Proceedings of the ASME 2024 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2024
August 25-28, 2024, Washington, DC, USA

IDETC/CIE 2024-143786

GRAPH NEURAL NETWORK-BASED LINK PREDICTION FOR HIGHLY IMBALANCED NETWORK DATA

Yinshuang Xiao
Walker Dept. of Mechanical Engineering
The University of Texas at Austin
Austin, Texas 78712-1591
Email: yinshuangxiao@utexas.edu

Zhenghui Sha*
Walker Dept. of Mechanical Engineering
The University of Texas at Austin
Austin, Texas 78712-1591
Email: zsha@austin.utexas.edu

ABSTRACT

The rapid advancement of graph neural networks (GNN) has revolutionized the integration of complex network and neural network models, fostering diverse engineering applications such as the classification of assembly parts and the prediction of customer preferences. GNN-based link prediction (LP), as a representative study, has shown utility in resource allocation and system performance evaluation, as evidenced in the existing literature. However, the inherent challenge of imbalanced network data (i.e., sparse network) from engineering domains poses obstacles to accurate link predictions due to the lack of training data of positive links, leading to biases and reduced performance. In contrast to existing studies that focus on enhancing model architecture, this paper concentrates on exploring preprocessing methods, e.g., the data undersampling strategies, and two post-processing methods for accuracy calculation based on 1) probability threshold-based labeling and 2) probability rank-based labeling, and studies their impacts on the prediction results with a given GNN-based LP model architecture. Real-world network data sourced from shared mobility systems and product market systems, which exhibit various levels of imbalance, are used as test cases. This study contributes to a better understanding of effective data undersampling strategies and post-processing methods for handling imbalanced network data when applying GNN-based LP to real-world complex systems.

Keywords: Graph neural network, link prediction, imbalanced binary classification, networked engineering systems.

1 INTRODUCTION

In the realm of engineering applications, complex networks play a pivotal role in comprehending intricate interactions within systems, guiding the design process. The representative works include but are not limited to system robustness analysis [1–3], system dynamic analysis [4–6], and network-based system predictive models [7–9]. These applications span various domains such as market systems [10–12], infrastructure systems [13, 14], and energy systems [15, 16], etc.

Moreover, the rapid evolution of advanced graph neural network (GNN) methods has significantly enhanced the computational efficiency and prediction accuracy of network data [17,18]. This results in a more powerful tool that enables researchers in system engineering and design domains to model, analyze, and optimize systems with increasingly intricate interplays [19, 20]. GNNs aggregate network topologies and attributes to generate new node/link/graph representations, enabling various downstream tasks such as node/link/graph classification, regression, node clustering, and link prediction (LP) [8, 18, 21]. Compared to traditional artificial neural networks (ANN) treating individual nodes as independent entities, GNNs often demonstrate superior performance (e.g., more accurate classification) by considering additional network structural information that considers interdependencies among nodes [8]. This superior performance has spurred the exploration of GNN applications in engineering domains [15, 22]. For example, in [19], Nobari *et al.* applied GNN in design automation applications to perform surface classification for 3D geometric models. In another work [20], Ferrero *et*

*Corresponding author.

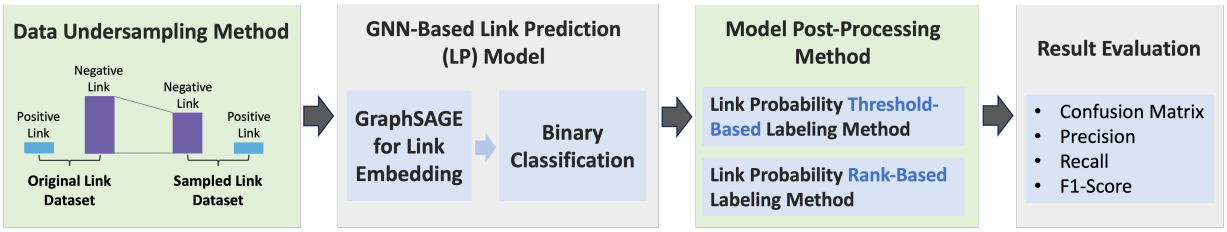


FIGURE 1: Experiment framework.

al. applied GNNs to classify the function of parts in an assembly. They found that GNNs outperformed other deep learning architectures and achieved competitive performance across different tiers of functions.

In addition to the above application, another important application in engineering research is GNN-based link prediction (LP). The significance of LP within the system engineering domain cannot be overstated, as it plays a crucial role in foreseeing connections and relationships within complex systems. In market systems, for instance, a link can symbolize the competitive relationship between products. An accurate link prediction model holds considerable benefits for companies, aiding in the identification of potential competitors. Similarly, regulators can leverage such models to monitor the dominant trend in market competition [23]. The importance of LP is further underscored by the existing literature. In our previous study [8], a GNN-based LP model was developed to predict station-to-station trips/demand in shared mobility systems, demonstrating its utility in supporting station capacity design and the decision on the location of new stations. In another study [7], Ahmed *et al.* applied the GNN-based LP model to predict co-consideration relationships between vehicles in the Chinese vehicle market system, validating its efficiency in identifying key vehicle attributes that promote the formation of co-consideration links. In summary, an accurate prediction of the links between entities holds substantial implications for improving efficiency, resource allocation, and overall system performance, establishing itself as a key component in the toolkit of engineering practices.

The core concept of GNN-based LP involves feeding the link embeddings generated by GNNs into a neural network for binary classification. Here, class 0 (negative link) signifies non-existing links (i.e., no link is observed from a pair of nodes), while class 1 (positive link) denotes existing links (i.e., a link exists between two ndoes) [24, 25]. Despite the advances in this approach, the sparse nature of positive links in comparison to the vast number of negative links poses a primary challenge in LP within real-world network data. This challenge manifests itself in the dominance of negative links, making it difficult for LP models to learn from and accurately predict limited positive instances, consequently leading to potential biases and diminished predictive performance [26]. To address this challenge, existing studies are motivated to develop various neural network models and/or model archetectures [26–28]. One representative model is the pairwise learning for neural link prediction (PLNLP) model proposed by Wang *et al.*. This model treats link prediction as a pairwise learning problem and advocates the use of ranking loss as an objective function to maximize the standard ranking metric, the area under the curve (AUC) [28]. However, in the LP pipeline, as shown in Figure 2, in addition to the models, both the training data (pre-model) and the prediction accuracy evaluation methods (after-model) are equally important. Therefore, there remains a knowledge gap regarding the impact of data sampling strategies and model post-processing methods (e.g., using probability threshold or probability rank to decide the positive links) on prediction results for a given model.

In this paper, we endeavor to bridge this research gap by conducting a comprehensive experiment aimed at investigating the influence of data undersampling with varied sampling ratios and model post-processing methods on prediction results given a GNN-based LP model. Furthermore, we utilize two highly imbalanced real-world network data sets sourced from different engineering systems to underscore the impact of the degree of imbalance on prediction performance. The first data are about shared mobility systems. It is from Chicago's Divvy Bike with a proportion of positive links around 7.4% [29]. This places the dataset within the moderate imbalance range ($1-20\%$) [30]. The second data are about product market systems, obtained from a recent survey conducted in the US vehicle market. This dataset exhibits a proportion of positive links around 1.1%, near the extreme imbalance range ($< 1\%$) [30]. Our study provides valuable insight into how the performance of GNN-based link prediction models can be optimized by adopting data sampling and post-processing methods without changing the model architecture. It contributes to a broader and better understanding of effective strategies for handling highly imbalanced datasets in the realm of real-world complex systems.

The remainder of the paper is organized as follows. In Section 2, we present the proposed experiment framework. Then, the details of the two datasets of the adopted case studies are introduced in Section 3. In the same section, we also illustrate how the experiment is set up. In Section 4, we present the experimental results, and in the last section, Section 5, we conclude the

paper with closing thoughts and future work.

2 EXPERIMENT FRAMEWORK

Figure 1 provides an overview of the experimental framework, encapsulating the data undersampling method, GNN-based LP models, model post-processing methods, and result evaluation strategies. The following sub-sections delve into each stage, unraveling the details from data sampling to result evaluation.

2.1 DATA UNDERSAMPLING METHOD

As aforementioned, a significant challenge for LP lies in the presence of too few positive links, leading to a severely imbalanced binary classification problem. One prevalent method to tackle this issue involves undersampling the majority class to achieve balance in training data [31]. Illustrated in Figure 2, the initial step of the undersampling process for LP involves obtaining the link set E, encompassing all possible links given the node set V within a training network G. For instance, if set V comprises $N_V = 10$ nodes, the total number of possible undirected links in V is $N_V(N_V-1)/2 = 45$ (for directed links, the total number is $N_V(N_V-1)$). Subsequently, all links are labeled according to their existence in the network, G. For example, in the instance of network G depicted in Figure 2, the link between node 1 and node 2 is observed, thus labeled "1", while the link between node 1 and node 4 does not exist and hence is labeled "0."

After labeling, E is divided into two subsets: E_p, which preserves all positive links, and E_n, which stores all negative links. Finally, the negative links are randomly sampled with respect to the given ratio $\gamma \in [1, N_{E_n}/N_{E_p}]$, where $\gamma = 1$ indicates that the sampled link set includes an equal number of positive and negative links. Conversely, $\gamma = N_{E_n}/N_{E_p}$ signifies no sampling process is carried out and all negative links are selected for training. Therefore, the γ value indicates the degree of imbalance within the training dataset. In this work, we are especially interested in how different data undersampling ratios (i.e., the γ values) influence prediction results with a given GNN-based LP model.

2.2 GNN-BASED LINK PREDICTION (LP) MODEL

In this study, we adhere to our previously proposed GNN-based LP model, which has exhibited superior performance compared to a basic artificial neural network, as detailed in [8]. Illustrated in Figure 3, the model architecture comprises two principal components: the first involves utilizing the GraphSAGE algorithm [24] for link embedding, while the second relates to binary classification for link prediction. In the first component, we employ the GraphSAGE algorithm [24] to generate a new vector representation of size M for each node, referred to as *node embedding*. This representation is derived by aggregating each node's individual features alongside its two-hop network

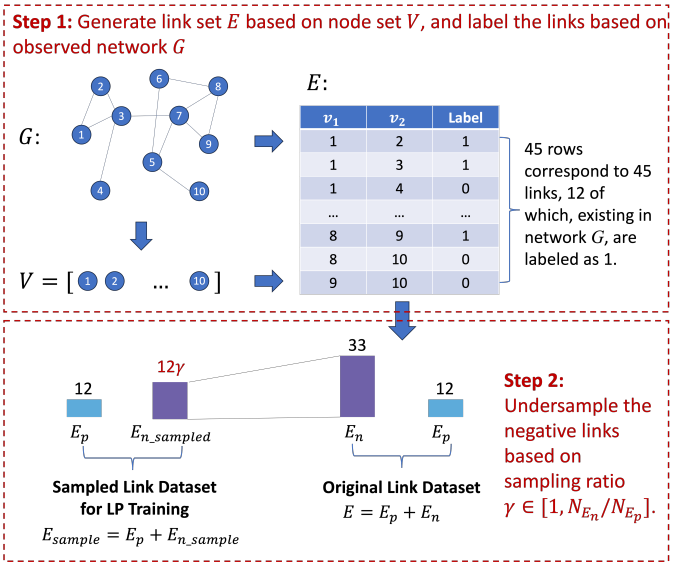


FIGURE 2: Undersampling process.

neighborhood information. It is important to note that the model shown in Figure 3 was created for directed networks. For undirected networks, the distinction between in- and out-neighbors is not applicable, and the network neighbors are treated uniformly. Additionally, training GraphSAGE necessitates providing network information for each node to aggregate neighborhood data, which can be approximated using the K-nearest neighbor method [7] or a regular artificial neural network [8].

Once individual node embeddings are obtained, they are concatenated to form a *link embedding*. In contrast to the directed network's link embedding shown in Figure 3, the undirected version is insensitive to the order of the start- and end-nodes. Moving on to the second component, the resultant link embedding of size $2M$ is fed into a fully connected neural network with one hidden layer. Both the input and hidden layers share the same size as the link embedding, and the output layer comprises a single neuron that utilizes the Sigmoid activation function. This output layer produces a probability of the input link's existence. The entire training process follows an end-to-end supervised learning approach, with the aim of minimizing binary cross-entropy loss using stochastic gradient descent (SGD) [25].

2.3 MODEL POST-PROCESSING METHODS

In this study, we introduce two model post-processing methods that convert the probability of link existence into binary labels. The first method is the threshold-based labeling method, and the second is the rank-based labeling method.

Link probability threshold-based labeling As depicted in Figure 4, the probability threshold-based labeling

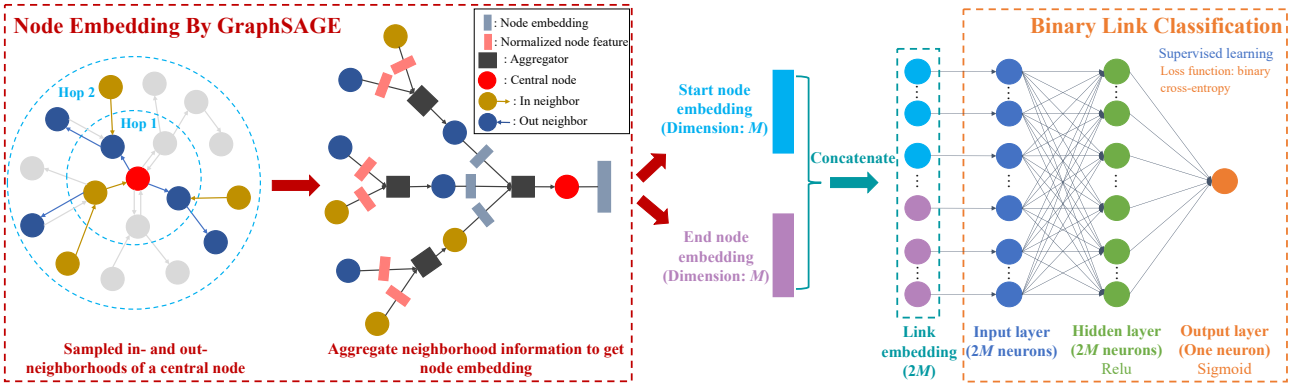


FIGURE 3: GNN-based LP model [8].

method begins by establishing an optimal threshold, denoted as $P_{\text{threshold}}$. Then, the predicted probabilities undergo a transformation into labels using the rule: links with probabilities higher than $P_{\text{threshold}}$ are labeled "1" while those below the threshold are labeled "0."

Once all labels are created, a confusion matrix [25] of this binary classification can be obtained from which we are able to compute the main accuracy metrics, such as true positive rate (recall) and precision. In this study, our focus is primarily on the model's proficiency in predicting minority (positive) links. Consequently, we adopt the optimal point on the precision-recall (PR) curve, specifically the point with the highest F1-Score. The PR curve represents the plot of precision versus recall at various thresholds of link probability. The F1-Score, expressed by Equation (1), is the harmonic mean of precision and recall, providing a balanced evaluation metric [8]. The probability threshold-based method emphasizes the optimal trade-off between precision and recall by setting a specific threshold value.

$$F1\text{-}Score = \frac{2 * Precision * Recall}{Precision + Recall}. \tag{1}$$

Link probability rank-based labeling In the rank-based labeling method, we commence by ranking the predicted probabilities from high to low, as illustrated in Figure 4. Here, we introduce the hit ratio at the top-K ranked links ($HR@K$), a metric commonly employed in recommendation systems to calculate the recall rate [32]. The formula for $HR@K$ is given by:

$$HR@K = \frac{N_p^K}{N_{E_p}}, \tag{2}$$

where N_p^K represents the number of true positive links included in the top-K ranked links, and N_{E_p} is the total number of true positive links in the test dataset. For this study, since we aim to keep the predicted network with the same density (N_{E_p}/N_E) as the ground-truth network, we set $K = N_{E_p}$[1]. To illustrate, in Figure 4, we set $K = 4$, which means the observation of 4 true positive links in E_{test}. The $HR@4$ in the lower-right plot of Figure 4 is 75%, indicating that the links with the top-4 probabilities can correctly predict 75% of true positive links. In simpler terms, if we label these top-K links as "1" and the rest as "0," we can achieve a recall of 75%. In contrast to the probability threshold-based method, which seeks a balance between precision and recall, this proposed rank-based method places greater emphasis on analogizing the size of the observed network.

2.4 RESULT EVALUATION

In this study, we employ widely recognized metrics for binary classification evaluation. We initiate the evaluation process by comparing the predicted labels with the true labels to obtain the confusion matrix. Subsequently, we calculate key metrics, including the true negative rate (TNR), false positive rate (FPR), true positive rate (TPR), false negative rate (FNR), and precision (which is equivalent to $TP/(TP+FP)$). The F1-Score is then computed using Equation (1), with values ranging from 0 to 1. A higher F1-Score indicates a superior performance of an LP model [31].

3 DATA SOURCE AND EXPERIMENT PREPARATION

In this section, we introduce two data sources utilized in our experiment, originating from real-world complex systems. The first dataset pertains to the shared mobility system, and the second dataset relates to the vehicle market system. We outline the detailed experiment settings corresponding to these two test cases below.

[1]It is worth noting that the rank-based labeling method is a general method. While density serves as an illustrative example, other metrics of interest, such as hit rate (recall), can also be seamlessly used to determine the top-K.

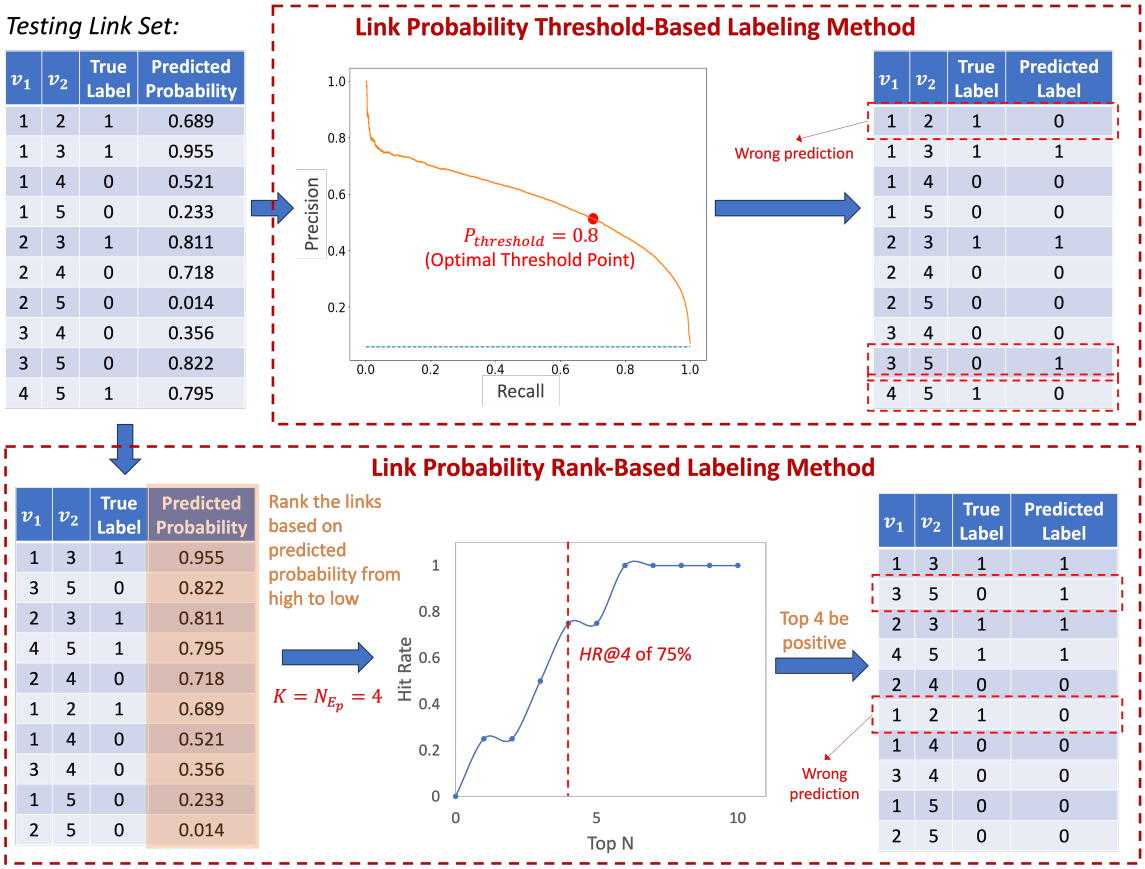


FIGURE 4: Illustration of model post-processing methods (Optimal Threshold Point: this point corresponds to the largest F1-Score which produces the best balance between precision and recall).

3.1 DATA SOURCE

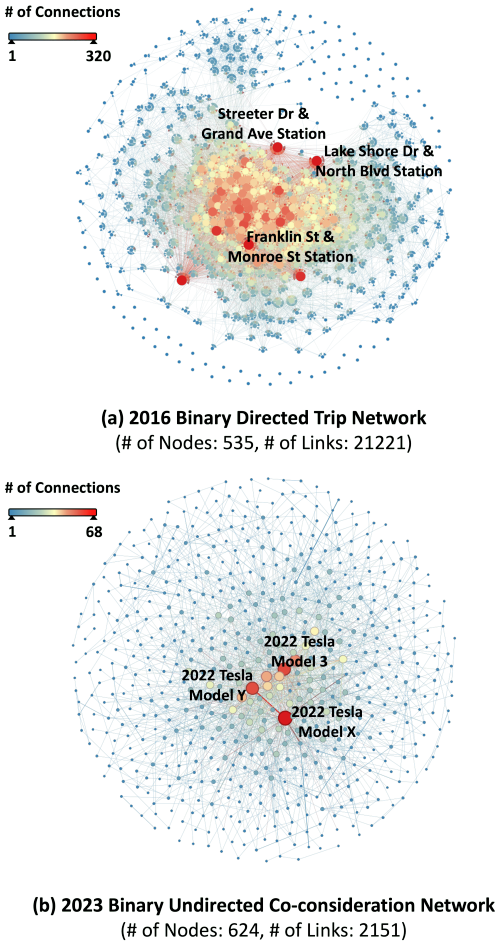
Divvy Bike shared mobility system data This data is comprised of two subsets. The first subset is the Divvy Bike data [29], specifically the data package for May 2016, encompassing both station and trip information. The station data encompasses unique identifiers, names, geographic coordinates, the number of docks, and online dates for each station. Trip data records include start- and end-station IDs, trip times and durations, along with users' demographic information. The second subset incorporates Point of Interest (POI) data, gathered using Overpass Turbo [33]. This dataset encompasses seven types of POIs (financial, education, recreational & tourism, residential, sustenance, healthcare, and transportation), totaling 2,269 POIs distributed around all the stations in the first subset [34]. Using the same data processing approach as described in our previous research [34], we derive a binary directed trip network illustrated in Figure 5 (a). This network comprises 535 station nodes, each characterized by ten features, including two geographic coordinate features, one capacity feature (number of docks), and seven POI features. Out of the 285,690 potential directed links, 7.4% (21,221 positive links) are observed.

US vehicle market survey data This dataset originates from our recent survey study on US new car buyers, which comprises two parts. The first part encompasses car attribute data, scraped from Cars.com, capturing 624 distinct car models with a total of 22 features (e.g., brand, fuel economy, base curb weight, etc.). The second part involves data from responses from new car buyers, where participants provided information on up to six cars they considered, including model year and name. In total, we collected consideration sets from 2,283 respondents. We constructed a binary undirected vehicle co-consideration network following the definition of the co-consideration network [23]. So, each node represents a unique vehicle model, and a link is established between two vehicles if they are co-considered by at least one buyer. The visualization of the resulting co-consideration network is presented in Figure 5 (b), featuring 624 nodes and 2,151 links, constituting 1.1% of all 194,376 possible undirected links.

TABLE 1: Experiment Scheme

Shared Mobility System		Vehicle Market System	
Undersampling Ratio γ	**Post-processing Method**	**Undersampling Ratio γ**	**Post-processing Method**
1, 3, 5, 8[1], No sampling[2]	Link probability *threshold-based* labeling method[3]	1, 3, 5, 50[1], No sampling	Link probability *threshold-based* labeling method
1, 3, 5, 8, No sampling	Link probability *rank-based* labeling method ($K = 7.4\% * N_E^{val}$)[4]	1, 3, 5, 50, No sampling	Link probability *rank-based* labeling method ($K = 1.1\% * N_E^{val}$)

[1]: These two values are chosen in each test case because they are nearest integer to the median of the range $[1, N_{E_n}/N_{E_p}]$.
[2]: No sampling implies that we include all positive and negative links from the obseved network (ground truth) as the training dataset.
[3]: The optimal probability threshold for each test is determined in the post PR curve analysis, as detailed in the results section.
[4]: N_E^{val} represents the total number of all potential links in the validation set.



(a) 2016 Binary Directed Trip Network
(# of Nodes: 535, # of Links: 21221)

(b) 2023 Binary Undirected Co-consideration Network
(# of Nodes: 624, # of Links: 2151)

FIGURE 5: Visualisations of the training networks.

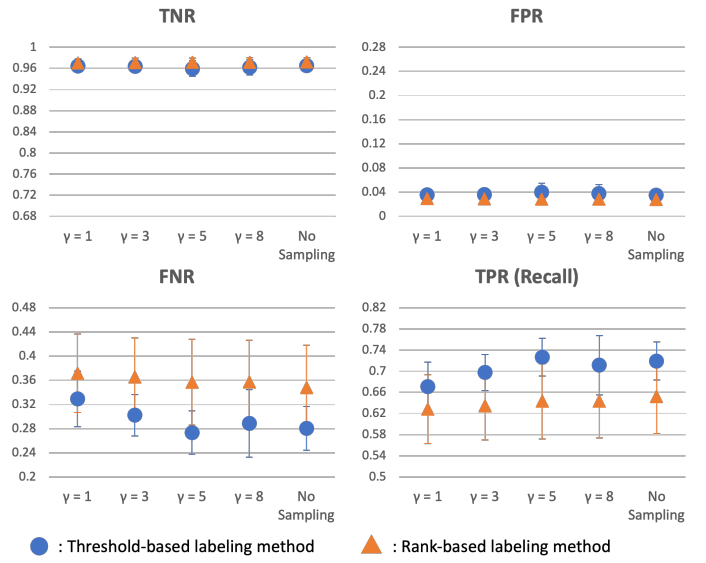
3.2 DESIGN OF EXPERIMENT

The detailed experimental scheme is presented in Table 1, encompassing a total of 20 tests. To enhance consistency and minimize variability, we employ the cross-fold validation approach [35] for each test. Taking the shared mobility system as an illustration, we divide the 535 nodes into five folds, each comprising 107 nodes. Subsequently, we iteratively designate one of these folds for validation and employ the remaining four folds for training. During this process, for both training and validation link sets, only the links that are observed in the original network in Figure 5 (a) are labeled as "1" and the rest as "0." This procedure yields five original link sets (before applying the undersampling strategies) for training and five validation link sets. It is important to note that those original link sets for training are rebalanced according to a specified undersampling ratio before being fed into the model for each test. Table 3 and Table 4 provide detailed statistics for each original set for training and validation link set for both cases.

Next, as outlined in Section 2.2, the training of the GraphSAGE LP model necessitates a reference network for the aggregation of network neighborhood information. In this study, to mitigate the potential consequences of inaccurate embedding of neighborhood information during experiments, we opt to directly input the original network, as depicted in Figure 5, into the LP model for both systems. Finally, we summarize the model settings and hyperparameter values in Table 2.

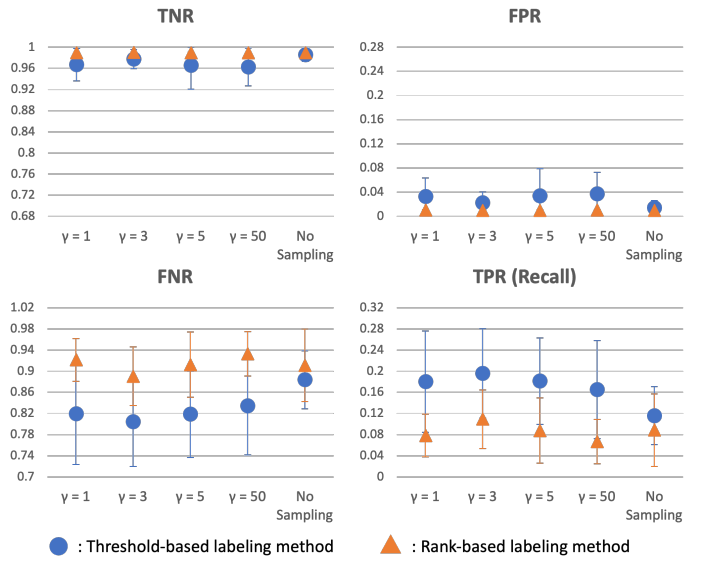
4 EXPERIMENT RESULTS

Confusion matrix In this section, we examine the results of the confusion matrices of all tests conducted on both the shared mobility system and the vehicle market system, which are summarized in Figure 6. As mentioned above, a five-fold cross-validation approach is employed for each test, and thus the figures illustrate the mean and standard deviation of the results over five rounds. Additionally, for the threshold-based labeling method, the probability thresholds vary across different tests due to the optimization process involved, as explained in Section 2.3. The purpose is to compare the best performance of each test at



Optimal probability threshold for threshold-based labeling method				
$\gamma = 1$	$\gamma = 3$	$\gamma = 5$	$\gamma = 8$	No Sampling
0.846	0.701	0.514	0.429	0.345

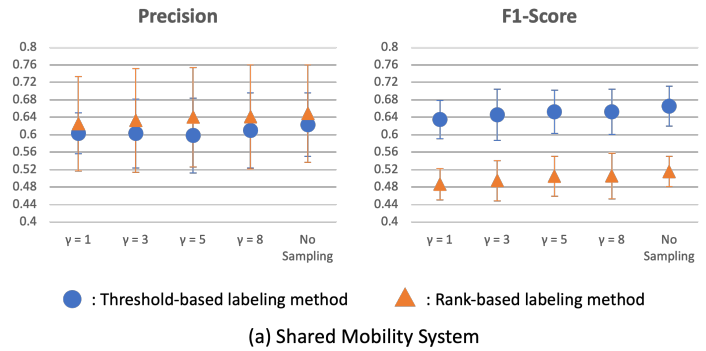
(a) Shared Mobility System



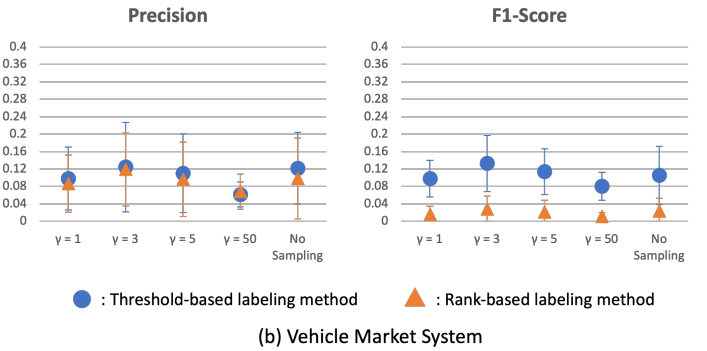
Optimal probability threshold for threshold-based labeling method				
$\gamma = 1$	$\gamma = 3$	$\gamma = 5$	$\gamma = 50$	No Sampling
0.768	0.617	0.447	0.046	0.046

(b) Vehicle Market System

FIGURE 6: Confusion matrix results statistics (mean $\pm$ standard deviation) for all tests conducted on both systems.



(a) Shared Mobility System



(b) Vehicle Market System

FIGURE 7: Precision and F1-Score statistics (mean $\pm$ standard deviation) of all tests for both systems.

the optimal threshold point with the largest F1-Score. The results reveal several insights in the following.

1) Regardless of a system exhibiting moderate or extreme data imbalance, the threshold-based labeling method consistently achieves higher recall, indicating better identification of true positive links compared to the rank-based method. However, this improvement comes at the cost of predicting more false positives, as evidenced by a higher FPR and a lower TNR.
2) In the case of the shared mobility system with a moderate imbalance issue, a t-test is conducted to assess the observed increasing trend of recall in both labeling methods. Our null hypothesis posits that there is no increase in recall. The resulting p-values for the threshold-based and rank-based methods are far less than 0.001. Hence, the hypothesis is rejected, Consequently, including more negative links (that is, as the γ value increases) is beneficial for increasing recall in both labeling methods, and these increases are statistically significant. Conversely, in the system with a more serious imbalance issue, the inclusion of more negative links may lead to a decrease in recall, especially for the threshold-based labeling method.

TABLE 2: Experiment parameter settings

System	Setting Items	Value
Shared Mobility System[1]	Neighborhood search depth	2
	# of Sampled in- and out-neighbors in two hops	10
	Node embedding size	30
	Input and hidden layer size for GraphSAGE	60
	Minibatch size	192
	Learning rate	4e-4
	Dropout	0
	Epoch	200
Vehicle Market System[2]	Neighborhood search depth	2
	# of Sampled neighbors in two hops	5
	Node embedding size	10
	Input and hidden layer size for GraphSAGE	10
	Minibatch size	32
	Learning rate	5e-4
	Dropout	0.3
	Epoch	500

[1]: We adopted these settings from our prior work [34], with the only alteration being the reduction of epochs from 500 to 200 to improve computational efficiency. As the primary objective of this paper is not to identify the optimal model performance, this epoch setting proves adequate for achieving convergence because a typical decrease is not observed beyond 200 epoches of training.

[2]: These settings are determined by trial and error. We guarantee the model's convergence. Similarly, we prioritize computational efficiency, since our focus does not extend to achieving the best model.

3) The exacerbated imbalance issue contributes to more dispersed performance, as reflected in the larger standard deviation observed in the plots for the vehicle market system. One potential reason could be attributed to the scarcity of positive samples relative to the overwhelming number of negative samples, resulting in greater variability in the model's predictions.

Precision and F1-Score Furthermore, upon examining the Precision and F1-Score results, we provide additional evi-

TABLE 3: Cross-fold training and validation data statistics for shared mobility system

Original Link Set for Training	# of Nodes	# of Positive Links (N_{E_p})	# of Negative Links (N_{E_n})
1	428	12,647	170,109
2	428	13,345	169,411
3	428	14,528	168,228
4	428	14,642	168,114
5	428	12,761	169,995
Validation Set	**# of Nodes**	**# of Positive Links (N_{E_p})**	**# of Negative Links (N_{E_n})**
1	107	1,077	10,265
2	107	887	10,455
3	107	663	10,679
4	107	598	10,744
5	107	1,035	10,307

TABLE 4: Cross-fold training and validation data statistics for vehicle market system

Original Link Set for Training	# of Nodes	# of Positive Links (N_{E_p})	# of Negative Links (N_{E_n})
1	500	1,253	123,497
2	500	1,482	123,268
3	500	1,369	123,381
4	500	1,411	123,339
5	496	1,379	121,381
Validation Set	**# of Nodes**	**# of Positive Links (N_{E_p})**	**# of Negative Links (N_{E_n})**
1	124	117	7,509
2	124	74	7,552
3	124	91	7,535
4	124	76	7,550
5	128	83	8,045

dence that, in systems with moderate imbalance issues, increasing the undersampling ratio enhances the overall model performance, leading to a higher F1-Score. A t-test is conducted, and resulting p-value is less than 0.001, indicating the statistical significance of this increase. This positive effect is observed for both labeling methods. The potential explanation lies in the increased information provided to the LP model when augmenting the number of negative links. In systems with more extreme imbalance data, one interesting observation is that an optimal undersampling ratio ($\gamma = 3$) appears to exist, particularly enhancing the predictive performance of the threshold-based method, resulting in the highest F1-Score. However, when no sampling method is applied, the vehicle market system exhibits the highest TNR and precision, along with the lowest recall. This low recall could be attributed to the escalating dominance of negative samples and the scarcity of positive samples, causing the LP model to exhibit bias toward predicting more samples as negative. Therefore, when dealing with large and sparse networks, implementing the proposed undersampling ratio strategy is recommended to determine the optimal γ, achieving the best trade-off between computational efficiency and model performance.

Lastly, the proposed rank-based labeling method exhibits inferior performance compared to the threshold-based method for both systems. This discrepancy is attributed to the distinct objectives of the two methods. The threshold-based method aims to achieve the best F1-Score, while the rank-based method strives to analogize the network size of the system. Consequently, the rank-based method tends to be conservative in predicting positive links, leading to higher precisions than the threshold-based method in the shared mobility system. This is visually represented in Figure 8, where it is apparent that the threshold-based method predicts more positive links for both systems, resulting in a higher density. Particularly for the vehicle market system, the mean network density for the threshold-based method is on average three times higher than that of the rank-based method.

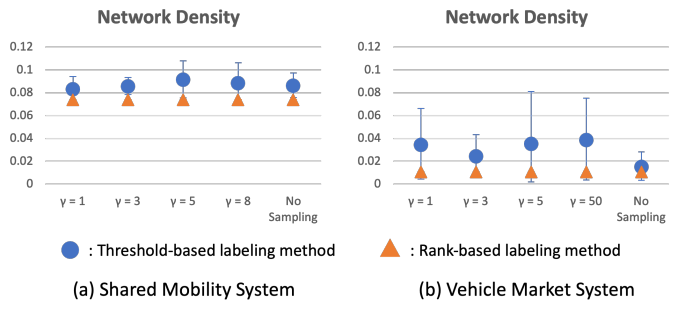


FIGURE 8: Predicted network density statistics (mean $\pm$ standard deviation) of all tests for both systems. The density here represents the ratio between the number of predicted positive links and the total number of possible links.

In summary, for a system characterized by moderately imbalanced network data, increasing the undersampling ratio proves effective in enhancing predictive performance, resulting in a larger F1-Score. Conversely, this strategy is not applicable to systems with extremely imbalanced network data. Moreover, in a broader context, the threshold-based labeling method consistently outperforms the proposed rank-based method. The rank-based approach aims to predict a network of comparable size to the original observed network but sacrifices a portion of prediction accuracy in the process.

5 CONCLUSION

In this study, we developed an experimental framework, exploring various combinations of data undersampling strategies with varying sampling ratios, along with two model post-processing methods, i.e., the probability threshold-based and rank-based labeling methods. We aim to scrutinize the impact of these combinations on prediction results given a GNN-based LP model. Furthermore, we utilized two real-world network datasets derived from shared mobility systems and product market systems. This choice aimed to illustrate the disparities in the influence of these combinations on engineering applications, particularly in the context of varied degrees of data imbalance. The results show that increasing the undersampling ratio improves predictive performance, particularly in systems with moderate data imbalance, as evident in a higher F1-Score. However, this approach is less effective in systems with extreme data imbalance. In cases of extreme imbalance resulting in a sparser network, it is advisable to undergo a tuning process to identify the optimal sampling ratio, thus achieving the best balance between computational efficiency and model performance. Overall, the threshold-based labeling method consistently outperforms the rank-based method, which sacrifices prediction accuracy for a network of comparable size to the original observed network.

While the pursuit of a high-performing link prediction (LP) model falls outside the scope of this study, one notable limitation in this paper is the absence of a definitive LP solution for systems characterized by extreme data imbalance, as evidenced by consistently low F1-Scores (below 0.2) across all experiments conducted using the vehicle market system example. This suggests that addressing LP for highly imbalanced network data remains a challenge. In future investigations, we aim to expand the scope of this research by testing a diverse range of engineering applications characterized by varying degrees of imbalance, ranging from extreme to mild imbalances. This extension seeks to improve our understanding of the challenges posed by data imbalance in LP within real-world systems. Additionally, this endeavor will contribute to the establishment of standardized benchmarks and evaluation metrics applicable to engineering practices. Furthermore, our future work endeavors to incorporate a broader selection of GNN-based LP models into the ex-

perimental framework. The goal is to explore and identify the optimal combination of data sampling strategies, post-processing methods, and LP models to comprehensively address the aforementioned limitation: the imbalance challenges inherent in networked data.

ACKNOWLEDGMENT

The authors greatly acknowledge the funding support from NSF CMMI #2203080.

REFERENCES

[1] Paparistodimou, G., Duffy, A., Whitfield, R. I., Knight, P., and Robb, M., 2020. "A network tool to analyse and improve robustness of system architectures". *Design Science,* ***6***, p. e8.

[2] Xiao, Y., and Sha, Z., 2022. "Robust design of complex socio-technical systems against seasonal effects: a network motif-based approach". *Design Science,* ***8***, p. e2.

[3] Panyam, V., Huang, H., Davis, K., and Layton, A., 2019. "Bio-inspired design for robust power grid networks". *Applied Energy,* ***251***, p. 113349.

[4] Fan, X., Dudkina, E., Gambuzza, L. V., Frasca, M., and Crisostomi, E., 2022. "A network-based structure-preserving dynamical model for the study of cascading failures in power grids". *Electric Power Systems Research,* ***209***, p. 107987.

[5] Gavino, P. A., Xiao, Y., Cui, Y., Chen, W., and Sha, Z., 2023. "Evolutionary co-mention network analysis via social media mining". In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Vol. 87301, American Society of Mechanical Engineers, p. V03AT03A045.

[6] Xie, J., Bi, Y., Sha, Z., Wang, M., Fu, Y., Contractor, N., Gong, L., and Chen, W., 2020. "Data-driven dynamic network modeling for analyzing the evolution of product competitions". *Journal of Mechanical Design,* ***142***(3), p. 031112.

[7] Ahmed, F., Cui, Y., Fu, Y., and Chen, W., 2022. "Product competition prediction in engineering design using graph neural networks". *ASME Open Journal of Engineering,* ***1***, p. 011020.

[8] Xiao, Y., Ahmed, F., and Sha, Z., 2023. "Graph neural network-based design decision support for shared mobility systems". *Journal of Mechanical Design,* ***145***(9).

[9] Xue, X., Sun, W., Wang, J., Li, Q., Luo, G., and Yu, K., 2020. "Rvfl-lqp: Rvfl-based link quality prediction of wireless sensor networks in smart grid". *IEEE Access,* ***8***, pp. 7829–7841.

[10] Sha, Z., Cui, Y., Xiao, Y., Stathopoulos, A., Contractor, N., Fu, Y., and Chen, W., 2023. "A network-based discrete choice model for decision-based design". *Design Science,* ***9***, p. e7.

[11] Wang, M., and Chen, W., 2015. "A data-driven network analysis approach to predicting customer choice sets for choice modeling in engineering design". *Journal of Mechanical Design,* ***137***(7), p. 071410.

[12] Wang, M., Sha, Z., Huang, Y., Contractor, N., Fu, Y., and Chen, W., 2018. "Predicting product co-consideration and market competitions for technology-driven product design: a network-based approach". *Design Science,* ***4***, p. e9.

[13] Gomez, C., Sanchez-Silva, M., Dueñas-Osorio, L., and Rosowsky, D., 2013. "Hierarchical infrastructure network representation methods for risk-based decision-making". *Structure and infrastructure engineering,* ***9***(3), pp. 260–274.

[14] Thongmak, P., Xiao, Y., Gavino, P., Zhang, M., and Sha, Z., 2024. "Geospatial network analysis of us megaregions in 40 years".

[15] Wu, J., and Wang, P., 2023. "Generative design for resilience of interdependent network systems". *Journal of Mechanical Design,* ***145***(3), p. 031705.

[16] Pagani, G. A., and Aiello, M., 2013. "The power grid as a complex network: a survey". *Physica A: Statistical Mechanics and its Applications,* ***392***(11), pp. 2688–2700.

[17] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M., 2020. "Graph neural networks: A review of methods and applications". *AI open,* ***1***, pp. 57–81.

[18] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y., 2020. "A comprehensive survey on graph neural networks". *IEEE transactions on neural networks and learning systems,* ***32***(1), pp. 4–24.

[19] Heyrani Nobari, A., Rey, J., Kodali, S., Jones, M., and Ahmed, F., 2024. "Meshpointnet: 3d surface classification using graph neural networks and conformal predictions on mesh-based representations". *Journal of Mechanical Design*, pp. 1–18.

[20] Ferrero, V., DuPont, B., Hassani, K., and Grandi, D., 2022. "Classifying component function in product assemblies with graph neural networks". *Journal of Mechanical Design,* ***144***(2), p. 021406.

[21] Sanchez-Lengeling, B., Reif, E., Pearce, A., and Wiltschko, A. B., 2021. "A gentle introduction to graph neural networks". *Distill*. https://distill.pub/2021/gnn-intro.

[22] Chen, Y.-h., Kara, L. B., and Cagan, J., 2024. "Bignet: A deep learning architecture for brand recognition with geometry-based explainability". *Journal of Mechanical Design,* ***146***(5), p. 051701.

[23] Xiao, Y., and Cui, Y., 2023. "Product competition analysis for engineering design: A network mining approach". In 2023 Conference on Systems Engineering Research (CSER).

[24] Hamilton, W., Ying, Z., and Leskovec, J., 2017. "Inductive representation learning on large graphs". *Advances in neural information processing systems,* ***30***.

[25] Nielsen, M. A., 2015. *Neural networks and deep learning*, Vol. 25. Determination press San Francisco, CA, USA.

[26] Li, B., Chaudhuri, S., and Tewari, A., 2016. "Handling class imbalance in link prediction using learning to rank techniques". In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 30.

[27] Jia, C., Ma, J., Liu, Q., Zhang, Y., and Han, H., 2020. "Linkboost: a link prediction algorithm to solve the problem of network vulnerability in cases involving incomplete information". *Complexity,* ***2020***, pp. 1–14.

[28] Wang, Z., Zhou, Y., Hong, L., Zou, Y., Su, H., and Chen, S., 2021. "Pairwise learning for neural link prediction". *arXiv preprint arXiv:2112.02936*.

[29] Divvy_Bike, 2020. Divvy system data. Last accessed 8 February 2022.

[30] Kumar, V., Lalotra, G. S., Sasikala, P., Rajput, D. S., Kaluri, R., Lakshmanna, K., Shorfuzzaman, M., Alsufyani, A., and Uddin, M., 2022. "Addressing binary classification over class imbalanced clinical datasets using computationally intelligent techniques". In Healthcare, Vol. 10, MDPI, p. 1293.

[31] Brownlee, J., 2020. *Imbalanced classification with Python: better metrics, balance skewed classes, cost-sensitive learning*. Machine Learning Mastery.

[32] Wu, H., Song, C., Ge, Y., and Ge, T., 2022. "Link prediction on complex networks: an experimental survey". *Data Science and Engineering,* ***7***(3), pp. 253–278.

[33] Wiki, O., 2022. Overpass turbo — openstreetmap wiki,. [Online; accessed 4-February-2022].

[34] Xiao, Y., Ahmed, F., and Sha, Z., 2022. "Travel links prediction in shared mobility networks using graph neural network models". In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Vol. 86212, American Society of Mechanical Engineers, p. V002T02A079.

[35] Bengio, Y., and Grandvalet, Y., 2003. "No unbiased estimator of the variance of k-fold cross-validation". *Advances in Neural Information Processing Systems,* ***16***.