Exploring Fairness in Heterogenous Graph Embeddings

Rohit Gajawada
rohitgajawada@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Sanjana Garg
sgarg96@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Rohit Mujumdar
rohitmujumdar@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

KEYWORDS

Graph Embeddings, Bipartite Networks, Fairness, Feature Learning

1 INTRODUCTION

1.1 Aim

In our project, we explore the idea of unbiased random walks to improve the fairness in the node embeddings. We plan to experiment with different sampling strategies in the generation of the random walks used in training. We will be focusing on transductive embedding strategies like metapath2vec[2] but if time permits we would also like to modify the inductive training strategy of GraphSAGE[5] to incorporate fairness.

Our objectives are to explore if existing graph embedding approaches can have their training strategy modified so that they are fair compared to vanilla counterparts. We will assess our results by measuring performance on the downstream task of recommendation in terms of fairness metrics like statistical parity and also precision and recall.

1.2 Challenges

There has not been any existing work where the notion of unbiased random walks are used to generate embeddings for bipartite graphs as most of the existing work for graph embeddings and fairness have been done for homogeneous networks or through adversarial training. Also to the best of our knowledge, there has not been any work in unbiased inductive methods for bipartite graphs.

1.3 Impact

With the increased in usage of artificially intelligent systems around us, it becomes important that these systems do not reinforce any existing social biases. Specifically, when using graph embeddings for solving real world tasks based on networks, it is paramount that these embeddings don't carry any inherent social and cultural biases thereby adversely affecting minority or discriminated populations. If we can develop such node embeddings for networks that are robust to bias and also information rich, they can be stored and the same embeddings can be used for multiple downstream tasks.

2 LITERATURE SURVEY AND BASELINES

One of the first attempts to study potential bias issues inherent within graph embeddings in a transductive way was when Rahman et. al. proposed Fairwalk [7], a fairness-aware embedding method that extended node2vec [4]. While node2vec uses a random walk over a graph to generate walk traces and then extracts features based on the learned traces, Fairwalk first partitions neighbors into groups based on their sensitive attribute values and samples nodes for random walk from each group with the same probability, thereby enforcing equality of representation. However, Fairwalk is designed for homogeneous networks and would invariably lose out on the semantics of bipartite (heterogeneous) graphs. We aim to improve upon this by employing metapath2vec[2] which extends node2vec [4] by conditioning the random walk sampling based on the node types. This demonstrate improvements over using naive homogeneous approaches for heterogeneous networks. We aim to introduce fairness in the embeddings generated by metapath2vec[2].

GraphSAGE is an inductive approach for generating learnt graph embeddings and it can be done in an unsupervised, semi-supervised or supervised way. The benefit of this method for generating embeddings is that it can scale well to unseen networks and nodes as it does not require all of them to exist during training. Popular works like PinSage [8] have adopted this for large scale recommender systems for bipartite graphs. However, there is no study about inherent bias present in the embeddings.

Bose et al.[1] introduced another inductive approach for learning unbiased embeddings by using filtering via adversarial training with respect to the sensitive attributes. It does not use the notion of unbiased random walks for unsupervised feature learning and the embeddings generated can only be used for that particular downstream task.

3 PLAN OF ACTION

3.1 Proposed Methodology - Fair Embeddings

We extend the idea used in Fairwalk [7] to the metapath2vec [2] approach. Assume there are two types of nodes V_1 and V_2 in a bipartite graph G and V_1 has sensitive demographic attributes like gender, race, etc (usually a user) and V_2 has attributes like genre, etc. We can enforce an unbiased step from V_2 to V_1 such that there is an equal likelihood of sampling users from different demographics $v_1 \in N(v_2)$ where $v_1 \in V_1$ in the neighbourhood of the node $v_2 \in V_2$. For the V_2 to V_1 step, there might not be obvious grouping of attributes which can affect bias but we will explore sampling strategies in this step as well. Let's say a_i are the distinct attributes present in the neighbors of v_i and $f_j(v_i)$ be the jth subgroup of neighbors grouped by a_i.

For V_2 to V_1:

$$p\left(v_1^{i+1} \mid v_2^i\right) = \begin{cases} \frac{1}{|f_j(N^{i+1}(v_2^i))|} \frac{1}{|A_2|} & \left(v_1^{i+1}, v_2^i\right) \in E,\ v_1^{i+1} \in f_j(N^{i+1}(v_2^i)) \\ 0 & \text{otherwise} \end{cases}$$

Since the metapath2vec approach is a transductive approach and this might not work well on unseen data, if time permits we would like to explore inductive and unbiased graph embeddings approaches similar to GraphSAGE [5]. node2vec [4] and metapath2vec could be thought of as DFS based approaches and GraphSAGE could be thought of as a BFS based approach, so essentially they are both learnt random walks. We plan to introduce an unbiased AGGREGATE step in GraphSAGE and also possibly add fairness based weighted components to the loss function.

3.2 Downstream Task - Recommendation

We will be adopting a supervised approach for this, where our classifier will take as input two embedding vectors, one for user and one for item and predict the presence or absence of an edge between them. An item v is recommended to a user u, if the recommendation score for the pair $(u, v) \notin E$ is within the top $k\%$ of all the scores received by all candidate pairs.The resulting set of recommendations given to a user u is denoted by $\rho(u)$.

3.3 Metrics

In this section, we explain the metrics we'll be employing for evaluating fairness and recommendation performance.

3.3.1 Notations. We have a bipartite graph $\mathbf{G} = (V, E)$ where $V = V_1 \cup V_2$ and $V_1 \cap V_2 = \emptyset$, and the set of edges $E \subseteq \{(u, v) : u \in V_1, v \in V_2\}$. A sensitive attribute is denoted by $\mathcal{S}$. User-item pairs $(u, v) \in V_1 \times V_2$ are partitioned into groups $G_{ij}^{\mathcal{S}}$

3.3.2 Fairness Metrics. [7]

- **Statistical Parity**- This measure requires the acceptance (recommendation) rates from two groups to be equal. Let $P(G_{ij}^{\mathcal{S}})$ denote the acceptance rate for group $G_{ij}^{\mathcal{S}}$, then

$$P(G_{ij}^{\mathcal{S}}) = \frac{|\{(u, v) : v \in \rho(u) \wedge (u, v) \in G_{ij}^{\mathcal{S}}\}|}{|G_{ij}^{\mathcal{S}}|}$$

  The bias or statistical parity between two groups is then the difference between their acceptance rates. Extending this to multiple groups it can be calculated as the variance of the acceptance rate of each group.

$$bias(\mathcal{G}^{\mathcal{S}}) = var(\{P(G_{ij}^{\mathcal{S}})\} : G_{ij}^{\mathcal{S}} \in \mathcal{G}^{\mathcal{S}})$$

- **Equality of Representation** - At the network level this can be measured my measuring bias between different groups, among all recommendations in the network.

$$bias^{ER}(\mathcal{G}^{\mathcal{S}}) = var(\{N(G_{ij}^{\mathcal{S}})\} : G_{ij}^{\mathcal{S}} \in \mathcal{G}^{\mathcal{S}})$$

  $N(G_{ij}^{\mathcal{S}})$ is number of recommendations from group $G_{ij}^{\mathcal{S}}$

3.3.3 Recommendation system metrics. We plan to evaluate the performance of recommendation system in terms of precision and recall.

3.4 Risks

A potential risk we anticipate is a performance drop when changing the walk strategy. We intend to tackle this by tuning the trade-off between fairness performance or by experimenting with our grouping strategy. This could give some improvement on the F1-score since we would be effectively reducing the number of groups.

3.5 Dataset Description

We will primarily be using the MovieLens dataset [6], which is rating data set collected and made available from the MovieLens web site by GroupLens Research. Specifically, we'll be using MovieLens 100K Dataset version of the dataset. The dataset consists of 100,000 ratings on a scale of 1 (lowest) to 5 (highest) from 943 users on 1682 movies. Each user has rated at least 20 movies. The dataset provides demographic information like age, gender, occupation, zip and features like movie title, release date, video release date, IMDb URL and genre for each movie.

We construct an undirected bipartite graph of movies and users as nodes and the edges between representing a user who rated a movie. The constructed graph has the properties described in Table1:

Property	Value
Number of nodes	2625
Number of edges	100000
Average degree	76.1905
Radius	3
Diameter	5
Density	0.02903

Table 1: Description of the Dataset as a bipartite graph

3.6 Existing Code

For metapath2vec we intend to modify the PyTorch Geometric [3] implementation of the algorithm as the codebase : source code from the *nn.models.metapath2vec* module and this example. For GraphSage we intend to use a PyTorch implementation of the algorithm. If we plan to add adversarial constraints in our GraphSAGE based approach, we intend to follow the codebase Flexible Fairness Constraints based on the paper by Bose et. al. [1].

REFERENCES

[1] A. Bose and William L. Hamilton. 2019. Compositional Fairness Constraints for Graph Embeddings. In *ICML*.

[2] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In *KDD '17*. ACM, 135–144.

[3] Matthias Fey and Jan Eric Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. http://arxiv.org/abs/1903.02428 cite arxiv:1903.02428.

[4] Aditya Grover and J. Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016).

[5] William L. Hamilton, Zhitao Ying, and J. Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *NIPS*.

[6] F. M. Harper and J. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5 (2015), 19:1–19:19.

[7] Tahleen A. Rahman, Bartlomiej Surma, M. Backes, and Y. Zhang. 2019. Fairwalk: Towards Fair Graph Embedding. In *IJCAI*.

[8] Rex Ying, Ruining He, K. Chen, Pong Eksombatchai, William L. Hamilton, and J. Leskovec. 2018. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining* (2018).