviour of earlier incoming nodes with the goal of interpolating the signal value. We study the non-convexity of the problem at hand, derive conditions when it can be marginally convexified, and propose an alternating projected descent approach between estimating the attachment probabilities and the edge weights. Numerical experiments with synthetic and real data dealing in cold start collaborative filtering corroborate our findings.

Index Terms— Incoming nodes, expanded graphs, graph signal interpolation, cold start.

1. INTRODUCTION

Graph Signal Processing (GSP) leverages the relationships between data points to subsequently process them for a multitude of classical applications [1, 2]. The graph is commonly considered of fixed size and can have either a fixed [3] or a changing number of edges [4]. However, graphs often grow in size with new nodes becoming available continuously [5, 6]. A typical setting is in cold start collaborative filtering. Here, a new item becomes available but we have no information to connect it with the available ones, thereby affecting the subsequent recommendation [7]. Hence, modelling the attachment behaviour of incoming nodes is paramount to interpolate the rating value at this new item [8, 9]. This is relevant not only in recommender systems [7] but also in data privacy over networks [10], which can manifest in inductive learning on graphs [11] where unseen nodes need to be classified.

There exists a set of diverse viewpoints to approach inference of nodal connections [12]. *Topology Identification* via GSP estimates a static [3, 13] or a time varying topology [4, 14] of fixed size by using different priors such as signal smoothness [15], realizations from diffusion [16], or Gaussian processes [17]. *Statistical* methods utilize stochastic attachment models based on the existing topology to drive the incoming node attachment. The Erdős-Rényi (ER) [5] and Barabasi-Albert [6] models are prime examples of this category, although more complex models also exist [18, 19]. *Link Prediction* approaches infer the existence of unobserved edges between existing nodes, given edges and/or feature information of these nodes

[20]. These include probabilistic approaches [21], similarity-based approaches [22, 23], and classifier-based approaches [24]. Some more recent works provide a synthesis between graphs growth models and processing signals over them. The works in [25, 26] estimate node connectivity for graphs drawn from ER and Bollobás-Riordan models by observing only on a subset of nodes signals that evolve according to an autoregressive process. The works in [10, 27] solve regression tasks over expanding graphs but require the connectivity of the incoming nodes.

Altogether, these contributions consider settings where we have a fixed number of nodes, available nodal features, or know the incoming node attachment. In absence of node feature signals and attachment information, both the GSP and the link prediction approaches become inapplicable. We may in these cases rely on stochastic models but they detach the connectivity modelling from the processing task, which results in sub-optimal performance. Likewise, we can also adapt the approaches in [25, 26] to account for incoming nodes but the considered growth and signal models do not always hold.

To overcome these limitations, we propose a stochastic attachment model for arbitrary graphs and utilize the information only on the existing graph to interpolate the signal value at the incoming node. The proposed model is parameterized by the attachment probabilities and the edge weights of the incoming node. These parameters are estimated in a data- and task-driven fashion from a training set of earlier incoming nodes. To estimate the parameters, we solve an empirical risk minimization problem by minimizing the signal reconstruction mean squared error on the incoming node regularized to the connectivity pattern. We study the convexity of the problem and provide an alternating projected descent algorithm to solve it. Finally, we corroborate the proposed model and compare it with baselines on synthetic and real applications dealing with the cold start problem in collaborative filtering.

2. PROBLEM FORMULATION

Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of N nodes in set $\mathcal{V} = \{v_1, \ldots, v_N\}$ and E edges in set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Let $\mathbf{A}$ be the graph adjacency matrix such that $A_{ij} \neq 0$ only if $(v_i, v_j) \in \mathcal{E}$. An incoming node v_+ connects to $\mathcal{G}$ and forms a directed expanded graph $\mathcal{G}_+ = (\mathcal{V}_+, \mathcal{E}_+)$ with node set $\mathcal{V}_+ = \mathcal{V} \cup v_+$ and edge set $\mathcal{E}_+ = \mathcal{E} \cup (v_+, v_i)$ for all new directed edges (v_+, v_i) landing at v_+. We represent the attachment pattern of node v_+ by $\mathbf{a}_+ \in \mathbb{R}^N$ where $[\mathbf{a}_+]_i = w_i$ is the weight of edge (v_+, v_i). The $(N+1) \times (N+1)$ adjacency matrix of graph $\mathcal{G}_+$ is

$$\mathbf{A}_+ = \begin{bmatrix} \mathbf{A} & \mathbf{0}_+ \\ \mathbf{a}_+^\top & 0 \end{bmatrix} \tag{1}$$

in which the last row and column represent the connectivity of v_+.[1] Node v_+ connects to any existing node $v_i \in \mathcal{V}$ independently with

---

Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands. e-mail: {b.das, e.isufi-1}@tudelft.nl

[1]We consider for simplicity of exposition the attachment of a single node. For multiple nodes, $\mathbf{a}_+$ becomes a matrix having a column per new node.

probability p_i. We collect the probabilities of attachment in vector $\mathbf{p} = [p_1, \ldots, p_N]^\top$. Then, the attachment pattern $\mathbf{a}_+$ is a random vector with each entry being an independent Bernoulli random variable weighted by scalar w_i. I.e., the ith element of $\mathbf{a}_+$ is

$$[\mathbf{a}_+]_i = \begin{cases} w_i & \text{with probability } p_i \\ 0 & \text{with probability } (1 - p_i) \end{cases} \quad (2)$$

for $i = 1, \ldots, N$. The expectation of $\mathbf{a}_+$ is $\mathbb{E}[\mathbf{a}_+] = \mathbf{p} \circ \mathbf{w}$ where vector $\mathbf{w} = [w_1, \ldots, w_N]^\top$ contains the weights of all edges (v_+, v_i). Likewise, the variance of $[\mathbf{a}_+]_i$ is $\text{var}([\mathbf{a}_+]_i) = w_i^2 p_i(1 - p_i)$ and the respective covariance matrix is

$$\mathbf{\Sigma}_+ = \text{diag}(\mathbf{w}^{\circ 2} \circ \mathbf{p} \circ (\mathbf{1} - \mathbf{p})) \quad (3)$$

where $\mathbf{a}^{\circ k} = \mathbf{a} \circ \ldots \circ \mathbf{a}$ is the Hadamard product of $\mathbf{a}$ with itself k times. The expected adjacency matrix of $\mathcal{G}_+$ is

$$\mathbb{E}[\mathbf{A}_+] = \begin{bmatrix} \mathbf{A} & \mathbf{0}_+ \\ (\mathbf{p} \circ \mathbf{w})^\top & 0 \end{bmatrix}. \quad (4)$$

Let $\mathbf{x} = [x_1, \ldots, x_N]^\top$ be the graph signal on $\mathcal{G}$ with x_i the signal at node v_i. Processing this signal by accounting for its coupling with $\mathcal{G}$ is key to several network data tasks [1]. E.g., in item-item collaborative filtering, the signal collects the ratings a specific user has provided to the existing items [8]. When a new item node v_+ becomes available, the task is to predict the signal rating value x_+ at this node. To solve such a task without knowing the exact connectivity of v_+, we rely on the stochastic models governed by $\mathbf{p}$ and $\mathbf{w}$, which in turn are unknown.

To identify a task-specific connectivity for the incoming nodes, we merge data-driven solutions with the above generic statistical model. Given graph $\mathcal{G}$ and a training set of attachment patterns for incoming nodes $\mathcal{T} = \{(v_{t+}, x_{t+}, \mathbf{a}_{t+}, \mathbf{b}_{t+})\}_t$ we infer the attachment probability vector $\mathbf{p}$ and weight vector $\mathbf{w}$ in an empirical risk minimization fashion. Each element in $\mathcal{T}$ comprises an incoming node v_{t+}, the signal at this node x_{t+}, the attachment vector $\mathbf{a}_{t+}$, and its binary attachment pattern $\mathbf{b}_{t+}$ –for a recommender system with cold start, we build an item-item graph $\mathcal{G}$ and treat some items as cold-starters whose ratings and connectivity are known – and define a task-specific loss $f_\mathcal{T}(\mathbf{p}, \mathbf{w}, \mathbf{a}_{t+}, \mathbf{x}_{t+})$ measuring the signal interpolation (rating) performance. Specifically, we solve the statistical optimization problem

$$\min_{\mathbf{p}, \mathbf{w}} \mathbb{E}\big[f_\mathcal{T}(\mathbf{p}, \mathbf{w}, \mathbf{a}_{t+}, \mathbf{x}_{t+})\big] + g_\mathcal{T}(\mathbf{p}, \mathbf{b}_{t+}) + h_\mathcal{T}(\mathbf{w}, \mathbf{a}_{t+})$$
$$\text{subject to} \quad \mathbf{p} \in [0, 1]^N, \mathbf{w} \in \mathcal{W} \quad (5)$$

where $g_\mathcal{T}(\mathbf{p}, \mathbf{b}_{t+})$ and $h_\mathcal{T}(\mathbf{w}, \mathbf{a}_{t+})$ are regularizers and set $\mathcal{W}$ constraints the edge-weights, e.g., non-negative or finite.

3. SIGNAL INTERPOLATION ON INCOMING NODES

We measure the signal interpolation performance via the mean square error (MSE) and use graph filters [28, 29] to diffuse the signal over $\mathcal{G}_+$. Graph filters are well-established local operators for processing graph data. They combine successive shift operations over the topology and have found applications in a variety of domains [30] Consider the expanded graph signal $\mathbf{x}_+ = [\mathbf{x}^\top, 0]^\top$, where zero is the signal value at node v_+. The output $\mathbf{y}_+$ of an order L graph filter is

$$\mathbf{y}_+ = \sum_{l=1}^{L} h_l \mathbf{A}_+^l \mathbf{x}_+ \quad (6)$$

Algorithm 1 Alternating projected gradient descent for (8).

1: **Input:** Graph $\mathcal{G}$, training set $\mathcal{T}$, graph signal $\mathbf{x}$, adjacency matrix $\mathbf{A}$, number of iterations K, cost C, learning rates λ_p, λ_w.
2: **Initialization**: $\mathbf{p} = \mathbf{p}^0$, $\mathbf{w} = \mathbf{w}^0$ randomly, $k = 0$.
3: **for** $k \leq K$ **do**
4:      $\mathbf{p}$ gradient: $\tilde{\mathbf{p}}^{k+1} = \mathbf{p}^k - \lambda_p \nabla_\mathbf{p} C(\mathbf{p}^k, \mathbf{w}^k)$;
5:      Projection: $\mathbf{p}^{k+1} = \underset{[0,1]^N}{\Pi}(\tilde{\mathbf{p}}^{k+1})$;
6:      $\mathbf{w}$ gradient: $\mathbf{w}^{k+1} = \mathbf{w}^k - \lambda_w \nabla_w C(\mathbf{p}^{k+1}, \mathbf{w}^k)$;
7:      Projection: $\mathbf{w}^{k+1} = \underset{\mathcal{W}}{\Pi}(\tilde{\mathbf{w}}^{k+1})$;
8: **end for**

---

where $\mathbf{h} = [h_1, \ldots, h_L]^\top$ are the filter coefficients. Nodes up to the L-hop neighborhood of v_+ contribute to its interpolated signal. Note that in (6), we ignore $l = 0$ because it does not contribute to the output $[\mathbf{y}_+]_{N+1}$ at v_+. Given the percolated signal $[\mathbf{y}_+]_{N+1}$, the following proposition quantifies the signal interpolation MSE as a function of model parameters $\mathbf{p}$ and $\mathbf{w}$.

Proposition 1. *Given $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with adjacency matrix $\mathbf{A}$ and signal $\mathbf{x}$ and let $\mathbf{A}_x = [\mathbf{x}, \ldots, \mathbf{A}^{L-1}\mathbf{x}]$. Given also an incoming node v_+ with true signal $x_+^\star$ attaching to $\mathcal{G}$ with probabilities $\mathbf{p}$ and edge weights $\mathbf{w}$, forming graph $\mathcal{G}_+$ with the expanded adjacency matrix $\mathbf{A}_+$ [cf.(1)]. The MSE of the interpolated signal $\mathbf{y}_+$ at node v_+ by an order L graph filter [cf.(6)] is*

$$\text{MSE}(\mathbf{p}, \mathbf{w}) = ||(\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_+^\star||_2^2 + \mathbf{h}^\top \mathbf{A}_x^\top \mathbf{\Sigma}_+ \mathbf{A}_x \mathbf{h} \quad (7)$$

Proof. See appendix. $\square$

Besides quantifying the MSE, Proposition 1 provides also insights on the role of parameters $\mathbf{p}$ and $\mathbf{w}$. The first term on the RHS of (7) captures the model bias w.r.t. the true signal $x_+^\star$. The prediction output is the dot product between the filtered output of $\mathbf{x}$ over $\mathcal{G}$, $\mathbf{A}_x \mathbf{h}$ with the expected attachment vector $\mathbf{w} \circ \mathbf{p}$. Minimizing the bias implies selecting a pair $(\mathbf{p}, \mathbf{w})$ that combines the signal at each $v \in \mathcal{V}$ to match $x_+^\star$. The second term $\mathbf{h}^\top \mathbf{A}_x^\top \mathbf{\Sigma}_+ \mathbf{A}_x \mathbf{h} = ||\mathbf{A}_x \mathbf{h}||_{\mathbf{\Sigma}_+}^2$ is the squared norm of the filtered signal weighted by the attachment variances. Minimizing this term might give trivial solutions such as $\mathbf{p} = \mathbf{1}$ and $\mathbf{p} = \mathbf{0}$. Thus, regularizers are needed for $\mathbf{p}$ and $\mathbf{w}$. We also remark that in (7) the Lth shift $\mathbf{A}^L \mathbf{x}_+$ does not contribute to the MSE because of the structure of matrix $\mathbf{A}_+$ in (1).

Optimization Problem. With this in place, we can formulate problem (5) as

$$\min_{\mathbf{p}, \mathbf{w}} \quad \text{MSE}_\mathcal{T}(\mathbf{p}, \mathbf{w}) + \sum_{t=1}^{|\mathcal{T}|}\left(\mu_p||\mathbf{p} - \mathbf{b}_{t+}||_q^q + \mu_w||\mathbf{w} - \mathbf{a}_{t+}||_q^q\right)$$
$$\text{subject to} \quad \mathbf{p} \in [0, 1]^N, \mathbf{w} \in \mathcal{W} \quad (8)$$

where $\text{MSE}_\mathcal{T}(\mathbf{p}, \mathbf{w})$ is the empirical MSE over the training set $\mathcal{T}$, $\sum_{t=1}^{|\mathcal{T}|} ||\mathbf{p} - \mathbf{b}_{t+}||_q^q$ and $\sum_{t=1}^{|\mathcal{T}|} ||\mathbf{w} - \mathbf{a}_{t+}||_q^q$ are regularizers with weights $\mu_w > 0$, $\mu_p > 0$. respectively and $q \in \{1, 2\}$. Problem (8) is non-convex in $\mathbf{w}$ and $\mathbf{p}$; it is marginally convex in $\mathbf{w}$ but not always in $\mathbf{p}$ due to the variance term in (7). We solve (8) with alternating projected gradient descent. Algorithm 1 summarizes the main steps. The gradients of the cost $C(\mathbf{p}, \mathbf{w})$ in (8) w.r.t. $\mathbf{p}$ and $\mathbf{w}$ for $q = 2$ are

$$\nabla_p C(\mathbf{p}, \mathbf{w}) = 2\sum_{t=1}^{|\mathcal{T}|}((\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_{t+})(\mathbf{w} \circ \mathbf{A}_x \mathbf{h})$$
$$+ |\mathcal{T}|(\mathbf{A}_x \mathbf{h})^{\circ 2} \circ (\mathbf{w}^{\circ 2}) \circ (\mathbf{1} - 2\mathbf{p}) + 2\mu_p \sum_{t=1}^{|\mathcal{T}|}(\mathbf{p} - \mathbf{b}_{t+}) \quad (9)$$
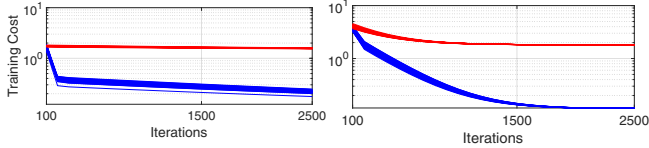
Fig. 1. Convergence of the training cost for Algorithm 1 solving Problem (8) being non-convex (blue) and marginally convex (red) [cf. (11)]. (Left) ER graph model; (Right) BA graph model.

$$\nabla_w C(\mathbf{p}, \mathbf{w}) = 2 \sum_{t=1}^{|\mathcal{T}|} ((\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_{t+})(\mathbf{p} \circ \mathbf{A}_x \mathbf{h})$$
$$+ 2|\mathcal{T}|(\mathbf{A}_x \mathbf{h})^{\circ 2} \circ \mathbf{w} \circ \mathbf{p} \circ (\mathbf{1} - \mathbf{p}) + 2\mu_w \sum_{t=1}^{|\mathcal{T}|} (\mathbf{w} - \mathbf{a}_{t+}) \qquad (10)$$

Instead, for $q = 1$, we replace terms $2\mu_p(\mathbf{p} - \mathbf{b}_{t+})$ and $2\mu_w(\mathbf{w} - \mathbf{a}_{t+})$ with $\text{sign}(\mathbf{p} - \mathbf{b}_{t+})$ and $\text{sign}(\mathbf{w} - \mathbf{a}_{t+})$ respectively. We have observed that norm one regularizers are more applicable to $\mathbf{p}$ than to $\mathbf{w}$ because some weights would be zero even if a node attaches with a high probability. While we can use Algorithm 1 to solve the non-convex case of problem (8), the following corollary provides a sufficient condition for problem (8) to be marginally convex also in $\mathbf{p}$. The proposed method has a complexity of order $\mathcal{O}(LE)$, where L is the filter order and E the number of edges in the existing graph.

Corollary 1. *The cost in Problem* (8) *is marginally convex in* $\mathbf{p}$ *if the regularization weight* $\mu_p > 0$ *satisfies*

$$\mu_p \geq w_h^2 \max_{i \in \{1,\dots,N\}} ([\mathbf{A}_x \mathbf{h}]_i)^2 - ||\mathbf{w} \circ \mathbf{A}_x \mathbf{h}||_2^2. \qquad (11)$$

Proof. See appendix. □

While guaranteeing convexity, condition (11) may lead to an optimum that is worse than the local optima of its non-convex counterpart due to a greater focus on the training attachment patterns than on the task-specific cost. We shall corroborate this next.

4. NUMERICAL RESULTS

In this section, we evaluate our approach on synthetic and real data. For comparison, we consider: *i*) *uniform attachment*: the incoming node attaches uniformly, i.e., $\mathbf{p}_{\text{rd}} = \frac{1}{N}\mathbf{1}$; *ii*) *preferential attachment*: the incoming node attaches with probability vector $\mathbf{p}_{\text{pf}} = \mathbf{d}/\mathbf{1}^\top \mathbf{d}$ where $\mathbf{d}$ is the degree vector; *iii*) *training attachment only*: a data-driven rule where we rely only on the attachment patterns available during training to build $\mathbf{p}_g = \frac{1}{|\mathcal{T}|} \sum_{t=1}^{|\mathcal{T}|} \mathbf{b}_{t+}$ and $\mathbf{w}_g = \frac{1}{|\mathcal{T}|} \sum_{t=1}^{|\mathcal{T}|} \mathbf{a}_{t+}$, i.e., we ignore the MSE costs. The first two serve as baselines to assess how the propsoed data-driven stochastic model compares with conventional statistical models, while the latter is considered to assess the importance of the task-specific cost.

4.1. Synthetic Data

For the synthetic experiments, we started with two undirected and unweighted random graphs following the Erdős-Rényi and Barabasi-Albert models each having $N = 100$ nodes. The respective graph signals were formed by randomly combining the first 30 eigenvectors of the corresponding graph Laplacian matrices. We then normalized the signal to have zero mean. The edge formation probabilities for these graphs and for the respective incoming nodes were set as $\mathbf{p}_{\text{rd}}$ and $\mathbf{p}_{\text{pf}}$ for the ER and BA graphs, respectively. We used a filter

Table 1. Averaged MSE and its standard deviation (Std) for synthetic data. (Upper) comparison with the uniformly random and preferential attachment; (Lower) joint training vs. individual training.

	Erdős-Rényi			Barabasi-Albert		
	Prop.	Pref.	Rand.	Prop.	Pref.	Rand.
MSE	**0.03**	0.06	0.06	**0.05**	0.1	0.08
Std.	**0.003**	0.003	0.003	**0.006**	0.006	0.006
	p,w	only **p**	only **w**	**p,w**	only **p**	only **w**
MSE	**0.03**	0.07	0.039	**0.05**	0.11	0.05
Std.	**0.003**	0.003	0.003	**0.006**	0.005	0.006

of order $L = 3$ with coefficients $h_l = \alpha^l$ and $\alpha = 0.3$ to diffuse the signal. The training set comprises 1000 data-points with a 800-200 train-test split and we selected μ_p, μ_w via ten-fold cross-validation from $[10^{-5}, 10^0]$. The learning rates λ_p, λ_w were fixed to 10^{-5}. We average the reconstruction MSE over 100 realizations per test node error and 100 train-test splits for a total of 10^4 runs.

First, we assess the convergence of Algorithm 1 under both the convex and non-convex settings. For the marginally convex condition also in $\mathbf{p}$ we set $\mu_p = 30$ to satisfy the convexity criterion in (11). Fig. 1 shows the training costs as a function of the number of iterations for 50 random initializations. We observe that the proposed approach converges always to local minima and it reaches a lower value for the non-convex case (blue) compared with the convex one (red) for all initialization. This is because a higher weight μ_p on the regularizer results in $\mathbf{p}$ adapting more to fit the training attachments $\mathbf{b}_{t+}$ rather than interpolating the signal, ultimately underfitting.

Next, we compare the interpolation MSE with the baselines for the ER ($\mu_p = 1, \mu_w = 1$) and BA ($\mu_p = 1, \mu_w = 0.1$). In the upper part of Table 1, we see the proposed approach outperforms alternatives in both settings. To further investigate the role of $\mathbf{p}$ and $\mathbf{w}$, we also train the proposed method for each of $\mathbf{p}$ and $\mathbf{w}$ while keeping the other fixed. The lower half of Table 1 suggests training only $\mathbf{w}$ provides a performance comparable to the joint training, which shows that the proposed approach still reaches the optimal value even without knowing the true attachment. However, training only $\mathbf{p}$ degrades the performance appreciably. This is because when $\mathbf{p}$ is known, we train on $\mathbf{w}$ with a convex cost and reach the global minima, as opposed to training only for $\mathbf{p}$ over a non-convex cost.

4.2. Cold start Collaborative Filtering

We now use the proposed method for rating prediction in cold start item-item collaborative filtering. We considered the Movielens 100K data-set [31] and removed all entities having less than 10 ratings leading to 943 users and 1152 items. For the cold start experiment, we start with a set of items with known ratings and use those to predict the ratings for unseen items for each user.

We built an item-item 35 nearest neighbour directed graph comprising 50 node items following [8]. The remaining items are divided into 700 for training and 402 for testing. To predict the ratings we used an order five graph filter, which coefficients are estimated to predict the ratings on the initial user-item set as in [8]. For our problem, we imposed a sparsity constraint on $\mathbf{p}$ and an ℓ_2−norm on $\mathbf{w}$. Algorithm 1 is run for 2000 iterations with learning rates $\lambda_p, \lambda_w = 10^{-4}$. We predict ratings on the test items for each user individually. We considered the Mean Absolute Error (MAE) as the evaluation criterion averaged over 100 samples drawn from $\mathbf{p}$.

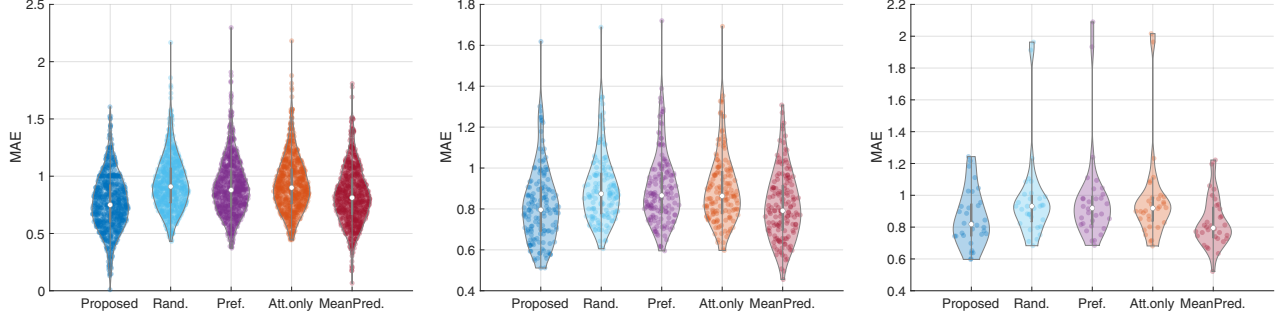In Fig. 2 we show the obtained results, where in addition to the

Fig. 2. Mean absolute error (MAE) violin plots for different methods and different rating densities. (Left) low ratings - proposed does best (0.75 ± 0.24), followed by mean (0.81 ± 0.24); (Centre) medium ratings - proposed and mean (0.79 ± 0.16) are tied; (Right) high ratings - mean does best (0.79 ± 0.15), followed by proposed (0.81 ± 0.17).

former baselines we also considered the user-based mean prediction. To better interpret the results, we divided users into three categories: $i)$ *low* containing users with less than 100 interactions; $ii)$ *medium* containing users with 100 to 200 interactions, and $iii)$ *high* containing users with more than 200 interactions. We observe that for users with low interactions the proposed approach achieves the lowest median error and personalizes the recommendations. For the medium and high interactions available, the proposed method performs better than the other attachment metrics. This shows the benefits of learning a graph and data-specific attachment pattern suited to a task. Second, for high rating users, the three starting baselines have a long tail in the MAE, whereas ours is more robust. The mean prediction method performs equally well in the high setting because in this case we have more ratings available and the cold start problem is less relevant. Instead, the proposed approach yield benefits in more data scarcity settings which pose the real challenge in recommender systems.

5. CONCLUSION

We proposed a data-driven attachment model for signal interpolation at the incoming nodes. The proposed model is characterized by probabilities of attachment and weights which we used training data to estimate for signal interpolation. We formulated a stochastic optimization problem w.r.t. the attachment parameters and used an alternating projected descent to solve it. We provided conditions when we can relax our interpolation requirements to make the convex marginally convex in both variables. The proposed approach outperforms related statistical models for interpolation over both synthetic and real experiments on cold start collaborative filtering. Future work will consider extending this approach to handle a sequence of nodes.

6. APPENDIX

Proof of Proposition 1: The interpolation MSE is $\mathbb{E}[([\mathbf{y}_+]_{N+1} - x_+^\star)^2]$ with the output at the incoming node. To express $[\mathbf{y}_+]_{N+1}$ in the model parameters, we consider the lth power of the adjacency matrix

$$\mathbf{A}_+^l = \begin{bmatrix} \mathbf{A}^l & \mathbf{0} \\ \mathbf{a}_+^\top \mathbf{A}^{l-1} & 0 \end{bmatrix} \quad (12)$$

and substitute it in the output $\mathbf{y}_+ = \sum_{l=1}^{L} \mathbf{A}_+^l \mathbf{x}_+$ [cf. (6)]. The output at the incoming node is thus $[\mathbf{y}_+]_{N+1} = \mathbf{a}_+^\top \sum_{l=1}^{L} \mathbf{A}_+^{l-1} \mathbf{x} = \mathbf{a}_+^\top \mathbf{A}_x \mathbf{h}$ where $\mathbf{A}_x = [\mathbf{x}, \dots, \mathbf{A}^{L-1}\mathbf{x}]$ and $\mathbf{h} = [h_1, \dots, h_L]^\top$.

The MSE is thus

$$\text{MSE}(\mathbf{p}, \mathbf{w}) = \mathbb{E}[(\mathbf{a}_+^\top \mathbf{A}_x \mathbf{h} - x_+^\star)^2]. \quad (13)$$

We add and subtract $(\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h}$ within the expectation and get

$$\text{MSE}(\mathbf{p}, \mathbf{w}) = \mathbb{E}[((\mathbf{a}_+ - \mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} + (\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_+^\star)^2] \quad (14)$$

where the RHS upon expanding becomes

$$\mathbb{E}[(\mathbf{a}_+^\top \mathbf{A}_x \mathbf{h} - (\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h})^2] + \mathbb{E}[(\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_+^\star)^2]$$
$$+ 2\mathbb{E}[(\mathbf{a}_+^\top \mathbf{A}_x \mathbf{h} - (\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h})((\mathbf{w} \circ \mathbf{p})^\top \mathbf{A}_x \mathbf{h} - x_+^\star)]. \quad (15)$$

In the first term, we expand the square, factor $\mathbf{A}_x \mathbf{h}$, and take the expectation inside to get $(\mathbf{A}_x \mathbf{h})^\top \mathbb{E}[(\mathbf{a}_+ - \mathbf{w} \circ \mathbf{p})(\mathbf{a}_+ - \mathbf{w} \circ \mathbf{p})^\top]\mathbf{A}_x \mathbf{h}$ which writes as $(\mathbf{A}_x \mathbf{h})^\top \boldsymbol{\Sigma}_+ \mathbf{A}_x \mathbf{h}$ [cf. (3)]. The second term is deterministic, thus we can drop the expectation. The third term instead is zero because $\mathbb{E}[\mathbf{a}_+] = \mathbf{w} \circ \mathbf{p}$. Combining these, we get (7). $\quad \square$

Proof of Corollary 1: To get marginal convexity, we check when the Hessian of the cost in (8) is positive semi-definite. The derivative of (8) w.r.t. $\mathbf{p}$ is shown in (9). The Hessian in $\mathbf{p}$ is

$$\nabla_p^2 C(\mathbf{p}, \mathbf{w}) = 2(\mathbf{w} \circ \mathbf{A}_x \mathbf{h})(\mathbf{w} \circ \mathbf{A}_x \mathbf{h})^\top - 2\text{diag}((\mathbf{w} \circ \mathbf{A}_x \mathbf{h})^{\circ 2}) + 2\mu_p \mathbf{I}_N. \quad (16)$$

The first term $(\mathbf{w} \circ \mathbf{A}_x \mathbf{h})(\mathbf{w} \circ \mathbf{A}_x \mathbf{h})^\top$ is a rank-one matrix with the only non-zero eigenvalue $2||\mathbf{w} \circ \mathbf{A}_x \mathbf{h}||^2$. The second matrix is a diagonal matrix with eigenvalues $\{-2(w_1[\mathbf{A}_x \mathbf{h}]_1)^2, \dots,$ $-2(w_N[\mathbf{A}_x \mathbf{h}]_N)^2\}$. The third matrix is also diagonal but with eigenvalues $2\mu_p$. The Hessian is the sum of a rank one matrix and two diagonal matrices. Its eigenvalues are the sum of the eigenvalues of the these matrices. By the semi-definite convexity condition [32], each of these eigenvalues now must be greater than or equal to zero. The condition in (11) is sufficient since all $w_i \leq w_h$ from the constraint set in (8). $\quad \square$

7. REFERENCES

[1] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.

[2] A. Ortega, P. Frossard, J. Kovačević, J. M. F. Moura, and P. Vandergheynst, "Graph Signal Processing: Overview, Challenges, and Applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, May 2018.

[3] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the Dots: Identifying Network Structure via Graph Signal Processing," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 16–43, May 2019.

[4] V. Kalofolias, A. Loukas, D. Thanou, and P. Frossard, "Learning time varying graphs," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2826–2830.

[5] P. Erdos, "On the evolution of random graphs," *Bulletin of the Institute of International Statistics*, vol. 38, pp. 343–347, 1961. [Online]. Available: https://ci.nii.ac.jp/naid/10025454140/

[6] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, Oct. 1999, publisher: American Association for the Advancement of Science.

[7] S. Liu, I. Ounis, C. Macdonald, and Z. Meng, "A heterogeneous graph neural model for cold-start recommendation," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 2029–2032.

[8] W. Huang, A. G. Marques, and A. R. Ribeiro, "Rating Prediction via Graph Signal Processing," *IEEE Transactions on Signal Processing*, vol. 66, no. 19, pp. 5066–5081, Oct. 2018.

[9] E. Isufi, M. Pocchiari, and A. Hanjalic, "Accuracy-diversity trade-off in recommender systems via graph convolutions," *Information Processing & Management*, vol. 58, no. 2, p. 102459, 2021.

[10] Y. Shen, G. Leus, and G. B. Giannakis, "Online Graph-Adaptive Learning With Scalability and Privacy," *IEEE Transactions on Signal Processing*, vol. 67, no. 9, pp. 2471–2483, May 2019.

[11] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Semi-Supervised Multiresolution Classification Using Adaptive Graph Filtering With Application to Indirect Bridge Structural Health Monitoring," *IEEE Transactions on Signal Processing*, vol. 62, no. 11, pp. 2879–2893, Jun. 2014.

[12] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs: Accounting for nonlinearities and dynamics," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 787–807, 2018.

[13] X. Dong, D. Thanou, M. Rabbat, and P. Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 44–63, 2019.

[14] A. Natali, M. Coutino, E. Isufi, and G. Leus, "Online time-varying topology identification via prediction-correction algorithms," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5400–5404.

[15] V. Kalofolias, "How to learn a graph from smooth signals," in *Artificial Intelligence and Statistics*, 2016, pp. 920–929.

[16] M. Coutino, E. Isufi, T. Maehara, and G. Leus, "State-space network topology identification from partial observations," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 211–225, 2020.

[17] E. Pavez and A. Ortega, "Generalized laplacian precision matrix estimation for graph signal processing," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6350–6354.

[18] G. Bianconi and A.-L. Barabási, "Competition and multiscaling in evolving networks," *EPL*, vol. 54, no. 4, p. 436, May 2001.

[19] V. N. Zadorozhnyi and E. B. Yudin, "Growing network: Models following nonlinear preferential attachment rule," *Physica A: Statistical Mechanics and its Applications*, vol. 428, pp. 111–132, Jun. 2015.

[20] V. Martínez, F. Berzal, and J.-C. Cubero, "A survey of link prediction in complex networks," *ACM computing surveys (CSUR)*, vol. 49, no. 4, pp. 1–33, 2016.

[21] A. Clauset, C. Moore, and M. E. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.

[22] G. Liu, J. Li, and L. Wong, "Assessing and predicting protein interactions using both local and global network topological metrics," in *Genome Informatics 2008: Genome Informatics Series Vol. 21*. World Scientific, 2008, pp. 138–149.

[23] W. Liu and L. Lü, "Link prediction based on local random walk," *EPL (Europhysics Letters)*, vol. 89, no. 5, p. 58007, 2010.

[24] W. Cukierski, B. Hamner, and B. Yang, "Graph-based features for supervised link prediction," in *The 2011 International Joint Conference on Neural Networks*, 2011, pp. 1237–1244.

[25] V. Matta, A. Santos, and A. H. Sayed, "Graph learning with partial observations: Role of degree concentration," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 1312–1316.

[26] M. Cirillo, V. Matta, and A. H. Sayed, "Learning bollobás-riordan graphs under partial observability," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5360–5364.

[27] A. Venkitaraman, S. Chatterjee, and B. Wahlberg, "Recursive Prediction of Graph Signals With Incoming Nodes," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 5565–5569.

[28] A. Sandryhaila and J. M. F. Moura, "Discrete Signal Processing on Graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[29] M. Coutino, E. Isufi, and G. Leus, "Advances in Distributed Graph Filtering," *IEEE Transactions on Signal Processing*, vol. 67, no. 9, pp. 2320–2333, May 2019.

[30] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[31] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm transactions on interactive intelligent systems (tiis)*, vol. 5, no. 4, pp. 1–19, 2015.

[32] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.