Supplementary Materials - Learning Graph Representation of Agent Diffusers [Accepted at AAMAS 2025]

1 Datasets Description

We evaluated our approach using a suite of datasets that are widely recognized and frequently utilized in literature [Xu et al.(2018), Zhang et al.(2021), Zhu et al.(2019)]. The selected datasets encompass a diverse range of images and include:

- **MSCOCO (Microsoft Common Objects in Context) [Cho et al.(2014)]**: A large-scale dataset that provides images with rich contextual information. It contains images spanning multiple categories and annotations for object detection, segmentation, and captioning tasks.
- **CUB (Caltech-UCSD Birds-200-2011) [Wah et al.(2011)]**: A dataset specifically focused on fine-grained bird species recognition. It includes high-resolution images of various bird species, along with detailed annotations.
- **LN-COCO (Large-scale Noisy-COCO) [Pont-Tuset et al.(2020)]**: A variation of the MSCOCO dataset designed to test the robustness of models against noisy data. It contains images similar to those in MSCOCO but with intentionally introduced noise and artifacts.
- **Multi-modal CelebA-HQ (MM CelebA-HQ) [Xia et al.(2021)]**: An extension of the CelebA-HQ dataset, which contains high-quality images of celebrities. This dataset provides multi-modal information, including various attributes and conditions for each image.

For our experiments, all images from these datasets were scaled to a resolution of 256×256 pixels to ensure consistency and comparability across different models and tasks. The detailed statistics and characteristics of these datasets are summarized in Table 1.

Table 1: Statistics of the datasets used in our experiments.

Dataset	Number of Images	Number of Classes
MSCOCO	123,287	80
CUB	11,788	200
LN-COCO	118,000	80
MM CelebA-HQ	30,000	1

2 Generalizability of LGR-ED

In this section, we will demonstrate how our findings can be generalized to any learning task that involves model ensembling. We introduce a conceptual framework called LGR-ME (Learning Graph Representation for Model Ensemble), which is designed to enhance the performance and applicability of ensemble methods across various tasks. The detailed architecture and workflow of the LGR-ME framework are depicted in Figure 1. The LGR-ME framework leverages graph representation learning to effectively combine multiple models, capturing complex interactions and dependencies between them. By representing each model as a node in a graph and the relationships between models as edges, LGR-ME can learn optimal ensemble strategies through graph-based techniques. This allows for a more nuanced and flexible approach to ensembling, accommodating a wide range of learning tasks and model types. Through comprehensive experiments, we will illustrate the robustness and adaptability of LGR-ME, showing its capability to improve ensemble performance in various scenarios. Our results indicate that LGR-ME not only enhances predictive accuracy but also provides insights into the contributions of individual models within the ensemble. This framework paves the way for more sophisticated and effective ensembling techniques, making it a valuable tool for researchers and practitioners in the field of machine learning.

3 Further Results

We also evaluated the LGR algorithm on seven datasets, five Kaggle classification datasets: Employee, Heart-attack, Titanic, Credit Card Approval, and Water Potability [Inc.(2023)], and the two following computer vision datasets:

1. The COCO dataset comprises approximately 118,000 training images within the train2017 set and 5,000 validation images in the val2017 set. The dataset encompasses a total of 80 object categories. Our evaluation metrics include box average precision calculated across various IoU thresholds, specifically at threshold values of 0.5 (AP50) and 0.75 (AP75).

2. The LVIS v1.0 dataset serves as an extensive object detection and instance segmentation dataset, featuring 100,000 training images and 20,000 validation images. Derived from the same source images as COCO, LVIS annotations reflect a long-tailed distribution across 1,203 categories. Evaluation in LVIS employs MS-COCO style box metrics, including AP, AP50,

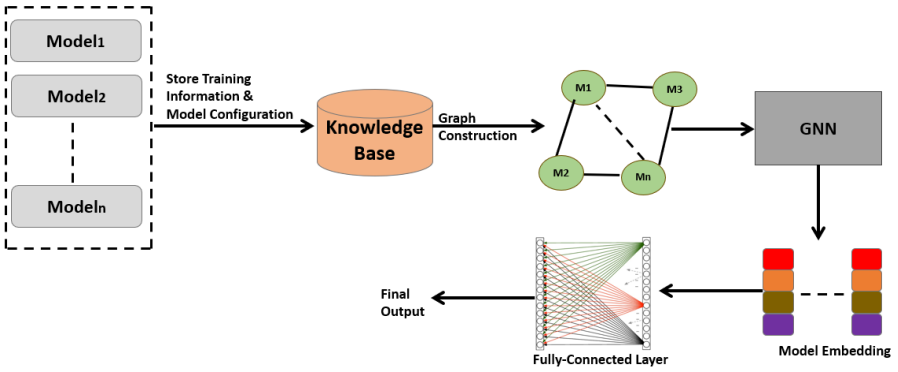

Figure 1: *LGR-ED's process starts by training machine learning models, we subsequently calculate the model specification, and output of each model. A model graph representation is created and learned using graph convolution neural network. A learned features are finally used to derive the final output of the desired task using fully connected layer.*

and AP75. The training schedule for LVIS consists of iterations at 210,000, 250,000, and 270,000.

We also used both shallow machine learning, and advanced deep learning models.

Shallow Machine Learning These include Random Forest with 10 estimators and a minimum samples split of 2, Gradient Boosting with 10 estimators and a 0.1 learning rate, Logistic Regression using L2 penalty and 'lbfgs' solver, SVC with an RBF kernel and 'scale' gamma, K-Nearest Neighbors with 5 neighbors, Decision Tree using Gini criterion, Gaussian Naive Bayes with a variance smoothing of 1×10^{-9}, and MLP with a 100-neuron hidden layer. Only the *PCF* function is used to assign weights to the graph edges.

Advanced Deep Learning We used DiffusionDet [Chen et al.(2023)], RetinaNet [Ke et al.(2020)], Sparse R-CNN [Sun et al.(2023)], Cascade-DETR [Ye et al.(2023)], and Decoupled-DETR [Zhang et al.(2023)]. For the hyperparameter optimization of the advanced deep learning models, we adopt the recent greedy search algorithm (GHO). In order to converge to the local optimal solution with the hope that this decision will result in a global optimal one, the GHO algorithm optimizes each hyperparameter while holding the others constant. Up until all of the hyperparameters are optimized, the local solution for each one is optimized iteratively.

Our GCNN processes graph data with three graph convolutional layers, transforming node features into 128, 32, and 18-dimensional spaces. Post-aggregation, features are averaged and passed through a fully connected layer, producing class probabilities.The GCNN is trained for 50 epochs with a batch size of 128 with a regularization parameters $\lambda = \gamma = 5.0$. To ensure a ro-

bust evaluation of the GCNN model, we employed KFold cross-validation where $K = 5$

Model	Accuracy	Precision	Recall	F1 Score
RandomForestClassifier	0.6368	0.6261	0.6368	0.6114
GradientBoostingClassifier	0.6275	0.6149	0.6275	0.6033
LogisticRegression	0.5958	0.4742	0.5958	0.4481
SVC	0.4291	0.5044	0.4291	0.3653
KNeighborsClassifier	0.5516	0.5316	0.5516	0.5354
DecisionTreeClassifier	0.5964	0.5967	0.5964	0.5963
GaussianNB	0.6169	0.5993	0.6169	0.5752
MLPClassifier	0.5629	0.6245	0.5629	0.4230
LGR-ME	**0.9975**	**0.9979**	**0.9979**	**0.9979**

Table 2: Average Metrics Over All Folds for Water Potability dataset

Model	Accuracy	Precision	Recall	F1 Score
RandomForestClassifier	0.7331	0.7328	0.7331	0.7285
GradientBoostingClassifier	0.7469	0.7435	0.7469	0.7390
LogisticRegression	0.7120	0.6961	0.7120	0.6973
SVC	0.6094	0.3891	0.6094	0.4734
KNeighborsClassifier	0.6775	0.6778	0.6775	0.6743
DecisionTreeClassifier	0.7605	0.7726	0.7605	0.7605
GaussianNB	0.6297	0.6725	0.6297	0.6218
MLPClassifier	0.6299	0.6030	0.6299	0.6097
LGR-ME	**0.9655**	**0.9717**	**0.9586**	**0.9650**

Table 3: Average Metrics Over All Folds for Titanic dataset

Model	Accuracy	Precision	Recall	F1 Score
RandomForestClassifier	0.8197	0.8174	0.8197	0.8148
GradientBoostingClassifier	0.8361	0.8373	0.8361	0.8296
LogisticRegression	0.7117	0.7015	0.7117	0.6808
SVC	0.5035	0.6116	0.5035	0.5016
KNeighborsClassifier	0.7800	0.7764	0.7800	0.7693
DecisionTreeClassifier	0.8114	0.8085	0.8114	0.8082
GaussianNB	0.6924	0.6809	0.6924	0.6831
MLPClassifier	0.6150	0.5261	0.6150	0.5259
LGR-ME	**0.8936**	**0.8932**	**0.8932**	**0.8932**

Table 4: Average Metrics Over All Folds for Employee dataset

3.1 Results Discussion for Shallow Machine Learning

The LGR-ME algorithm's performance across various datasets underscores its ability to adeptly combine the strengths of multiple weak classifiers. As illustrated in Tables 2, 3, 4, 5, and 6, LGR-ME often results in achieving superior performance metrics compared to any single classifier in its ensemble. For the Water Potability dataset for instance, LGR-ME's performance is nothing short of remarkable, achieving an accuracy of 0.9975, which is significantly higher than any other model, including RandomForest with an accuracy of 0.6368 and GradientBoosting at 0.6275. This demonstrates LGR-ME's capability to handle intricate patterns and nuances in datasets. The convergence of the loss

Model	Accuracy	Precision	Recall	F1 Score
RandomForestClassifier	0.7727	0.7779	0.7727	0.7730
GradientBoostingClassifier	0.7440	0.7517	0.7440	0.7439
LogisticRegression	0.8140	0.8166	0.8140	0.8133
SVC	0.6662	0.6782	0.6662	0.6562
KNeighborsClassifier	0.6781	0.6877	0.6781	0.6752
DecisionTreeClassifier	0.7484	0.7562	0.7484	0.7482
GaussianNB	**0.8142**	**0.8188**	**0.8142**	**0.8138**
MLPClassifier	0.6859	0.6698	0.6859	0.6661
LGR-ME	0.6401	0.6401	0.6401	0.6401

Table 5: Average Metrics Over All Folds for Heart-attack dataset

Model	Accuracy	Precision	Recall	F1 Score
RandomForestClassifier	0.8747	0.8793	0.8747	0.8748
GradientBoostingClassifier	0.8166	0.8180	0.8166	0.8164
LogisticRegression	0.8348	0.8389	0.8348	0.8334
SVC	0.6115	0.7108	0.6115	0.5269
KNeighborsClassifier	0.6787	0.6815	0.6787	0.6756
DecisionTreeClassifier	0.8220	0.8242	0.8220	0.8218
GaussianNB	0.7985	0.8140	0.7985	0.7926
MLPClassifier	0.7839	0.7913	0.7839	0.7825
GCN	**0.9074**	**0.9074**	**0.9074**	**0.9074**

Table 6: Average Metrics Over All Folds for Credit Card Approvals dataset

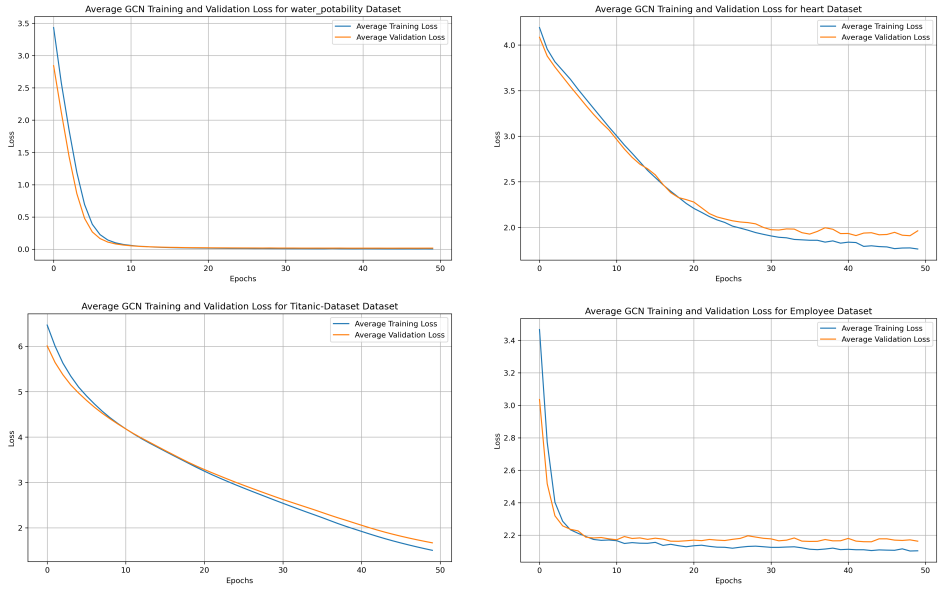

Figure 2: Training/Validation Loss for all datasets

function is a crucial indicator of the stability and reliability of a learning algorithm. It is essential to note that LGR-ME's performance isn't universally dominant. For instance, on the Heart-attack dataset, GaussianNB achieves the highest accuracy of 0.8142, while LGR-ME lags behind with 0.6401. This sug-

Model	AP	AP_{50}	AP_{75}
DiffusionDet	46.8	65.3	51.8
RetinaNet	38.7	58.0	41.5
Sparse R-CNN	40.3	61.2	43.9
Cascade-DETR	43.8	62.6	47.7
Decoupled-DETR	43.9	62.8	48.1
LGR-ME	**47.1**	**66.2**	**52.1**

Table 7: Comparisons with different object detectors on COCO 2017 val set.

Model	AP	AP_{50}	AP_{75}
DiffusionDet	31.5	43.4	33.5
RetinaNet	34.6	42.1	40.9
Sparse R-CNN	35.5	33.1	30.2
Cascade-DETR	26.5	37.9	27.9
Decoupled-DETR	28.4	39.1	29.2
LGR-ME	**48.3**	**49.1**	**48.5**

Table 8: Comparisons with different object detectors on LVIS v1.0 val set.

gests that while LGR-ME is generally adept at combining weak classifiers, its performance can still be contingent on the nature of the dataset and the quality of the base classifiers. For the Credit Card Approvals dataset, LGR-ME is not the top performer, with GCN taking the lead at 0.9074. However, LGR-ME's ensemble approach still allows it to achieve competitive results across the board, emphasizing the power of collaborative learning. For all the datasets, the loss function exhibits a clear trend towards convergence, indicating that the model is learning the underlying patterns effectively. For the datasets under consideration, we observed consistent convergence patterns, as illustrated in figure 2, below. For each of the highlighted datasets, the loss function exhibits a clear trend towards convergence, indicating that the model is learning the underlying patterns effectively. The detailed results and further discussions are provided in the appendix. On the Titanic dataset, LGR-ME again showcases its prowess with an accuracy of 0.9655, outperforming other models such as RandomForest (0.7331) and GradientBoosting (0.7469). Similarly, for the Employee dataset, LGR-ME achieves an accuracy of 0.8936, surpassing RandomForest (0.8197) and DecisionTree (0.8114).

3.2 Results Discussion for Advanced Deep Learning

In Table 7, we provide an extensive comparative analysis of our LGR-ME model in relation to several contemporary state-of-the-art object detectors, as evaluated on the COCO dataset. This comprehensive examination aims to highlight the performance and efficacy of our proposed LGR-ME in comparison to existing cutting-edge detection models. In further elucidation, our LGR-ME model demonstrates a notable achievement with an Average Precision (AP) score of 47.1, establishing a significant lead over several established methodologies. To delve into the intricacies of its performance enhancement, LGR-ME exhibits an elevated level of superiority, which can be fine-tuned and magnified by systematically increasing the number of iterations and expanding the scope of eval-

uation boxes. This nuanced approach not only underscores the robustness of our methodology but also provides a pathway for optimizing performance across diverse scenarios. Table 8 showcases the experimental outcomes on the LVIS dataset. Once more, the findings substantiate the exceptional performance of LGR-ME when compared to contemporary state-of-the-art object detection solutions. The efficacy of LGR-ME can be attributed to its adept learning representation, which efficiently strives to implicitly eliminate non-relevant models throughout the learning process. This intricate approach contributes significantly to the model's superiority in achieving robust results on the LVIS dataset.

References

[Chen et al.(2023)] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. 2023. Diffusiondet: Diffusion model for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 19830–19843.

[Cho et al.(2014)] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[Inc.(2023)] Kaggle Inc. 2023. Kaggle: Your Home for Data Science.

[Ke et al.(2020)] Wei Ke, Tianliang Zhang, Zeyi Huang, Qixiang Ye, Jianzhuang Liu, and Dong Huang. 2020. Multiple anchor learning for visual object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10206–10215.

[Pont-Tuset et al.(2020)] Jordi Pont-Tuset, Jasper Uijlings, Soravit Changpinyo, Radu Soricut, and Vittorio Ferrari. 2020. Connecting vision and language with localized narratives. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, 647–664.

[Sun et al.(2023)] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Zehuan Yuan, and Ping Luo. 2023. Sparse R-CNN: An end-to-end framework for object detection. *IEEE transactions on pattern analysis and machine intelligence* (2023).

[Wah et al.(2011)] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset. (2011).

[Xia et al.(2021)] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. 2021. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2256–2265.

[Xu et al.(2018)] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2018. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1316–1324.

[Ye et al.(2023)] Mingqiao Ye, Lei Ke, Siyuan Li, Yu-Wing Tai, Chi-Keung Tang, Martin Danelljan, and Fisher Yu. 2023. Cascade-DETR: Delving into High-Quality Universal Object Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6704–6714.

[Zhang et al.(2021)] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 833–842.

[Zhang et al.(2023)] Manyuan Zhang, Guanglu Song, Yu Liu, and Hongsheng Li. 2023. Decoupled detr: Spatially disentangling localization and classification for improved end-to-end object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6601–6610.

[Zhu et al.(2019)] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. 2019. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5802–5810.