Domain-Aware Label Smoothing for Robust Abstract Visual Reasoning

Seungyu Choi[1], Soobin Cha[1], Jongmin Lim[2] and Kwangsu Kim[3]

[1] Sungkyunkwan University, Suwon, Korea, {csg1718, chasoobin99}@ gmail.com
[2] Sungkyunkwan University, Suwon, Korea, jm.lim@g.skku.edu
[3] Sungkyunkwan University, Suwon, Korea, kim.kwangsu@skku.edu

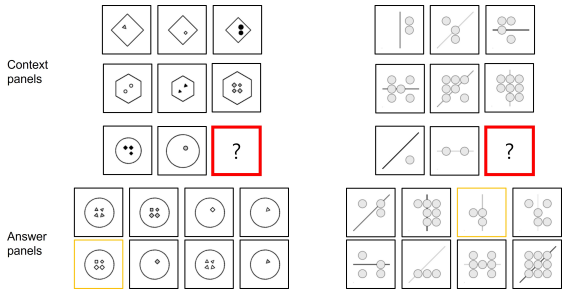
Abstract

The Raven's Progressive Matrices(RPM) problem involves discovering rules within a set of images, serving as an evaluation metric for AI models' visual reasoning capabilities. Noisy Contrast and Decentralization(NCD) was introduced by Tao et al. to tackle this task. However, we have identified certain limitations with the NCD approach. Specifically, it causes imbalanced label distribution in the preprocessing process, negatively affecting the model's robustness and generalization power. Meanwhile, the panels from other answers could be considered panels in different domains. To address these challenges, we propose a novel approach incorporating domain-aware label smoothing. By selectively applying label smoothing to specific rows based on the distinction of domains, we aim to enhance the model's robustness. Our experimental results on various RPM problem datasets demonstrate the efficacy of domain-aware label smoothing method in improving overall performance and show that the robustness of model calibration improves performance.

Keywords— *Abstract visual reasoning, Label smoothing, Class imbalance, Model calibration*

I. Introduction

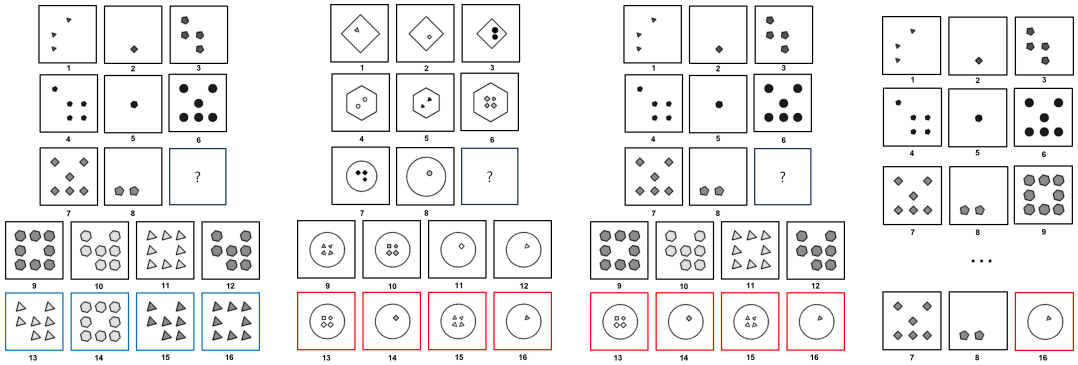
For humans, solving Raven's Progressive Matrices(RPM) problem is a task that only deduces relationships from the object's attribute without the influence of cultural and linguistic factors. An example of integrating the RPM problem is shown in Fig 1. A single problem contains eight context panels and eight answer panels including the target panel. Each panel has objects with attributes, and eight context panels are positioned in a 3*3 matrix with the bottom right missing(red panel). Each matrix has at least one rule, which is the relation between row-wise or column-wise panels. The task is to predict the correct panel(orange panel) from the answer panels that satisfies the rule of the matrix. RPM problem is an indicator to measure individual intelligence through visual information only. The ability to deduce hidden rules from visual information contributes to the advancement of science. One of the ultimate goals of vision AI is to make models have human's visual reasoning ability. In that perspective, the RPM problem can be used as an indicator to measure the model's ability to reason, a field of study called Abstract Visual Reasoning(AVR) is born [1].



(a) I-RAVEN problem (b) PGM problem

Fig. 1. **Examples of I-RAVEN dataset and PGM dataset.** Each RPM problem has context panels which compose a matrix, and answer panels. Matrix has at least one rule, and it is shared with rows in the case of the I-RAVEN dataset and rows or columns in the case of the PGM dataset.

NCD[11] is the first unsupervised learning method to tackle the RPM problem. The preprocessing process of NCD is shown in Fig 2. A new matrix with ten rows is generated based on the first two rows of the original matrix, which contain only context panels. The remaining eight rows are filled by putting answer panels into the missing piece. The training is done using pseudo labels. Specifically, the first two rows sharing the same rule are assigned to a positive label(1), and the other rows are assigned to a negative label(0). However, one of the eight rows is a correct answer, but it is miss labeled as a negative label, causing label noise. To reduce label noise, some answer panels are replaced with answer panels from another problem. However, there is still room for improvement in this approach. The main problem is that the class distribution is imbalanced. Class imbalance can cause overfitting because the model is prone to be biased towards the majority



(a) Original problem (b) Random other problem (c) Replaced answer panels (d) Generated rows

Fig. 2. **Problem setup of NCD.** Some of the answer panels in the original problem(blue rectangles) are replaced with answer panels of another problem(red rectangles). Then ten rows are generated by iteratively filling modified answer panels into the missing piece. So there are answer panels from different problems in a generated row from a single problem.

classes [10]. In addition, there is difference of domain between different problems. Each problem has unique rules, and the wrong answers are generated with slightly modified attributes from the correct answer. For a single problem, NCD replaces some answer panels with panels from other problems. So panels from different domains consist of a single problem.

In this paper, we propose LS-NCD that selectively applies label smoothing based on domain difference [8]. This method can effectively reduce overfitting and overconfidence by calibrating the model [7]. Specifically, it lowers the target label value of the context panel row and increases the target label value of the original answer panel row. Experiments with various RPM datasets [1, 9, 3] show an overall accuracy improvement compared to the existing methods. Moreover, by measuring the expected calibration error[5] on the PGM dataset, we demonstrate that the performance improvement results from resolving overconfidence and obtaining robustness through model calibration with our domain-aware label smoothing.

II. Related Work

Noisy Contrast and Decentralization(NCD). NCD is the first unsupervised method to solve the RPM problem. The training procedure of NCD starts with replacing some answer panels between problems, as shown in Fig 2. ResNet-18 [2] backbone extracts features from generated ten rows. And the location of each feature is decentralized using the newly defined feature centroid, which is calculated as the average of the first two rows. By adapting to various RPM problems, the distributed feature enhances the model's generalization capability. We conduct additional research on this and propose to improve the generalization performance. A problem with this method is an imbalance of instances between negative and positive labels. It can induce overconfidence and impair the robustness of the model. This paper aims to mitigate this challenge by using label smoothing.

Label Smoothing. When there is a class imbalance, the model tends to be biased towards the majority classes, which can lead to overconfidence [10]. Overconfident predictions of the model pose potential risks [6] such as precision degradation and reduce the model's reliability. Therefore, calibration can be used to make the model's confidence reflect its accuracy. Label smoothing is one of the widely used techniques to improve model calibration. Assuming that the model is trained with cross-entropy loss, let z be the logit vector of the penultimate layer. z will be used to compute the final output probability $\hat{y}$ through softmax, and the cross entropy loss L_{CE} will be calculated as follows.

$$L_{CE} = \hat{y} - y = softmax(z) - y, \tag{1}$$

To reduce this loss, the model will push z to the value of y. If y is a hard label, the model will create a large gap between the logits of each class to match z to y, which means that the model is overconfident about its logits. However, if label smoothing is applied to y, the logit gap will decrease. In this paper, we note the existence of class imbalance in the preprocessing setup of NCD. To address this problem, we perform model calibration using label smoothing.

III. Method

Domain-Aware Label Smoothing. In the problem setup of the NCD method, the ratio of positive to negative instances is 2:8, which leads to class imbalance. It can lead to overconfidence, as the model may exhibit a biased inclination towards the majority class, resulting in inaccurate predictions. To address this issue, we propose a Domain-Aware Label Smoothing method that calibrates the model by regularizing the overconfidence, leading to robustness. Fig 4 illustrates the various forms that the first two rows

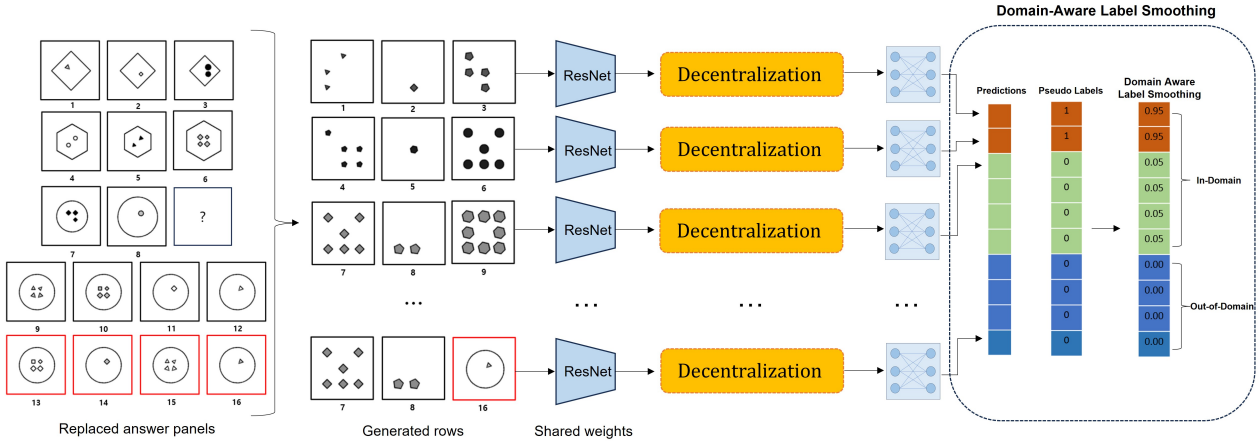


Fig. 3. **Overall architecutre of LS-NCD.**

can take. The rules for the positive rows of this example are as follows.

- The number of objects in the third panel is the sum of the number of objects in the first and second panels.
- The angle of objects increases as the panel moves to the right.
- The color of objects is consistent across the panels.

In this case, changing panel 3 or panel 6 as follows does not violate the rule for the row. However, NCD assigns a label value of 1 to the fixed form of the first two rows. Therefore, the label value of the first two rows should be reduced.

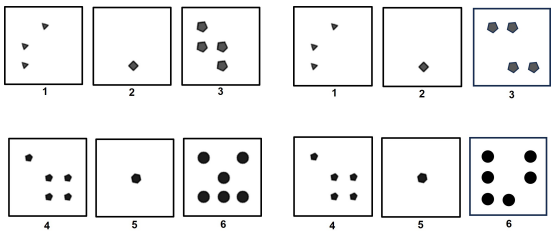
Moreover, label smoothing can be applied differently based on the distinction of domains of the eight rows that contain the answer panel. Fig 5 illustrates the difference between rows that contain the original answer panel and rows that contain replaced answer panels. Each problem has different rules. When generating the answer set for the RPM problem datasets, having too many easy negatives in the answer panel prevents the model from learning the representation sufficiently. Therefore, negative answers are created by modifying some attributes of the correct answer. NCD replaces the answer panel with one from a different problem, in which case a row that contains a replaced answer panel has different domain. Based on this, label smoothing is applied selectively depending on the domain distinctions of each row. Let $l_{i,j}$ be the value of the pseudo label for jth row of ith problem. In this case $j \in \{1,...,10\}$. Originally,

$$\begin{cases} l_{i,j} = 1, & j \in \{1,2\} \\ l_{i,j} = 0, & j \in \{3,...,10\}. \end{cases} \tag{2}$$

Then let k be the number of replaced answer panels from another problem and α be the label smoothing value. In LS-NCD,

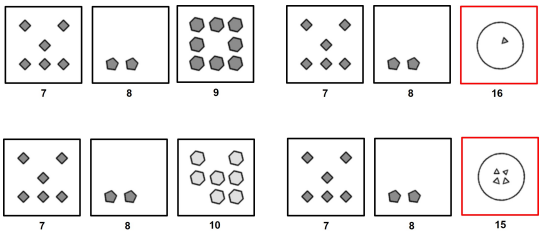
$$\beta = \alpha/(10-k), \tag{3}$$

$$\begin{cases} l_{i,j} = 1-\alpha/2+\beta, & j \in \{1,2\} \\ l_{i,j} = \beta, & j \in \{2,...,9-k\} \\ l_{i,j} = 0, & j \in \{10-k,...,10\}. \end{cases} \tag{4}$$



(a) Original form of rows (b) Alternative form of rows

Fig. 4. **Various possible forms of first two rows.** In the existing method, the label value of the first two rows is calculated according to the top line of Eqn 2, whereas in our method, considering that the first two rows can have the form of (b) as well as (a), the label value of the first two rows is calculated according to the top line of Eqn 4.



(a) Original rows (b) Replaced rows

Fig. 5. **Comparison between original rows and replaced rows.** During the NCD process, the label value of both (a) and (b) is calculated according to the bottom line of Eqn 2, whereas in our method, taking into account the domain distinction between the rows, the label value of (a) is calculated according to the second line of Eqn 4, and the label value of (b) is calculated according to the third line of Eqn 4.

Network Architecture. Our model's overall architecture is based on NCD, as shown in Fig 3. Specifically,

Configuration	Total	Center	2*2Grid	3*3Grid	L-R	U-D	O-IC	O-IG
NCD	30.05 / 43.34	41.35 / 56.45	27.9 / 27.7	33.05 / 25.05	26.4 / 50.45	25.9 / 50.9	30.45 / 54.7	25.3 / 36.55
LS-NCD	31.33 / 44.42	41.4 / 57.35	28.35 / 28.65	32.8 / 27.2	28.9 / 51.65	27.75 / 51.9	33.95 / 55.35	26.15 / 38.85
Difference	1.28 / 1.08	0.05 / 0.9	0.45 / 0.95	-0.25 / 2.15	2.5 / 1.2	1.85 / 1.0	**3.5** / 0.65	0.85 / **2.3**

Table 1. **Accuracy(%) with NCD and LS-NCD on RAVEN/I-RAVEN.**

for a single problem, some parts of the answer panels are replaced with those from another problem. Then a 10*3 matrix is constructed by appending two rows that consist of the context panel only, and the other eight rows are obtained by iteratively filling the missing piece with one answer panel. A pretrained ResNet-18 backbone extracts feature from each row, and features are decentralized. The decentralized features are fed into the classifier. Binary cross-entropy is used during training to calculate the loss between the predicted outputs and domain-aware label smoothed pseudo labels.

IV. EXPERIMENT

Dataset. We performed experiments on PGM, RAVEN, and I-RAVEN datasets. Each problem in the PGM dataset has rules across rows or columns. It serves as a benchmark to measure the model's generalization performance with regimes, which will be explained later. RAVEN dataset has seven different problem types. For example, four objects are in each panel in distribute-four and nine in each panel in distribute-nine. Each problem in RAVEN dataset has rules across a row. However, there is a drawback in RAVEN dataset as the wrong answers in the answer panel are obtained from the correct answer by changing only one attribute so that the model can predict the answer with high accuracy without considering the context panel. To solve this issue, I-RAVEN dataset is introduced. It has the same characteristics as RAVEN dataset, except for the process of generating wrong answers.

Experiment Setup. We used a computing environment with a 12-Core AMD Ryzen 9 3900X CPU, a single GEFORCE RTX 2080 Ti GPU, and 32GB RAM for hardware. We used pretrained ResNet-18 [2] as the backbone for the feature extractor. We used Adam optimizer [4] with a fixed learning rate of 0.00002. We set the smoothing value, a hyperparameter for NCD and LS-NCD, to 0.2. For the number of replacements, another hyperparameter, we used the default values of 4 and 6 for NCD and LS-NCD, respectively.

Configuration	Neutral	Interpolation	Extrapolation
NCD	18.61	14.58	13.79
LS-NCD	19.2	15.68	15.46
Difference	0.59	1.1	**1.67**

Table 2. **Accuracy(%) with NCD and LS-NCD on PGM.**

Result. The testing accuracy of LS-NCD is 31.33% on RAVEN dataset and 44.42% on I-RAVEN dataset. The results are shown in Table 1. We first compare LS-NCD with NCD. There is an overall improvement in performance when domain-aware label smoothing is applied to the original method with enhanced robustness. However, the degree of performance improvement by applying LS-NCD varies depending on the problem types. One of these problem types is *Out-InGrid*(O-IG). It has a single large object in each panel that contains a 2*2 grid of small colored objects within it, as shown in example (b) in Figure 2. In O-IG, there is a large gap in performance between NCD and LS-NCD on I-RAVEN dataset. This gap is because it uses not only attributes of large object, but also attributes of small objects.

In Table 2, we report the results of evaluating NCD and LS-NCD on PGM dataset evaluation. Here, LS-NCD achieved an accuracy of 19.2%

Ablation Study. To verify the model calibration performance by domain-aware label smoothing, we measured the expected calibration error (ECE) [5] for NCD and LS-NCD using the PGM dataset. ECE is computed by dividing the model's predicted values into M-equispaced bins, then summing up the differences between the model's confidence and the expected accuracy calculated from the test dataset instances for each bin. Formally, let M be the number of bins, n be the number of instances, and B_m be the number of instances in a specific bin, then ECE is calculated as follows.

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} \left| acc(B_m) - conf(B_m) \right|, \qquad (5)$$

Table 3 shows the results of calculating ECE for NCD and LS-NCD evaluated on the PGM dataset. The results show that the ECE value is much lower in the interpolation and extrapolation regimes than in the neutral regime, where the accuracy improvement is relatively small. It demonstrates that applying model calibration to NCD using domain-aware label smoothing reduces overconfidence and generalizes the model, leading to accuracy improvement.

Configuration	Neutral	Interpolation	Extrapolation
NCD	0.073	0.061	0.062
LS-NCD	0.074	0.038	0.040
Error difference	-0.001	0.023	0.022

Table 3. **Expected calibration error with NCD and LS-NCD on PGM.** The value in parentheses of the difference indicates the accuracy gain.

V. CONCLUSION.

We address the label imbalance problem with the NCD method and propose domain-aware label smoothing. With this method, the model's robustness and generalization power become more powerful in various RPM datasets, as proved by our experimental results. Especially in the extrapolation regime of PGM dataset, it showed much higher performance than the NCD. We hope that our approach can be applied to other methods for solving RPM problems and improve their performance due to the robustness of our model.

VI. ACKNOWLEDGEMENT

This work was supported by Korea Internet & Security Agency(KISA) grant funded by the Korea government(PIPC) (No.RS-2023-00231200, Development of personal video information privacy protection technology capable of AI learning in an autonomous driving environment)

REFERENCES

[1] D Barrett, F Hill, A Santoro, A Morcos, and T Lillicrap. Measuring abstract reasoning in neural networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 511–520, Stockholm, Sweden, 2018.

[2] K He, X Zhang, S Ren, and J Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[3] S Hu, Y Ma, X Liu, Y Wei, and S Bai. Stratified rule-aware network for abstract visual reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[4] D P Kingma and J Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[5] V Kuleshov and P Liang. Calibrated structured prediction. *NeurIPS*, 28:3476–3484, 2015.

[6] T Kapur et al M Ghafoorian, A Mehrtash. Transfer learning for domain adaptation in mri: Application in brain lesion segmentation. *Medical image computing and computer-assisted intervention*, 10435:516–524, 2017.

[7] R Müller, S Kornblith, and G Hinton. When does label smoothing help? *NeurIPS*, 32:4694–4703, 2019.

[8] C Szegedy, V Vanhoucke, S Ioffe, J Shlens, and Z Wonja. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.

[9] C Zhang, F Gao, and B et al. Raven: A dataset for relational and analogical visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5317–5327, 2019.

[10] W Zheng and M Jin. The effects of class imbalance and training data size on classifier learning: an empirical study. *SN Computer Science*, 1(2):71, 2020.

[11] T Zhuo, Q Huang, and M Kankanhalli. Unsupervised abstract reasoning for raven's problem matrices. *IEEE Transactions on Image Processing*, 30:8332–8341, 2021.

SUMMARY OF THIS PAPER

A. *Problem Setup*

In this paper, we define the class imbalance in the ten rows from NCD(Noisy Contrast and Decentralization) preprocessing, one of the unsupervised methods for solving the RPM problem. To prevent the model from becoming overconfident and losing reliability due to class imbalance, we apply model calibration using label smoothing. In this case, we apply label smoothing selectively according to the domain difference between the ten rows.

B. *Novelty*

To the best of our knowledge, this is the first method that tackles the RPM problem from the perspective of class imbalance. And it is also the first try to apply label smoothing in this research area.

C. *Algorithms*

We apply domain-aware label smoothing based on the NCD method. Since the rows which contain context panels can have various forms without violating the given rules, the label value of those rows is reduced. And since there is a domain difference between the rows which contain the original answer panel and the rows which contain replaced answer panel, only the label value of the former is increased.

D. *Experiments*

When we measured the accuracy on the RAVEN, I-RAVEN, and PGM datasets, we observed that LS-NCD, which applies domain-aware label smoothing, outperformed the original method NCD in overall accuracy. Moreover, our method showed a much lower error when we measured the expected calibration error for NCD and LS-NCD on the PGM dataset. It confirmed that the performance improvement was due to the model calibration by the effect of label smoothing, which reduced overconfidence and obtained robustness.