

FINAL PROJECT

Artificial Intelligence

(CIS 5603)

**Graph Identifying relationship between genes and human ontology**

Rushabh Patel

Department of Computer & Information Science

Email: rushabh.patel5@temple.edu

28 April 2022

**Abstract**

**Background**

Deep phenotyping is the study of genotype-phenotype relationships and the history of human illness via a rigorous and accurate investigation of phenotypic anomalies. Understanding and detecting this phenotype-genotype relationship is a crucial step in bringing precision medicine into clinical practice. Machine learning techniques have made it possible to anticipate these relationships between anomalous human traits and genes.

**Methods**

The framework was created to anticipate relationships between human phenotype ontology (HPO) and genes in this study. Annotation data from diverse knowledge resources, such as Orphanet, is utilized to parse human phenotype-gene relationships. The node2vec [1] technique was used to produce the embeddings for the nodes (HPO & genes). It samples nodes from this graph using random walks [2] and then learns features across these sampled nodes to build embeddings. These embeddings were utilized to predict the presence of a connection between these nodes using five different supervised machine learning techniques.

**Results:**

The downstream link prediction task shows that Gradient Boosting Decision Tree based model (LightGBM) achieved an optimal AUROC 0.904 & AUCPR 0.784. In addition, LightGBM achieved an optimal weighted F1 score of 0.87. Compared to other 4 methods LightGBM is able to find more accurate interaction/link between human phenotype & gene pairs.

1. **Introduction**

Many people now suffer from illnesses caused by irregularities in the genome, and many diseases go misdiagnosed as a result of this nonuniformity. Human phenotyp analysis is important in medical research and clinical practice [3]. The HPO can play a significant role in deep phenotyping by converting precision medicine into clinical practice. Deep phenotyping is an in-depth, precise investigation of phenotypic anomalies in which the phenotype component for an individual is seen and reported [5]. The Human Phenotype Ontology (HPO) is a widely used resource for defining and organizing human phenotypes in a systematic and logical manner [4]. The HPO is mostly derived from medical literature and various information resources, such as DECIPHER [6], OMIM [7], and Orphanet [8], as well as a database of chromosomal imbalance and human phenotypes. The bulk of studies in this field primarily use topological and ancestorial links between any two nodes in a directed acyclic network to discover relevant traits. These studies do not use feature learning to find relationships through embeddings of various nodes, which is difficult to deduce from network structure. The interactions and connections between HPO-gene terms were predicted using standard classification methods. Because target values are predicted without addressing inherited connections within the ontology, these techniques created discrepancies. If we want to predict a relationship between human phenotype and gene, for example, a typical classifier will correlate the HPO phrase "Squamous Cell Carcinoma" with a gene, but not "Abnormality of the Skin," resulting in an incorrect prediction. We employ node embeddings to manage the hierarchical interactions between HPO concepts that truly represent HPO. The node embeddings provide a method for mapping graph nodes to distributed representations and enable the link between graph and embedding space to be translated. Node2Vec is one of the most widely used ways for creating node embeddings. It uses a biased random walk to produce a flexible neighborhood sampling technique and then passes the sample data to the word2vec model as an input. We are developing a framework to predict human phenotype-gene interactions in this study. The dataset for this study was compiled from a variety of information sites, including Orphanet. To learn the features, we turned the knowledge resource data to an undirected network and used that graph to construct node embeddings. These node characteristics were utilized to train machine learning models to anticipate interactions in the future. We used many indicators to do quantitative analysis on the output of these models.

1. **Methodology:**
   1. **Data Preparation:**

Based on the currently observable incomplete network, link prediction seeks to detect missing links or future link interactions between nodes. Predicting node-to-node linkages has been a major study area in the field of graphs and networks. To apply machine learning to predict the interaction between the graph's disconnected node-pairs, we must first describe the graph as a structured dataset with a collection of characteristics. Figure 1 depicts a network with seven nodes and disconnected node pairs: AF, BD, BE, BG, and EG. Assume we examined the graph data at time t and discovered a few additional connections that had developed in the graph at time t+n (red links in figure 2). So, to extract a set of features from the graph at time ***t***, we extract unconnected node pairs ***B-E, B-D , B-G, E-G, A-F, , ,*** and then we look at the graph, at time ***t+n*** and label these three new links (red lines in Fig 2) in the graph for the node pairs***, B-D***, ***B-E, & A-F*** as 1 and the node-pairs ***BG*** & ***EG*** will be labeled as 0, because still at time ***t+n*** no links were found for these nodes.

![A picture containing chart

Description automatically generated]() ![A screenshot of a computer

Description automatically generated with medium confidence]()

**Figure 1**  **Figure 2**

*Graph at time t Graph at time t+n*

We had access to the graph at time t+n in this case, which is why we were able to obtain the labels for the target variable. We would, however, have access to a single enormous graph in real-world networks or graphs. We must first understand that connections between nodes in a network are formed gradually over time, and that by keeping this in mind, we may solve our problem by randomly concealing some of the edges in the supplied graph and then producing labels. However, while eliminating connections or edges, we must avoid removing edges that might result in an isolated node, i.e. a node with no edges or an isolated network.

Negative population or disconnected node-pairs make up a large portion of the undirected graph derived from the Orphanet HPO annotation dataset. We develop an adjacency matrix, as illustrated in Figure 3, to discover these node pairings. This adjacency matrix is a square matrix with nodes of the graph defining both rows and columns. Edges or linkages are represented by the value in the matrix. If there is one, there is an edge, and if there is zero, there is no edge between the node-pairs. In figure 3, the adjacency matrix for nodes HP:0000951 and ORPHA:763 has a value of 0 at the cross junction, and there is no edge or connection between them in the graph.

![A picture containing table

Description automatically generated]()

**Figure 3**

*Finding unconnected node pairs using Adjacency Matrix*

We'll iterate above or below the diagonal in this square matrix because both supply comparable information and traversing the entire matrix would be superfluous. The traversal will assist us in locating the negative sample spots. Similarly, we drop a few edges at random to acquire positive samples, however this may result in weakly linked fragments and nodes. To deal with this, we first look for graph splitting while deleting a node-pair or verifying the overall node counts. We can safely drop the node-pair if these requirements are met. Following these methods, we discovered a total of 1665146 disconnected node pairings, of which only 125304 positive samples, or around 7.5 percent, were detected (highly imbalanced dataset).

* 1. **Feature Extraction:**

The node2vec technique is used to extract features from the graph G. To represent node characteristics, it constructs vector space embeddings. Node2Vec starts with weighted random walks from each and every node in the graph and interprets them as terms that can be embedded into Euclidean Space using the skip-gram model [9]. The goal of this method is to maximize the probability of node n in context K within a contextual window of length l:

The ith term or node sequence created by random walk is denoted by ki. p(kj | ki) is the output of the skip-gram technique, which is specified by the SoftMax function.

where ***nj***, ***ni***are vector representations of terms***kj*** , ***ki***in the hidden layer of the skip-gram model. The node2vec package was used to train a model with 30 nodes per walk, 200 walks per node, and 128 embedding dimensions. This node2vec model will be applied to every node pair in the dataset.

* 1. **Downstream Prediction using ML algorithms:**

Machine learning models are fed with node properties retrieved via the node2vec technique. We divided the entire dataset into training (80%) and testing (the remaining 20%) to evaluate the algorithm's performance. For this supervised link prediction challenge, we employed the approaches listed below.

* + 1. **Logistic Regression:**

In the discipline of statistics, logistic regression is a frequently used approach for categorizing binary outcomes. It employs the sigmoid activation function to keep the final result between 0 and 1. The coefficients are calculated using the maximum-likelihood learning technique and our training set. It's a popular algorithm that makes assumptions about how data is distributed. We utilized the L-BFGS solver for this research.[10]

* + 1. **Neural Network:**

Recent advancements have changed the area of machine learning. This program seeks to imitate the human brain when it comes to detecting patterns in data. Neural networks may be used to perform a range of tasks, such as grouping related data, categorizing various objects, and so on. Random weights and thresholds are used to start the neural network. The training data is fed as vectors into the input layer, and it is allowed to pass through the subsequent hidden layers by tweaking the weights and thresholds until the outputs resemble true labels. To get output values between 0 and 1, we used a Neural Network with two hidden layers, each with relu activation, and sigmoid activation at the output layer. The Adam optimizer, which is an extension of SGD (Stochastic Gradient Descent) that lets us change the network weight depending on training data, was utilized at a learning rate of 1e-3. [12].

1. **Results** 
   1. **Evaluation Metrics**

Because of the extreme class imbalance, the model assessment criteria we chose are critical. Understanding how each class performs rather than relying on aggregate metrics is critical in this study. The metrics we utilized to validate the performance of various machine learning approaches are listed below.

* + 1. **AUROC (Area Under the Receiver Operating Characteristics):**

AUROC is a fairly typical summary statistic used to measure the quality of a predictor in a binary classification problem. The probability curve represents the Receiver Operating Characteristics, and the Area Under Curve indicates the degree of separability in AUROC. It gives information on the model's class distinguishability. The True Positive Rate (Sensitivity) is displayed against the False Positive Rate in this ROC curve (1 - Specificity).

* + 1. **AUCPR** (Area Under the Precision-Recall Curve):

For verifying extremely skewed data, this is the most extensively used statistic in the machine learning field. By displaying Precision versus Recall, this is used to evaluate the performance of binary classification models. In comparison to AUROC, this statistic allows us to see the performance of positive samples more carefully.

* 1. **Comparing Performances of different models.**

To evaluate the predictive performance of these five models, Logistic Regression & Neural Network, on human phenotype-gene dataset we use all the many different metrics including AUROC & AUCPR. To appropriately evaluate the imbalance nature of the dataset, we calculate important metrics for each class instance in our case, we just have two classes 0 & 1.

Chart, line chart

Description automatically generatedGraphical user interface, chart

Description automatically generated

Chart, line chart

Description automatically generatedGraphical user interface, chart

Description automatically generated

**Figure 6**

*AUCROC & AUCPR*

Calendar

Description automatically generatedCalendar

Description automatically generated

**Figure 7**

*Micro, Macro & Weighted Metrics*

1. **Discussion**

In this paper, we introduced a method for predicting correlations between human phenotypes and genes utilizing diverse information resources, such as Orphanet [1]. The main element of this study is to describe data as a graph and then discover a way to express this graph as an acceptable feature set, which will allow us to use it for downstream tasks such as prediction. In essence, we presented a method for obtaining the embedding vectors through an algorithm known as node2vec and then used these embeddings to construct five distinct machine learning models. We examined and compared the results using several quantitative indicators such as AUROC and AUCPR. Some of these metrics were computed for each class instance in order to better grasp the situation for an imbalanced class, in our case positive samples. Based on these metrics, we found very interesting results. If we want to just focus on positive samples meaning the measure of the link that we correctly identify having associations of all the actual association in the graph (we refer to it as Precision), then we may either use Deep Neural Network algorithm. On the other hand, if we just want to focus on accurately identifying positives from True Positives, i.e., actual links in the graph (Recall), then use Logistic Regression.

**References:**

1. Liao, L., He, X., Zhang, H., & Chua, T. S. (2018). Attributed Social Network Embedding. *IEEE Transactions on Knowledge and Data Engineering*, *30*(12), 2257–2270. https://doi.org/10.1109/tkde.2018.2819980
2. Peng C, Dieck S, Schmid A, et al. CADA: Phenotype-driven gene prioritization based on a case-enriched knowledge graph. medRxiv; 2021. DOI: 10.1101/2021.03.01.21251705.
3. Owen, P. “Clinical practice and medical research: bridging the divide between the two cultures.” *The British journal of general practice: the journal of the Royal College of General Practitioners* vol. 45,399 (1995): 557-60.
4. Köhler S, Gargano M, Matentzoglu N, et al. The Human Phenotype Ontology in 2021. *Nucleic Acids Res*. 2021;49(D1): D1207-D1217. doi:10.1093/nar/gkaa1043
5. Robinson PN. Deep phenotyping for precision medicine. *Hum Mutat*. 2012;33(5):777-780. doi:10.1002/humu.22080
6. Firth HV, Richards SM, Bevan AP, Clayton S, Corpas M, Rajan D, et al. DECIPHER: database of chromosomal imbalance and phenotype in humans using ensemble resources. The American Journal of Human Genetics. 2009; 84:524–33. [PubMed: 19344873]
7. Hamosh A, Scott AF, Amberger JS, Bocchini CA, McKusick VA. Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. Nucleic acids research. 2005;33: D514–D7. [PubMed: 15608251]
8. Aymé S, Schmidt J. Networking for rare diseases: a necessity for Europe. Bundesgesundheitsblatt-Gesundheitsforschung-Gesundheitsschutz. 2007; 50:1477–83. [PubMed: 18026888]
9. Doğan T HPO2GO: prediction of human phenotype ontology term associations for proteins using cross ontology annotation co-occurrences. *PeerJ*. 2018;6: e5298.
10. Mathur S, Dinakarpandian D. Finding disease similarity based on implicit semantic similarity. *Journal of biomedical informatics*. 2012; 45:363–71.
11. Hoehndorf R, Dumontier M, Gkoutos GV. Identifying aberrant pathways through integrated analysis of knowledge in pharmacogenomics. *Bioinformatics*. 2012; 28:2169–75.
12. Gao, J., Liu, L., Yao, S. *et al.* HPOAnnotator: improving large-scale prediction of HPO annotations by low-rank approximation with HPO semantic similarities and multiple PPI networks. *BMC Med Genomics* **12,**187 (2019). https://doi.org/10.1186/s12920-019-0625-1
13. Aditya Grover and Jure Leskovec. 2016. Node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. DOI: https://doi.org/10.1145/2939672.2939754
14. Wright, R. E. (1995). *Logistic regression.* In L. G. Grimm & P. R. Yarnold (Eds.), *Reading and understanding multivariate statistics* (p. 217–244). American Psychological Association.
15. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
16. McCulloch, W.S. & Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), pp.115–133.
17. Chen, T. & Guestrin, C., 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD’16. New York, NY, USA: ACM, pp. 785–794.