Ranking Based Approach for Individual Fairness in GNNs

The main objective of this paper is to ensure individual fairness within graph neural networks while also preserving the performance of GNNs. Individual fairness refers to the notion of similar outcomes for individuals who share similar characteristics. To accomplish this, the research conducts experiments on two specific tasks: Node Classification and Link Prediction. These experiments are carried out using three different datasets, incorporating two backbone models, as well as two distinct similarity metrics and ranking metrics for each task.

What is Individual Fairness?

Firstly, what does individual fairness mean for Graph Neural Networks (GNNs)? We can discuss two fairness paradigms from this standpoint, namely individual and group fairness. Group fairness pertains to whether the predictions of models are similar for similar members of different subgroups (e.g. Hispanic, Asian). On the other hand, individual fairness focuses on individuals rather than groups. In this context, models are expected to provide similar outcomes for similar individuals. In this paper, the authors prefer to adopt a ranking-based approach instead of a distance-based approach for the sake of fairness.

In the distance-based approach, the main objective of fairness optimization is to ensure that the distances between two nodes in the input and output spaces remain as similar as possible. On the other hand, the ranking-based approach aims to maximize the similarity between rankings, which is determined by the distances between one node and other nodes in both the input and output spaces. In the ranking-based approach, each node is associated with two ranking lists. The first ranking list (S_G) is generated from the input space, while the second list (S_Y) is obtained from the model's prediction. The node with the closest distance to the target node holds the first rank in the ranking list. The ultimate goal, in order to achieve individual fairness, is to minimize the dissimilarity between these two ranking lists.

Node Similarity and Ranking Similarity

So, we need to calculate similarity matrices for the input and output spaces. Additionally, we need to employ a metric for comparing these similarity matrices. The authors of this paper utilized two similarity metrics to assess the similarities between two nodes. They employed cosine similarity to evaluate feature-based similarities, while Jaccard similarity was used for structural comparisons. In order to compare two distinct rankings, they adopted Expected Reciprocal Rank (ERR) and Cumulative Gain-based approach (NDCG). Furthermore, they focused on the first k elements within the ranking lists to mitigate algorithmic complexity.

Backbone Models and Their Performances

REDRESS is a modular framework consisting of two main modules. The first module aims to maximize the usefulness of the backbone model, while the second module focuses on achieving fairness. At the end, the loss function incorporates two terms for these modules. Additionally, the contribution of the fairness module can be adjusted by setting the hyperparameter gamma (γ). In this paper, Graph Convolutional Network (GCN) and Simplifying Graph Convolutional Network (SGC) are employed for the Node Classification task, while GCN and Variational Graph Auto-Encoders (GAE) are utilized for the Link Prediction task.

To assess the performance of the backbone models, accuracy (ACC) is employed for Node Classification, and the area under the receiver operating characteristic curve (AUC) is used for Link Prediction.

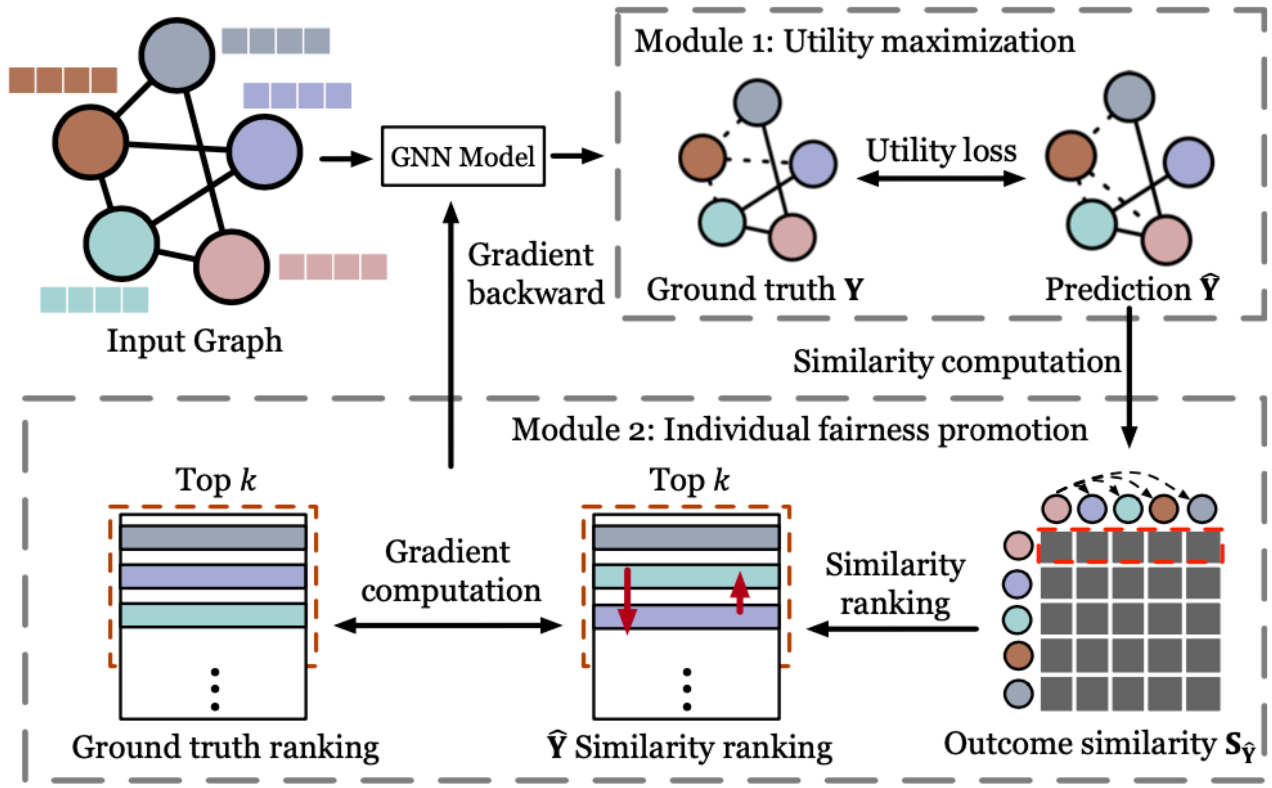

Figure 2: An illustration of the proposed REDRESS structure. Module 1 and 2 is utilized for model utility maximization and individual fairness promotion, respectively.

Datasets

We utilize three datasets for each downstream task. For the Node Classification task, we employ one citation network (ACM) and two co-authorship networks (Co-author-CS and Co-author-Phy). On the other hand, for the Link Prediction task, we utilize three social networks (BlogCatalog, Flickr, and Facebook).

In the citation network, each node represents a paper, and the edges between nodes indicate the citation relationships between papers. In the co-authorship networks, each node represents an author, and the edges represent collaboration relationships between them.

To calculate node attributes, we utilize the bag-of-words model, which involves analyzing the abstracts of the papers. For more detailed statistics regarding the datasets, please refer to the table below.

	Dataset	# Nodes	# Edges	# Features	# Classes
NC	ACM	16,484	71,980	8,337	9
	CS	18,333	81,894	6,805	15
	Phy	34,493	247,962	8,415	5
LP	BlogCatalog	5,196	171,743	8,189	N/A
	Flickr	7,575	239,738	12,047	N/A
	Facebook	4,039	88,234	1,406	N/A

Reproducibility and Our Results

All datasets are accessible to the public, and the authors of the paper have already shared their repository. Additionally, they have provided detailed explanations for the sake of reproducibility. As a result, I have obtained similar scores to those mentioned in the paper, and I have not noticed any inconsistencies. I have successfully replicated some of the results.

Instead of rerunning all 48 experiments, which would take approximately two days to complete due to running 24 trials for each task, I decided to include a subset of experiments for reporting regenerated results. The selection of which experiments to reproduce for reporting was based on factors such as diversity, dataset sizes, and the time required to complete the experiments.

By considering diversity, I aimed to include experiments that cover a wide range of scenarios, ensuring a representative sample of the overall experimental space. This approach allows for capturing different aspects and variations present in the tasks.

Additionally, dataset sizes played a role in the decision-making process. I considered the importance of including experiments that cover different dataset scales or distributions. By including a mix of large and small datasets, I could assess the generalizability and performance of the models across various data sizes.

Lastly, the time required for experiments to converge and produce results was taken into account. Since running all 48 experiments would take a considerable amount of time, I prioritized selecting experiments that yielded results relatively quickly. This approach allowed for a more timely analysis and reporting of the findings.

By considering these factors, I ensured that the subset of experiments chosen for reproduction and reporting would provide meaningful insights while optimizing the use of time and resources.

In [ ]:
```python
import pandas as pd
import subprocess
from collections import defaultdict
import json
import re
```

In [ ]:
```python
node_classification_setup = {
    "dataset": ["ACM", "coauthor-cs", "coauthor-phy"],
    "model": ["SGC", "GCN"],
    "node_similarity": ["feature", "structural"],
    "ranking_similarity": ["NDCG", "ERR"]
}

link_prediction_setup = {
    "dataset": ["BlogCatalog", "facebook", "Flickr"],
    "model": ["GCN", "GAE"],
    "node_similarity": ["feature", "structural"],
    "ranking_similarity": ["NDCG", "ERR"]
}
```

In [ ]:
```
def log_parser(log, fair_pattern, utility_pattern):
    fair = re.findall(fair_pattern, log[-1])
    utility = re.findall(utility_pattern, log[-1])
    if fair and utility:
        return log[:-1] + [round(float(fair[0]), 2), round(float(ut
ility[0]), 2)]
    else:
        return log[:-1] + [None, None]
```

In [ ]:
```
with open("node classification_results.json") as f:
    nc_result = json.load(f)

parsed_nc_result = [
    log_parser(
        experiment,
        "@k =  ([0-9.]+)\nTest set results",
        "accuracy= ([0-9.]+)\n"
    )
    for experiment in nc_result
]
```

In [ ]:
```
with open("link prediction_results.json") as f:
    lp_result = json.load(f)

parsed_lp_result = [
    log_parser(
        experiment,
        "fair_after\[([0-9.]+)\]",
        "auc_after\[([0-9.]+)\]",
    )
    for experiment in (lp_result)
]
```

In [ ]:
```
def get_dict_for_df(parsed_result):
    parsed_dict = defaultdict(list)
    for r in parsed_result:
        parsed_dict["Dataset"].append(r[3])
        parsed_dict["Backbone"].append(r[4])
        parsed_dict["Node Similarity"].append(r[1])
        parsed_dict["Ranking Similarity"].append(r[2])
        parsed_dict["Utility"].append(r[6])
        parsed_dict["Fairness"].append(r[5])
    return parsed_dict
```

Node Classification Reproduced Results

In [ ]:
```
pd.DataFrame(get_dict_for_df(parsed_nc_result)).dropna()
```

Out[ ]:

	Dataset	Backbone	Node Similarity	Ranking Similarity	Utility	Fairness
0	ACM	SGC	feature	NDCG	0.67	0.60
1	ACM	GCN	feature	NDCG	0.70	0.31
2	coauthor-cs	SGC	feature	NDCG	0.89	0.76
3	coauthor-cs	GCN	feature	NDCG	0.92	0.47
6	ACM	SGC	feature	ERR	0.66	0.83
7	ACM	GCN	feature	ERR	0.70	0.76
8	coauthor-cs	SGC	feature	ERR	0.91	0.92
9	coauthor-cs	GCN	feature	ERR	0.92	0.79
12	ACM	SGC	structural	NDCG	0.67	0.39
13	ACM	GCN	structural	NDCG	0.68	0.25
18	ACM	SGC	structural	ERR	0.68	0.49
19	ACM	GCN	structural	ERR	0.70	0.39

Table 3: Node classification results on ACM, Co-author-CS (CS) and Co-author-Phy (Phy) datasets. BB represents the backbone GNN model. Vanilla denotes the vanilla GNN. All values are reported in percentage. The relative improvement of each entry compared with the corresponding backbone performance is denoted in the parentheses. Best performance is marked in bold.

	BB	Model	Feature Similarity		Structural Similarity	
			Utility: ACC	Fairness: NDCG@10	Utility: ACC	Fairness: NDCG@10
ACM	GCN	Vanilla	**72.49 ± 0.6** (—)	47.33 ± 1.0 (—)	**72.49 ± 0.6** (—)	25.42 ± 0.6 (—)
		InFoRM	68.03 ± 0.3 (−6.15%)	39.79 ± 0.3 (−15.9%)	69.13 ± 0.5 (−4.64%)	12.02 ± 0.4 (−52.7%)
		PFR	67.88 ± 1.1 (−6.36%)	31.20 ± 0.2 (−34.1%)	69.00 ± 0.7 (−4.81%)	23.85 ± 1.3 (−6.18%)
		REDRESS (Ours)	71.75 ± 0.4 (−1.02%)	**49.13 ± 0.4** (+3.80%)	72.03 ± 0.9 (−0.63%)	**29.09 ± 0.4** (+14.4%)
	SGC	Vanilla	68.40 ± 1.0 (—)	55.75 ± 1.1 (—)	**68.40 ± 1.0** (—)	37.18 ± 0.6 (—)
		InFoRM	**68.81 ± 0.5** (+0.60%)	48.25 ± 0.5 (−13.5%)	66.71 ± 0.6 (−2.47%)	28.33 ± 0.6 (−23.8%)
		PFR	67.97 ± 0.7 (−0.62%)	34.71 ± 0.1 (−37.7%)	67.78 ± 0.1 (−0.91%)	37.15 ± 0.6 (−0.08%)
		REDRESS (Ours)	67.16 ± 0.2 (−1.81%)	**58.64 ± 0.4** (+5.18%)	67.77 ± 0.4 (−0.92%)	**38.95 ± 0.1** (+4.76%)
CS	GCN	Vanilla	90.59 ± 0.3 (—)	50.84 ± 1.2 (—)	**90.59 ± 0.3** (—)	18.29 ± 0.8 (—)
		InFoRM	88.66 ± 1.1 (−2.13%)	53.38 ± 1.6 (+5.00%)	87.55 ± 0.9 (−3.36%)	19.18 ± 0.9 (+4.87%)
		PFR	87.51 ± 0.7 (−3.40%)	37.12 ± 0.9 (−27.0%)	86.16 ± 0.2 (−4.89%)	11.98 ± 1.3 (−34.5%)
		REDRESS (Ours)	**90.70 ± 0.2** (+0.12%)	**55.01 ± 1.9** (+8.20%)	89.16 ± 0.3 (−1.58%)	**21.28 ± 0.3** (+16.4%)
	SGC	Vanilla	87.48 ± 0.8 (—)	74.00 ± 0.1 (—)	87.48 ± 0.8 (—)	32.36 ± 0.3 (—)
		InFoRM	88.07 ± 0.1 (+0.67%)	74.29 ± 0.1 (+0.39%)	88.65 ± 0.4 (+1.34%)	32.37 ± 0.4 (+0.03%)
		PFR	88.31 ± 0.1 (+0.94%)	48.40 ± 0.1 (−34.6%)	84.34 ± 0.3 (−3.59%)	28.87 ± 0.9 (−10.8%)
		REDRESS (Ours)	**90.01 ± 0.2** (+2.89%)	**76.60 ± 0.1** (+3.51%)	**89.35 ± 0.1** (+2.14%)	**34.24 ± 0.2** (+5.81%)
Phy	GCN	Vanilla	**94.81 ± 0.2** (—)	34.83 ± 1.1 (—)	**94.81 ± 0.2** (—)	1.57 ± 0.1 (—)
		InFoRM	89.33 ± 0.8 (−5.78%)	31.25 ± 0.0 (−10.3%)	94.46 ± 0.2 (−0.37%)	1.77 ± 0.0 (+12.7%)
		PFR	89.74 ± 0.5 (−5.35%)	24.16 ± 0.4 (−30.6%)	87.26 ± 0.2 (−7.96%)	1.20 ± 0.1 (−23.6%)
		REDRESS (Ours)	94.63 ± 0.7 (−0.19%)	**43.64 ± 0.5** (+25.3%)	93.94 ± 0.3 (−0.92%)	**1.93 ± 0.1** (+22.9%)
	SGC	Vanilla	**94.45 ± 0.2** (—)	49.63 ± 0.1 (—)	**94.45 ± 0.2** (—)	3.61 ± 0.1 (—)
		InFoRM	92.01 ± 0.1 (−2.58%)	43.87 ± 0.2 (−11.6%)	94.27 ± 0.3 (−0.19%)	3.64 ± 0.0 (+0.83%)
		PFR	89.74 ± 0.3 (−4.99%)	28.54 ± 0.1 (−42.5%)	89.73 ± 0.3 (−5.00%)	2.62 ± 0.1 (−27.4%)
		REDRESS (Ours)	94.30 ± 0.1 (−0.16%)	**53.40 ± 0.1** (+7.60%)	93.94 ± 0.2 (−0.54%)	**4.03 ± 0.0** (+11.6%)

Table 5: Node classification results on ACM, Co-author-CS (CS) and Co-author-Phy (Phy) datasets. BB represents the backbone GNN model. Vanilla denotes the vanilla GNN. All values are reported in percentage. The relative improvement of each entry compared with the corresponding backbone performance is denoted in the parentheses. Best performance is marked in bold.

	BB	Model	Feature Similarity		Structural Similarity	
			Utility: ACC	Fairness: ERR@10	Utility: ACC	Fairness: ERR@10
ACM	GCN	Vanilla	72.49 ± 0.6 (—)	75.70 ± 0.6 (—)	**72.49** ± 0.6 (—)	37.55 ± 0.4 (—)
		InFoRM	67.65 ± 1.0 (−6.68%)	73.49 ± 0.5 (−2.92%)	65.91 ± 0.2 (−9.07%)	19.96 ± 0.6 (−46.8%)
		PFR	68.48 ± 0.6 (−5.53%)	76.28 ± 0.1 (+0.77%)	70.22 ± 0.7 (−3.13%)	36.54 ± 0.4 (−2.69%)
		REDRESS (Ours)	**73.46 ± 0.2** (+1.34%)	**82.27 ± 0.1** (+8.68%)	71.87 ± 0.4 (−0.86%)	**43.74 ± 0.0** (+16.5%)
	SGC	Vanilla	**68.40 ± 1.0** (—)	80.06 ± 0.1 (—)	**68.40 ± 1.0** (—)	45.95 ± 0.3 (—)
		InFoRM	67.96 ± 0.5 (−0.64%)	75.63 ± 0.5 (−5.53%)	66.16 ± 0.6 (−3.27%)	39.79 ± 0.1 (−13.4%)
		PFR	67.69 ± 0.4 (−1.04%)	76.80 ± 0.1 (−4.07%)	66.69 ± 0.3 (−2.50%)	46.99 ± 0.5 (+2.26%)
		REDRESS (Ours)	66.51 ± 0.3 (−2.76%)	**82.32 ± 0.3** (+2.82%)	67.10 ± 0.7 (−1.90%)	**49.02 ± 0.2** (+6.68%)
CS	GCN	Vanilla	**90.59 ± 0.3** (—)	80.41 ± 0.1 (—)	**90.59** ± 0.3 (—)	26.69 ± 1.3 (—)
		InFoRM	88.37 ± 0.9 (−2.45%)	80.63 ± 0.6 (+0.27%)	87.10 ± 0.9 (−3.85%)	29.68 ± 0.6 (+11.2%)
		PFR	87.62 ± 0.2 (−3.28%)	76.26 ± 0.1 (−5.16%)	85.66 ± 0.7 (−5.44%)	19.80 ± 1.4 (−25.8%)
		REDRESS (Ours)	90.06 ± 0.5 (−0.59%)	**83.24 ± 0.2** (+3.52%)	89.81 ± 0.2 (−0.86%)	**32.42 ± 1.6** (+21.5%)
	SGC	Vanilla	87.48 ± 0.8 (—)	90.58 ± 0.1 (—)	87.48 ± 0.8 (—)	43.28 ± 0.2 (—)
		InFoRM	87.31 ± 0.5 (−0.19%)	90.64 ± 0.1 (+0.07%)	88.21 ± 0.9 (+0.83%)	43.37 ± 0.1 (+0.21%)
		PFR	87.95 ± 0.2 (+0.54%)	79.85 ± 0.2 (−11.8%)	86.93 ± 0.1 (−0.63%)	38.83 ± 0.8 (−10.3%)
		REDRESS (Ours)	**90.48 ± 0.2** (+3.43%)	**92.03 ± 0.1** (+1.60%)	**90.39 ± 0.1** (+3.33%)	**45.81 ± 0.0** (+5.85%)
Phy	GCN	Vanilla	**94.81 ± 0.2** (—)	73.25 ± 0.3 (—)	**94.81 ± 0.2** (—)	2.58 ± 0.1 (—)
		InFoRM	88.67 ± 0.7 (−6.48%)	73.80 ± 0.6 (+0.75%)	94.68 ± 0.2 (−0.14%)	2.45 ± 0.1 (−5.04%)
		PFR	88.79 ± 0.2 (−6.35%)	73.32 ± 0.4 (+0.10%)	89.69 ± 1.0 (−5.40%)	1.67 ± 0.1 (−35.3%)
		REDRESS (Ours)	93.71 ± 0.1 (−1.16%)	**80.23 ± 0.1** (+9.53%)	93.91 ± 0.4 (−0.95%)	**3.22 ± 0.3** (+24.8%)
	SGC	Vanilla	94.45 ± 0.2 (—)	77.48 ± 0.2 (—)	94.45 ± 0.2 (—)	4.50 ± 0.1 (—)
		InFoRM	92.06 ± 0.2 (−2.53%)	75.13 ± 0.4 (−3.03%)	94.27 ± 0.1 (−0.19%)	4.44 ± 0.0 (−1.33%)
		PFR	87.39 ± 1.2 (−7.47%)	73.42 ± 0.2 (−5.24%)	89.16 ± 0.3 (−5.60%)	3.41 ± 0.2 (−24.2%)
		REDRESS (Ours)	**94.81 ± 0.2** (+0.38%)	**79.57 ± 0.2** (+2.70%)	**94.54 ± 0.1** (+0.10%)	**4.98 ± 0.1** (+10.7%)

Link Prediction Reproduced Results

In [ ]:
```
pd.DataFrame(get_dict_for_df(parsed_lp_result)).dropna()
```

Out[ ]:

	Dataset	Backbone	Node Similarity	Ranking Similarity	Utility	Fairness
2	facebook	GCN	feature	NDCG	0.96	0.29
3	facebook	GAE	feature	NDCG	0.95	0.29
8	facebook	GCN	feature	ERR	0.95	0.65
9	facebook	GAE	feature	ERR	0.96	0.65
14	facebook	GCN	structural	NDCG	0.93	0.31
20	facebook	GCN	structural	ERR	0.93	0.44
21	facebook	GAE	structural	ERR	0.93	0.45

Table 4: Link prediction results on BlogCatalog (Blog), Flickr and Facebook (FB) datasets. BB represents the backbone GNN model. Vanilla denotes the vanilla GNN. All values are reported in percentage. The relative improvement of each entry compared with the corresponding backbone performance is denoted in the parentheses. Best performance is marked in bold.

	BB	Model	Feature Similarity		Structural Similarity	
			Utility: AUC	Fairness: NDCG@10	Utility: AUC	Fairness: NDCG@10
Blog	GCN	Vanilla	85.87 ± 0.1 (—)	16.73 ± 0.1 (—)	85.87 ± 0.1 (—)	32.47 ± 0.5 (—)
		InFoRM	79.85 ± 0.6 (−7.01%)	15.57 ± 0.2 (−6.93%)	84.00 ± 0.1 (−2.18%)	26.18 ± 0.3 (−19.4%)
		PFR	84.25 ± 0.2 (−1.89%)	16.37 ± 0.0 (−2.15%)	83.88 ± 0.0 (−2.32%)	29.60 ± 0.4 (−8.84%)
		REDRESS (Ours)	**86.49 ± 0.8** (+0.72%)	**17.66 ± 0.2** (+5.56%)	**86.25 ± 0.3** (+0.44%)	**34.62 ± 0.7** (+6.62%)
	GAE	Vanilla	**85.72 ± 0.1** (—)	17.13 ± 0.1 (—)	85.72 ± 0.1 (—)	41.99 ± 0.4 (—)
		InFoRM	80.01 ± 0.2 (−6.66%)	16.12 ± 0.2 (−5.90%)	82.86 ± 0.0 (−3.34%)	27.29 ± 0.3 (−35.0%)
		PFR	83.83 ± 0.1 (−2.20%)	16.64 ± 0.0 (−2.86%)	83.87 ± 0.1 (−2.16%)	35.91 ± 0.4 (−14.5%)
		REDRESS (Ours)	84.67 ± 0.9 (−1.22%)	**18.19 ± 0.1** (+6.19%)	**86.36 ± 1.5** (+0.75%)	**43.51 ± 0.7** (+3.62%)
Flickr	GCN	Vanilla	**92.20 ± 0.3** (—)	13.10 ± 0.2 (—)	92.20 ± 0.3 (—)	22.35 ± 0.3 (—)
		InFoRM	91.39 ± 0.0 (−0.88%)	11.95 ± 0.1 (−8.78%)	91.73 ± 0.1 (−0.51%)	23.28 ± 0.6 (+4.16%)
		PFR	91.91 ± 0.1 (−0.31%)	12.94 ± 0.0 (−1.22%)	91.86 ± 0.2 (−0.37%)	19.80 ± 0.4 (−11.4%)
		REDRESS (Ours)	91.38 ± 0.1 (−0.89%)	**13.58 ± 0.3** (+3.66%)	**92.67 ± 0.2** (+0.51%)	**28.45 ± 0.5** (+27.3%)
	GAE	Vanilla	89.98 ± 0.1 (—)	12.77 ± 0.0 (—)	89.98 ± 0.1 (—)	23.58 ± 0.2 (—)
		InFoRM	88.76 ± 0.7 (−1.36%)	12.07 ± 0.1 (−5.48%)	**91.51 ± 0.2** (+1.70%)	15.78 ± 0.3 (−33.1%)
		PFR	**90.30 ± 0.1** (+0.36%)	12.12 ± 0.1 (−5.09%)	90.10 ± 0.1 (+1.33%)	20.46 ± 0.3 (−13.2%)
		REDRESS (Ours)	89.45 ± 0.5 (−0.59%)	**14.24 ± 0.1** (+11.5%)	89.52 ± 0.3 (−0.51%)	**29.83 ± 0.2** (+26.5%)
FB	GCN	Vanilla	95.60 ± 1.7 (—)	23.07 ± 0.2 (—)	95.60 ± 1.7 (—)	16.55 ± 1.1 (—)
		InFoRM	90.26 ± 0.1 (−5.59%)	23.23 ± 0.3 (+0.69%)	**96.66 ± 0.6** (+1.11%)	15.18 ± 0.7 (−8.28%)
		PFR	87.11 ± 1.2 (−8.88%)	21.83 ± 0.2 (−5.37%)	94.87 ± 1.9 (−0.76%)	19.53 ± 0.5 (+18.0%)
		REDRESS (Ours)	**96.49 ± 1.6** (+0.93%)	**29.60 ± 0.1** (+28.3%)	92.66 ± 0.4 (−3.08%)	**27.73 ± 1.1** (+67.5%)
	GAE	Vanilla	**98.54 ± 0.0** (—)	26.75 ± 0.1 (—)	**98.54 ± 0.0** (—)	27.03 ± 0.1 (—)
		InFoRM	90.50 ± 0.4 (−8.16%)	22.77 ± 0.2 (−14.9%)	95.03 ± 0.1 (−3.56%)	15.38 ± 0.2 (−43.1%)
		PFR	96.91 ± 0.1 (−1.65%)	23.52 ± 0.1 (−12.1%)	98.28 ± 0.0 (−0.26%)	22.89 ± 0.3 (−15.3%)
		REDRESS (Ours)	95.98 ± 1.5 (−2.60%)	**28.43 ± 0.3** (+6.28%)	94.07 ± 1.7 (−4.54%)	**33.53 ± 0.2** (+24.0%)

Table 6: Link prediction results on BlogCatalog (Blog), Flickr and Facebook (FB) datasets. BB represents the backbone GNN model. Vanilla denotes the vanilla GNN. All values are reported in percentage. The relative improvement of each entry compared with the corresponding backbone performance is denoted in the parentheses. Best performance is marked in bold.

	BB	Model	Feature Similarity		Structural Similarity	
			Utility: AUC	Fairness: ERR@10	Utility: AUC	Fairness: ERR@10
Blog	GCN	Vanilla	**85.87 ± 0.1** (—)	67.95 ± 0.1 (—)	85.87 ± 0.1 (—)	38.63 ± 0.2 (—)
		InFoRM	80.14 ± 0.1 (−6.67%)	68.55 ± 0.1 (+0.88%)	83.68 ± 0.0 (−2.55%)	34.26 ± 0.9 (−11.3%)
		PFR	83.65 ± 0.0 (−2.59%)	68.04 ± 0.3 (+0.13%)	84.72 ± 0.1 (−1.34%)	37.28 ± 0.4 (−3.49%)
		REDRESS (Ours)	83.90 ± 0.2 (−2.29%)	**72.83 ± 0.2** (+7.18%)	**86.44 ± 0.0** (+0.66%)	**42.16 ± 0.1** (+9.14%)
	GAE	Vanilla	**85.72 ± 0.1** (—)	67.92 ± 0.1 (—)	85.72 ± 0.1 (—)	44.23 ± 0.2 (—)
		InFoRM	81.87 ± 0.1 (−4.49%)	68.36 ± 0.4 (+0.65%)	82.50 ± 0.1 (−3.76%)	33.98 ± 0.5 (−23.2%)
		PFR	83.49 ± 0.1 (−2.60%)	67.89 ± 0.0 (−0.04%)	84.31 ± 0.1 (−1.64%)	39.89 ± 0.2 (−9.81%)
		REDRESS (Ours)	85.30 ± 1.5 (−0.49%)	**69.62 ± 0.4** (+2.50%)	**85.77 ± 2.0** (+0.06%)	**47.44 ± 0.3** (+7.26%)
Flickr	GCN	Vanilla	92.20 ± 0.3 (—)	70.39 ± 0.1 (—)	92.20 ± 0.3 (—)	38.44 ± 0.5 (—)
		InFoRM	91.28 ± 0.0 (−1.00%)	72.17 ± 0.0 (+2.53%)	**92.24 ± 0.0** (+0.04%)	39.03 ± 0.4 (+1.53%)
		PFR	**92.43 ± 0.2** (+0.25%)	71.36 ± 0.2 (+1.38%)	92.06 ± 0.2 (−0.15%)	37.29 ± 0.7 (−2.99%)
		REDRESS (Ours)	87.89 ± 0.4 (−4.67%)	**73.90 ± 0.3** (+4.99%)	91.39 ± 0.0 (−0.88%)	**44.82 ± 0.5** (+16.6%)
	GAE	Vanilla	89.98 ± 0.1 (—)	70.34 ± 0.2 (—)	89.98 ± 0.1 (—)	36.98 ± 0.3 (—)
		InFoRM	90.56 ± 1.4 (+0.64%)	71.54 ± 0.1 (+1.71%)	**91.55 ± 0.2** (+1.74%)	35.58 ± 0.4 (−3.79%)
		PFR	90.44 ± 0.2 (+0.51%)	71.65 ± 0.2 (+1.86%)	90.09 ± 0.2 (+0.12%)	33.89 ± 0.3 (−8.36%)
		REDRESS (Ours)	**93.06 ± 0.3** (+3.42%)	**72.41 ± 0.2** (+2.94%)	87.96 ± 0.4 (−2.24%)	**44.00 ± 0.1** (+19.0%)
FB	GCN	Vanilla	95.60 ± 1.7 (—)	61.52 ± 0.5 (—)	**95.60 ± 1.7** (—)	32.18 ± 1.7 (—)
		InFoRM	90.66 ± 0.0 (−5.17%)	61.49 ± 0.2 (−0.05%)	94.65 ± 1.3 (−0.99%)	30.03 ± 1.7 (−6.68%)
		PFR	89.85 ± 2.0 (−6.01%)	62.02 ± 0.3 (+0.81%)	92.30 ± 0.5 (−3.45%)	30.62 ± 1.8 (−4.85%)
		REDRESS (Ours)	**95.99 ± 1.9** (+0.41%)	**64.08 ± 0.1** (+4.16%)	92.93 ± 0.8 (−2.79%)	**43.74 ± 1.5** (+35.9%)
	GAE	Vanilla	**98.54 ± 0.0** (—)	63.19 ± 0.1 (—)	**98.54 ± 0.0** (—)	42.17 ± 0.4 (—)
		InFoRM	92.80 ± 0.1 (−5.83%)	62.29 ± 0.0 (−1.42%)	94.75 ± 0.2 (−3.85%)	31.93 ± 0.6 (−24.3%)
		PFR	96.85 ± 0.1 (−1.72%)	61.71 ± 0.1 (−2.34%)	98.18 ± 0.1 (−0.37%)	39.04 ± 0.3 (−7.42%)
		REDRESS (Ours)	95.10 ± 0.7 (−3.49%)	**64.40 ± 0.7** (+1.91%)	92.35 ± 0.3 (−6.28%)	**44.54 ± 0.3** (+5.62%)

Possible Extension Idea

I am considering the creation of artificial nodes based on the concepts explored in the Computer Vision field. The concept revolves around generating new nodes by merging existing nodes from the training dataset. By taking into account the attributes of the nodes, including their relevant features, as well as the similarities observed among other nodes, my objective is to produce nodes that are located within close proximity to each individual node.