

Kernel-Whitening: Overcome Dataset Bias with Isotropic Sentence Embedding

Songyang Gao¹, Shihan Dou¹, Qi Zhang^{12*}, Xuanjing Huang¹²

¹ School of Computer Science, Fudan University, Shanghai, China

² Shanghai Key Laboratory of Intelligent Information Processing, Shanghai, China
{gaosy21, shdou21}@m.fudan.edu.cn

Abstract

Dataset bias has attracted increasing attention recently for its detrimental effect on the generalization ability of fine-tuned models. The current mainstream solution is designing an additional shallow model to pre-identify biased instances. However, such two-stage methods scale up the computational complexity of training process and obstruct valid feature information while mitigating bias. To address this issue, we utilize the representation normalization method which aims at disentangling the correlations between features of encoded sentences. We find it also promising in eliminating the bias problem by providing isotropic data distribution. We further propose Kernel-Whitening, a Nyström kernel approximation method to achieve more thorough debiasing on nonlinear spurious correlations. Our framework is end-to-end with similar time consumption to fine-tuning. Experiments show that Kernel-Whitening significantly improves the performance of BERT on out-of-distribution datasets while maintaining in-distribution accuracy.

1 Introduction

Despite remarkable performance on NLP tasks, pre-trained language models, like BERT, suffer sharp performance degradation in out-of-distribution (OOD) settings (McCoy et al., 2019). The above defect roots in the excessive reliance on spurious correlations, which is widely found in crowdsourcing-built datasets (Gururangan et al., 2018). These phenomena are donated as dataset bias problem (He et al., 2019). A line of works attempts to tackle this problem by down-weighting bias training examples to discourage the main model from adopting recognized biases, including example reweighting (Schuster et al., 2019), confidence regularization (Utama et al., 2020a), or model ensembling (Clark et al., 2019).

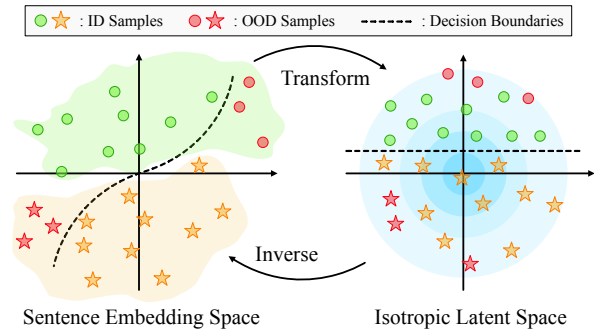


Figure 1: Illustration of Kernel-Whitening. The vertical and horizontal axes represent the valid and invalid features, respectively. Uneven sample distribution induces a bias decision boundary, resulting in errors on out-of-distribution data. The normalization method maps the data to isotropic latent space, where the new boundary is uncorrelated to redundant features, providing better generalization.

However, aforementioned methods over-depend on researchers’ intuition and task-specific insights to characterize spurious correlations, causing unrecognized bias patterns to remain in individual dataset (Sharma et al., 2018). Such assumption that dataset biases are known as a prior has been relaxed by limited capacity models (Utama et al., 2020b) or early training (Tu et al., 2020) in recent works. These approaches still rely on extra shallow models, which are not end-to-end, and weak-weighted bias samples simultaneously obstruct learning from their non-bias parts (Wen et al., 2021).

Instead of designing an extra model as previous attempts did, in this work, we propose a novel end-to-end framework, Kernel-Whitening, to significantly improve OOD performance while maintaining a similar computational cost as conventional BERT fine-tuning. The BERT-whitening (Su et al., 2021) and BERT-flow (Li et al., 2020) methods are effective normalization techniques to obtain better semantic representation. BERT-whitening calculates a linear operator¹ with SVD decompo-

* Corresponding author.

¹A positive definite symmetric matrix.

sition (GOLUB and REINSCH, 1970) to transform the sentence representation to follow a distribution with respect to the standard normal distribution. BERT-flow introduces normalized flow (Rezende and Mohamed, 2015) to perform similar transformations. Particularly, we find that the normalization method is also promising in improving the generalization ability of fine-tuned models by eliminating spurious correlations in training datasets. Despite the significant improvement over the OOD datasets, the linear transformation of BERT-whitening is not capable of dealing with non-linear dependencies between features. Meanwhile, flow-based methods require a complex inference process, which scales up the training costs.

In an attempt to eliminate nonlinear correlations while maintaining low training expenditure, we introduce kernel methods to naturally reconstruct a set of sentence representations only with linear correlation (Achlioptas et al., 2001). However, traditional kernel methods focus only on data similarities without providing explicit mapping operators, therefore, we use the Nyström approximation (Xu et al., 2015) to obtain low-rank kernel estimations. In general, we transform training data to an isotropic Gaussian distribution without affecting topological relationships between data points.

Kernel-Whitening² achieves competitive performance on generalization tasks. Experiments on eight datasets demonstrate that our method can improve the accuracy by 7%-11% on OOD datasets. In addition, the analysis of sentence representation proves that our method effectively removes the spurious correlations between dimensional features, which are known to be the direct cause of the dataset bias problem. Overall, our main contributions are as follows:

- We propose a novel framework, Kernel-Whitening, which ameliorates the bias problem by transforming sentence representation into isotropic distribution with similar time to fine-tuning.
- We introduce a kernel estimation algorithm, *i.e.*, Nyström approximation, to alleviate normalization methods from the trade between complex arithmetic and disentangle effects.
- We conduct comprehensive experiments on debiasing tasks to verify the effectiveness of

²Our code is available at <https://github.com/SleepThroughDifficulties/KernelWhitening>.

normalization methods for overcoming spurious correlations.

2 How Do Normalization Methods Provide Better Generalization

In this section, we discuss the negative impacts of dataset bias on the model’s generalization ability, and subsequently, how normalization methods lead to better performance in OOD settings.

2.1 Illustrate Dataset Bias from Feature Perspective

We first interpret dataset bias as triggered by the imbalance distribution of training data in feature space. Figure 2 shows an empirical analysis on MNLI dataset (Williams et al., 2018). Both train and test sets exhibit a strong positive correlation between the word overlap ratio and the class portion of the entailment label, which enable models to achieve high accuracy scores without modelling semantic information. McCoy et al. (2019) proposed HANS, a label-fair test set (*i.e.*, labels are proportionally consistent with different degrees of word overlap) to investigate the generalizability of models fine-tuned with above-mentioned bias, the model provides over 20% accuracy degradation when the specific literal heuristic could not be utilized for prediction.

Formally, given input data (X, Y) and bias dataset D , the training process can be formulated as follows:

$$\begin{aligned} P(Y|X^L, X^P) &= \frac{P(Y|X^L)P(X^P|X^L, Y)}{P(X^P)} \\ &= P(Y|X^L), \end{aligned} \quad (1)$$

where X^L represents the task-related features, and X^P represents other irrelevant features. That is, the model learns X^L from the label pair (X, Y) . By modeling the conditional distribution of labels relative to the input, the model extracts valid features for specific tasks.

However, previous works argue that construction process of bias dataset introduces spurious correlations between $(X^L|D)$ and $(X^P|D)$ (Gururangan et al., 2018):

$$\begin{aligned} P(X^P|X^L, D) &= \frac{P(X^P, X^L|D)}{P(X^L|D)} \\ &> P(X^P|D). \end{aligned} \quad (2)$$

Therefore, the actual training objective on dataset D is the posterior between feature

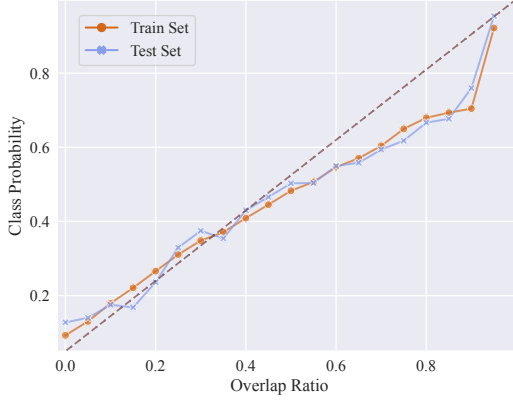


Figure 2: Lable probability with increasing lexical overlap for entailment label on MNLI dataset.

distribution $(X^L, X^P|D)$ and label Y , *i.e.*, $P(Y|X^L, X^P; D)$. By Eq.2, irrelevant features increase the confidence for specific labels:

$$\begin{aligned} P(Y|X^L, X^P, D) &= \frac{P(Y|X^L, D)P(X^P|X^L, D)}{P(X^P)} \\ &> P(Y|X^L, D) \\ &= P(Y|X^L). \end{aligned} \quad (3)$$

Such overconfidence does not perturb the model effect on the test set, which has a similar distribution to the training set. However, out-of-distribution data follow the correct distribution $P(Y|X^L)$, and are therefore classified as the bias label (*e.g.*, "entailment" in MNLI dataset), even if they have different relations.

2.2 Isotropic Representation Leads to Better Generalization

According to the definition described in Eq.3, dataset bias causes deep networks to fit the dataset-specific distribution, which impairs the generalization performance. Normalization methods intervene in the above problem by reconstructing the feature space. When sentences are encoded by pre-train model, the embedding representations are transformed into isotropic distribution, *e.g.*, standard normal distribution. Suppose data x and prior u satisfy:

$$x = f_\theta(u), \quad u \sim P_U(u), \quad (4)$$

where \mathcal{U} represents the latent space of isotropic u , and f is an invertible function. The probabilistic density function of original data on the transformed space can be calculated as follows:

$$P_{\mathcal{X}}(\mathbf{x}) = \left| \det \frac{\partial f_\theta^{-1}(\mathbf{x})}{\partial \mathbf{x}} \right| P_U(f_\theta^{-1}(\mathbf{x})). \quad (5)$$

By Eq.5, the distribution of training data is transformed into isotropic ones. Subsequently, suppose X_u^L and X_u^P are the latent representations of X^L and X^P , the spurious correlations between valid features and invalid features are eliminated:

$$\begin{aligned} P_U(Y|X_u^P, X_u^L, D) &= \frac{P_U(Y|X_u^L, D)P_U(X_u^L, X_u^P|D)}{P_U(X_u^P)P_U(X_u^L|D)} \\ &= P_U(Y|X_u^L), \end{aligned} \quad (6)$$

With isotropic data distribution, a decision boundary independent of the redundant features is obtained, which provides better generalization on the OOD samples.

2.3 Minor Weakness for BERT-flow and BERT-whitening

The BERT-flow method learns a flow-based generative model to fit the transform function f_θ , and BERT-whitening method computes the inverse linear operator with SVD decomposition of the covariance matrix. Despite the decent effect of representing normalization methods, BERT-flow requires multiple convolutional layers to find the appropriate transformation function, which increases the difficulty and time consumption of the training process. When fine-tuned on a tiny dataset, the flow layer encountered obstacles in providing reasonable transform results. Moreover, the BERT-whitening method focuses only on eliminating linear correlations between features, which is ineffective in alleviating the nonlinear correlation problem. In an attempt to ease the effort of training and provide faster but thorough transformation, we propose a novel normalization framework by kernel approximation, which is detail discussed in the next section.

3 Distribution Generalization with Kernel Approximation

In this section, we show our end-to-end framework named Kernel-Whitening. We first introduce Nyström kernel estimation algorithm, and subsequently how to apply such approximation method on debiasing tasks.

3.1 Nyström Kernel Estimation

We first elaborate kernel trick, which constructs new linearly differentiable properties by mapping the original feature space onto a high-dimensional RKHS (Alvarez et al., 2012). Given a set of training data $X = \{x_i \in \mathbb{R}^d, i = 1, \dots, n\}$, the kernel

method maps X onto a dot product space \mathcal{H} using $\phi : X \rightarrow \mathcal{H}$. Generally, the dimension of \mathcal{H} can be so large that the mapping function cannot be obtained explicitly. Nevertheless, the dot product result can be represented by a positive definite kernel k , *i.e.*, a function k satisfies:

$$\begin{aligned} k(x_k, x_j) &= \langle \phi(x_k), \phi(x_j) \rangle \\ &= \sum_{i=1}^N \lambda_i \phi_i(x_k) \phi_i(x_j), \end{aligned} \quad (7)$$

where λ_i and ϕ_i denotes the eigenvalues and eigenfunctions of kernel operator k , and N denotes their number. With finite dataset $\{x_i \in \mathbb{R}^d, i = 1, \dots, n\}$, such decomposition can be replaced with empirical estimation as follows:

$$\frac{1}{n} \sum_{j=1}^n k(\mathbf{x}_k, \mathbf{x}_j) \phi_i(\mathbf{x}_j) = \lambda_i \phi_i(\mathbf{x}_k). \quad (8)$$

Eq.8 indicates a spectral decomposition of kernel matrix G , which satisfy $G_{k,j} = k(x_k, x_j)$. Considering the SVD decomposition of G as

$$G = W \Sigma W^T, \quad (9)$$

where W is an orthogonal matrix, and Σ is a diagonal matrix with positive diagonal elements. When matching Eq.8 with Eq.9, the mapping operator ϕ is therefore denoted as:

$$\phi(x_j) = G_{j,:} W \Sigma^{\frac{1}{2}}. \quad (10)$$

However, existing datasets often contain thousands to hundreds of thousands samples, which makes it impossible to directly calculate the SVD decomposition. Therefore, we introduce Nyström method (Williams and Seeger, 2001) to provide a low-rank estimation of kernel matrix.

Suppose S to be a sampled subset of X , the kernel matrix G can be represented as:

$$G = \begin{bmatrix} G_s & G_r^T \\ G_r & G_x \end{bmatrix}, \quad (11)$$

where G_s denotes the gram matrix of subset S . The W and Σ can subsequently approximated by:

$$W \approx [G_s \ G_r]^T, \ \Sigma \approx G_s. \quad (12)$$

The reconstructed Nyström representation of single example \mathbf{x} is as follows:

$$\phi(\mathbf{x}) = G_{\mathbf{x},S} G_s^{-\frac{1}{2}}, \quad (13)$$

where $G_{\mathbf{x},S} = [k(x_1, \mathbf{x}), \dots, k(x_s, \mathbf{x})]^T$ for $x_i \in S$. By estimating high-dimensional representations of the training samples we obtain a linearly divisible distribution, which can be normalized with a linear transformation.

3.2 Batch Iterate for Global Approximation

In Section 3.1, we elaborate the standard Nyström approach to processing input data. The difference is, the subset S in the traditional approach, *i.e.*, kernel SVM, usually contains hundreds of elements, while deep networks are trained on smaller batches (*e.g.*, 32 for Kernel-Whitening) with stochastic gradient descent (SGD) optimizer. The insufficient samples compromise the information of reconstructed representations, making the improvement inconspicuous when directly applying Nyström methods to debiasing tasks.

In attempt to introduce global information while processing batch data, we design preservation and reloading structures to extend the dimension of low-rank kernel matrix. For each batch, we calculate Nyström matrix with batch features Z_L^t and global features Z_f^t , which represent the principal components of data distribution in step t . Giving input data Z_L^t contains L instances, the extended representation is given with Eq.13:

$$\phi(Z_L^t) = G_{0:L,:} G^{-\frac{1}{2}}, \quad (14)$$

where G denotes the kernel matrix generated by $[Z_L^t, Z_f^t]^T$, and $G_{0:L,:}$ denotes the first L rows of G . Especially, we select Radial Basis Function (RBF) kernel in Kernel-Whitening method.

Noticing that the kernel estimation method only projects the data points into a linearly separable space, we further normalize the distribution with a linear transformation. The reconstructed representation $\phi(Z_L^t)$ is subsequently weighted under the supervised signal of Hilbert-Schmidt independence criterion (HSIC) (Wang et al., 2021), which is an adequate indicator for estimating the mutual independence between features. The optimal weight W^* is calculated by:

$$W^* = \arg \min_{W \in \mathcal{R}^L} \left\| \tilde{\Sigma}_{G,W} \right\|_F^2, \quad (15)$$

where $W = \{W_i\}_{i=1}^n$ denotes the sample weight vector, and $\tilde{\Sigma}_{G,W}$ represents the empirical estima-

tion for covariance between features:

$$\tilde{\Sigma}_{G,W} = \frac{1}{L-1} \sum_{1 < i < j < L} F_i^T F_j \quad (16)$$

$$F_i = W G_{:,i} - \frac{1}{n} \sum_{j=1}^n W_j G_{j,:} \quad (17)$$

At the end of each iteration, we update the global features with local information Z_L^t to catch reasonable basis vectors:

$$Z_f^{t+1} = \alpha_i Z_f^t + (1 - \alpha_i) Z_L^t, \quad (18)$$

where α_i denotes the attenuated factor controlling the importance of local information.

3.3 Training Objective

In Section 3.2, we show how to obtain the reconstructed feature representation $\phi(Z_L^t)$ and weighting parameters W^* . Subsequently, we use above results to train on original BERT models. The final train loss of Kernel-Whitening is denoted as:

$$Loss = \sum_{i=1}^L W_i^* f(\phi(Z_L^t)_i, y_i), \quad (19)$$

where $f(\cdot, \cdot)$ represents the cross-entropy loss with input $\phi(Z_L^t)_i$ and its corresponding label y_i . Our detailed algorithm implementation is shown in Algorithm 1.

Algorithm 1 Framework of Kernel-Whitening for our system.

Input: The set of pooler output for current batch,

$Z = \{Z_i\}_{i=1}^n$; The set of global features, Z_f^t ;

The classifier for specific task, $f(\cdot, \cdot)$.

- 1: Compute kernel matrix G by $[Z, Z_f^t]^T$
- 2: Compute W, Σ by SVD(G)
- 3: Compute representation $\phi(Z)$ by G, W, Σ with Eq.10, 14
- 4: Compute weight W^* by HSIC(G, W) with Eq.15
- 5: Update global features Z_f^t with Z with Eq.18
- 6: Compute training loss L by classifier $f(\cdot, \cdot)$, W^* and $\phi(Z)$ with Eq.19

Output:

Updated global features, Z_f^{t+1} ;

Training loss, L

4 Experiments

In this section, we provide a comprehensive analysis of Kernel-Whitening and the other two normalization methods (*i.e.*, BERT-flow (Li et al., 2020)³ and BERT-whitening (Su et al., 2021)) through extensive experiments on three tasks.

4.1 Baseline Methods

Our method is compared with previous works as follows:

- Clark et al. (2019) (Reweighting and Learned-Mixin), which predicts confidence for each sample and down-weights problematic data.
- Sanh et al. (2020) (Product-of-Experts and PoE_{cross-entropy}), which trains limited capacity models as experts to debias without explicitly identifying dataset bias.
- Utama et al. (2020b) (PoE_{self-debias} and Conf-reg_{self-debias}), which uses a shallow model to identify biased samples and focus the main model on them.
- Utama et al. (2020a) (Conf-reg), which uses confidence regularization to discourage models from exploiting biases.
- Xiong et al. (2021) (MoCaD), which produces uncertainty estimations to achieve a three-stage ensemble-based debiasing framework.

4.2 Datasets and Metrics

We conduct experiments on three tasks: natural language inference, fact verification, and paraphrase identification. Each task contains in-distribution and out-of-distribution datasets.

For the NLI task, we conduct experiments on the Multi-Genre Natural Language Inference (MNLI) dataset (Nangia et al., 2017) and HANS (McCoy et al., 2019). We train the model on MNLI, and choose MNLI-mismatch and HANS as the ID and OOD test set.

For the fact verification task, we use FEVER⁴ as the training dataset provided by (Thorne et al., 2018). We train the model on FEVER, and evaluate the model performance on ID test set and FEVER Symmetric (Schuster et al., 2019) (version 1 and 2) as our OOD test set.

³The code of BERT-flow is available at <https://github.com/bohanli/BERT-flow>.

⁴<https://github.com/TalSchuster/FeverSymmetric>

Model	MNLI		FEVER			QQP		
	ID	HANS	ID	Symm. v1	Symm. v2	ID	PAWS dupl	\neg dupl
BERT-base	84.3	61.1	85.4	55.2	63.1	91	96.9	9.8
Reweighting	83.5	69.2	84.6	61.7	66.5	85.5	49.7	51.2
Product-of-Experts	84.1	66.3	82.3	62.0	65.9	88.8	50.3	61.2
PoE _{cross-entropy}	83.6	67.3	85.7	57.7	61.4	-	-	-
PoE _{self-debias}	80.7	68.5	85.4	59.7	65.3	77.4	44.1	69.4
Learned-Mixin	84.2	64.0	83.3	60.4	64.9	86.6	69.7	51.7
Conf-reg	84.3	69.1	86.4	60.5	66.2	89.1	91.0	19.8
Conf-reg _{self-debias}	84.3	67.1	87.6	59.8	66.0	85.0	48.8	28.7
MoCaD	82.3	70.7	87.1	65.9	69.1	-	-	-
BERT-flow*	82.5	69.1	85.7	64.4	67.2	-	-	-
BERT-whitening-64*	84.3	69.9	86.7	65.7	69.5	87.5	82.5	34.7
BERT-whitening-384*	83.7	70.4	85.9	65.9	68.3	87.3	75.3	41.3
Kernel-Whitening-64*	84.4	70.1	87.4	65.1	69.8	90.8	91.2	32.4
Kernel-Whitening-384*	83.9	70.9	87.8	66.2	70.3	87.6	72.7	43.2

Table 1: Models evaluation on MNLI, FEVER, QQP, and their respective challenge test sets. The performance of the three normalized models is shown in cyan, the model name with asterisks represents the experimental results on our machine. The best results on each dataset are bolded.

Model	Requires Prior Knowledge	Extra Model
Reweighting	✓	✓
Product-of-Experts	✓	✓
PoE _{cross-entropy}	✓	✓
PoE _{self-debias}	✗	✓
Learned-Mixin	✓	✓
conf-reg	✓	✓
Conf-reg _{self-debias}	✗	✓
MoCaD	✓	✓
Kernel-Whitening	✗	✗

Table 2: Details of the state-of-the-art debiasing methods used to compare with Kernel-Whitening. Our model is end-to-end while not requiring prior knowledge of biases or additional shallow models.

For the paraphrase identification task, we perform the evaluation using Quora Question Pairs (QQP) as ID dataset and PAWS (Zhang et al., 2019) as OOD dataset which consists of two types of data including *duplicate* and *non-duplicate*.

Evaluation Metrics

Following previous works, we measured the accuracy score on the in-distribution and out-of-distribution test sets to compare the results of different models.

4.3 Implementation Details

Following previous debiasing methods, we apply our debiasing method on the BERT-base (Devlin et al., 2019). The hyperparameters of BERT are consistent with previous research papers. The learning rate is $2e-5$ for MNLI dataset and $1e-5$ for FEVER and QQP, the batch size is 32 and the optimizer is AdamW with a weight decay of 0.01. Note that previous methods (Sanh et al., 2020; Xiong et al., 2021) have shown high variance in experiment results under different settings, we evaluate the performance of our model by four random seeds and report the averaged result. We use the [CLS] vector as sentence embedding for all three methods. The model is trained in an NVIDIA GeForce RTX 3090 GPU. All models are trained five epochs, and checkpoints with top-2 performance are finally evaluated on the challenge test set.

4.4 Experimental Results

The extensive results of all the above-mentioned methods are summarized in table 1. Compared with other baseline methods, Kernel-Whitening significantly improves the model performance on challenge sets, and achieves state-of-the-art results on seven of the eight benchmarks. On the MNLI and FEVER datasets, our framework achieves the best performance with about 10 percentage points

higher than the accuracy of BERT-base, which outperforms other debiasing methods. This proves that our framework has the best results and generalizability among these methods.

Moreover, our approach can effectively eliminate the dataset bias while mitigating the damage to generalizable features. The vast majority of debiasing methods improve the performance of out-of-distribution datasets by sacrificing the performance of in-distribution datasets, which means that current debiasing methods attempt to achieve a trade-off between ID performance and OOD performance. However, our approach achieves the best performance on OOD datasets for natural language inference and fact verification tasks with better results on ID datasets. For QQP dataset, our proposed approach also achieves decent generalization in PAWS without excessive performance degradation on ID datasets.

In general, the normalization methods perform well on both in-distribution and out-of-distribution datasets for all tasks. All five models of three methods are end-to-end approaches and do not rely on any prior knowledge of the dataset. That is to say, they achieve better utility and scalability while providing more effective debiasing. For BERT-whitening and Kernel-Whitening, a larger hidden dimension indicates better performance on OOD datasets, and Kernel-whitening performs better when parameters are constant to BERT-whitening, which strongly supports our analysis of normalization methods. BERT-flow shows an acceptable performance on OOD datasets, but is inferior to the whitening-based approach. We argue that flow model requires more samples as reference, and the original hyperparameters are not capable for the additional network layers.

5 Analysis and Discussion

In this section, we construct supplementary experiments to further analyze the effectiveness of normalization methods, especially our Kernel-Whitening framework.

5.1 Effect of Latent Dimension L

The dimensionality of reconstructed features is a key feature. The reduction of vector size brings smaller memory occupation and faster inference downstream layers, while the missing information may impair the ability of the model. To further illustrate the effect of low-rank kernel approxima-

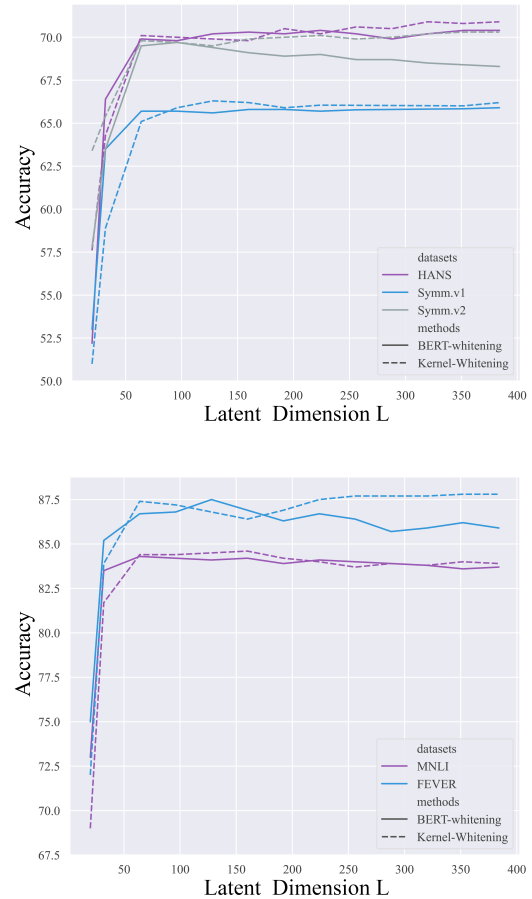


Figure 3: Effect of different dimensionality L with whitening methods on each aforementioned tasks. The x axis is the latent dimension of sentence embeddings. The two images are model performance on out-of-distribution and in-distribution test sets, respectively.

tion, we conduct a sensitivity analysis of latent dimension L . Figure 3 shows the variation curve of performance change for two whitening-based methods. For both in-distribution and out-of-distribution tasks. A latent dimension of double the batch size provides promising performance. As the dimensionality rises, Kernel-Whitening maintains a stable debiasing effect, while BERT-whitening fluctuates on FEVER and Symm. v2 dataset. We argue that this phenomenon is due to that high-dimensional features are more prone to nonlinear correlations, where Kernel-Whitening is designed to show better results. Moreover, Kernel-Whitening always performs better when the dimensionality is greater than 300, which illustrates the stability and generality of our method.

5.2 Independence Study

In Section 2 we analyse how isotropic data distribution leads to better generalization. To check

whether normalization methods remove the dependencies between features, we conduct experiments on the covariance between features during training process. As shown in Figure 4, All three normalization methods exhibit a suppression effect on feature correlation, while our method achieves the optimal performance at the end of training. As the iterations increase, the covariance first decreases rapidly and converges to a low point. All method’s performance fluctuations around certain steps, we believe such fluctuations are related to biased samples in the data.

Overall, Kernel-Whitening largely remits dependencies between features, and such independence effectively contributes to the generalization ability of deep network models.

5.3 Time Consumption

Besides outstanding debiasing performance, we compute the time consumption with baseline methods to further demonstrate the strength of Kernel-Whitening. We train each model equally on an NVIDIA RTX 2080Ti GPU with the same batch size. We compare three normalization methods with the best baseline work, MoCaD (Xiong et al., 2021), which trains a bias model to produce model calibrating. To give a horizontal comparison between different datasets, we set the time consumption of fine-tune 100 as a baseline. As shown in Table 3, the time consumption of Kernel-Whitening is nearly the same as fine-tuning and costs 6 times less extra time than MoCad. Although BERT-whitening only uses a linear transformation to obtain reconstruction representations, our method is still faster. Because our method performs SVD decomposition on a matrix of $L * L$ while BERT-whitening handles the same operations on a matrix of $L * N$, where L is the hidden dimension and N is the output dimension of BERT (*e.g.*, 768).

Method	MNLI	FEVER
BERT-base	100	100
MoCad	264	227
BERT-flow	198	190
BERT-whitening	146	139
Kernel-Whitening	138	134

Table 3: Time consumption (percentages) of training one epoch on the whole dataset. Whitening-based methods cost much less time than previous works.

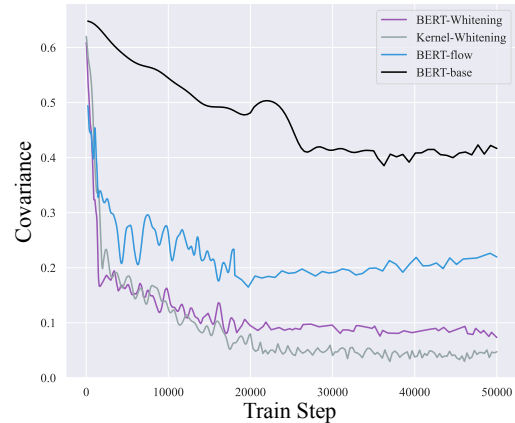


Figure 4: The independence study on covariance between reconstructed features. The hidden dimension of BERT-whitening and Kernel-Whitening are set to 64. All models are trained 50000 steps with batch 32.

6 Related Work

6.1 Dataset Bias

Recent observations (McCoy et al., 2019; Naik et al., 2018) show that, natural language understanding models tend to over-rely on specific shallow heuristics, resulting in inadequate generalization capability in out-of-distribution (OOD) settings (Schuster et al., 2019). Sinha et al. (2021); Pham et al. (2021) have reported the insensitivity to word-order permutations among transformer-based models. The original and out-of-order examples elicit the same classification label when permuted randomly, which contradicts the conventional understanding of semantics. Such phenomena are studied as dataset bias problems.

Existing methods train additional models to identify biased training data (Clark et al., 2019; Utama et al., 2020a; Schuster et al., 2019) or use the above bias model to calibrate the classification results of test data (Utama et al., 2020b; Sanh et al., 2020; Xiong et al., 2021). The so-called bias model refers to classifiers who use only a portion of input data for prediction, *e.g.*, hypothesis-only model in NLI task which only predict from specific linguistic phenomena in hypothesis sentences such as negation. These methods are not end-to-end and face difficulty in fully identifying all bias patterns.

Recently, another line of works notice the connection between dataset bias and feature distribution, and try to tackle the dataset bias problem by identifying features with better generalizability. Dou et al. (2022) use an loss function based on information bottleneck (IB) to focus the model on task-relevant features, and Wu and Gui (2022)

similarly achieve such feature filtering by mapping sentence embedding into a specific low-dimension subspace.

6.2 Unsupervised Semantic of Sentence Embedding

Previous works suggest that the word representations of pre-train language model are not isotropic (Gao et al., 2018; Ethayarajh, 2019), leading model to poorly capture the underlying semantic of sentences (Li et al., 2020). Such anisotropic causes the difficulty of using sentence embedding directly through simple similarity metrics. Gao et al. (2018) propose word embedding matrix regularization methods to mitigate the degeneration problem. Recently, researchers attempt to transform BERT sentence embedding into an isotropic Gaussian distribution through normalizing flow (Li et al., 2020) or whitening methods (Su et al., 2021). As supervised learning also suffers from uneven data distribution of train sets, we are the first to normalize the data distribution on supervised training to eliminate dataset bias problem.

7 Conclusion

In this work, we propose a novel framework, Kernel-Whitening, to tackle the spurious correlation from a feature perspective. We analyze how to introduce isotropic sentence embedding for eliminating dataset bias and propose a promising and computationally kernel estimation, to obtain an approximation of disentangled sentence embedding. Experiments on various datasets demonstrate that Kernel-Whitening achieves better performance on both ID and OOD datasets than comparative works. This implies that a shallow model, or prior knowledge of dataset bias, is not must for the improvement of generalization.

8 Limitations

In this section, we discuss the potential limitations of our work. The analysis of model effects in this paper is focusing on commonly used benchmarks for natural language understanding debiasing works, and they may carry confounding factors that affect the performance of our model. Therefore, it is worth further exploring the performance of our model on more tasks, *e.g.*, the WikiGender-Bias dataset for gender bias on relation extraction task. In addition, this presented work is inspired by unsupervised semantic learning methods, such as

BERT-whitening, and it will be better to test the performance of our approach on unsupervised tasks. We leave these two problems to further work.

Acknowledgements

The authors wish to thank the anonymous reviewers for their helpful comments. This work was partially funded by National Natural Science Foundation of China (No. 62076069, 61976056, 61906176), Program of Shanghai Academic Research Leader (No. 22XD1401100), and Beijing Academy of Artificial Intelligence (BAAI).

References

- Dimitris Achlioptas, Frank McSherry, and Bernhard Schölkopf. 2001. Sampling techniques for kernel methods. *Advances in neural information processing systems*, 14.
- Mauricio A Alvarez, Lorenzo Rosasco, Neil D Lawrence, et al. 2012. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266.
- Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don’t take the easy way out: Ensemble based methods for avoiding known dataset biases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4069–4082.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **BERT: Pre-training of deep bidirectional transformers for language understanding**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Shihan Dou, Rui Zheng, Ting Wu, Songyang Gao, Qi Zhang, Yueming Wu, and Xuanjing Huang. 2022. Decorrelate irrelevant, purify relevant: Overcome textual spurious correlations from a feature perspective. *arXiv preprint arXiv:2202.08048*.
- Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65.
- Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tieyan Liu. 2018. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

- GH GOLUB and C REINSCH. 1970. Handbook series linear algebra. singular value decomposition and least squares solutions. *Numerische Mathematik*, 14:403–420.
- Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.
- He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. *EMNLP-IJCNLP 2019*, page 132.
- Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130.
- Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.
- Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353.
- Nikita Nangia, Adina Williams, Angeliki Lazaridou, and Samuel Bowman. 2017. The repeval 2017 shared task: Multi-genre natural language inference with sentence representations. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 1–10.
- Thang Pham, Trung Bui, Long Mai, and Anh Nguyen. 2021. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1145–1160.
- Danilo Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR.
- Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M Rush. 2020. Learning from others’ mistakes: Avoiding dataset biases without modeling them. In *International Conference on Learning Representations*.
- Tal Schuster, Darsh Shah, Yun Jie Serene Yeo, Daniel Roberto Filizzola Ortiz, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3419–3425.
- Rishi Sharma, James Allen, Omid Bakhshandeh, and Nasrin Mostafazadeh. 2018. Tackling the story ending biases in the story cloze test. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 752–757.
- Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. 2021. Unnatural language inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7329–7346.
- Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.
- James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819.
- Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633.
- Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020a. Mind the trade-off: Debiasing nlu models without degrading the in-distribution performance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8717–8729.
- Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020b. Towards debiasing nlu models from unknown biases. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7597–7610.
- Tinghua Wang, Xiaolu Dai, and Yuze Liu. 2021. Learning with hilbert–schmidt independence criterion: A review and new perspectives. *Knowledge-Based Systems*, 234:107567.
- Zhiquan Wen, Guanghui Xu, Mingkui Tan, Qingyao Wu, and Qi Wu. 2021. Debaised visual question answering from feature and sample perspectives. *Advances in Neural Information Processing Systems*, 34.
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1112–1122.

Chris Williams and Matthias Seeger. 2001. Using the nystroem method to speed up kernel machines. *Advances in Neural Information Processing Systems 13*.

Ting Wu and Tao Gui. 2022. Less is better: Recovering intended-feature subspace to robustify nlu models. *arXiv preprint arXiv:2209.07879*.

Ruibin Xiong, Yimeng Chen, Liang Pang, Xueqi Cheng, Zhi-Ming Ma, and Yanyan Lan. 2021. Uncertainty calibration for ensemble-based debiasing methods. *Advances in Neural Information Processing Systems*, 34.

Zenglin Xu, Rong Jin, Bin Shen, and Shenghuo Zhu. 2015. Nystrom approximation for sparse kernel methods: Theoretical analysis and empirical evaluation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.