Modeling Oncology Drug Combinations using Graph Representation Learning

SURA Research Project

Grace Hu

I. Introduction

Social networks, drug-drug interactions etc. can often be visualized as graphical networks. In the latter case, the drugs themselves can be represented as nodes in a graph, and interactions between two drugs are represented as edges [1].

Link prediction, or the prediction of the possibility of a relationship existing between two nodes, is an important part of analyzing these graphical networks. One way to do so is by using similarity-based algorithms, where the more similar two nodes are the more likely an edge exists between them. The degree of similarity of two nodes is measured via **Local Similarity Indices** such as Common Neighbours (CN), Jaccard Index (JC), Adamic-Adar Index (AA) [2].

Another algorithm that can be used for link prediction is **node2vec**, where graphical nodes are represented by embeddings [3]. These embeddings are learned using a random walk procedure, which maintains neighbourhood structures while mapping the nodes to mathematical vectors. node2vec is said to outperform other algorithms such as DeepWalk [4] in link prediction tasks [5].

The goal of this research project is to evaluate the accuracy of the node2vec algorithm against Local Similarity Indices in a link prediction task based on a drug-drug interaction dataset [1]. Area Under the Receiver Operating Characteristic Curve (AUC) is used as an evaluation metric.

II. Results and Discussion

The AUC scores were calculated, with DeepWalk, Common Neighbours and Adamic-Adar performing the best at scores of 0.838, 0.851 and 0.854 respectively on the same validation set. Node2Vec notably underperformed on both the training set and validation set, though according to literature it should perform better in link prediction tasks [5]. Meanwhile, DeepWalk also did not score as well as the Local Similarity Indices on average.

Figure 1: Graph of Drug-Drug Interaction Network

	Drug-Drug Interaction Graph
Number of nodes	1514
Number of edges	48514
Average node degree	32
Graph density	0.04235769085
Clustering coefficient	0.3039679581007187

Table I: Drug-Drug Interaction Network Properties

This discrepancy is possibly because an unbiased random walk procedure was used for the model to learn the embeddings in the first place. Meanwhile, 500 random walks, a walk length of 5 and a window size of 3 were also used, and it is unknown whether this allows for proper mapping of node neighbourhoods within the graph.

Formula for edge density:

$$D = \frac{2|E|}{|V|(|V|-1)}$$

Algorithm	AUC Score (validation set)
node2vec	0.5082075225
DeepWalk	0.83849796125
CN	0.85187161499999
JC	0.825520147499999
AA	0.8543261299999998

Table II: Comparison of AUC Scores

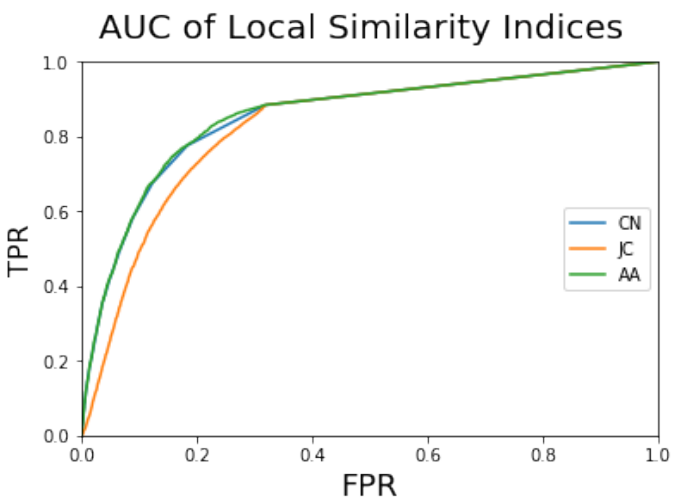

Figure 2: AUC Curves of the three LSI

Similarly, the drug-drug interaction network is a rather sparse graph, with 1514 nodes and 48514 edges. The average degree per node is 32, the edge density is 0.04 (compared to a maximum of 1), and the clustering coefficient is 0.3.

All these values point to a sparse graph, which may be why node2vec (which relies heavily on local neighbourhoods in networks) may not be the most efficient. Also, it is highly possible that there is a bug in the training of the node2vec embedding model in the first place.

III. Methodology

An algorithm for link prediction was devised using a **logistic regression model** with one linear layer, with **Binary Cross Entropy** (BCE) as the loss function. **Stochastic Gradient Descent** (SGD) was used as the optimizer algorithm. The graph was divided into the training set and the validation set, edges chosen as part of the validation set removed from and edges in the validation set were removed from the training graph.

The calculation of the AUC score requires both a dataset of both positive (existing) edges and also negative (non-existing) edges from a graph. Thus sets of negative edges were appended to both the training set and the validation set by randomly sample the list of non-edges (nx.non_edges) in both the training graph and validation graph respectively.

Three Local Similarity Indices were compared against node2vec performance: **Common Neighbours (CN), Jaccard Index (JC), and Adamic-Adar Index (AA)**. The formulas to calculate the three indices for two select nodes are as follows [2]:

(1) Common Neighbours (CN)

$$s_{xy}^{CN} = |\Gamma(x) \cap \Gamma(y)|$$

where $\Gamma(x)$ is the set of neighbours of node x

(2) Jaccard Index (JC)

$$s_{xy}^{Jaccard} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

(3) Adamic-Adar Index (AA)

$$s_{xy}^{AA} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k_z}$$

where k_z is the degree of node z

For the node2vec algorithm, an **unbiased random walk** procedure was used to generate network neighborhood [5]. Pairs of neighboring nodes generated from the random walks were then used to train the encoder function, which generated vector embeddings (of dimension 64x1) for each node.

These embeddings were then used as part of a logistic regression model mentioned above, and used to generate two metrics: the **dot product** (DP), and the **element-wise multiplication product** (EM). The AUC scores of both metrics were then calculated using the built-in Scikit-learn function.

Meanwhile, embeddings were also generated using a DeepWalk algorithm [4], and DP and EM were calculated using those embeddings as well.

The AUC scores of node2vec were then compared against the AUC scores of the three Local Similarity Indices and DeepWalk.

References

[1] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda *et al.*, "Drugbank 5.0: a major update to the drugbank database for 2018," *Nucleic acids research*, vol. 46, no. D1, pp. D1074–D1082, 2017.

[2] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150–1170, Mar 2011.

[3] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," *CoRR*, vol. abs/1709.05584, 2017. [Online]. Available: http://arxiv.org/abs/1709.05584

[4] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. New York, NY, USA: ACM, 2014, pp. 701–710. [Online]. Available: http://doi.acm.org/10.1145/2623330.2623732

[5] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," *CoRR*, vol. abs/1607.00653, 2016. [Online]. Available: http://arxiv.org/abs/1607.00653