Federal state autonomous
educational institution for higher education
«National research university
«Higher school of economics»
Faculty of Graduate School of Business
Master's Programme
«Business Analytics and Big Data Systems»

Project

on the topic

"LastFM Users Country Prediction"

Performed by 1 course students,
Shirnin Alexander, Markin Nikita, Sukharkov Alexander

Moscow 2023

Contents

Abstract

In this project, we aim to predict the country of Last.fm users based on the artists they like and their social connections. We plan to use machine learning and graph-based deep learning methods, including graph neural networks (GNNs), to learn meaningful representations of users and their relationships in the graph. By leveraging these representations, we aim to develop accurate prediction models for each user's country. We will compare the performance of these methods in terms of accuracy and efficiency to determine the most effective approach for predicting user countries in Last.fm. The most effective approach has the potential to improve user profiling and recommendation systems, which can enhance the user experience on the platform.
Key words - GNN, Gradient boosting, Embedding Space, Node2Vec.

1 Introduction

Music is an essential part of human life and culture, and it has become even more accessible in the digital age. Last.fm was one of the largest music streaming platforms in the world, with millions of users and vast amounts of data about their listening habits. Understanding the preferences and behaviors of Last.fm users can provide valuable insights into the global music scene and help improve personalized music recommendations.

In this research project, we aim to predict the country of Asian Last.fm users based on the users they follow and the musicians they like. We believe that music preferences can reveal important information about a person's cultural background, and that this information can be used to accurately predict a user's country of origin. By analyzing the listening habits of Asian users on Last.fm, we can gain a deeper understanding of the music tastes and cultural differences in the region.

We are going to use the LastFM Asia Social Network dataset [1]. It consists of 7,624 nodes and has 27,806 edges. Each node has a label, that represents a country. There are 18 countries in total.

We plan to use graph-based machine learning techniques to analyze the dataset and develop predictive models. Our approach will involve feature extraction from the graph structure, such as computing centrality measures and clustering coefficients, as well as incorporating embeddings of the artist preferences of each user. We will evaluate the performance of our models using the accuracy metric and compare various methods to analyze them.

The outcomes of our research can have significant implications not only for the music industry but also beyond it. By developing accurate predictive models, our work can aid music streaming platforms in enhancing their recommendation systems, providing users with more personalized and relevant content.

2 Literature Review

2.1 DeepWalk

DeepWalk [2] is an algorithm that learns latent representations of vertices in a network. These latent representations capture the social relations between vertices in a continuous vector space, which can be used by other models to make predictions.

DeepWalk uses truncated random walks to generate local information about each vertex's neighborhood. It treats these random walks as the equivalent of sentences in natural language processing, and applies the same methods used to learn latent representations for words in language modeling to the network data. Specifically, it uses a skip-gram model with negative sampling to learn representations that encode structural regularities of the network. One of the advantages of DeepWalk is that it can handle missing information in the network, which is common in many real-world scenarios. In fact, the paper showed that DeepWalk outperformed other baseline methods, especially in the presence of missing information. Additionally, DeepWalk is scalable and parallelizable, making it suitable for large-scale network classification tasks.

The paper demonstrated the efficacy of DeepWalk for multilabel classification tasks on several social network datasets, including BlogCatalog, Flickr, and YouTube. In these experiments, DeepWalk's learned representations provided F1 scores up to 10% higher than competing methods when labeled data was sparse. The competing methods included SpectralClustering, Modularity, EdgeCluster, wvRN, and the Majority (a naive method that simply chooses the most frequent labels in the training set). Additionally, in some cases, DeepWalk's learned representations outperformed all baseline methods while using 60% less training data. These results suggest that DeepWalk can effectively learn latent representations of vertices in a network for multilabel classification tasks, outperforming existing methods.

In conclusion, DeepWalk offers a powerful and flexible approach for learning

latent representations of vertices in a network, which can be used for a wide range of tasks including multilabel classification. For multilabel classification tasks, DeepWalk's learned representations can be used as features for traditional machine learning models such as logistic regression or support vector machines. Its ability to handle missing information, scalability, and parallelizability make it particularly well-suited for real-world applications.

2.2 Node2Vec

Authors of Node2Vec paper [3] propose a new algorithmic framework for learning continuous feature representations for nodes in networks. The authors highlight the need for careful feature engineering in prediction tasks over nodes and edges in networks, and how recent advances in representation learning have been able to automate this process by learning the features themselves. However, the present feature learning approaches are not expressive enough to capture the diversity of connectivity patterns observed in networks.

To address this limitation, node2vec algorithm is introduced, which learns a mapping of nodes to a low-dimensional space of features that maximizes the likelihood of preserving network neighborhoods of nodes. The node2vec algorithm builds upon previous methods that rely on fixed definitions of network neighborhoods by introducing a more flexible concept of a node's neighborhood. The algorithm achieves this by designing a biased random walk approach that can efficiently explore various types of neighborhoods. According to the authors, this added flexibility in neighborhood exploration is crucial in creating more complex and comprehensive node representations.

In the paper, the authors show that node2vec is more effective than existing techniques on multi-label classification and link prediction in real-world networks from various domains. They highlight the benefits of using a search-based optimization perspective, which provides an explanation for search strategies based on exploration-exploitation trade-off and allows for interpretability of the learned rep-

resentations. The authors also discuss the scalability and robustness of node2vec to perturbations, and they show how extensions of node embeddings to link prediction outperform popular heuristic scores designed specifically for this task. Finally, they compare node2vec with other methods such as DeepWalk and LINE, demonstrating that node2vec's flexible and controllable approach outperforms these rigid search strategies.

Overall, Node2Vec can be useful for node classification tasks as it can learn low-dimensional representations of nodes that capture the complex connectivity patterns in networks. Thereafter, other machine learning algorithms can be trained on these representations.

2.3 Graph Attention Networks

Graph Attention Networks (GATs) [4] are a neural network architecture designed to operate on graph-structured data. They leverage masked self-attention layers to overcome the limitations of prior methods based on graph convolutions or their approximations. By allowing nodes to attend over their neighborhood's features, GATs enable assigning different weights to different nodes in a neighborhood without requiring costly matrix operations or depending on knowing the graph structure upfront.

The architecture of GATs consists of multiple graph attention layers, where each layer attends to a neighborhood of nodes in the graph. Within each layer, a node's features are combined with the features of its neighbors, and then a self-attention mechanism is applied to compute the attention coefficients for each node in the neighborhood. These coefficients determine the importance of each neighbor's features for the node in question. Finally, a weighted sum of the neighbor features is computed and concatenated with the original node features to produce the output of the layer.

GATs have demonstrated state-of-the-art performance on various node classification benchmarks, both inductive and transductive. For example, they have

once achieved state-of-the-art results on the Cora, Citeseer, and Pubmed citation network datasets, as well as on a protein-protein interaction dataset. In these tasks, GATs ability to assign different weights to different nodes in a neighborhood without requiring costly matrix operations has proven especially useful.

In general, GATs are a powerful neural network architecture for graph-structured data that overcome the limitations of prior methods based on graph convolutions or their approximations. GATs use the attention mechanism to selectively aggregate the information from neighboring nodes. This allows them to focus on the most relevant nodes and edges, capturing the local and global patterns in the graph. By incorporating multiple attention mechanisms, GATs can also learn to attend to different aspects of the graph, such as the node's structural position, the node's attributes, or the relationship between the nodes.

3 Anticipated Methods

We would to use methods, that we have described in the literature review. Thus, we propose three approaches to solve thath problem:

- **Node2Vec** is a popular graph embedding method that learns low-dimensional representations of nodes in a graph by optimizing a skip-gram objective function. It generates node embeddings by sampling random walks. Node2Vec uses a biased random walk approach, where each node has a probability distribution over the types of walks it can take. This allows Node2Vec to balance between exploring and exploiting the graph structure, and to capture both local and global node properties. The embeddings can then be used as input features for a downstream machine learning task such as node classification. In this case, we will use the Node2Vec algorithm to learn node embeddings from the lastfm users' graph dataset and then train a logistic regression or XGBoost model on the learned embeddings to predict the country of the users.

- **DeepWalk** is another graph embedding method that is based on the idea of using random walks to learn node representations. However, it uses an unbiased random walk approach, where each walk is generated uniformly at random, without any preference towards any particular direction or type of node. Thereafter, simirarly embeddings can then be used as input features for the node classification. We will likewise use the DeepWalk algorithm to learn node embeddings from the lastfm users' graph dataset and then train a logistic regression or XGBoost model on the them to make predictions.

- For the **Graph Attention Network (GAT) + Node Embeddgins**, you can use it to directly predict the country of users in the lastfm dataset. GAT operates on the graph-structured data of the lastfm users and their features. The nodes of the graph are the unique users, and the edges between them represent following relationships. However, since in our case there are

different number of features in each node, we will use some algorithms for creating embeddings first. For node embeddings we might use different methods: Node2Vec, fully connected layers or GraphSAGE. GAT allows assigning different importances to different nodes within a neighborhood and does not depend on knowing the entire graph structure upfront. By stacking attentional layers, GAT can learn node representations that capture the underlying structure of the graph and make predictions based on the learned representations. In this case, GAT can be trained on the lastfm users' graph dataset with their artist preferences as features and their country as the target label.

4 Expected Results

We expect that Node2Vec will perform better than DeepWalk in our case, because it employs a more advanced random walk strategy. While DeepWalk generates random walks that are entirely random and unguided, Node2Vec generates random walks that are biased towards exploring both local and global network neighborhoods. The skip-gram model then learns node embeddings based on these biased walks, which can better capture the underlying structure of the network. This means that Node2Vec can better capture complex network structures and relationships between nodes, which can result in improved performance on downstream tasks such as node classification. However, the relative performance of Node2Vec and DeepWalk may depend on the specific characteristics of the network being studied, so it's always important to experiment with multiple methods to determine the best approach.

GAT is expected to get the best performance, since it is a more advanced neural network architecture that is specifically designed to operate on graph-structured data, making it better suited for tasks such as node classification. GAT uses an attention mechanism to selectively focus on important nodes in a graph, allowing it to effectively model relationships between nodes that are distant from each other in the graph. It is difficult to say, which node embedding method might work the best in this case. Fully connected layer is a simple method, however, it will be trained together with the GAT. Otherwise, Node2Vec and GraphSAGE won't be trained together with GAT. But due to their implementation, they might give a better node representation.

5 References

References

1. B. Rozemberczki and R. Sarkar, "Characteristic Functions on Graphs: Birds of a Feather, from Statistical Descriptors to Parametric Models," in *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*, p. 1325–1334, ACM, 2020.

2. B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, aug 2014.

3. A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," 2016.

4. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2018.