Fine-tuning Pre-trained Language Models for Few-shot Intent Detection: Supervised Pre-training and Isotropization

Haode Zhang[1] **Haowen Liang**[1] **Yuwei Zhang**[2]
Liming Zhan[1] **Xiao-Ming Wu**[1*] **Xiaolei Lu**[3] **Albert Y.S. Lam**[4]
Department of Computing, The Hong Kong Polytechnic University, Hong Kong S.A.R.[1]
University of California, San Diego[2] Nanyang Technological University, Singapore[3]
Fano Labs, Hong Kong S.A.R.[4]
{haode.zhang,michaelhw.liang,lmzhan.zhan}@connect.polyu.hk, zhangyuwei.work@gmail.com
xiao-ming.wu@polyu.edu.hk, xiaolei.lu@ntu.edu.sg, albert@fano.ai

Abstract

It is challenging to train a good intent classifier for a task-oriented dialogue system with only a few annotations. Recent studies have shown that fine-tuning pre-trained language models with a small set of labeled utterances from public benchmarks in a supervised manner is extremely helpful. However, we find that supervised pre-training yields an anisotropic feature space, which may suppress the expressive power of the semantic representations. Inspired by recent research in isotropization, we propose to improve supervised pre-training by regularizing the feature space towards isotropy. We propose two regularizers based on contrastive learning and correlation matrix respectively, and demonstrate their effectiveness through extensive experiments. Our main finding is that it is promising to regularize supervised pre-training with isotropization to further improve the performance of few-shot intent detection. The source code can be found at https://github.com/hdzhang-code/isoIntentBert.

1 Introduction

Intent detection is a core module of task-oriented dialogue systems. Training a well-performing intent classifier with only a few annotations, i.e., few-shot intent detection, is of great practical value. Recently, this problem has attracted considerable attention (Vulić et al.; Zhang et al., b) but remains a challenge.

To tackle few-shot intent detection, earlier works employ induction network (Geng et al.), generation-based methods (Xia et al., a), metric learning (Nguyen et al.), and self-training (Dopierre et al., b), to design sophisticated algorithms. Recently, pre-trained language models (PLMs) have emerged as a simple yet promising solution to a wide spectrum of natural language processing (NLP) tasks, triggering the surge of PLM-based solutions for few-shot intent detection (Wu et al.; Zhang et al., a,b; Vulić et al.; Zhang et al., b), which typically fine-tune PLMs on conversation data.

A PLM-based fine-tuning method (Zhang et al., a), called IntentBert, utilize a small amount of labeled utterances from public intent datasets to fine-tune PLMs with a standard classification task, which is referred to as *supervised pre-training*. Despite its simplicity, supervised pre-training has been shown extremely useful for few-shot intent detection even when the target data and the data used for fine-tuning are very different in semantics. However, as will be shown in Section 3.2, IntentBert suffers from severe anisotropy, an undesirable property of PLMs (Cai et al., 2020; Ethayarajh; Li et al.).

Anisotropy is a geometric property that semantic vectors fall into a narrow cone. It has been identified as a crucial factor for the sub-optimal performance of PLMs on a variety of downstream tasks (Gao et al., a; Arora et al., b; Cai et al., 2020; Ethayarajh; Li et al.), which is also known as the representation degeneration problem (Gao et al., a). Fortunately, isotropization techniques can be applied to adjust the embedding space and yield significant performance improvement in many tasks (Li et al.; Su et al.; Rajaee and Pilehvar, 2021a).

Hence, this paper aims to answer the question:

- Can we improve supervised pre-training via *isotropization* for few-shot intent detection?

Many isotropization techniques have been developed based on transformation (Su et al.; Huang et al.), contrastive learning (Gao et al., b), and top principal components elimination (Mu and Viswanath). However, these methods are designed for off-the-shelf PLMs. When applied on PLMs that have been fine-tuned on some NLP task such as semantic textual similarity or intent classification, they may introduce an adverse effect, as observed

* Corresponding author.

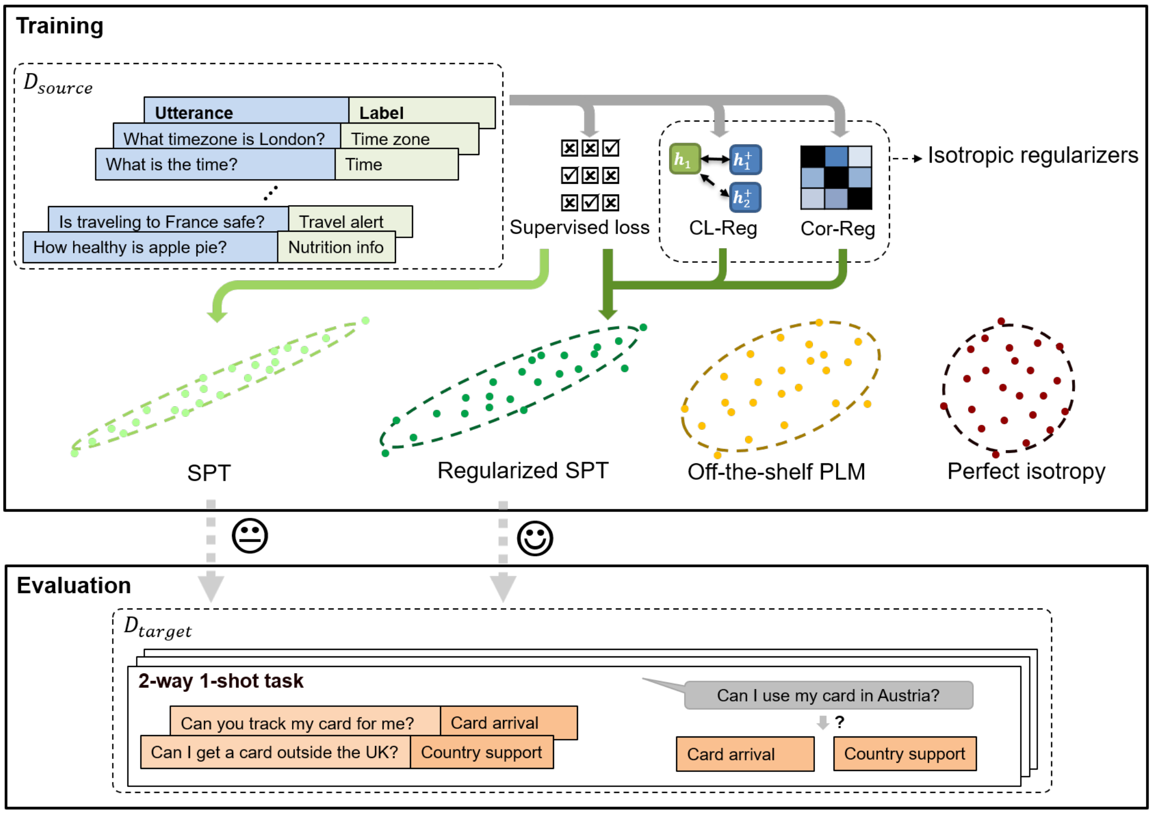


Figure 1: Illustration of our proposed regularized supervised pre-training. SPT denotes supervised pre-training (fine-tuning an off-the-shelf PLM on a set of labeled utterances), which makes the feature space more anisotropic. CL-Reg and Cor-Reg are designed to regularize SPT and increase the isotropy of the feature space, which also leads to better performance on few-shot intent detection.

in Rajaee and Pilehvar (2021b) and our pilot experiments.

In this work, we propose to regularize supervised pre-training with isotropic regularizers. As shown in Fig. 1, we devise two regularizers, a contrastive-learning-based regularizer (CL-Reg) and a correlation-matrix-based regularizer (Cor-Reg), each of which can increase the isotropy of the feature space during supervised training. Our empirical study shows that the regularizers can significantly improve the performance of standard supervised training, and better performance can often be achieved when they are combined.

The contributions of this work are three-fold:

- We present the first study on the isotropy property of PLMs for few-shot intent detection, shedding light on the interaction of supervised pre-training and isotropization.
- We improve supervised pre-training by devising two simple yet effective regularizers to increase the insotropy of the feature space.
- We conduct a comprehensive evaluation and analysis to validate the effectiveness of the proposed approach.

2 Related Works

2.1 Few-shot Intent Detection

With the surge of few-shot learning (Finn et al.; Vinyals et al.; Snell et al.), few-shot intent detection has started to receive attention. Earlier works mainly focus on model structures, such as capsule network (Geng et al.), variational autoencoder (Xia et al., a), metric function (Yu et al.; Nguyen et al.), usually leading to complicated solutions. Recently, PLMs-based methods are becoming more attractive due to their simplicity and promising performance. Zhang et al. (c) cast the problem into natural language inference (NLI) problem and fine-tune PLMs on the NLI dataset. Zhang et al. (b) fine-tune PLMs on unlabeled utterances in an unsupervised manner. Zhang et al. (a) fine-tune PLMs on large public annotated intent detection dataset. On the other hand, the study is extended to other settings including semi-supervised learning (Dopierre et al., b,a), generalized setting (Nguyen et al.), multi-label classification (Hou et al.) and incremental learning (Xia et al., b). This work focuses on the most basic setting, i.e., transferring learning from intents with abundant annotations to intents with limited annotations, further improving fine-tuning via isotropization.

2.2 Pre-training for Tasked-oriented Dialogue

Recently, a line of efforts try to adapt pre-trained models to task-oriented dialogue tasks (Henderson et al., b; Peng et al., 2021) by continue pre-training. Specifically, TOD-BERT (Wu et al.) conducts self-supervised learning on diverse dialogue corpus. ConvBERT (Mehri et al., 2020) is pre-trained on 700 million open-domain dialogue corpus. Vulić et al. study further task-wise fine-tuning after the adaptation to conversational corpus. This work follows the same methodology of continue pre-training of PLMs, but focuses on few-shot intent detection.

2.3 Anisotropy of PLMs

Isotropy is a crucial desired geometric property of the semantic space of PLMs. Recent studies identify the anisotropy problem of PLMs (Cai et al., 2020; Ethayarajh; Li et al.; Mu and Viswanath; Rajaee and Pilehvar, 2021b), which is also known as the representation degeneration problem (Gao et al., a): word embeddings concentrate in a narrow cone, which suppresses the expressive capability. To resolve the problem, various methods have been proposed, including spectrum control (Wang et al.), flow-based mapping (Li et al.), whitening transformation (Su et al.; Huang et al.), contrastive learning (Gao et al., b) and cluster-based method (Rajaee and Pilehvar, 2021a). Despite the significant improvement on various tasks, these methods are designed for off-the-shelf PLMs. The interaction between isotropization and fine-tuning remains under-explored. Most recently, Rajaee and Pilehvar reveal the potential contradiction between the two operations on the semantic textual similarity (STS) task. Zhou et al. propose to fine-tune PLMs with isotropic batch normalization on some supervised tasks, which requires a large amount of training data.

3 Pilot Experiments

Before introducing the approach, we present pilot experiments to gain some insights into the interaction between fine-tuning and isotropization.

Dataset	BERT	IntentBERT
BANKING	.96	.71(.04)
HINT3	.95	.72(.03)
HWU64	.96	.72(.04)

Table 1: The impact of fine-tuning on the isotropy. Fine-tuning renders the semantic space notabley more anisotropic. The mean and standard deviation are reported for experiments randomly repeated by 5 times.

3.1 Measuring isotropy

Following Mu and Viswanath; Biś et al., after making embeddings zero-mean, we adopt the measurement of isotropy as follows:

$$\mathrm{I}(\mathbf{V}) = \frac{\min_{\mathbf{c}\,\in\,C} \mathrm{Z}(\mathbf{c}, \mathbf{V})}{\max_{\mathbf{c}\,\in\,C} \mathrm{Z}(\mathbf{c}, \mathbf{V})}, \tag{1}$$

where $\mathbf{V} \in \mathbb{R}^{N\times d}$ is the matrix of stacked N utterance embeddings, C is the set of unit eigenvectors of $\mathbf{V}^\top\mathbf{V}$, and $\mathrm{Z}(\mathbf{c}, \mathbf{V})$ is the partition function (Arora et al., b) defined as:

$$\mathrm{Z}(\mathbf{c}, \mathbf{V}) = \sum_{i=1}^{N} \exp\left(\mathbf{c}^\top \mathbf{v}_i\right), \tag{2}$$

where $\mathbf{v}_i$ is the i_{th} row vector in $\mathbf{V}$. $\mathrm{I}(\mathbf{V})$ ranges in $[0, 1]$, and the value of 1 indicates perfect isotropy.

3.2 Fine-tuning Leads to Anisotropy

To observe the impact of fine-tuning on isotropy, we follow IntentBERT (Zhang et al., a), a highly effective solution to few-shot intent detection, to fine-tune BERT (Devlin et al.) on OOS (Larson et al.), a huge public intent detection dataset (details are given in Section 4.1), and then observe the isotropy change on target datasets. As shown in Table 1, after fine-tuning, the model's isotropy is notably deteriorated consistently on all datasets. The change in the covariance matrix of the semantic space agrees with the above observation[1]. Therefore, *fine-tuning renders the semantic space anisotropic*.

3.3 Isotropization after Fine-tuning May Have an Adverse Effect

To examine how isotropization affects the fine-tuned model, we apply two strong isotropization techniques to IntentBERT: dropout-based contrastive learning (Gao et al., b) and whitening transformation (Su et al.). The former fine-tunes PLMs

[1]For details, please refer to the appendix.

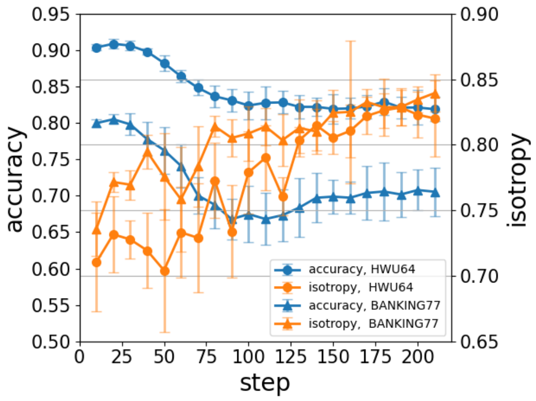


Figure 2: The training process of contrastive learning on IntentBERT. The isotropy (orange) is improved, but the performance (blue) drops down. The data is collected on dataset HWU64 and BANKING77.

in a contrastive manner[2], while the latter transforms the semantic space into an isotropic space via matrix multiplication. When applied to off-the-shelf PLMs, both of them have been demonstrated highly effective (Gao et al., b; Su et al.), but when we apply them to fine-tuned models, they may be frustrated. As visualized in Fig. 2, contrastive learning improves the isotropy, but it significantly deteriorates the performance consistently on two large-scale datasets. As for whitening transformation, its effectiveness is data-dependent as shown in Fig. 3 – it hurts the performance on HWU64 (Fig. 3a), but yields better result on BANKING77 (Fig. 3b), although the transformation produces perfect isotropy. The above observations indicate that *isotropization may hurt fine-tuned models*, which agrees with recent findings of Rajaee and Pilehvar.

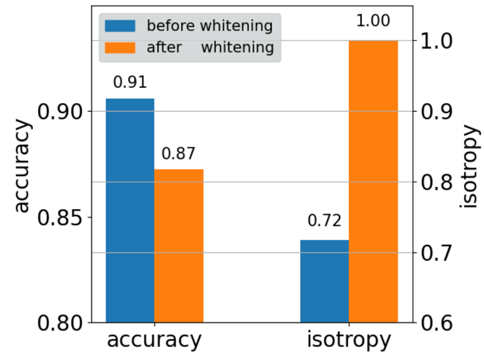
4 Method

The pilot experiment reveals the anisotropy of the fine-tuned PLM and the challenge of isotropization after fine-tuning. In this work, we propose joint fine-tuning and isotropization. Specifically, we propose two regularizers to endow the feature space with isotropy during fine-tuning. Before illustrating the technique, we first define the problem and give details of supervised pre-training.

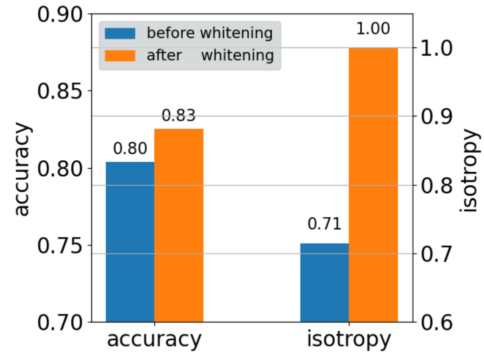
4.1 Preliminaries

Problem Definition Few-shot intent detection refers to intent classification given only a few

[2]We refer the reader to the original paper for details.



(a) HWU64



(b) BANKING77

Figure 3: The impact of whitening transformation on the IntentBERT. The transformation generates perfect isotropy on both HWU64 and BANKING77, but brings inconsistent impact on the performance.

labeled data. To tackle the problem, we leverage a dataset $\mathcal{D}_{\text{source}} = \{(x_i, y_i)\}_{N_s}$, where N_s is the number of data, x_i denotes the i_{th} utterance and y_i is the label. The target is to train a model with decent performance on another dataset $\mathcal{D}_{\text{target}} = \{(x_i, y_i)\}_{N_t}$, where N_t is the number of data. There is no overlap between the label spaces of the two datasets. Fig. 1 gives further illustrations with examples.

Supervised Pre-training Given $\mathcal{D}_{\text{source}}$, we follow Zhang et al. (a) to attach a linear layer on top of the utterance representation extracted by the PLM as the classifier:

$$\mathrm{p}(y|\mathbf{h}_i) = \text{softmax}\left(\mathbf{W}\mathbf{h}_i + \mathbf{b}\right) \in \mathbb{R}^L, \quad (3)$$

where $\mathbf{h}_i \in \mathbb{R}^d$ is the representation of the i_{th} utterance in $\mathcal{D}_{\text{source}}$, $\mathbf{W} \in \mathbb{R}^{L\times d}$ and $\mathbf{b} \in \mathbb{R}^L$ are parameters of the linear layer and L denotes the class number. The model parameters $\theta = \{\phi, \mathbf{W}, \mathbf{b}\}$, with ϕ being parameters of PLM, are trained on $\mathcal{D}_{\text{source}}$ with a cross-entropy loss:

$$\theta = \arg\min_{\theta} \mathcal{L}_{\text{ce}}\left(\mathcal{D}_{\text{source}}; \theta\right). \quad (4)$$

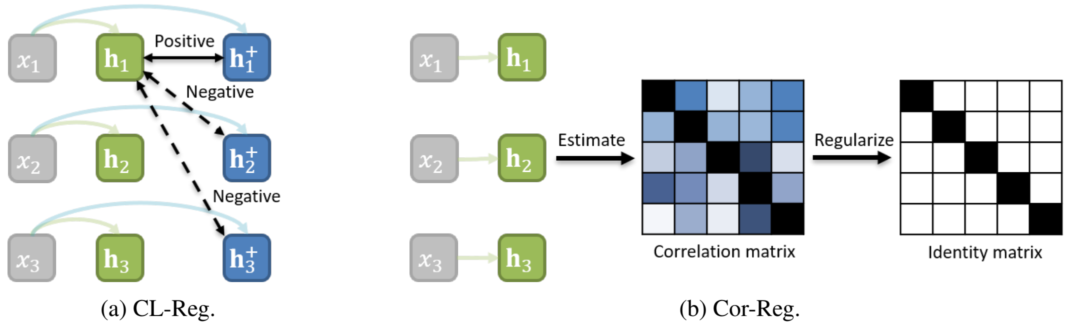


Figure 4: Illustration of CL-Reg and Cor-Reg. x_i is the i_{th} utterance in a batch of size 3. In the left figure, x_i is passed to the PLM with built-in dropout twice to produce representations $\mathbf{h}_i$ and $\mathbf{h}_i^+$, respectively. Positive and negative pairs are composed for each x_i. Take x_1 for example, its two representations $\mathbf{h}_1$ and $\mathbf{h}_1^+$ compose the positive pair, while $\mathbf{h}_1$ and the second representation (blue) of other utterances, $\mathbf{h}_2^+$ and $\mathbf{h}_3^+$, compose negative pairs. In the right figure, correlation matrix is estimated from data in the batch, and then is pushed towards identity matrix.

The fine-tuned PLM is endowed with general intent detection skills (Zhang et al., a). However, as analyzed in Section 3.2, such a process yields undesirable anisotropy. To mitigate such anisotropy, we propose two regularizers.

4.2 Contrastive-learning-based Regularizer

Inspired by the recent success of contrastive learning in mitigating the anisotropy (Yan et al.; Gao et al., b), we employ the dropout-based contrastive learning loss designed by Gao et al. (b) as the regularizer. Via minimizing the value of the regularizer, semantically close (positive) pairs are pulled together, while semantically irrelevant (negative) pairs are pushed away:

$$\mathcal{L}_{\text{reg}} = -\frac{1}{N_b}\sum_{i}^{N_b}\log\frac{e^{\text{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{h}_i,\mathbf{h}_j^+)/\tau}}, \quad (5)$$

where $\mathbf{h}_i \in \mathbb{R}^d$ and $\mathbf{h}_i^+ \in \mathbb{R}^d$ are the feature vectors of the same utterance x_i, composing the positive pair. $\mathbf{h}_i^+$ is generated via standard dropout (Srivastava et al., 2014). To be specific, utterance x_i is passed to the backbone with built-in dropout for a second time to generate $\mathbf{h}_i^+$. Dropout serves as the minimal form of data augmentation to generate $\mathbf{h}_i^+$ that differs from $\mathbf{h}_i$ only in dropout masks (Gao et al., b). $\text{sim}(\mathbf{h}_1, \mathbf{h}_2)$ denotes the cosine similarity between $\mathbf{h}_1$ and $\mathbf{h}_2$. τ is the temperature. N_b is the batch size. In Fig. 4a, we give an example where $N_b = 3$ for illustration. Gao et al. (b) applied the contrastive loss to off-the-shelf PLMs in an unsupervised scenario, while we employ it jointly with fine-tuning in a few-shot setting.

4.3 Correlation-matrix-based Regularizer

Besides the implicit contrastive-learning-based regularizer, we propose a regularizer based on the explicit characterization of isotropy. The perfect isotropy is characterized by zero covariance and uniform variance (Su et al.; Zhou et al.), i.e., a covariance matrix with uniform diagonal elements and zero non-diagonal elements. By endowing the feature space with such statistical property, we can achieve isotropization. However, as will be shown in Section 5.3, the appropriate variance value is difficult to determine. Therefore, we loose the restriction and base the regularizer on *correlation matrix*, leaving variances free to be learned. The matrix is pushed towards the identity matrix during training:

$$\mathcal{L}_{\text{reg}} = \|\mathbf{\Sigma} - \mathbf{I}\|, \quad (6)$$

where $\|\cdot\|$ denotes Frobenius norm, $\mathbf{I} \in \mathbb{R}^{d\times d}$ is the identity matrix. $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$ is the correlation matrix with $\mathbf{\Sigma}_{ij}$ denoting the Pearson correlation coefficient between the i_{th} dimension and the j_{th} dimension. As shown in Fig. 4b, $\mathbf{\Sigma}$ is estimated based on utterances in current batch.

4.4 Regularizing Supervised Pre-training with Isotropization

Ultimately, the overall loss is a combination of the cross-entropy loss $\mathcal{L}_{\text{ce}}$ and the regularizer:

$$\mathcal{L} = \mathcal{L}_{\text{ce}}(\mathcal{D}_{\text{source}};\theta) + \lambda\mathcal{L}_{\text{reg}}(\mathcal{D}_{\text{source}};\theta), \quad (7)$$

where λ is the weight, $\theta = \{\phi, \mathbf{W}, \mathbf{b}\}$ and ϕ denotes the parameters of the PLM. $\mathcal{L}_{\text{reg}}$ is implemented by either CL-Reg or Cor-Reg. Driven by the above loss function, the model learns the intent detection knowledge while maintaining an appropriate degree of isotropy. We also propose adopting the two regularizers simultaneously, which is demonstrated more effective in our experiments:

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\text{ce}}(\mathcal{D}_{\text{source}};\theta) + \lambda_1\mathcal{L}_{\text{cl}}(\mathcal{D}_{\text{source}};\theta) \\ +\lambda_2\mathcal{L}_{\text{cor}}(\mathcal{D}_{\text{source}};\theta),\end{aligned} \tag{8}$$

where λ_1 and λ_2 denote the weight, $\mathcal{L}_{\text{cl}}$ and $\mathcal{L}_{\text{cor}}$ denote CL-Reg and Cor-Reg, respectively.

4.5 Few-shot Intent Classification

After fine-tuning on $\mathcal{D}_{\text{source}}$, the linear classifier is removed, and the frozen PLM can be immediately used as a feature extractor for novel few-shot intent classification tasks. A parametric classifier can be fit with the few labeled examples and make predictions on queries. Our experiment shows that the simple logistic regression classifier, suffices to achieve promising performance, thanks to the effective utterance representations produced by the regularized supervised pre-training.

5 Experiments

To validate the effectiveness of the approach, we conduct extensive experiments.

5.1 Experimental Setup

Datasets. To train the model, we follow Zhang et al. to employ OOS (Larson et al.) which contains diverse semantics of 10 domains, providing rich resources to learn from. Domains “Banking” and “Credit Cards” are excluded to avoid semantic leakage due to the proximity of training data with test data. In the remaining domains, 6 are used for training and 2 for validation, as shown in Table 2. For evaluation, we employ three datasets: **BANKING77** (Casanueva et al.) is a fine-grained intent detection dataset focusing on “Banking”; **HINT3** (Arora et al., a) contains 3 domains, “Mattress Products Retail”, “Fitness Supplements Retail” and “Online Gaming”. **HWU64** (Liu et al.) is a large-scale dataset containing 21 domains. Dataset statistics are summarized in Table 3.

Training	Validation
“Utility”, “Auto commute”, “Work”, “Home”, “Meta”, “Small talk”	“Travel”, “Kitchen dining”

Table 2: Domain split of OOS.

Dataset	#domain	#intent	#data
OOS	10	150	22500
BANKING77	1	77	13083
HINT3	3	51	2011
HWU64	21	64	10030

Table 3: Dataset statistics.

Our Method. Our method does not presume the PLM in use. We conduct experiments on two popular PLMs, BERT (Devlin et al.) and RoBERTa (Liu et al., 2019). For both PLMs, the embedding of $[CLS]$ is used as the utterance representation in Eq. 3. We employ logistic regression as the classifier. Hyperparameters $\lambda, \lambda_1, \lambda_2$ and τ are determined by validation. The best hyperparameters are provided in Table 4.

Method	Hyperparameter
CL-Reg	$\lambda = 1.7, \tau = 0.05$
Cor-Reg	$\lambda = 0.04$
CL-Reg + Cor-Reg	$\lambda_1 = 1.7, \lambda_2 = 0.04, \tau = 0.05$

(a) BERT-based method.

Method	Hyperparameter
CL-Reg	$\lambda = 2.9, \tau = 0.05$
Cor-Reg	$\lambda = 0.06$
CL-Reg + Cor-Reg	$\lambda_1 = 2.9, \lambda_2 = 0.13, \tau = 0.05$

(b) RoBERTa-based method.

Table 4: Hyperparameters selected via validation.

Baselines. We compare our method to following strong baselines. For BERT-based baselines, **BERT-Freeze** freezes the off-the-shelf PLM; **CONVBERT** (Mehri et al., 2020), **TOD-BERT** (Wu et al.) and **DNNC-BERT** (Zhang et al., c) further pre-train BERT on conversational corpus or natural language inference tasks. **USE-ConveRT** (Henderson et al., a; Casanueva et al.) is a transformer-based dual-encoder pre-trained on conversational corpus. **CPFT-BERT** (Zhang et al., b) further pre-trains BERT in an unsupervised manner on precisely the same training data and validation data as our method. **IntentBERT** (Zhang et al., a) further pre-trains BERT via supervised pre-training described in Section 4.1. To guarantee a fair comparison, we provide **IntentBERT-ReImp**, the re-implemented version of IntentBERT, which employs the same random seed, training data and validation data as our methods. For RoBERTa-based baselines, **RoBERTa-Freeze**

Method	BANKING77		HINT3		HWU64		Val.	
	2-shot	10-shot	2-shot	10-shot	2-shot	10-shot	2-shot	10-shot
BERT-Freeze	57.10	84.30	51.95	80.27	64.83	87.99	74.20	92.99
CONVBERT¶	68.30	86.60	72.60	87.20	81.75	92.55	90.54	96.82
TOD-BERT¶	77.70	89.40	68.90	83.50	83.24	91.56	88.10	96.39
USE-ConveRT¶	–	85.20	–	–	–	85.90	–	–
DNNC-BERT¶	67.50	89.80	64.10	87.90	73.97	90.71	72.98	95.23
CPFT-BERT	72.09	89.82	74.34	90.37	83.02	93.66	89.33	97.30
IntentBERT¶	82.40	91.80	**80.10**	90.20	–	–	–	–
IntentBERT-ReImp	80.38(.35)	92.35(.12)	77.09(.89)	89.55(.63)	90.61(.44)	95.21(.15)	93.62(.38)	97.80(.18)
BERT-White	72.95	88.86	65.70	85.70	75.98	91.26	87.33	96.05
IntentBERT-White	82.52(.26)	92.29(.33)	78.50(.59)	90.14(.26)	87.24(.18)	94.42(.08)	**94.89(.21)**	98.07(.12)
CL-Reg	**83.45(.35)**	**93.66(.22)**	79.30(.87	**91.06(.30)**	**91.46(.15)**	**95.84(.12)**	94.43(.22)	**98.43.02)**
Cor-Reg	**83.94(.45)**	**93.98(.26)**	**80.16(.71)**	**91.38(.55)**	**90.75(.35)**	**95.82(.14)**	**95.02(.22)**	**98.47(.07)**
CL-Reg + Cor-Reg	**85.21(.58)**	**94.68(.01)**	**81.20(.45)**	**92.38(.01)**	**90.66(.42)**	**95.84(.19)**	**95.41(.25)**	**98.58(.01)**

Table 5: 5-way evaluation using BERT. Mean value and standard deviation are reported for our methods. CL-Reg, Cor-Reg and CL-Reg + CorReg denote supervised pre-training regularized by the corresponding regularizer. Top 3 results are highlighted. ¶ denotes results from (Zhang et al., a).

Method	BANKING77		HINT3		HWU64		Val.	
	2-shot	10-shot	2-shot	10-shot	2-shot	10-shot	2-shot	10-shot
RoBERTa-Freeze	60.74	82.18	57.90	79.26	75.30	89.71	74.86	90.52
WikiHowRoBERTa	32.88	59.50	31.92	54.18	30.81	52.47	34.10	60.59
DNNC-RoBERTa	74.32	87.30	68.06	82.34	69.87	80.22	58.51	74.46
CPFT-RoBERTa	80.27(.11)	93.91(.06)	79.98(.11)	**92.55(.07)**	83.18(.11)	92.82(.06)	86.71(.10)	96.45(.05)
IntentRoBERTa	81.38(.66)	92.68(.24)	78.20(1.72)	89.01(1.07)	**90.48(.69)**	94.49(.43)	95.33(.54)	98.32(.15)
RoBERTa-White	79.27	93.00	73.13	89.02	82.65	94.00	89.90	97.14
IntentRoBERTa-White	83.75(.45)	92.68(.31)	79.64(1.38)	90.13(.66)	86.52(1.33)	93.82(.53)	96.06(.58)	98.35(.21)
CL-Reg	**84.63(.68)**	**94.43(.34)**	**81.10(.49)**	91.65(.13)	**91.67(.20)**	**95.44(.28)**	**96.32(.14)**	**98.79(.05)**
Cor-Reg	**86.92(.71)**	**95.07(.41)**	**82.20(.48)**	**92.11(.41)**	**91.10(.18)**	**95.69(.12)**	**96.82(.03)**	**98.89(.03)**
CL-Reg + Cor-Reg	**87.96(.31)**	**95.85(.02)**	**83.55(.30)**	**93.17(.23)**	90.47(.39)	**95.64(.28)**	**96.35(.19)**	**98.85(.07)**

Table 6: 5-way evaluation using RoBERTa. Mean value and standard deviation are reported for our methods. CL-Reg, Cor-Reg and CL-Reg + CorReg denote supervised pre-training regularized by the corresponding regularizer. Top 3 results are highlighted.

freezes the model. **WikiHowRoBERTa** (Zhang et al., d) further pre-trains RoBERTa on synthesized intent detection data. **DNNC-RoBERTa** and **CPFT-RoBERTa** are similar to DNNC-BERT and CPFT-BERT except for the underlying PLM. **IntentRoBERTa** is the re-implemented version of IntentBERT based on RoBERTa, with precisely the same random seed, training data and validation data as our method. To show the superiority of the joint fine-tuning and isotropization, we compare our method to the following baselines adopting whitening transformation (Su et al.) upon fine-tuned models. **BERT-White** and **RoBERTa-White** apply the transformation to BERT and RoBERTa, respectively. **IntentBERT-White** and **IntentRoBERTa-White** apply the transformation to IntentBERT-ReImp and IntentRoBERTa, respectively.

All baselines use logistic regression as the classifier except DNNC-BERT and DNNC-RoBERTa, wherein we follow the original implementation[3] to pre-train a BERT-style pairwise encoder for nearest neighbor classification.

Training Details. We use PyTorch library and python to build the experiment flow. We employ Hugging Face implementation[4] of *bert-base-uncased* and *roberta-base*. Adam (Kingma and Ba) is used as the optimizer with learning rate of $2e-05$ and weight decay of $1e-03$. The model is trained with Nvidia RTX 3090 GPUs. Early-stop is adopted. The training is stopped if no improvement in the validation accuracy is observed for consecutive 100 steps. The same set of random seeds, $\{1, 2, 3, 4, 5\}$, are employed for IntentBERT-ReImp, IntentRoBERTa and our methods.

Evaluation. The performance is evaluated by

[3]https://github.com/salesforce/DNNC-few-shot-intent
[4]https://github.com/huggingface/transformers

C-way K-shot tasks. For each task, we randomly sample C classes and K examples per class to fit the classifier without further fine-tuning. Then we sample extra 5 examples per class as queries for evaluation. Fig. 1 gives an example where $C = 2$, $K = 1$. We report the accuracy averaged over 500 such tasks randomly sampled from $\mathcal{D}_{\text{target}}$.

5.2 Main Results

The main results are provided in Table 5 (BERT-based) and Table 6 (RoBERTa-based). First, incorporating the isotropic regularizer, either CL-Reg or Cor-Reg, consistently outperforms all baselines by a notable margin in most cases, indicating the robustness of the proposed methods against data distribution shift. The gain is attributed to a more isotropic feature space, as to be shown in Section 5.3. It is also noted that the regularization outperforms whitening transformation, demonstrating the superiority of the joint fine-tuning and isotropization. Second, Cor-Reg outperforms CL-Reg on most datasets, thanks to the explicit characterization of isotropy with the correlation matrix. Ultimately, CL-Reg and Cor-Reg show complementarity on most datasets, especially on BANKING77. The above observations are consistent for both BERT and RoBERTa. Moreover, when RoBERTa is adopted, the margin is even larger, as shown in Table 6.

Method	BANKING77	HINT3	HWU64
IntentBERT-ReImp	.71(.04)	.72(.03)	.72(.03)
SPT+CL-Reg	.77(.01)	.78(.01)	.75(.03)
SPT+Cor-Reg	.79(.01)	.76(.06)	.80(.03)
SPT+CL-Reg+Cor-Reg	.79(.01)	.76(.05)	.80(.02)

Table 7: The impact of the proposed regularizers on the isotropy. The data is collected based on BERT. SPT denotes supervised pre-training.

5.3 Analysis

CL-Reg and Cor-Reg render the feature space more isotropic. We report the impact of the proposed regularizers on isotropy in Table 7. Both regularizers manage to make the feature space more isotropic compared to IntentBERT-ReImp where only supervised pre-training is employed. In addition, Cor-Reg achieves similar or better isotropy compared to CL-Reg, which is consistent with Cor-Reg's relative superiority in performance. Interestingly, when the two regularizers are adopted simultaneously, the isotropy is not further improved, although better performance is observed.

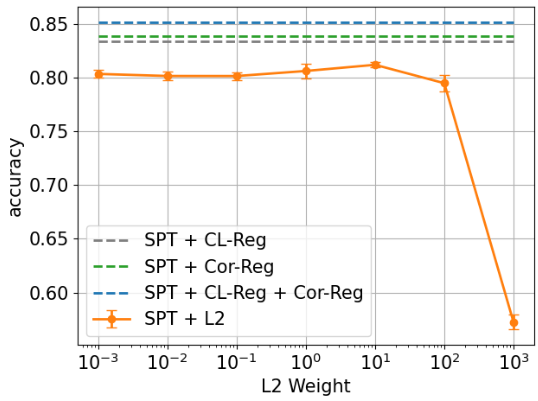


Figure 5: Comparison between proposed methods and L2 regularization. L2 regularization fails to achieve comparable performance with our methods. The data is collected based on BERT with 5-way 2-shot tasks on BANKING77. SPT denotes superivsed pre-training.

The gain is not from the reduction in the model variance. Regularization techniques such as L1 regularization (Tibshirani, 1996) and L2 regularization (Hoerl and Kennard, 1970) are popularly employed to improve model's performance via the reduction in model's variance. We argue that the gain of the proposed regularization method is ascribed to the improved isotropy (Table 7) rather than the reduction in the model variance. To demonstrate it, we compare our method against L2 regularization with a wide range of weights, and it is observed that merely reducing model variance cannot yield performance that is comparable to our method, as shown in Fig. 5.

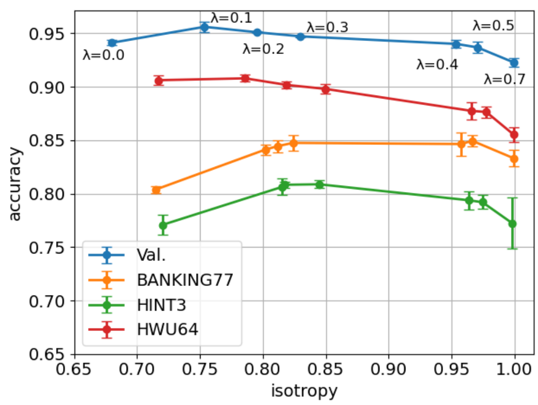


Figure 6: The relation between performance and isotropy. The experiment employs BERT and adopt 5-way 2-shot tasks for the evaluation.

Moderate isotropy is helpful. To investigate the relation between isotropy and performance, we tune the weight on Cor-Reg to yield different degrees of isotropy and examine the corresponding performance. As shown in Fig. 6, a typical pattern is observed for most datasets – the best performance is achieved only when the isotropy is moderate. This observation indicates an appropriate trade-off between the knowledge learned by supervised pre-training and isotropy, which is achieved by our method.

Correlation matrix is advantageous over covariance matrix. In the design of Cor-Reg (Section 4.3), the correlation matrix, rather than the covariance matrix, is employed to characterize isotropy, although the latter contains more information – variance. We argue that the design choice is advantageous since it is hard to determine the variance scale. We conduct experiments using the covariance matrix, pushing the non-diagonal elements (covariances) towards 0, and the diagonal elements (variances) towards 1, 0.5 and the mean value, respectively, which are denoted by Cov-Reg-1, Cov-Reg-0.5 and Cov-Reg-mean in Table 8. As shown in the table, all these configurations underperform Cor-Reg.

Method	BANKING77	Val.
Cov-Reg-1	82.19(.84)	94.52(.19)
Cov-Reg-0.5	82.62(.80)	94.52(.26)
Cov-Reg-mean	82.50(1.00)	93.82(.39)
Cor-Reg (ours)	**83.94(.45)**	**95.02(.22)**

Table 8: The advantage of Cor-Reg over the covariance-matrix-based design. The experiment employs BERT and 5-way 2-shot evaluation.

Method	BANKING77	Val.
SPT+BatchNorm	82.38(.38)	94.78(.24)
SPT+CL-Reg	83.45(.35)	94.43(.22)
+BatchNorm	**84.18(.28)**	**95.10(.20)**
SPT+Cor-Reg	83.94(.45)	95.02(.22)
+BatchNorm	**84.67(.51)**	**95.22(.18)**
SPT+Cor-Reg+CL-Reg	85.21(.58)	95.41(.25)
+BatchNorm	**85.64(.41)**	**95.57(.25)**

Table 9: The complementarity of the proposed methods and batch normalization. The experiment employs BERT and 5-way 2-shot evaluation. SPT denotes supervised pre-training. BatchNorm denotes batch normalization.

The proposed methods are complementary with batch normalization. Batch normalization (Ioffe and Szegedy) can potentially mitigate the anisotropy via normalizing each dimension into a scalar with unit variance. We find that combining the proposed method with batch normalization yields better performance, as shown in Table 9.

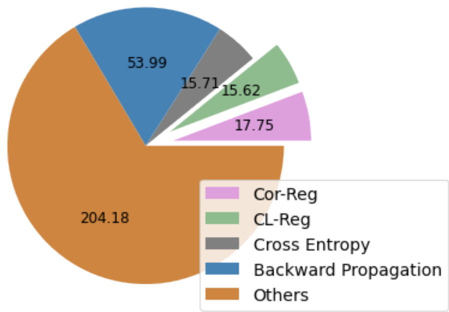


Figure 7: Run time decomposition of a single epoch. The unit is second.

The computational overhead is small. We decompose the duration of one epoch to analyze the computational overhead incurred by CL-Reg and Cor-Reg. As shown in Fig. 7, the overhead occupies a small proportion in the entire epoch, consuming around 17.75 seconds and 15.62 seconds, respectively. Thus, both regularizers cause acceptable computational overhead. We leave the overhead optimization as future work.

6 Conclusion

We identify the anisotropy of the feature space rendered by supervised pre-training for few-shot intent detection. To mitigate this issue, we propose two regularizers that notably improve the performance by regularizing the feature space towards isotropy. Combining them yields better performance on most datasets. The study may have a broad implication for other tasks besides intent detection, to which fine-tuning PLMs is a solution.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments. This research was supported by the grants of HK ITF UIM/377 and PolyU DaSAIL project P0030935 funded by RGC.

References

Gaurav Arora, Chirag Jain, Manas Chaturvedi, and Krupal Modi. a. HINT3: Raising the bar for intent detection in the wild. In *EMNLP, 2020*.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. b. A latent variable model approach to PMI-based word embeddings. *TACL, 2016*, 4:385–399.

Daniel Biś, Maksim Podkorytov, and Xiuwen Liu. Too much in common: Shifting of embeddings in transformer language models and its implications. In *NAACL, 2021*.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2020. Isotropy in the contextual embedding space: Clusters and manifolds. In *ICLR, 2020*.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In *ACL, 2020*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL, 2019*.

Thomas Dopierre, Christophe Gravier, and Wilfried Logerais. a. ProtAugment: Intent detection meta-learning through unsupervised diverse paraphrasing. In *ACL-IJCNLP, 2021*.

Thomas Dopierre, Christophe Gravier, Julien Subercaze, and Wilfried Logerais. b. Few-shot pseudo-labeling for intent detection. In *COLING, 2020*.

Kawin Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *EMNLP-IJCNLP, 2019*.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML, 2017*.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. a. Representation degeneration problem in training natural language generation models. In *ICLR 2019*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. b. SimCSE: Simple contrastive learning of sentence embeddings. In *EMNLP, 2021*.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. Induction networks for few-shot text classification. In *EMNLP-IJCNLP, 2019*.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Tsung-Hsien Wen, and Ivan Vulić. a. ConveRT: Efficient and accurate conversational representations from transformers. In *Findings of EMNLP 2020*, Online.

Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. b. Training neural response selection for task-oriented dialogue systems. In *ACL, 2019*.

Arthur E Hoerl and Robert W Kennard. 1970. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Yutai Hou, Yongkui Lai, Yushan Wu, Wanxiang Che, and Ting Liu. Few-shot learning for multi-label intent detection. *AAAI, 2021*.

Junjie Huang, Duyu Tang, Wanjun Zhong, Shuai Lu, Linjun Shou, Ming Gong, Daxin Jiang, and Nan Duan. WhiteningBERT: An easy unsupervised sentence embedding approach. In *Findings of EMNLP, 2021*.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML, 2015*.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR 2015*.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. An evaluation dataset for intent classification and out-of-scope prediction. In *EMNLP-IJCNLP, 2019*.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. On the sentence embeddings from pre-trained language models. In *EMNLP, 2020*.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. Benchmarking natural language understanding services for building conversational agents. In *IWSDS,2019*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *arXiv preprint arXiv:2009.13570*.

Jiaqi Mu and Pramod Viswanath. All-but-the-top: Simple and effective postprocessing for word representations. In *ICLR 2018*.

Hoang Nguyen, Chenwei Zhang, Congying Xia, and Philip Yu. Dynamic semantic matching and aggregation network for few-shot intent detection. In *Findings of EMNLP 2020*.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. *TACL*, 9:807–824.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. Few-shot natural language generation for task-oriented dialog. In *Findings of EMNLP 2020*.

Sara Rajaee and Mohammad Taher Pilehvar. How does fine-tuning affect the geometry of embedding space: A case study on isotropy. In *Findings of EMNLP 2021*.

Sara Rajaee and Mohammad Taher Pilehvar. 2021a. A cluster-based approach for improving isotropy in contextual embedding space. In *ACL-IJCNLP, 2021*.

Sara Rajaee and Mohammad Taher Pilehvar. 2021b. An isotropy analysis in the multilingual bert embedding space. *arXiv preprint arXiv:2110.04504*.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS, 2017*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(56):1929–1958.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc., Series B (Methodological)*, 58(1):267–288.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NeurIPS, 2016*.

Ivan Vulić, Pei-Hao Su, Samuel Coope, Daniela Gerz, Paweł Budzianowski, Iñigo Casanueva, Nikola Mrkšić, and Tsung-Hsien Wen. ConvFiT: Conversational fine-tuning of pretrained language models. In *EMNLP, 2021*.

Lingxiao Wang, Jing Huang, Kevin Huang, Ziniu Hu, Guangtao Wang, and Quanquan Gu. Improving neural language generation with spectrum control. In *ICLR, 2019*.

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *EMNLP, 2020*.

Congying Xia, Caiming Xiong, Philip Yu, and Richard Socher. a. Composed variational natural language generation for few-shot intents. In *Findings of EMNLP 2020*.

Congying Xia, Wenpeng Yin, Yihao Feng, and Philip Yu. b. Incremental few-shot text classification with multi-round new classes: Formulation, dataset and system. In *NAACL, 2021*.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. ConSERT: A contrastive framework for self-supervised sentence representation transfer. In *ACL-IJCNLP, 2021*.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. Diverse few-shot text classification with multiple metrics. In *NAACL, 2018*.

Haode Zhang, Yuwei Zhang, Li-Ming Zhan, Jiaxin Chen, Guangyuan Shi, Xiao-Ming Wu, and Albert Y.S. Lam. a. Effectiveness of pre-training for few-shot intent classification. In *Findings of EMNLP 2021*.

Jianguo Zhang, Trung Bui, Seunghyun Yoon, Xiang Chen, Zhiwei Liu, Congying Xia, Quan Hung Tran, Walter Chang, and Philip Yu. b. Few-shot intent detection via contrastive pre-training and fine-tuning. In *EMNLP, 2021*.

Jianguo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip Yu, Richard Socher, and Caiming Xiong. c. Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In *EMNLP, 2020*.

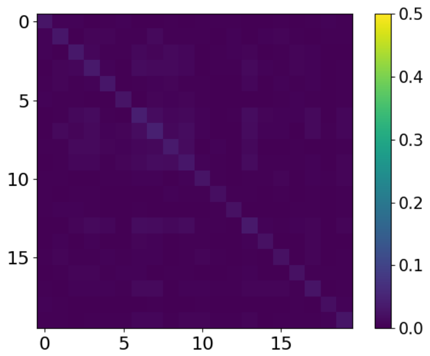
Li Zhang, Qing Lyu, and Chris Callison-Burch. d. Intent detection with WikiHow. In *AACL, 2020*.

Wenxuan Zhou, Bill Yuchen Lin, and Xiang Ren. Isobn: Fine-tuning bert with isotropic batch normalization. In *AAAI, 2021*.

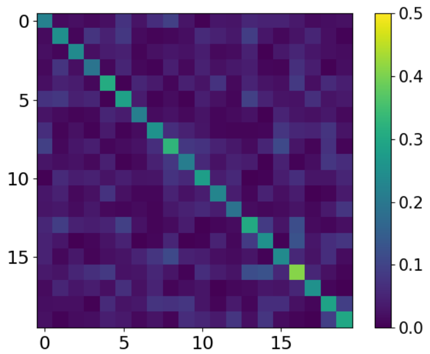
A Appendix

A.1 More Results of Pilot Experiments

To show the impact of fine-tuning on isotropy, we visualize the covariance matrix of the feature space in Fig. 8. It is found that after fine-tuning, the covariance is strengthened in general. Pefect isotropy requires uniform variance and zero covariance, and thus the changes in the covariance matrix agrees with the isotropy measurement presented in Table 1.



(a) BERT



(b) IntentBERT

Figure 8: Covariance matrix of the first 20 dimensions of the feature space. Absolute values in the matrices are visualized. Data is collected on BANKING77.