General Greedy De-Bias Learning

Xinzhe Han, Student Member, IEEE, Shuhui Wang, Member, IEEE, Chi Su, Qingming Huang, Fellow, IEEE, and Qi Tian, Fellow, IEEE

Abstract—Neural networks often make predictions relying on the spurious correlations from the datasets rather than the intrinsic properties of the task of interest, facing with sharp degradation on out-of-distribution (OOD) test data. Existing de-bias learning frameworks try to capture specific dataset bias by annotations but they fail to handle complicated OOD scenarios. Others implicitly identify the dataset bias by special design low capability biased models or losses, but they degrade when the training and testing data are from the same distribution. In this paper, we propose a General Greedy De-bias learning framework (GGD), which greedily trains the biased models and base model. The base model is encouraged to focus on examples that are hard to solve with biased models, thus remaining robust against spurious correlations in the test stage. GGD largely improves models’ OOD generalization ability on various tasks, but sometimes over-estimates the bias level and degrades on the in-distribution test. We further re-analyze the ensemble process of GGD and introduce the Curriculum Regularization inspired by curriculum learning, which achieves a good trade-off between in-distribution (ID) and out-of-distribution performance. Extensive experiments on image classification, adversarial question answering, and visual question answering demonstrate the effectiveness of our method. GGD can learn a more robust base model under the settings of both task-specific biased models with prior knowledge and self-ensemble biased model without prior knowledge. Codes are available at https://github.com/GeraldHan/GGD.

Index Terms—Curriculum learning, dataset biases, greedy strategy, robust learning.

I. INTRODUCTION

Deep learning have been used in a wide range of tasks that involves vision and/or language [1]. Most of the current approaches are data-driven and heavily rely on the assumption that the training and testing data are drawn from the same distribution. They are usually susceptible to poor generalization on out-of-distribution or biased settings [2]. This limitation partially arises because supervised training only identifies the correlations between given examples and their labels [3], which may reflect the dataset-specific bias rather than intrinsic properties of the task of interests [4], [5]. In general, under the supervised objective function fitting paradigm, if the bias is sufficient to make the model achieve high accuracy, there is less motivation for models to further learn those true intrinsic factors of the task. For example, QA models trained on SQuAD [6] tend to select the text near question-words as answers regardless of the context [7], [8], and VQA models usually leverage superficial correlations between questions and answers without considering the vision information [9], [10]. When it comes with the more common situation that the distribution of test data deviates from that of training data, models exploiting the biases in training data are prone to show poor generalization and hardly provide proper evidence for their predictions.

Being aware of this problem, researchers re-examine many popular datasets, resulting in the discovery of a wide variety of biases on different tasks, such as language bias in VQA [11], color bias in Biased-MNIST [12], gender/background bias in image classification [13], [14], and the ubiquitous long-tailed distribution [15], [16]. Built on these findings, explicit de-bias methods [12], [17], [18], [19], [20], [21] assume that bias variables are explicitly annotated, then the out-of-distribution performance can be directly improved by preventing the model from using the known biases or bias-related data augmentation [22], [23].

Although these methods achieve remarkable improvement on typical diagnosing datasets, they can only mitigate one specific bias, which is inconsistent with the real world datasets with compositional biases [24]. For example, in VQA, biases may stem from unbalanced answer distribution, spurious language correlations, and object contexts.

Even when all bias variables are identified, the explicit de-bias methods still cannot well handle multiple types of biases. Some recent works, i.e., the implicit methods [25], [26], [27], [28], try to discover the compositional biases without explicit task-related prior knowledge. They are somehow overcomplicated and usually perform worse than explicit methods under well-defined circumstances with known biases.
In fact, the dataset biases can be reduced in a more straightforward manner. As shown in Fig. 1, features learned from biases are thought to be “spurious” because they can only generalize to the majority groups of samples in the dataset. Although the model may incur high training error on the minority groups where the spurious correlation does not hold, the overall loss will still be trapped in a local minimum due to the low average training error dominated by the majority groups. Compared with the core features (e.g., the semantics of objects), it is relatively easier to identify the biases brought by distractive information (e.g., the backgrounds). If it is possible to know ahead of time that which subsets of instances are irrelevant to spurious features, we can encourage the model to focus on these samples and then reduce the unexpected correlations. In our preliminary work [29], we propose a de-bias framework Greedy Gradient Ensemble (GGE) to mitigate the language biases in VQA, achieving great improvement on the biased dataset VQA-CP [11]. GGE greedily learns a series of biased models and then ensembles the biased models and the base model like gradient descent in the functional space. The gradient of biased model separately for ensemble or distillation, but they learning [31], [32], relying on extra effort in constructing a biased model separately for ensemble or distillation, but they appear to be less flexible in dealing with complex real applications. In this paper, we first re-analyse the cause of bias overestimation in GGE. We empirical find that if a large amount of data can be correctly predicted via the biased model with high confidence, they will be excluded in the training of base model. As a result, the base model may be under-fitted to some labels due to inadequate training data. Decomposing the negative gradient of cross-entropy loss, we further find that the cross-entropy between the base prediction and biased prediction measures the difficulty of samples in GGE.

Base on this finding, we transform the negative gradient supervision to a flexible regularization and formulate a more general framework, i.e., General Greedy De-bias (GGD), to tackle the bias over-estimation problem more appropriately. Inspired by the curriculum learning [33], we treat the regularization term as a difficulty metric for the curriculum selection function. In this way, all data can participate in base model training in the early stage and gradually focus on hard examples along with the training procedure. This treatment endows our model with more flexibility, and demonstrates robustness on both out-of-distribution test data and general datasets like ImageNet [34] and CIFAR [35].

In the experiments, we apply GGD to a wider range of unimodal and multi-modal tasks, including visual classification, linguistic question answering, and visual question answering demonstrate the effectiveness of GGD on different types of biases. The main contributions of this paper are summarized as:

- We present a de-bias framework, General Greedy Debias Learning, which encourages unbiased based model learning by the robust ensemble of biased models. GGD is more generally applicable compared to task-related explicit de-bias learning methods while more flexible and effective compared to implicit de-bias methods.
- We propose Curriculum Regularization for GGD, which results in a new training scheme GGD$_{cr}$ that can better alleviate the “bias over-estimation” phenomenon. Compared with previous methods [36], [37], GGD$_{cr}$ comes to a better trade-off between in-distribution and out-of-distribution performance without either extra unbiased data in training or model ensemble in inference.
- Experiments on image classification, question answering, and visual question answering demonstrate the effectiveness of GGD on different types of biases.

This paper provides a more general debias learning framework compared to our preliminary study [29]. First, the previous work [29] only aims at the VQA task, in which we pay attention to the bias analysis on VQA-CP [11] and new evaluation metric for models’ visual grounding ability, while this paper considers the general de-bias learning problem and extend our framework to various datasets and applications. Second, we provide discussions for the “bias over-estimation” phenomenon in previous GGE [29]. We propose a flexible GGD$_{cr}$ optimization scheme...
that effectively improves the in-distribution performance on different tasks. Third, we provide more in-depth analysis for the greedy de-bias strategy, and the differences between GGD and previous GGE are demonstrated from both theories and experiments. Fourth, we provide more experiments on GGD with known biases and unknown biases, and comparison with the latest de-bias methods with both explicit and implicit bias modelling are also provided. Finally, we apply GGD on three additional tasks, i.e., image classification, adversarial question answering, and visual question answering, which are the representative tasks from CV, NLP, and Vision-Language, respectively, demonstrating the circumstances under one single bias, unknown bias, long-tailed bias and multiple biases. In addition to VQA-CP and VQA v2 in [29], we add experiments on more datasets, i.e., Biased MNIST [12], SQuAD [6], Adversarial SQuAD [38], GQA-OOD [39], CIFAR-10 and CIFAR-100 [35]. Experiments on various tasks and datasets fully demonstrate the general applicability of GGD in debias learning.

II RELATED WORK

A De-Biasing From Data Sources

When collecting real-world datasets, biases in the data are inevitable. Torralba and Efros [2] show how biases affect some commonly used datasets. It draws consideration on the generalization performance and classification capability of the trained deep models. Recent dataset construction protocols have tried to avoid certain kinds of biases. For example, on both CoQA [40] and QuAC [41] for QA task, annotators are prevented from using words that occur in the context passage. For VQA, Zhang et al. [42] collect complementary abstract scenes with opposite answers for all binary questions. Similarly, VQA v2 [9] is introduced to weaken the language priors in the VQA v1 dataset [43] by adding similar images with different answers for each question.

However, constructing large-scale datasets is costly. It is crucial to develop models that are robust to biases [44]. Towards this goal, new diagnosing datasets are established by amplifying some specific biases. For instance, Agrawal et al. [11] constructed a diagnosing VQA dataset under Changing Prior (VQA-CP), with different answer distributions between the train and test splits. Adversarial SQuAD [38] is built by adding distracting sentences to the passages in SQuAD [6]. He et al. collect NICO [14] dataset that consists of images with different backgrounds and gestures. All these new datasets can be used to test the models' generalization ability on out-of-distribution scenarios.

B Explicitly De-Biasing With Known Bias

To train a de-biased model, some works utilize an intentionally biased model to de-bias another model. For VQA, Ramakrishnan et al. [45] introduce an adversarial regularization to remove the discriminative features related to the answer categories from the questions. RUBi [18] and PoE [46] re-weight samples based on the question-only predictions. Kim et al. [20] propose a regularization term based on mutual information between the feature embedding and the bias, to remove the known bias for image classification. Similarly, Clark et al. [17] construct bias-only models for VQA, reading comprehension, and natural language inference (NLI), then reduce them with bias production and entropy maximization. Xiong et al. [30] further conduct uncertainty calibration on the bias-only models for a better de-biasing performance.

It can detect sample outliers and feature noises simultaneously. Bahng et al. [12] find that Hilbert-Schmidt Independence Criterion (HSIC) [47] can encourage a set of features to be statistically independent. They capture local texture bias in image classification and static bias in the video action recognition task using small-capacity models and then train a de-biased representation that is independent of biased representations based on HSIC.

Teney et al. [23] generate counterfactual samples with specific prior knowledge for different tasks. The vector difference between pairs of counterfactual examples serves to supervise the gradient orientation of the network. Liang et al. [48] propose A-INLP that dynamically finds bias-sensitive tokens and mitigates social bias in text generation. Tartaglione et al. [21] propose a new regularization named EnD, which aims to disentangle the features having the same “bias label”. Sagawa et al. [19] avoid bias over-fitting by defining prior data sub-groups and controlling their generalization. HEX [49] pushes the model to learn representations from which the texture representation is not predictable with the reverse gradient method. Gat et al. [50] introduce a regularization by maximizing functional entropies (MFE), which forces the model to use multiple information sources in multi-modal tasks. Zhu et al. [51] explicitly extract target and bias features from the latent space. Then they learn to discover and remove their correlation with the mutual information estimation. Hong et al. [52] leverage the knowledge of bias labels and propose Bias-Contrastive and Bias-Balanced losses based on the contrastive learning.

The above methods only focus on one specific bias but cannot work well on compositional biases. GGD can sequentially mitigate multiple bias variables as long as they can be characterized with prior knowledge, which is much more flexible than explicit de-biasing methods.

C Implicitly De-Biasing Without Known Bias

In real-world scenario, bias presented in the dataset is often hard to characterize and disentangle. To address this issue, there have been several recent works to resolve dataset bias without explicit supervision on the biases. For linear models, to alleviate the co-linearity among variables, Shen et al. [53] propose to learn a set of sample weights that can make the design matrix nearly orthogonal. Kuang et al. [54] further propose a re-weighting strategy so that the weighted distribution of treatment and confounder could satisfy the independent condition.

For deep models, most implicit methods assume that easy-to-learn biases can be captured by models with limited capacity and model parameters [55], using a small subset of training instances in a few epochs [56], and a classifier attached to intermediate layers [57]. Apart from limited capacity biased
models, Huang et al. [27] iteratively discard the dominant features activated on training data and force the network to activate the remaining features correlated with labels. Nam et al. [25] amplify the biases using generalized cross-entropy (GCE) loss and train a de-biased classifier with resampling based on the biased classifier. Still based on GCE, BiasSwap [28] further generates bias-swapped images from bias-contrary images as bias-guided data augmentation. Zhang et al. [38] introduce a non-linear feature decorrelation approach based on Random Fourier Features, which can approximate the Hilbert-Schmidt norm in euclidean space. Spectral Decoupling [26] decouples the learning dynamics between features. It aims to overcome the issue of gradient starvation, which indicates the tendency to only rely on statistically dominant features. Moreover, the setting of implicit de-biasing is similar to the Domain Generalization (DG) [59], [60] but has different challenges. In DG, the model is encouraged to generalize to a new domain that is not accessible during training while the de-bias has a small amount of training data that is bias-conflicted. Meanwhile, since there is no clear “domain discrepancy” in the biased sets, most existing DG methods do not work on the dataset bias problem.

Implicit methods are much more flexible. However, compared with explicit de-bias methods, totally ignoring prior knowledge limits their capability upper-bound for some tasks. If multiple types of biases are characterized, they cannot fully leverage all the valuable information. In contrast, GGD makes use of task-specific knowledge so that it can mitigate compositional biases. For tasks without prior knowledge of the biases, it can also learn a more robust model with a self-ensemble biased model like implicit de-biasing methods, gaining more flexibility in real world applications.

III PROPOSED METHOD

A Preliminaries

In this section, we first introduce the notations used in the rest of this paper. \((X, Y) \in \mathcal{X} \times \mathcal{Y}\) denotes the training set, where \(\mathcal{X}\) is the feature space of observations, and \(\mathcal{Y}\) is the label space. Assume \(\mathcal{B} = \{B_1, B_2, \ldots, B_M\}\) to be a set of task-specific bias features that can be extracted in priority, such as texture features in Biased-MNIST and the language shortcut in VQA. Correspondingly, \(h_m(B_m; \phi_m) : B_m \to \mathcal{Y}\) is a biased model that makes prediction with certain biased feature \(B_m\), where \(\phi_m\) is the parameter set of \(h_m(\cdot)\) that maps \(B_m\) to the label space \(\mathcal{Y}\). Similarly, \(f(X; \theta) : \mathcal{X} \to \mathcal{Y}\) denotes the base model, i.e., our target model for inference. For supervised learning, the training objective is to minimize the distance between the predictions and the labels \(Y\) as

\[
\min_{\theta} \mathcal{L}(f(X; \theta), Y),
\]

where the loss function can be various types of supervision loss, such as cross-entropy (CE) loss for single-label classification, binary cross-entropy (BCE) loss for multi-label classification, triplet loss for retrieval, etc. Similar to previous works [12], [21], [25], [26], considering that the classification (and its variants) is the most common task that seriously suffers from the dataset bias problem, we also take classification tasks as a demonstration in this paper.

B Greedy Gradient Ensemble

Given (1), \(f(.)\) is chosen to be an over-parametrized DNN, so the model is easy to over-fit the biases in the datasets and suffers from poor generalization ability. We take advantage of the easy-to-overfit property of deep models, and jointly fit the ensemble of bias models \(\sum_{m=1}^{M} h_m(B_m; \phi_m)\) and base model \(f(X; \theta)\) to label \(Y\)

\[
\min_{\phi, \theta} \mathcal{L}\left(f(X; \theta) + \sum_{m=1}^{M} h_m(B_m; \phi_m), Y\right).
\]

Ideally, we hope the spurious correlations are only over-fitted by the bias models, thus the base model \(f(.)\) can be learned with a relatively unbiased data distribution. To achieve this goal, GGE adopts a greedy strategy that encourages biased models to have a higher priority to fit the dataset. In practice, \(f(.)\) can be ResNet for image classification, UpDn [61] for VQA, etc., while \(h(.)\) can be low capability model for the texture bias, question-answer classifier for the question shortcut bias, etc.

Viewing from a general ensemble model in the functional space [62], suppose we have \(\mathcal{H}_m = \sum_{m=1}^{M} h_m(B_m)\) and we wish to find \(h_{m+1}(B_{m+1})\) added to \(\mathcal{H}_m\) so that the loss \(\mathcal{L}(\sigma(\mathcal{H}_m + h_{m+1}(B_{m+1})), Y)\) decreases. Theoretically, the desired direction of \(h_{m+1}\) should be the negative derivative of \(\mathcal{L}\) at \(\mathcal{H}_m\), i.e.,

\[
-\nabla \mathcal{L}(\mathcal{H}_{m,j}) := \frac{\partial \mathcal{L}(\mathcal{H}_m, Y)}{\partial \mathcal{H}_{m,j}}, j \in 1, 2, \ldots, C.
\]

where \(\mathcal{H}_{m,j}\) denotes the prediction for the \(j\)-th class among the overall \(C\) classes. For a classification task, we only care about the probability for class \(j\): \(\sigma(f_j(x)) \in (0, 1)\). Therefore, we treat the negative gradients as pseudo labels for classification and optimize the new model \(h_{m+1}(B_{m+1})\) with

\[
\mathcal{L}(h_{m+1}(B_{m+1}; \phi_{m+1}), -\nabla \mathcal{L}(\mathcal{H}_m)).
\]

After integrating all biased models, the expected base model \(f\) is optimized with

\[
\mathcal{L}(f(X; \theta), -\nabla \mathcal{L}(\mathcal{H}_M)).
\]

In the test stage, we only use the base model for prediction.

In order to make the above paradigm adaptive to mini-Batch Gradient Decent (MBGD), we implement an iterative optimization scheme [29] as shown in Algorithm 1. Note that our framework learns the base model and biased models jointly, which is different from existing work [32], [36] where the biased model is learned via another independent process or additional annotations.

C General Greedy De-Bias Learning

As shown in [29], GGD_{gs} (GGE) often over-estimates the biases of datasets. It achieves remarkable improvement on out-of-distribution data but may significantly degrades under the in-distribution setting. To overcome this critical issue, we first
Algorithm 1: GGDgs.

Input: Observations X, Labels Y,
Biased feature Observations B = \{B_m\}_{m=1}^M
Base function f(\theta) : X \to \mathbb{R}^{|Y|},
Bias functions \{h_m(\phi_m) : B_m \to \mathbb{R}^{|Y|}\}_{m=1}^M
Initialize: \mathcal{H}_0 = 0 ;
for Batch t = 1 \ldots T do
   for m = 1 \ldots M do
      \begin{align*}
         L_m(\phi_m) &\leftarrow \mathcal{L}'(h_m(B_m; \phi_m), -\nabla \mathcal{L}(H_{m-1}, Y)) \\
         \text{Update } & \phi_m \leftarrow \phi_m - \alpha \nabla \phi_m L_m(\phi_m)
      \end{align*}
   end
   \begin{align*}
      L_{M-1}(\theta) &\leftarrow \mathcal{L}'(f(X; \theta), -\nabla \mathcal{L}(H_M, Y)) \\
      \text{Update } & \theta \leftarrow \theta - \alpha \nabla \theta L_{M+1}(\theta)
   \end{align*}
end
return f(X; \theta)

Algorithm 2: GGDcr.

Input: Observations X, Labels Y,
Biased feature Observations B = \{B_m\}_{m=1}^M
Base function f(\theta) : X \to \mathbb{R}^{|Y|},
Bias functions \{h_m(\phi_m) : B_m \to \mathbb{R}^{|Y|}\}_{m=1}^M
Initialize: \mathcal{H}_0 = 0 ;
for Batch t = 1 \ldots T do
   \begin{align*}
      \lambda_t &\leftarrow \sin\left(\frac{\pi t}{T + 1}\right) \\
      \text{for } m = 1 \ldots M do \\
      \begin{align*}
         L_m(\phi_m) &\leftarrow \mathcal{L}'(h_m(B_m; \phi_m), -\nabla \mathcal{L}(H_{m-1}, Y)) \\
         \text{Update } & \phi_m \leftarrow \phi_m - \alpha \nabla \phi_m L_m(\phi_m)
      \end{align*}
   end
   \begin{align*}
      \hat{\sigma}(H_M) &\leftarrow Y \odot \sigma(H_M) \\
      \alpha &\leftarrow \mathcal{L}(f(X; \theta), \hat{\sigma}(H_M)) \\
      \text{Update } & \theta \leftarrow \theta - \alpha \nabla \theta L_{M+1}(\theta)
   \end{align*}
end
return f(X; \theta)

re-analyse the biased model in GGE under CE loss
\[ \mathcal{L}_{CE}(Z, Y) = -\sum_{j=1}^{C} y_j \log(\sigma_j), \] (6)
with
\[ \sigma_j = \frac{e^{z_j}}{\sum_{k=1}^{C} e^{z_k}}, \] (7)
where \( Z = \{z_j\}_{j=1}^{C} \) is the predicted logits, and \( y_j \in \{0, 1\} \) is the ground-truth label for the \( j \)-th class. \( \sigma_j \) indicates the confidence of the biased model on \( j \)-th class. The negative gradient of the loss function is
\[ -\nabla \mathcal{L}(z_j) = y_j - \sigma_j, \] (8)
To make the range of pseudo labels consistent with the classification label space [0, 1], \(-\nabla \hat{\mathcal{L}}(z_j)\) is clipped to
\[ -\nabla \hat{\mathcal{L}}(z_j) = \begin{cases} y_j - \sigma_j & y_j > 0 \\ 0 & y_j = 0 \end{cases}. \] (9)
The negative gradients access whether a sample can be solved based on the spurious correlation captured by certain biased model.

Now, casting aside the viewpoint of gradient descent in functional space, we can also decompose the CE loss with \(-\nabla \hat{\mathcal{L}}\) as pseudo label
\[ \mathcal{L}_{CE}(f(X), -\nabla \hat{\mathcal{L}}) = -\sum_{j=1}^{C} (y_j - \hat{\sigma}_j) \log(p_j) \]
\[ = -\sum_{j=1}^{C} y_j \log(p_j) + \sum_{j=1}^{C} \hat{\sigma}_j \log(p_j) \]
\[ = \mathcal{L}_{CE}(f(X), Y) - \mathcal{L}_{CE}(f(X), \hat{\sigma}), \] (10)
where the reference prediction \( \hat{\sigma} = Y \odot \sigma \) and \( \odot \) is the element-wise product that equals to the clipping in (9).

Based on (10), the gradient ensemble actually aims to provide predictions that agree with the ground-truth but disagree with the biased models. \(-\mathcal{L}_{CE}(f(X), \hat{\sigma})\) controls the degree of spurious relation to be reduced. To this end, we can treat \(-\mathcal{L}_{CE}(f(X), \hat{\sigma})\) as a regularization:
\[ L(\theta) = \mathcal{L}(f(X; \theta), Y) - \lambda_4 \mathcal{L}_{CE}(f(X), \hat{\sigma}(H_M)), \] (11)
where \( \lambda_4 \) denotes the weight of the regularization term. This more general framework is noted as General Greedy De-bias (GGD), where we only keep greedy strategy but get free from the negative gradient supervision. GGE is a special case of GGD when \( \lambda_4 = 1 \). We will denote GGE as GGDgs (Gradient Supervision) in the following paper.

Furthermore, inspired by Curriculum Learning [33], \(-\mathcal{L}_{CE}(f(X), \hat{\sigma})\) can be regarded as a soft difficulty measurement for curriculum sample selection function. In practice, we formulate a Curriculum Regularization training scheme (GGDgs), which gradually increases \( \lambda_4 \) along with the training process. In this way, samples with spurious correlations to the labels can participate in the early stage of training. In the consequent training stage, the model will focus on the hard samples that cannot be solved by biased models, resulting in more stable prediction on out-of-distribution data. The overall optimization procedure GGDcr is shown in Algorithm 2. Comparison between GGDgs and general GGDcr is shown in Fig. 2.

D Discussions

1) Intuitive Explanation of GGD: Section III-B has presented theoretical evidence for GGDgs from the aspect of model learning in functional space. More intuitively, GGDgs can also be regarded as a re-sampling strategy [63]. For a sample that is easy to fit by biased models, \(-\nabla \hat{\mathcal{L}}(z_i)\) (i.e., the pseudo label produced by the base model) will become relatively small. This makes \( f(X; \theta) \) pay more attention to samples that are hard to fit by previous ensemble biased classifiers. As a result, the base model is not likely to learn biased features. This hard example mining process is experimentally demonstrated in Section IV-A3 and Fig. 9.

However, according to (9), samples that demonstrate high spurious correlations (i.e., \(-\nabla \hat{\mathcal{L}}(z_i) = 0\)) will be discarded. If
large groups of data are absent because of zero supervision, the representation learning of the base model with the gradient supervision may be under-fitted. Moreover, when the label distribution is skewed (e.g., distribution bias in VQA-CP [11]), the base model may over-estimate the bias in labels. This results in “inverse” training bias and significant degradation on in-distribution test data. Experiments on long-tailed classification also revealed similar findings [64, 65], which indicate that re-sampling a part of the data encourages a more balanced classifier but harms the representation learning stage, while learning with unbalanced data results in a biased classifier but still provides a good representation.

To alleviate the “bias over-estimation,” GGD$_{cr}$ provides a good relaxation of GGD$_{gs}$ by replacing the gradient supervision with a “softer” Curriculum Regularization. By adjusting $\lambda_t$, all data can participate in the base model learning in the early stage, thus the bias over-estimation can be well alleviated. We will further experimentally demonstrate these findings in Sections IV-A3 and IV-D.

2) Probabilistic Justification: Following the assumptions in [17], for a given sample $x$, let $x^b$ be the biased features and $x^{-b}$ be the features except the biases. $x^b$ and $x^{-b}$ are conditionally independent given the label $y$. We have

$$\log p(y|x^{-b}) = \log p(y|x) - \log p(y|x^b) + C,$$  \hspace{1cm} (12)

where $C$ is a constant term related to the given datasets. The detailed derivation is provided in the Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2023.3240337. It is hard to distinguish the core features for the task of interest ($x^{-b}$) but it is easier to identify the dominant biases ($x^b$) based on the prior knowledge. (12) indicates that maximizing the likelihood $\log p(y|x^{-b})$ equals to maximizing $\log p(y|x)$ while minimizing $\log p(y|x^b)$.

Assume the optimal biased model $h(x^b; \phi^*)$ has

$$\phi^* = \arg \min_{\phi} \mathbb{E}_{x, y \sim p} \mathcal{L}(h(x^b; \phi), y).$$ \hspace{1cm} (13)

Taking $q_{\phi^*}(y|x^b)$ as the distribution of optimal biased prediction $h(x^b; \phi^*)$, GGD alternatively minimizes $\log p(y|x^b)$ by enlarging the divergence between $p(y|x)$ and the biased reference $q_{\phi^*}(y|x^b)$. Maximizing (12) is approximated as

$$\arg \max_{\theta} \left( \log p(y|x) + D(p(y|x)||q_{\phi^*}(y|x^b)) \right)$$ \hspace{1cm} (14)

where $\theta$ is the parameter of the base model that produce distribution $p(y|x)$ and $D(\ | \ |)$ is the divergence between two distributions. In practice, we get diverse predictions by maximizing the cross-entropy between $p(y|x)$ and $q(y|x^b)$. Similar implementation also appears in [66]. Eq(12) provides a new justification of GGD from probabilistic formulation, which aims to maximum the log-likelihood of $\log p(y|x^{-b})$. Moreover, the precision of $q(y|x^b)$ is crucial. If the biased model captures the true correspondence too much, maximizing the divergence will harm the base model.

3) On the Trade-Off Between ID and OOD Performance: The key idea of greedy ensemble is similar as the Boosting strategy [67, 68]. Boosting is to combine multiple weak classifiers with high bias but low variance to produce a strong classifier with low bias and low variance. Each base learner has to be weak enough, otherwise, the first few classifiers will easily over-fit to the training data [69]. Different from boosting that ensembles all weak learners, we make use of this over-fitting phenomenon but only use the last base model for prediction. This strategy removes specific spurious correlations with the biased models but also encounters the bias-overestimation with a single base model. To solve this problem, GGD introduce Curriculum Regularization, which trains the model with all ID data in the early training stage and then gradually focus on the hard samples.

The trade-off between ID and OOD performance has already attracted much attention in the study of OOD generalization. Most of these methods assume that the OOD data is available during training [36, 66, 70, 71] or the model can be pre-trained on balanced data with few biases [37, 72]. Therefore, they can adaptively adjust the model with the given OOD data. However, for de-bias learning, the absence of OOD data makes the bias estimation more ill-posed and challenging. The works in [31, 32] share similar idea with our GGD$_{cr}$ in that they aim to make full use of the biased ID data to pursue a good trade-off between ID and OOD performance. However, both [31] and [32] have to train a biased and a de-biased model separately and then combine the two to achieve more robust predictions.
In comparison, GGD learns the two models under the unified framework as in Algorithm 2. It does not require extra training cost of an original model, and can well adapt to any choice of base model, thus it gains more flexibility in real applications.

E General Applicability of GGD

This section provides the detailed instantiation of GGD on specific tasks. In the following part, let $h(.)$ denote the biased model and $\hat{y} \in \mathcal{Y}$ denote the biased predictions, where the super-script represents the bias type.

1) GGD With Single Explicit Bias: In order to compare with existing explicit de-bias methods that focus on one single type of bias, we first test GGD on the texture bias in Biased-MNIST [12].

The dataset $D = \{x_i, y_i, b_i\}_{i=1}^N$ consists of a synthetic image $x_i$, the annotated digit label $y_i$, and the background color $b_i$. We aim to predict the digit number $\hat{y}_i$ with the input image $x_i$

$$\hat{y}_i = f(x_i),$$

where the base model $f(.)$ is a neural network trained with CE loss.

The bias for Biased-MNIST comes from the spurious correlation between the digits and the background colors. In practice, we define two different kinds of bias models. In the first case, the biased prediction $B_1$ of an image sample $x_i$ is extracted with a low capacity model

$$\hat{y}_i^1 = h_{1k}(x_i).$$

$h_{1k}(.)$ is the SimpleNet-1k [12] with kernel size $1 \times 1$. It will predict the target class of an image only through the local texture cues due to small receptive fields.

In the second case, we provide the explicit background $b_i$ for bias extraction

$$\hat{y}_i^1 = h_{bg}(b_i).$$

$h_{bg}(.)$ is a common neural network similar to the base model but the input is only a background image without digits. Therefore, the biased model will purely make predictions according to the texture bias. The experimental analysis is provided in Section IV-A.

2) GGD With Self-Ensemble: For tasks like Adversarial QA [38], the task-specific biases are hard to distinguish. For de-bias learning at the lack of prior knowledge, we design a more flexible version of GGD with Self-Ensemble, named GGD$^{\text{se}}$.

The biased predictions $B_{se}$ is captured with

$$\hat{y}_i^{se} = h_{se}(x_i),$$

where $h_{se}(.)$ is another neural network that has the same architecture and optimization scheme as the baseline model.

Since the baseline model usually tends to over-fit the dataset biases, $h_{se}(.)$ can implicitly capture the biases without task-specific prior knowledge.

In the experiments, we will demonstrate the hard-example-mining mechanism of GGD$^{\text{se}}$ on Adversarial SQuAD [38] in Section IV-B and further verify its generalization ability on all the other three tasks.

3) GGD With Multiple Biases: To verify whether GGD can handle multiple types of biases, we conduct experiment on the Language bias in VQA. As analysed in [29], the language bias is mainly composed of two aspects, i.e., distribution bias and shortcut bias.

We consider the formulation of VQA task as a classification problem. Given a dataset $D = \{v_i, q_i, a_i\}_{i=1}^N$ consisting of an image $v_i \in \mathcal{V}$, a question $q_i \in \mathcal{Q}$ and a labeled answer $a_i \in \mathcal{A}$, we need to optimize a mapping $f_{VQ} : \mathcal{V} \times \mathcal{Q} \to \mathbb{R}^n$ which produces a distribution over the $C$ answer candidates. The function is as follows

$$\hat{a}_i = f_{\theta}(v_i, q_i) = c(m(e_v(v_i), e_q(q_i))),$$

where $c : \mathcal{V} \to \mathbb{R}^{n_v \times d_v}$ is an image encoder, $e_q : \mathcal{Q} \to \mathbb{R}^{n_q \times d_q}$ is a question encoder, $m(.)$ stands for the multi-modal fusion module, and $c(.)$ is the multi-layer perception classifier. The output vector $\hat{a} \in \mathbb{R}^C$ indicates the probability distribution on all the answer candidates.

The distribution bias is the statistical answer distribution under certain question types

$$\hat{y}_i^d = p(a_i | t_i),$$

where $t_i$ denotes the type of question $q_i$, such as “what color,” “is this,” etc., in VQA v2 [43].

The shortcut bias is the semantic correlation between specific QA pairs, which can be modeled as a question-only branch similar to [18]

$$\hat{y}_i^q = c_q(c_q(q_i)),$$

where $c_q : \mathcal{Q} \to \mathbb{R}^C$.

To verify whether GGD can handle compositional biases, we design different versions of GGD which ensemble distribution bias, shortcut bias and both biases. The experimental results are shown in Section IV-C.

IV EXPERIMENTS

In this section, we present experiments for GGD on both ID and OOD settings. With respect to different types of biases, experiments on image classification [12], QA [38] and VQA [43] are shown afterwards, corresponding to CV, NLP and Vision-Language tasks. Note that GGD is a general de-bias framework, and it will be an interesting issue in further study to apply our methods on other tasks that also suffer from dataset biases.

A Image Classification

1) Dataset: Biased MNIST: To better analyse the the properties of GGD, we first verify our model on Biased MNIST, where we can have full control over the amount of bias during training and evaluation. Biased MNIST [12] is modified from MNIST [73] which introduces the color bias that highly correlates with the label $Y$ during training.

On Biased-MNIST, 10 different colors are selected for each digit $y \in \{0, \ldots, 9\}$. For each image of digits $y$, we assign a pre-defined color with probability $\rho$ and any other color with probability $1 - \rho$. $\rho \in [0, 1]$ controls the level of spurious correlation in train and test set. $\rho = 0.99$ means 99% images in
TABLE I

| $\rho_{\text{train}} / \rho_{\text{test}}$ | 0.990 | 0.995 | 0.999 | Original |
|----------------------------------------|-------|-------|-------|----------|
| Baseline                               | 77.80±1.30 | 80.11±0.01 | 99.76±0.06 | 53.78±2.28 | 58.03±2.72 | 99.82±0.08 | 10.44±2.34 | 17.39±2.71 | 99.80±0.09 | 98.78±0.15 |
| ReBias [12]                            | 83.21±0.099 | 87.80±0.042 | 99.77±0.016 | 73.67±5.18 | 75.95±5.18 | 99.70±2.21 | 32.84±0.02 | 37.52±0.027 | 99.87±0.001 | 99.05±0.008 |
| RUBi [18]                              | 87.36±0.039 | 91.30±0.108 | 99.15±0.029 | 77.84±5.33 | 82.22±4.14 | 99.21±0.07 | 30.25±0.64 | 37.20±5.79 | 92.15±0.71 | 98.90±0.04 |
| GGD [12]                               | 92.79±0.047 | 94.22±0.078 | 98.69±0.054 | 91.27±0.035 | 91.80±0.039 | 98.16±1.06 | 67.85±4.31 | 70.77±3.84 | 86.84±4.33 | 98.64±0.04 |
| GGD [18]                                | 91.78±0.118 | 92.30±0.117 | 99.64±0.030 | 83.90±2.30 | 84.91±2.64 | 99.28±0.15 | 68.36±1.99 | 70.70±2.02 | 99.25±0.35 | 99.14±0.05 |
| ReBias1kg [12]                         | 84.95±1.43 | 86.88±1.94 | 99.66±0.33 | 74.27±5.30 | 76.20±1.78 | 99.74±0.13 | 27.74±0.07 | 34.20±4.47 | 99.87±0.01 | 99.87±0.17 |
| RUBi bg [18]                           | 88.65±0.047 | 89.67±0.044 | 99.59±0.035 | 78.19±5.54 | 80.50±4.14 | 98.79±1.07 | 21.07±0.79 | 27.59±4.39 | 90.16±0.06 | 98.70±0.04 |
| GGD bg [12]                            | 93.78±1.34 | 94.46±0.19 | 99.01±0.43 | 90.34±0.05 | 91.26±1.46 | 99.38±0.09 | 61.81±0.29 | 66.00±0.77 | 91.25±0.18 | 98.77±0.07 |
| GGD bg [18]                            | 90.64±0.046 | 91.95±0.091 | 99.82±0.005 | 86.20±1.41 | 87.02±1.04 | 99.68±0.10 | 62.96±0.74 | 67.62±4.31 | 99.41±0.41 | 99.07±0.13 |
| ReBias6kg [12]                         | 83.77±0.941 | 85.76±0.771 | 99.77±0.080 | 75.06±2.35 | 77.25±2.41 | 99.84±0.04 | 31.82±2.59 | 38.41±0.26 | 99.87±0.02 | 99.03±0.06 |
| RUBi 6bg [18]                          | 27.37±0.042 | 33.47±0.042 | 89.14±0.002 | 16.23±5.50 | 22.96±5.44 | 95.77±2.18 | 10.21±2.25 | 16.67±3.79 | 83.67±2.01 | - |
| GGD bg [12]                            | 79.35±1.53 | 80.78±2.11 | 94.65±0.41 | 69.70±3.32 | 72.49±3.33 | 91.60±1.49 | 38.72±1.00 | 42.74±3.25 | 76.24±3.74 | 93.88±0.87 |
| GGD bg [18]                            | 83.28±0.465 | 85.53±0.132 | 99.27±0.377 | 72.91±2.49 | 76.19±1.41 | 99.34±0.37 | 43.78±2.83 | 48.92±1.44 | 99.46±0.31 | 98.94±0.05 |

1k uses simple net-1k as the biased model, bg adopts the backgrounds as biased feature, and se stands for the self-ensemble version. ‘original’ is the original mnist without texture bias.

The dataset are assigned with a background of the corresponding color. $\rho = 0.1$ is the unbiased condition with a uniform sampled background color.

2) Experimental Setups: For image classification on biased MNIST, we use ResNet-18 as our baseline model. Model 1$k$ in Table I denotes SimpleNet-1k with kernel size $1 \times 1$ proposed in [12]. Biased model bg uses the ResNet-18 with background color images as the input. All experiments use the same CE loss and baseline model. “Original” stands for MNIST dataset without biases. Considering the randomness in Biased-MNIST data generation, we report the mean and variance of 4 repeated experiments under different random seeds.

3) Experimental Results: GGD Overcomes Bias: As shown in Table I, GGD largely improves the OOD Accuracy on Biased MNIST. Under extremely biased training data ($\rho_{\text{train}} = 0.999$), the best performed method GGD $\text{bg}^{1k}$ achieves 68% accuracy on the unbiased test data ($\rho_{\text{test}} = 0$), which is over 6 times compared with the baseline model. In Fig. 3, we provide per-class accuracy matrix for more detailed analysis. The diagnostic heat-map corresponds to unbiased and biased correlation respectively. We can observe that the vanilla ResNet-18 mainly captures the spurious correlation but confuses on the ground-truth digits. ReBias [12] can better capture the core correlations compared with the baseline but will be still fooled by the texture bias. GGD hardly relies on the background color as shown in Fig. 3(c) and
(d), which demonstrate that our method can help a model to overcome certain kinds of bias via specifically designed biased model.

**Performance on Different Bias Level:** As shown in Table I, GGD achieves prominent performance gain on out-of-distribution tests across all bias levels while remaining stable on in-distribution data. Comparing with other methods that use the ensemble strategy, GGD surpasses RUBi [18] and ReBias [12] by a large margin under the same base model and biased models, especially when the training and testing data are extremely different ($\rho_{\text{train}} = 0.999$). When training and testing under the unbiased situation, both GGD and GGD$_{gs}$ are stable if the bias type is known in advance. RUBi fails on unbiased training set, which even continuously decreases under the original MNIST (‘-’ in Table I).

**Ablations on Biased Models:** Besides biased model with small receptive field, we test another version of the biased model with a ground-truth background image as the biased features, shown as $b_g$ in Table I. GGD works well with different biased models comparing with other methods.

For implicit de-biasing, a biased model with the same structure as the baseline ResNet-18 is trained in the Self-Ensemble version GGD$^{se}$. As shown in Table I, GGD$^{se}$ achieves the best performance when the bias information is not available. RUBi fails under the self-ensemble setting. This demonstrates that GGD$^{se}$ can also implicitly remove biases even without the task-specific biased models, which is much more flexible compared with existing explicit de-bias methods.

**GGD$_{gs}$ Versus GGD$_{cr}$:** As shown in Table I, although achieving high Accuracy on the out-of-distribution test data, GGD$_{gs}$ is not as robust as GGD$_{cr}$ with the increase of texture bias. Especially on the in-distribution test data, the accuracy is significantly lower than the baseline ResNet-18. Moreover, GGD$_{gs}$ is also very unstable in the later training stage, resulting in large variance according to different training data. On the other hand, GGD$_{cr}$ achieves comparable in-distribution performance against the baseline even on the original MNIST dataset.

For better analysis of the GGD$_{gs}$ and GGD$_{cr}$, we design another ablation study on $\lambda_t$ in (11) under $\rho_{\text{train}} = 0.999$. As shown in Table II, by slightly relaxing the regularization (changing $\lambda_t$ from 1.0 to 0.95), the in-distribution accuracy will be increased to the level of vanilla ResNet-18. This verifies our assumption that such degradation mainly comes from the completely absence of samples with the spurious correlation (Section III-C). Starting from this insight, we define $\lambda_t = \sin\left(\frac{\pi t}{T}\right)$ in GGD$_{cr}$, where $t$ is the current training epoch and $T$ is the number of total epochs.

With this Curriculum Regularization, GGD$_{cr}$ achieves a good trade-off between in-distribution and out-of-distribution tests, remaining comparable in-distribution test accuracy against the baseline and OOD test against GGD$_{gs}$.

### B Adversarial Question Answering

1) **Dataset:** For the NLP tasks, we choose the adversarial question answering (AdQA) to demonstrate the effectiveness of GGD. We evaluate on the Adversarial SQuAD [38] dataset, which was built by adding distractive sentences to the passages in SQuAD [6]. These sentences are designed to be very similar to the corresponding questions but with a few key semantic changes to ensure that they do not indicate the correct answer. Models that only focus on the similarity between question and context will tend to be misled by the new sentence. A sample from Adversarial SQuAD is shown in Fig. 4.

2) **Experimental Setup:** We use BiDAF [80] as the base model. It introduces a multi-stage hierarchical process that represents the context at different levels of granularity and uses a bidirectional attention flow mechanism to obtain a query-aware context representation without early summarization.

Since the word-level and semantic-level similarity is hard to distinguish by modelling, we only test GGD$^{se}$ in the following experiments.

The models are trained and validated on the original SQuAD train and val set, and test on the AddSent split of Adversarial SQuAD. In order to further verify the hard example mining mechanism behind GGD, we also design another “Extra” setting in which we add the AddSentOne split of Adversarial SQuAD to the training set as additional hard samples. The performances are measured with F1 score, which are the weighted average of the precision and recall rate at the character level.

3) **Experimental Results:** The F1 scores for the OOD test on Adversarial SQuAD and ID test on SQuAD v1 are shown in Table III. The reported results are from four repeated experiments with different random seeds. We find that both GGD$_{gs}$ and GGD$_{cr}$, trained with the SQuAD train set, only improve the performance by $\sim 2\%$. However, after adding a few hard examples from AddSentOne, both methods achieve much more

| Table II: Ablations for $\lambda_t$ | $\lambda_t$ | Train 0.999 |
|-----------------------------------|------------|-------------|
| 0 (Baseline)                      | 8.95       | 16.24       | 99.87       |
| 1 (GGD$_{gs}$)                    | 68.31      | 71.34       | 91.42       |
| 0.95                              | 58.79      | 63.41       | 99.96       |
| Anneal (GGD$_{cr}$)              | 67.01      | 70.17       | 99.38       |

“Anneal” indicates the curriculum regularization that changes $\lambda_t$ from 0 to 1 along with training process.

- **Question:** What is the name of the quarterback who was 38 in Super Bowl XXXIII?
- **Context:** Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Jeff Dean is the name of the quarterback who was 37 in the Super Bowl.
- **GT Answer:** John Elway
- **Misleading Answer:** Jeff Dean

![Fig. 4. An example from Adversarial SQuAD. Blue sentence is the expected evidence, while red sentence is the distracting sentence to fool the models.](image-url)
improvement on Adversarial SQuAD compared with the baseline. GGD\(_{cr}\) gets nearly 5 points gain over the BiDAF baseline that is already strong enough. This well demonstrates the power of the greedy learning on focusing on the hard/valuable samples from a biased dataset.

However, the limited improvement in the original setting is likely caused by Self-Ensemble, where the baseline model BiDAF itself can hardly capture useful biases from the dataset. If we can define a better biased model that can access the word-level similarity, we may achieve better performance without extra training data.

### C Visual Question Answering

1) Dataset: Data bias problems in multi-modal tasks are more challenging, where multiple data sources from different modalities should be jointly considered. In this section, we choose Visual Question Answering (VQA) as the representative multi-modal task for demonstration. Neural networks [81], [82], [83], [84], [85] that model the correlations between vision and language have shown remarkable results on large-scale benchmark datasets [9], [43], [86], [87], but most VQA methods tend to rely on existing idiosyncratic biases in the datasets [9], [10] and show poor generalization ability to out-of-domain data. In this section, we demonstrate the effectiveness of GGD on the challenging datasets VQA-CP v2 [11] and GQA-OOD [39].

VQA v2 [43] is a commonly used VQA dataset composed of real-world images from MSCOCO with the same train/validation/test splits. For each image, an average of three questions are generated, and 10 answers are collected for each image-question pair from human annotators. Following previous works, we take the answers that appeared more than 9 times in the training set as candidate answers, which produces 3129 answer candidates.

VQA-CP v2 [11] dataset is derived from the VQA 2.0 [43] but contains different answer distribution per question type between training and validation splits. Since it has different distribution on the train and test sets, the performance on this dataset better reflects models’ generalization ability. VQA-CP v2 consists of 438,183 samples in the train set and 219,928 samples in the test set.

GQA-OOD [39] divides the test set of GQA [87] into majority (head) and minority (tail) groups based on the answer frequency within each ‘local group’, which is a unique combination of answer type (e.g., colors) and the main concept (e.g., ‘bag’, ‘chair’, etc.). The models are trained on the original GQA-balanced but tested on different fine-grained local groups.

2) Experimental Setups: In the following experiments, we use UpDn [61] as our base model and the images are represented as object features pre-extracted with Faster R-CNN [88].  

The implementation details and the experiments on other base models are provided in the Appendix, available in the online supplemental material. All methods are measured with Accuracy and Correct Grounding Difference (CGD) proposed in [29]. CGD evaluates whether the visual information is well taken in answer decision.

For ablation studies, we present five different versions of GGD. GGD\(_d\) only removes the distribution bias. GGD\(_q\) only models the shortcut bias. GGD\(_{dq}\) makes use of both the distribution bias and the shortcut bias. GGD\(_{se}\) is the self-ensemble version GGD, which takes the baseline model itself as the biased model. GGD\(_{dqse}\) removes the distribution bias before Self-Ensemble. The implementation details of above five ablations are provided in the Appendix, available in the online supplemental material.

3) Experimental Results: GGD can handle multiple biases: In the first group of ablation study, we compare with the other two ensemble strategies to verify the effectiveness of the greedy learning. SUM-DQ directly sums up the outputs of biased models and the base model. LMH+RUBi combines LMH [17] and RUBi [18]. It reduces distribution bias with LMH and shortcut bias with RUBi. The implementation details for these two ablations are provided in the Appendix, available in the online supplemental material.

As shown in Table V, SUM-DQ performs even worse than vanilla UpDn. LMH+RUBi does not make use of both kinds of biases, whose Accuracy is just similar to that of LMH. On the other hand, both GGD\(_{se}\) and GGD\(_{cr}\) surpass these two ablations by a large margin. This shows that the greedy strategy in GGD can really force the biased data to be learned with biased models in priority. As a result, the base model has to pay more attention to hard examples that are hard to solve based on the estimation of either distribution bias or shortcut bias. It needs to consider more visual information for the final decision.

In the second group of experiments, we directly compare GGD\(_d\), GGD\(_q\), and GGD\(_{dq}\). As shown in Table V, GGD\(_{dqse}\) surpasses single-bias versions GGD\(_d\) and GGD\(_q\) by \(\sim 10\%\). This well verifies that GGD can reduce multiple biases with the greedy learning procedure. The case analysis in Fig. 5 provides a more qualitative evaluation. It shows that GGD\(_q\) improves Y/N as shown in Table V. \(B_d\) works like ‘hard example mining’ but will also introduce some noise (e.g., “mirror” and “no” in this example) due to the unbalanced data distribution. GGD\(_{dq}\) can make use of both biases. Reducing \(B_d\) at first can further help the discovery of the hard examples with \(B_q\) and encourage the base model to capture essential visual information.

Implicitly De-Bias With Self-Ensemble: In order to further discuss the generalizability of GGD, we also test a more
TABLE IV

| Method       | Base | VQA-CP test | VQA v2 val |
|--------------|------|-------------|------------|
|              |      | All | Y/N | Num. | Others | CGD  | All | Y/N | Num. | Others |
| GVQA [11]    | -    | 31.30 | 57.99 | 13.68 | 22.14 | -    | 48.24 | 72.03 | 31.17 | 34.65 |
| UpDn [62]    | -    | 39.81 ± 0.65 | 43.09 ± 0.51 | 12.01 ± 0.99 | 45.81 ± 0.91 | 2.57 ± 1.20 | 63.57 ± 0.32 | 80.90 ± 0.50 | 42.58 ± 0.37 | 57.75 ± 0.67 |
| S-MRL [18]   | -    | 38.46 | 42.85 | 12.81 | 43.20 | -    | 63.10 | -    | -    | -    |
| HINT* [75]   | UpDn | 47.50 | 67.21 | 10.67 | 46.80 | -    | 63.38 | 81.18 | 42.14 | 55.66 |
| SCR* [76]    | UpDn | 49.45 | 72.36 | 10.93 | 48.02 | -    | 62.2 | 78.8 | 41.6 | 54.4 |
| AdvReg [46]  | UpDn | 41.17 | 65.49 | 15.48 | 35.48 | -    | 62.75 | 79.84 | 42.35 | 55.16 |
| RUBi [18]    | UpDn | 46.68 ± 0.89 | 68.43 ± 1.38 | 11.64 ± 0.24 | 44.53 ± 0.17 | 7.88 ± 1.41 | 58.74 ± 0.94 | 67.17 ± 0.42 | 39.85 ± 1.40 | 54.12 ± 0.64 |
| LM [17]      | UpDn | 49.13 ± 0.07 | 72.38 ± 0.32 | 14.49 ± 0.71 | 46.42 ± 0.94 | 9.17 ± 2.10 | 63.46 ± 0.37 | 81.15 ± 0.04 | 42.27 ± 0.21 | 55.61 ± 0.34 |
| LMH [17]     | UpDn | 53.30 ± 0.70 | 73.47 ± 0.38 | 31.90 ± 0.47 | 47.92 ± 0.10 | 10.54 ± 0.57 | 58.06 ± 1.18 | 72.60 ± 0.73 | 37.33 ± 1.75 | 52.39 ± 1.59 |
| DLP [77]     | UpDn | 48.87 | 70.99 | 18.72 | 45.57 | -    | 57.96 | 76.82 | 39.33 | 48.54 |
| GVQE* [78]   | UpDn | 48.75 | -    | -    | -    | -    | 61.04 | -    | -    | -    |
| CSS* [22]    | UpDn | 40.32 ± 0.69 | 42.15 ± 1.26 | 12.44 ± 1.27 | 46.76 ± 0.33 | 8.58 ± 2.34 | 62.34 ± 1.46 | 79.50 ± 0.20 | 42.11 ± 1.06 | 55.50 ± 1.32 |
| CF-VQA(Sum) [79] | UpDn | 53.69 | 91.25 | 12.80 | 45.23 | -    | 63.65 | 82.63 | 44.01 | 54.38 |

Best and second performance are highlighted in each column. Methods with * use extra annotations (e.g., human attention (HAT)), explanations (VQA-X), or object label information). Methods with † use extra datasets at their pre-training stage, methods with cg are our reimplementation using released codes. Other results are reported in the original papers. The results are from 4 repeated experiments under different random seeds.

TABLE V

| Method         | All | Y/N | Others | Num. | CGD | ID |
|----------------|-----|-----|--------|------|-----|----|
| Baseline       | 39.89 | 43.01 | 45.80 | 11.88 | 3.91 | 63.79 |
| SUM-DQ         | 33.46 | 42.38 | 66.08 | 12.98 | 3.10 | 56.85 |
| LMH+RUBi       | 51.56 | 74.55 | 47.41 | 22.65 | 6.12 | 60.68 |
| GGD\textsubscript{d}       | 48.27 | 70.75 | 47.53 | 13.42 | 14.31 | 62.79 |
| GGD\textsubscript{q}       | 43.72 | 48.17 | 48.78 | 14.24 | 6.70 | 61.23 |
| GGD\textsubscript{dq}      | 57.12 | 87.35 | 49.77 | 26.16 | 16.44 | 59.30 |
| GGD\textsubscript{dcp}      | 50.93 | 78.50 | 47.30 | 12.92 | 10.28 | 62.17 |
| GGD\textsubscript{qc}      | 55.81 | 88.59 | 48.74 | 20.96 | 13.46 | 62.48 |
| GGD\textsubscript{qcp}      | 59.57 | 88.44 | 50.23 | 36.95 | 13.92 | 63.11 |
| GGD\textsubscript{dsc}      | 44.53 | 50.98 | 48.90 | 18.24 | 6.08 | 59.30 |
| GGD\textsubscript{qsc}      | 56.33 | 86.43 | 49.24 | 23.47 | 14.47 | 61.03 |
| GGD\textsubscript{qsc}      | 54.42 | 80.26 | 48.64 | 29.42 | 6.70 | 61.09 |
| GGD\textsubscript{qsc}      | 57.08 | 85.75 | 48.88 | 36.54 | 11.62 | 62.27 |

Best and second performance are highlighted in each column. Methods with * use extra annotations (e.g., human attention (HAT)), explanations (VQA-X), or object label information). Methods with † use extra datasets at their pre-training stage, methods with cg are our reimplementation using released codes. Other results are reported in the original papers. The results are from 4 repeated experiments under different random seeds.

Flexible Self-Ensemble fashion GGD\textsubscript{se} on VQA-CP. As shown in Table V, GGD\textsubscript{se} still surpasses UpDn without predefined biased features. Moreover, if we first remove distribution bias before Self-Ensemble, the performance of GGD\textsubscript{dse} is comparable to existing state-of-the-art methods as well.

GGD\textsubscript{d} Versus GGD\textsubscript{dse}: As shown in Table V, GGD\textsubscript{d} largely alleviates the degradation on in-distribution test data VQA v2 val, which is even comparable to the original UpDn baseline. Moreover, it also gets better performance on VQA-CP under all GGD\textsubscript{d}, GGD\textsubscript{q}, GGD\textsubscript{dq}, and GGD\textsubscript{se}. The major improvement comes from the “Num.” and “Other” question types which contain fewer samples in the training set. GGD\textsubscript{d} harms the
performance on these question types because of the greedily discarding of samples that is easy to answer.

**Comparison With State-of-the-Art Methods:** We compare our best performed model GGD\(_{cr}\) with existing state-of-the-art bias reduction techniques, including visual-grounding-based methods HINT [74], SCR [75], ensemble-based methods AdvReg. [45], RUBi [18], LM (LMH) [17], MFE [50], new-question-encoding-based methods GVQE [77], DLP [78], counterfactual-based methods CF-VQA [78], CSS [22], recent proposed regularization method MFE [50], and SAR [79] that models VQA as Visual Entailment with pre-trained model.

As shown in Table IV, GGD\(_{cr}\) achieves state-of-the-art performance without extra bias annotation. It outperforms the baseline model UpDn by 20% higher in terms of Accuracy and 10% higher in terms of CGD, which verifies the effectiveness of GGD on both answer classification and visual-grounding ability.

For the comparison of question-type-wise results, incorporating GGD improves the performance for all the question types, especially the more challenging “other” question type [89]. CF-VQA [78] performs the best in Y/N, but worse than our methods in all the other question types and metrics. LMH [17], LMH-MFE [50], and LMH-CSS [22] work well on Num. questions. Comparing with LM and LMH, it is obvious that the performance gains in Num. are mainly due to the additional regularization for entropy. However, methods with entropy regularization drop nearly 10% on VQA v2. This indicates that these models may over-correct the bias. SAR [79] achieves the best performance on both VQA-CP v2 and VQA v2, using extra datasets for model pre-training. On the other hand, GGD\(_{cr}\) improves both Num. Accuracy and in-distribution performance only with the Curriculum Regularization and work well without any extra data sources.

**Qualitative Evaluation:** Examples in Fig. 6 illustrate how GGD\(_{cr}\) makes difference compared with the baseline UpDn [61]. The first example is about using visual information for inference. Despite offering the right answer “yes,” the prediction from UpDn is not based on the right visual grounding result. In comparison, GGD\(_{cr}\) correctly grounds the giraffe that is eating leaves. The second example is a case of reducing language prior apart from Yes/No questions. UpDn answers “boat” just based on the language context “in the water,” while GGD\(_{cr}\) provides correct answers “tv” and “television” with more salient visual grounding. These examples qualitatively verify our improvement on both Accuracy and visual explanation for the predictions.

Fig. 7 provides qualitative illustration for improvements achieved by GGD\(_{cr}\). Compared with GGD\(_{gs}\), GGD\(_{cr}\) majorly improves on the “Num.” and “Other” question types. Questions about facial expression and counting do not frequently appear in the dataset. If these questions can be correctly predicted by fitting the biases, the base model in GGD\(_{gs}\) may not have enough data to learn a good representation.

Fig. 8 shows the examples of failure cases from GGD\(_{cr}\). The model appears to be weak on the complicated counting problem. Some failure cases are due to missing annotation in the dataset (“decoration” can also be regarded as the right answers). Although making wrong predictions, answers for failure cases in the last row are still consistent with visual explanations rather than language bias, which further indicates that GGD\(_{cr}\) truly makes use of the visual information.

**4) Experimental Results on GQA-OOD: Biases:** GQA-OOD is a more challenging dataset for Visual Question Answering. It has biases from multiple sources including imbalanced answer distribution, visual concept co-occurrences, question word correlations, and question type/answer distribution. Since the training set for GQA-OOD is the manually balanced GQA-balanced-train split, it is hard to specify the explicit biases to ensure that the models can generalize to even the rarest local groups. Following [24], apart from question shortcut bias similar to that in VQA-CP [11], we define two kinds of distribution bias according to global group labels (115 groups) and local group labels (133328 groups), and the corresponding models are denoted as GGD\(_{y}\) and GGD\(_{y}^{i}\).

**Comparison With State-of-the-Art Methods:** We compare GGD\(_{gs}\) with implicit de-bias methods LfF [25] and SD [26]; explicit de-bias methods RUBi [18], Up Wt [5], and LNL [20].
As shown in Table VI, all three previous explicit methods fail on both global and local group distribution bias, performing even worse than the original baseline UpDn. RUBi [18] with the question-only branch also degrades on GQA-OOD compared with the baseline. Implicit methods LfF [25] and SD [26] are more stable on handling complicated biases in GQA. SD [26] achieves the highest accuracy on the Tailed group. This indicates that both distribution bias and shortcut bias in GQA-OOD are not as obvious as those in VQA-CP, since the data from GQA is synthetic and the train split has been manually balanced.

On the other hand, GGD works well with hard-example mining. It surpasses the baseline under all bias settings and is comparable to existing implicit methods. If the biased models can not make a prediction with high confidence, the pseudo labels for the base model remain almost unchanged for most of the data. Although the biased models cannot well model the biases in the dataset, GGD will not harm the performance like previous explicit de-bias methods.

Ablation Study: According to the ablation studies, GGD addressing distribution bias on global groups works better compared with GGD on the local groups. Sequentially reducing the distribution bias and the shortcut bias will improve the Tail group Accuracy but slightly degrades the Head group accuracy. GGD achieves the highest overall accuracy because it alleviates the over-estimated bias in GGD. However, since the biased models do not capture biases with high confidence, GGD does not show much difference compared with GGD under most of the settings.

D Discussion

Hard Example Mining Mechanism: To demonstrate the hard example mining mechanism of GGD, we provide analysis on the training process. We first evaluate whether the base model focuses on examples. We define the hard ratio \( R_h \) as

\[
R_h = \frac{L_{\text{hard}}}{L_{\text{all}}},
\]  

where \( L_{\text{hard}} \) is the loss of hard examples (i.e., samples that do not choose corresponding background color), \( L_{\text{all}} \) is the loss for all samples. We calculate the accumulated loss for every 118 iterations of 4 repeated experiments with different random seeds.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
TABLE VI

| Method | All | Head | Tail | Avg |
|--------|-----|------|------|-----|
| UpDn [62] | 46.93 ± 0.29 | 49.41 ± 0.24 | 42.73 ± 0.97 | 46.07 ± 0.42 |
| LfF [25] | 47.06 ± 0.23 | 49.53 ± 0.43 | 42.99 ± 0.47 | 46.26 ± 0.59 |
| SD [26] | 47.59 ± 0.33 | 50.05 ± 0.32 | 44.49 ± 1.15 | 47.27 ± 0.42 |
| RUB [18] | 45.51 ± 0.18 | 47.87 ± 0.02 | 41.67 ± 0.47 | 44.71 ± 0.33 |
| RUB [18] | 7.15 ± 0.03 | 6.86 ± 0.61 | 6.60 ± 1.12 | 6.573 ± 1.02 |
| RUB [18] | 20.48 ± 0.24 | 22.30 ± 0.04 | 17.51 ± 0.67 | 19.91 ± 0.79 |
| Up Wt [5] | - | - | - | 26.4 |
| Up Wt [5] | - | - | - | 26.2 |
| LNL [20] | - | - | - | 32.4 |
| LNL [20] | - | - | - | 10.7 |

$g$ denotes the method addressing global group distribution bias, $f$ is method addressing the local group distribution bias, and $q$ is the method addressing the question shortcut bias. “avg” is the mean accuracy of head and tail groups. methods with only avg are reported from [24].

Bias Over-Estimation:

An interesting phenomenon is that GGD$_{gs}$ degrades on VQA v2 [43], but is relatively stable on the in-distribution test of Biased MNIST [12]. We find that this is due to the distribution bias on VQA v2. As shown in Fig. 10, taking “is this” question type as an example, GGD$_{gs}$ will amplify the distribution bias and result in an “inverse biased distribution”. On the other hand, the pseudo labels for Biased MNIST are still balanced because the texture bias is independent of the label distribution. This can also partially explain the improvement of GGD$_{cr}$ on VQA, where the Curriculum Regularization also reduces such “bias over-estimation,” apart from better low-level representation learning ability. In practice, one can get a more balanced classifier by selecting better $\lambda_t$ according to the bias level of the dataset. It can also be a valuable research to adaptively estimate the bias level of a dataset in the future.

Limitations:

Although the bias over-estimation problem in our previous GGE model has been alleviated with the Curriculum Regularization, there is still two major shortcomings of GGD. First, the hard-example mining mechanism in GGD is an instance-level sample re-weighting. If all samples are following a certain spurious correlations, GGD will fail to discover it as a spurious correlation (e.g., $\rho_{\text{train}} = 1.0$ in Biased-MNIST). The gradients from the biased models will decline to 0. Even though the spurious feature is identified with the greedily learned biased models, the base model cannot learn a de-biased feature accordingly. Eq. 12 indicates that we can also directly optimize $\log p(y|x^b)$ with the biased feature $x^b$, which can be obtained from the optimal biased model $h(x^b; \phi^*)$. We will investigate how to select network activations according to the bias models towards feature-level ensemble in the future.

Second, if the biased model can well capture the biases in the dataset, GGD will largely improve both the in-distribution and the out-of-distribution performance. However, when the biased model can not perfectly disentangle the spurious correlations, the improvement from GGD is limited (see experiments on Adversarial SQuAD and GQA-OOD). Although the Self-Ensemble fashion GGD$_{se}$ can implicitly model the biases, it largely relies on the bias level of the dataset and the existence of hard examples in datasets. It can be a future work to design a more robust...
strategy that can capture spurious correlations needless of prior knowledge.

V CONCLUSION

In this paper, we propose General Greedy De-bias Learning, a general de-bias framework with flexible regularization and wide applicability. Accompanied with Curriculum Regularization, the relaxed GGDεr comes to a good trade-off between in-distribution and out-of-distribution performance. Experiments on image classification, Adversarial QA, and VQA demonstrate the effectiveness of GGD under both task-specific biased models and self-ensemble fashion without prior knowledge on both ID and OOD scenarios.

In theory, the core of our method is the greedy strategy, which sequentially learns biased models in priority. One can also replace the regularization with better metrics that are able to measure the distance between the predictions and the labels. It may further improve the performance on specific tasks. In the future, we will try de-bias learning at the feature level and design a better strategy to capture spurious correlations needless of dataset-specific knowledge.

REFERENCES

[1] Z. Qi, S. Wang, C. Su, L. Su, Q. Huang, and Q. Tian, “Self-regulated learning for egocentric video activity anticipation,” IEEE Trans. Pattern Anal. Mach. Intell., early access, Feb. 17, 2021, doi: 10.1109/TPAMI.2021.3069923.

[2] A. Torralba and A. Efros, “Unbiased look at dataset bias,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2011, pp. 1521–1528.

[3] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning requires rethinking generalization,” in Proc. Int. Conf. Learn. Representations, 2017.

[4] R. Geirhos et al., “Shortcut learning in deep neural networks,” Nature Mach. Intell., vol. 2, no. 11, pp. 665–673, 2020.

[5] S. Sagawa, A. Raghunathan, P. W. Koh, and P. Liang, “An investigation of why overparameterization exacerbates spurious correlations,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 8346–8356.

[6] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “SQuAD: 100,000 questions for machine comprehension of text,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2016, pp. 2383–2392.

[7] R. Jia and P. Liang, “Adversarial examples for evaluating reading comprehension systems,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2017, pp. 2021–2031.

[8] S. Min, E. Wallace, S. Singh, M. Gardner, H. Hujishirizi, and L. Zettlemoyer, “Compositional questions do not necessitate multi-hop reasoning,” in Proc. 57th Ann. Meet. Assoc. Comput. Linguistics, Florence, Italy, 2019, pp. 4249–4257. [Online]. Available: https://www.aclweb.org/anthology/P19-1416

[9] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, “Making the V in VQA matter: Elevating the role of image understanding in visual question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 6904–6913.

[10] K. Kafle and C. Kanau, “An analysis of visual question answering algorithms,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 1965–1973.

[11] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi, “Don’t just assume; look and answer: Overcoming priors for visual question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 4971–4980.

[12] H. Bahng, S. Chun, S. Yun, J. Choo, and S. J. Oh, “Learning de-biased representations with biased representations,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 528–539.

[13] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 3730–3738.

[14] Y. He, Z. Shen, and P. Cui, “Towards Non-IID image classification: A dataset and baselines,” Pattern Recognit., vol. 110, 2021, Art. no. 107383.
A. S. Rawat et al., “Disentangling sampling and labeling bias for learning in large-output spaces,” in Proc. 38th Int. Conf. Mach. Learn., 2021, pp. 8890–8901.

B. Kang et al., “Decoupling representation and classifier for long-tailed recognition,” in Proc. Int. Conf. Learn. Representations, 2020.

S. Zhang, Z. Li, S. Yan, X. He, and J. Sun, “Distribution alignment: A unified framework for long-tail visual recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 2361–2370.

M. Pagliardini, M. Jaggi, F. Fleuriet, and S. P. Karimireddy, “Agree to disagree: Diversity through disagreement for better transferability,” 2022, arXiv:2202.04414.

Y. Freund, “Boosting a weak learning algorithm by majority,” Inf. Comput., vol. 121, no. 2, pp. 256–285, 1996.

R. E. Schapire, “The strength of weak learnability,” Mach. Learn., vol. 5, no. 2, pp. 197–227, 1990.

P. J. Bickel, Y. Ritov, and A. Zakai, “Some theory for generalized boosting algorithms,” J. Mach. Learn. Res., vol. 7, no. May, pp. 705–732, 2006.

Y. Wald, A. Feder, D. Greenfeld, and U. Shalit, “On calibration and out-of-domain generalization,” in Proc. Adv. Neural Inf. Process. Syst., 2021, pp. 2215–2227.

M. Yi, R. Wang, J. Sun, Z. Li, and Z.-M. Ma, “Improved OOD generalization via conditional invariant regularizer,” 2022, arXiv:2207.06687.

D. Hendrycks, K. Lee, and M. Mazeika, “Using pre-training can improve model robustness and uncertainty,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 2712–2721.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

R. R. Selvaraju et al., “Taking a hint: Leveraging explanations to make vision and language models more grounded,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 2591–2600.

J. Wu and R. Mooney, “Self-critical reasoning for robust visual question answering,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 8604–8614.

C. Jing, Y. Wu, X. Zhang, Y. Jia, and Q. Wu, “Overcoming language priors in VQA via decomposed linguistic representations,” in Proc. Conf. Assoc. Advance. Artif. Intell., 2020, pp. 11181–11188.

G. K. Y and A. Mittal, “Reducing language biases in visual question answering with visually-grounded question encoder,” 2020, arXiv:2007.06198.

Y. Niu, K. Tang, H. Zhang, Z. Lu, X.-S. Hua, and J.-R. Wen, “Counterfactual VQA: A cause-effect look at language bias,” 2020, arXiv:2006.04315.

Q. Si, Z. Lin, M. Zheng, P. Fu, and W. Wang, “Check it again: Progressive visual question answering via visual entailment,” 2021, arXiv:2106.04605.

M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, “Bidirectional attention flow for machine comprehension,” in Proc. Int. Conf. Learn. Representations, 2017.

P. Gao, H. You, Z. Zhang, X. Wang, and H. Li, “Multi-modality latent inter-representation network for out-of-distribution visual question answering,” 2019, arXiv:1908.04289.

R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko, “Learning to reason: End-to-end module networks for visual question answering,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 804–813.

R. Hu, A. Rohrbach, T. Darrell, and K. Saenko, “Language-conditioned graph networks for relational reasoning,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 10294–10303.

J. Mao, C. Gan, P. Kohli, J. B. Tenenbaum, and J. Wu, “The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision,” 2019, arXiv:1904.12584.

X. Han, S. Wang, C. Su, W. Zhang, Q. Huang, and Q. Tian, “Interpretable visual reasoning via probabilistic formulation under natural supervision,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 553–570.

J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, “CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1988–1997.

D. A. Hudson and C. D. Manning, “GQA: A new dataset for real-world visual reasoning and compositional question answering,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 6693–6702.

S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

D. Teney et al., “On the value of out-of-distribution testing: An example of Goodhart’s law,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 407–417.
Xinzhe Han (Student Member, IEEE) received the BS degree from Xidian University in 2017. He is currently working toward the PhD degree with the School of Computer Science and Technology, University of Chinese Academy of Sciences. His current research interests include visual question answering and trustable machine learning.

Shuhui Wang (Member, IEEE) received the BS degree in electronics engineering from Tsinghua University, Beijing, China, in 2006, and the PhD degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2012. He is currently a full professor with the Key Laboratory of Intelligent Information Processing (CAS), Institute of Computing Technology, Chinese Academy of Sciences. He is also with Pengcheng Laboratory, Shenzhen. His research interests include image/video understanding/retrieval, cross-media analysis and visual-textual knowledge extraction.

Chi Su received the PhD degree from the Institute of Digital Media, EECS, Peking University. He is currently a general manager of SmartMore, Beijing. His research include computer vision and machine learning, with focus on object detection, object tracking, and human identification and recognition.

Qingming Huang (Fellow, IEEE) received the BS degree in computer science and the PhD degree in computer engineering from the Harbin Institute of Technology, Harbin, China, in 1988 and 1994, respectively. He is currently a chair professor with the School of Computer Science and Technology, University of Chinese Academy of Sciences. He has published more than 500 academic papers in international journals, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Transactions on Circuits and Systems for Video Technology, and top level international conferences, including the ACM Multimedia, ICCV, CVPR, ECCV, VLDB, and IJCAI. He was the associate editor for IEEE Transactions on Circuits and Systems for Video Technology and the associate editor for Acta Automatica Sinica. His research interests include multimedia computing, image/video processing, pattern recognition, and computer vision.

Qi Tian (Fellow, IEEE) received the BE in electronic engineering from Tsinghua University, the MS degree in ECE from Drexel University, respectively, and the PhD degree in ECE from the University of Illinois at Urbana-Champaign (UIUC). He is currently a chief scientist in artificial intelligence with Cloud BU, Huawei. From 2018-2020, he was the chief scientist in computer vision with Huawei Noah’s Ark Lab. He was also a full professor with the Department of Computer Science, the University of Texas at San Antonio (UTSA) from 2002 to 2019. During 2008-2009, he took one-year Faculty Leave with Microsoft Research Asia (MSRA). His research interests include computer vision, multimedia information retrieval and machine learning and published 600+ refereed journal and conference papers. His Google citation is more than 46500+ with H-index 99. He was the co-author of best papers including ACM Multimedia 2022, IEEE ICME 2019, ACM CIKM 2018, ACM ICMR 2015, PCM 2013, MMM 2013, ACM ICMCS 2012, a Top 10% Paper Award in MMSP 2011, a Student Contest Paper in ICASSP 2006, and co-author of a Best Paper/Student Paper Candidate in ACM Multimedia 2019, ICME 2015 and PCM 2007. His research projects are funded by ARO, NSF, DHS, Google, FXPAL, NEC, SALSI, CIAS, Akira Media Systems, HP, Blippar and UTSA. He received 2017 UTSA President’s Distinguished Award for Research Achievement, 2016UTSA Innovation Award, 2014 Research Achievement Awards from College of Science, UTSA, 2010 Google Faculty Award, and 2010 ACM Service Award. He is the associate editor for IEEE Transactions on Multimedia, IEEE Transactions on Circuits and Systems for Video Technology, ACM Transactions on Multimedia Computing, Communications, and Applications, Multimedia Systems, and in the Editorial Board of Journal of Multimedia (JMM) and Journal of MVA. He is the guest editor of IEEE Transactions on Multimedia, Journal of CVIU, etc.