DICS-Net: Dictionary-guided Implicit-Component-Supervision Network for Few-Shot Classification

Shuai Shao, Lei Xing, Weifeng Liu, Yanjiang Wang, Baodi Liu

Abstract—The few-shot classification (FSC) task has recently been a hot research topic. It aims to address the classification problem with insufficient labeled data on a cross-category basis. Typically, researchers pre-train a feature extractor with base data, then use it to extract the features of novel data and recognize them. Notably, the novel set only has a few annotated samples and has non-overlapped categories from the base set, which leads to the pre-trained feature extractor not adapting to the novel data flawlessly. We dub this problem as Feature-Extractor-Maladaptive (FEM) problem. Starting from the root cause of this problem, this paper presents a new scheme, Dictionary-guided Implicit-Component-Supervision Network (DICS-Net), to improve the performance of FSC. We believe that although the categories of base and novel sets are different, the composition of the sample's components is similar. For example, both cats and dogs contain leg and head components. Actually, such entity components are intra-class stable. They have fine cross-category versatility and new category generalization. However, in many real-world scenarios, common information of different categories (such as cats and airplanes) is not easy to find, which hinders the possibility of modeling based on this assumption. Therefore, we first design a Dictionary-based Implicit-Component Generator (DICG) to mine common information of different sets; then construct an implicit-component-based auxiliary task to improve the adaptability of the feature extractor. We conduct experiments on three benchmark datasets (mini-ImageNet, tiered-ImageNet, and FC100). The improvements of 0.9%-10.1% compared with state-of-the-arts have evaluated the efficiency of our DICS-Net.

Index Terms—few-shot classification (FSC), Feature-Extractor-Maladaptive (FEM) problem, Dictionary-guided Implicit-Component-Supervision Network (DICS-Net), Dictionary-based Implicit-Component Generator (DICG).

I. INTRODUCTION

In industrial applications, classification tasks have always been the focus of attention. Convolutional Neural Network (CNN) based methods in traditional classification tasks have achieved fantastic performance, such as [1]–[3]. Their successes are inseparable from the support of rich and accessible labeled data. However, collecting labeled data is a heavy-priced work or even impractical. Therefore, a new investigation – Few-Shot Classification (FSC), targets to address this problem with scarce labeled samples, has attracted growing attention in recent years. In a standard FSC paradigm, the employed data includes two parts, i.e., base set and novel set. There are many labeled samples in the base set, but very few in the novel set (typically, for the general FSC setting, each novel category only has 1 or 5 labeled samples). We need to pre-train a feature extractor through the base set, then employ it to extract features of novel data, and finally design a classifier to recognize the novel data’s category. Notably, the categories contained in the base set are non-overlapped from those in the novel set.

From the setting of FSC, we observe that one of the essential factors affecting the performance is that: the pre-trained feature extractor (based on the base set) can not adapt to the novel data flawlessly due to the different categories between the base set and novel set. Precisely, the feature extractor captures the novel feature according to the focus trend on the base data, which causes the novel feature to have weak discrimination. We dub this problem as Feature-Extractor-Maladaptive (FEM) problem.

Sound like fine-tuning the pre-trained network for novel data is a suitable way to deal with the challenge. But unfortunately, [4] has demonstrated that this strategy with few labeled samples easily causes the overfitting problem. Hence, it usually provides limited improvements for FSC or even negatively affects the results. Thus researchers usually give up fine-tuning and choose the decoupled paradigm, i.e., freezing the feature extractor after pre-training and directly extracting novel features. As for the other recent proposed classical algorithms, such as metric-learning based methods [5], graph based methods [6], they have not focused on this particular issue. Therefore, it is necessary to develop a dedicated approach for the specific FEM problem.

In this paper, we attempt to address the FEM problem from the root cause, that is, assigning the same labels to different sets. How to achieve this purpose? Let’s think from nature, if we decompose some objects, it’s not hard to find that the components that make up different objects usually have many similarities. For example, suppose dogs are in the base set, and cats are in the novel set. Although their categories are different, they all have some same components, such as head, body, and leg. We illustrate an example in Figure 1(a). If we define these components as labels, then the class information of base and novel will have a lot of overlapping parts, so we can achieve our goal.

How to generate generalized component-based multi-labels? The most intuitive way is to directly obtain the real component information in each class of samples through manual statistics or some toolkits. But this approach has a fatal flaw: it only makes sense when the base and novel classes are not much different. However, this assumption is difficult to hold in
real applications, as shown in the Figure 1(b); therefore, this intuitive method is difficult to work with.

But if we think deeply: do samples of different categories with deep gaps really have no common information? In our opinion, the answer is negative. Here is an extreme example: just like the cat, dog, flower, and plane shown in Figure 1(b), if we trace the source, they are actually composed of hydrogen atoms, carbon atoms, etc., so we can use hydrogen atoms and carbon atoms as the common components. Therefore, this paper proposes a hypothesis that those very different samples still have some implicit common information, dubbed as Implicit-Component. Then we propose the Dictionary-guided Implicit-Component-Supervision Network (DICS-Net) to reasonably use them to improve the generalization ability of the pre-trained model. There are two strategies in our DIML-Net that focus on two key challenges: (1) How to achieve the implicit-component? (2) How to use the implicit-component?

(1) To overcome the first challenge, we design a simple-yet-effective module, Dictionary-based Implicit-Component Generator (DICG), which consists of two stages. We first use the pre-trained BERT model [7] to encode the base data’s class labels to the semantic embeddings. After that, we introduce the popular dictionary learning theory [8] to mine the base data’s implicit-component information. Dictionary learning is a classic representation learning method that can help to mine critical information from data. In this task, we map the semantic embeddings to the dictionary space to learn their base vectors and re-represent them through the base vectors. The base vectors can be treated as the implicit-component information. We illustrate the flowchart in Figure 2. Notably, the base and novel data share the same implicit-component and the insight behind is that: The semantic embeddings obtained from the BERT model [7] are all independent and identically distributed (i.i.d.). That is to say, we can pull the known base category label and the unknown novel category label into the same semantic space, so the base and novel semantic embeddings can share the same implicit-component.

(2) To solve the second problem, we present an Implicit-Component-Supervised Auxiliary Task. Here, we define the generated implicit-components as the multi-label information for base samples and introduce the multi-label classification loss as the auxiliary loss to update the network. A flowchart is illustrated in Figure 3. Besides, in order to control the impact of multi-label accuracy in the paper, we introduce an empirical parameter for our implicit-component-supervised auxiliary loss. We believe that when the accuracy of generated-multi-label is sufficiently high, this parameter should be scaled large for larger loss weight and conversely small.

In summary, the main contributions focus on:

- We point out that the feature-extractor-maladaptive (FEM) problem is one of the core issues that affect the few-shot classification (FSC). In response, we put forward a new point of view: the components that make up different categories are similar. Inspired by this assumption, we design the Dictionary-guided Implicit-Component-Supervision Network (DICS-Net) to improve the adaptability of the feature extractor.
- In order to construct discriminative component-based multi-labels, we design a Dictionary-based Implicit-Component Generator for querying the components of the target. In addition, a simple-yet-effective auxiliary task is present to reasonably employ the implicit-components.
- We evaluate our DICS-Net on three benchmark datasets (mini-ImageNet, tiered-ImageNet, FC100) and achieve significant improvements of at least 0.9%-10.1% compared with other state-of-the-art methods, which has demonstrated the efficiency of our method. We design experiments to analyze our method and evaluate that our DICS-Net is a simple, easy-to-implement, and easy-to-understand method.

II. RELATED WORK

A. Few-Shot Classification

The work of few-shot classification (FSC) is one of the most concerning issues at present. Researchers usually pay efforts from two perspectives to solve it. On the one hand, hoping the designed feature extractor is robust enough to achieve a feature extracting process on cross-category situation. Many classical strategies have been proposed, for examples, [9] introduce knowledge distillation; [10] introduce self-supervision; [5] introduce metric-learning. On the other hand, strengthening
the final classifier and make it ignore the influence of inaccurate feature representation to some extent, as examples, [11] introduce distribution calibration; [12] introduce self-training. Our DICS-Net can be viewed as focusing on the first point. We introduce implicit-component-based multi-labels to make our feature extractor more adaptable for novel data. For the final classifier, we just employ one of the simplest one, shown in the following section.

B. Label-Semantics-Related Few-Shot Classification

Some recent works also use the label semantics information, such as [13]–[16]. The similarity between our method with them is that: they all use the label semantic information. But on (1) the way of application, and (2) the way to obtain label semantics are very different from each other. (1) On the way of application: These methods all need to design an extra word2vector network to convert the label semantic information into label embedding, and then splices it with visual embedding. While our DICS-Net directly uses the label semantics, no need for additional networks, only need to design a multi-class loss in the pre-train network. (2) On the way to obtain label semantics: In the comparison methods, each sample has a variety of labels. These labels are manually annotated, which are very accurate but difficult to obtain. Therefore, these methods lack universality. While in our DICS-Net, we combine the pre-trained BERT [7] with dictionary learning to generate new labels for samples of the same category. The accuracy of the generated multi-labels is definitely far from the manually annotated labels, but it greatly reduces the labor cost and makes our method more practical.

C. Dictionary Learning

Dictionary learning can be seen as the most popular machine learning theory, which was first proposed by [8]. It is capable of mapping samples’ original feature embeddings to the dictionary space. Using the reconstructive dictionary bases to represent samples is helpful to reduce the redundant information of samples. Many strategies have been proposed to improve the quality of the to-be-learned dictionary, such as using non-negative constraint and self-taught learning [17]; designing the fisher discrimination criterion [18] to learn a structured dictionary. Each kind of method has its advantages and limitations. This article merely uses the principle dictionary learning framework to encode the labels.

III. METHODOLOGY

In this section, we first introduce the setting of few-shot classification and split the complete framework into two phases, pre-training and meta-testing; then design the Dictionary-based Implicit-Component Generator (DICS-G) to capture the common information between base and novel sets; next propose our Dictionary-guided Implicit-Component-Supervision Network (DICS-Net) as the feature extractor for pre-training; finally introduce the to-be-used classifier in meta-testing phase for few-shot classification.

A. Problem Setup

In the few-shot classification (FSC) task, we have two kinds of datasets, i.e., base set $D_b = \{(x_i, y_i) | y_i \in \mathcal{C}_b\}_{i=1}^{N_b}$ and novel set $D_n = \{(x_j, y_j) | y_j \in \mathcal{C}_n\}_{j=1}^{N_n}$, where $x$ and $y$ indicate the sample and corresponding label; $N_b$ and $N_n$ denote the total number of base data and novel data; $\mathcal{C}_b$ and $\mathcal{C}_n$ denote the base category and novel category. Besides, $\mathcal{C}_b \cap \mathcal{C}_n = \emptyset$. Note that there exist a large number of labeled images in $D_b$ and only a few labeled images in $D_n$.

The FSC task consists of two phases: (1) Pre-training. Researchers employ the base data $D_b$ to pre-train a CNN-based model, which consists of feature extractor $\mathcal{J}(\cdot)$ and base classifier $\mathcal{M}_b(\cdot)$. (2) Meta-testing. We freeze the pre-trained feature extractor and utilise it to extract the feature of novel data $D_n = \{S, Q\}$, where $S$ and $Q$ denote support set (labeled training samples), and query set (to-be-classified testing samples). $S \cap Q = \emptyset$. Researchers split the FSL into two settings: transductive setting and inductive setting. The former needs to know the query data in advance when designing the classifier, this setting usually goes against most real application scenarios. On the contrary, the latter does not need to know the query information in advance, so this article...
only focuses on this setting. In classification stage, we follow the standard paradigm with \( C \)-way-\( N \)-shot per episode as [12], where \( C \)-way denotes \( C \) classes, and \( N \)-shot indicates \( N \) samples per class. The reported results are the average of accuracies of all the episodes with 95\% confidence intervals.

B. Dictionary-based Implicit-Component Generator

As mentioned above, one of the key challenges of this paper is how to mine the implicit common information of non-overlapped categories of samples. This section proposes the Dictionary-based Implicit-Component Generator (DICG) to solve this problem, which consists of two steps. The flowchart is shown in Figure 2.

(i) Collecting their semantic labels and sending them into a pre-trained BERT model [7] to get their semantic embeddings, which can be defined as \( \mathbf{L} \in \mathbb{R}^{\text{dim}_L \times C_b} \), where \( \text{dim}_L \) denotes the dimension size, \( C_b \) denotes the base class number.

(ii) Introducing dictionary learning theory [8] to obtain the implicit-components, which can be formulated as:

\[
\begin{align*}
\arg\min_{\mathbf{U}, \mathbf{V}} & \quad \| \mathbf{L} - \mathbf{U} \mathbf{V} \|_F^2 + 2\gamma \| \mathbf{V} \|_{\ell_1} \\
\text{s.t.} & \quad \| \mathbf{u}_k \|_2^2 \leq 1 \quad (k = 1, 2, \ldots, K)
\end{align*}
\]

where \( \mathbf{U} \in \mathbb{R}^{\text{dim}_L \times K} \) denotes the to-be-learned base vectors (i.e., dictionary), corresponds to the implicit-components, \( K \) denotes the number of implicit-components; \( \mathbf{V} \in \mathbb{R}^{K \times C_b} \) represents the sparse codes for dictionary, also denotes the labels’ reconstructed representations based on the base vectors; \( \gamma \) is a hyperparameter, which is a positive scalar constant.

We update \( \mathbf{U} \) and \( \mathbf{V} \) alternately until the objective function does not descend. \( \mathbf{V} \) can be solved as:

\[
\mathbf{V}_{kc} = \frac{\max (\mathbf{J}, \gamma) + \min (\mathbf{J}, \gamma)}{(\mathbf{U}^T \mathbf{U})_{kk}}
\]

where

\[
\mathbf{J} = (\mathbf{U}^T \mathbf{L})_{kc} - \sum_{l=1, l\neq k}^{K} \left((\mathbf{U}^T \mathbf{U})_{kl}\right) \mathbf{V}_{lc}
\]

Then, the BCD [19] is introduced to update \( \mathbf{U} \) as:

\[
\mathbf{U}_{*k} = \frac{\mathbf{L} (\mathbf{V}_{*k})^T - \tilde{\mathbf{U}}^k \mathbf{V} (\mathbf{V}_{*k})^T}{\| \mathbf{L} (\mathbf{V}_{*k})^T - \tilde{\mathbf{U}}^k \mathbf{V} (\mathbf{V}_{*k})^T \|_2}
\]

where \( \tilde{\mathbf{U}} = \left\{ \begin{array}{ll} \mathbf{U}_p & p \neq k \\ 0 & p = k \end{array} \right. \), \( 0 \) denotes zero matrix.

C. Dictionary-guided Implicit-Component-Supervision Network

In this section, we propose the Dictionary-guided Implicit-Component-Supervision Network (DICS-Net) for the few-shot classification. To simply and effectively use the generated implicit-components, we design a implicit-component-supervised auxiliary task, and integrate it with a standard classification task. We illustrate a flowchart in Figure 3 and introduce the details as follows.

1) Standard Classification Task: Assume we have a base image \( x \). Input it to the pre-trained feature extractor \( \mathcal{F}(\cdot) \) and achieve the feature vector \( \mathbf{x} = \mathcal{F}(x) \in \mathbb{R}^{\text{dim}} \), where \( \text{dim} \) denotes the dimension of the feature vector. Next, input the feature into the base classifier \( \mathcal{M}_b(\cdot) \) to predict the soft label. There are lots of choices, such as Support Vector Machine, Nearest Neighbor, Softmax. In the standard classification task, we select the softmax activation function as our classifier. Specifically, we project the feature vector into a label space, i.e., \( x \rightarrow \mathbf{z} \), where \( \mathbf{z} = [z_1, z_2, \cdots, z_{C_b}] \in \mathbb{R}^{C_b} \), \( C_b \) denotes the number of base category. Then transform it to the probability distribution by:

\[
y^{sc}_c = \frac{e^{z^{sc}_c}}{\sum_{c=1}^{C_b} e^{z^{sc}_c}}
\]

where \( \mathbf{y}^{sc} = [y^{sc}_1, y^{sc}_2, \cdots, y^{sc}_{C_b}] \in \mathbb{R}^{C_b} \) can be viewed as the predicted soft label of sample \( x \) in the standard classification task. Introduce categorical cross entropy function \( \mathcal{F}_{\text{cc}} \) to calculate the standard classification loss \( \mathcal{L}^{sc} \) as:

\[
\mathcal{L}^{sc}(x, \mathbf{y}^{sc}) = \mathcal{F}_{\text{cc}}(\mathcal{M}_b(\mathcal{F}(x)), \mathbf{y}^{sc})
\]

\[
= \mathcal{F}_{\text{cc}}(\mathbf{y}^{sc}, \mathbf{\hat{y}}^{sc})
\]

where \( \mathbf{\hat{y}}^{sc} = [\hat{y}^{sc}_1, \hat{y}^{sc}_2, \cdots, \hat{y}^{sc}_{C_b}] \in \mathbb{R}^{C_b} \) denotes the one-hot truth label vector of \( x \) in the standard classification task.

2) Implicit-Component-Supervised Auxiliary Task: As described before, the novel set \( D_n \) has totally different categories with the base set \( D_b \). Therefore, the pre-trained feature extractor \( \mathcal{F}(\cdot) \) (based on base set) is not applicable for the novel set. But fortunately, the components that make up samples of different categories are similar, just like both dogs and cats have legs, body, head (some examples are shown in Figure 1). Inspired by this assumption, we construct new implicit-component-based labels for the base set. Since each sample has multiple components, the problem can be diverted to a multi-label classification task.

From DICG, we collect \( K \) implicit-components and generate the multi-labels. Each sample belongs to a part of the multi-labels. In this implicit-component-supervised auxiliary task,
Algorithm 1: DICS for FSC

Input: Base set \( D_b \), Base category set \( C_b \), Novel set \( D_n = \{ \mathcal{S}, \mathcal{Q} \} \).

Output: Query label

1. Sending the base category labels \( C_b \) to the pre-trained BERT model to get their semantic embeddings \( L \).
2. Employing the dictionary encoder to obtain the implicit-components \( U \) and reconstructive representation \( V \) by Equation (1).
3. Designing the feature extractor \( \mathcal{J}(\cdot) \) through \( D_b \) and \( V \).
4. Obtaining novel data feature by \( X_s = \mathcal{J}(\mathcal{S}) \), \( X_q = \mathcal{J}(\mathcal{Q}) \).
5. Training the classifier \( \mathbf{W} \) by Equation (11).
6. Utilizing the optimal classifier to predict the query label by Equation (12).

we choose the sigmoid activation function as the classifier for base data. Similar as the standard classification task, we project the feature vector into a implicit-component-based label space, i.e., \( x \rightarrow z^{cs} \), where \( z^{cs} = [z^{cs}_1, z^{cs}_2, \ldots, z^{cs}_K] \in \mathbb{R}^K \), then transform it to the binomial distribution by:

\[
y^{cs}_k = \frac{1}{1 + e^{-y^{cs}_k}}
\]

where \( y^{cs} = [y^{cs}_1, y^{cs}_2, \ldots, y^{cs}_K] \in \mathbb{R}^K \) can be regarded as the predicted soft multi-label of sample \( x \). Then compute the component-supervised auxiliary loss \( L^{cs} \) by introducing binary cross entropy function \( \mathcal{F}_{bce} \), which can be formulated as:

\[
\begin{align*}
L^{cs}(x, \hat{y}^{cs}) &= \mathcal{F}_{bce}(\mathcal{M}_b(\mathcal{J}(x)), \hat{y}^{cs}) \\
&= \mathcal{F}_{bce}(\hat{y}^{cs}, \hat{y}^{cs}) \\
&= -\sum_{k=1}^{K} \hat{y}^{cs}_k \log(y^{cs}_k) + (1 - \hat{y}^{cs}_k) \log(1 - y^{cs}_k)
\end{align*}
\]

where \( \hat{y}^{cs} = [\hat{y}^{cs}_1, \hat{y}^{cs}_2, \ldots, \hat{y}^{cs}_K] \in \mathbb{R}^K \) denotes the truth component-based multi-label vector of \( x \).

3) Overall Loss Function: Finally, we achieve the resultant loss for our DICS-Net, which can be simply formulated as:

\[
L = L^{sc} + \alpha L^{cs}
\]

where \( \alpha \) is an empirical parameter to control the influence of implicit-component-supervised auxiliary loss. We evaluate the influence of the parameter in Section IV-D.

D. Few-Shot Classification

Through Equation (9), the feature extractor \( \mathcal{J}(\cdot) \) is more suitable for the novel data than before. Send the novel data to \( \mathcal{J}(\cdot) \) and achieve novel feature \( X = \mathcal{J}(D_n) \), where \( X = [X_s, X_q] \). \( X_s = \mathcal{J}(\mathcal{S}) \in \mathbb{R}^{\text{dim} \times N_s} \) denotes the support feature, \( X_q = \mathcal{J}(\mathcal{Q}) \in \mathbb{R}^{\text{dim} \times N_q} \) denotes the query feature, \( N_s \) and \( N_q \) indicate the number of support and query data, \( \text{dim} \) represents the dimension of samples. Next, we try to construct a novel classifier \( \mathcal{M}_n(\cdot) \). Here, we only consider the inductive setting.

Multiple kinds of traditional classifiers are suitable, such as support vector machine, linear regression, logistic regression. We select the linear regression as the example to introduce our method. We formulate the objective function as:

\[
\arg\min_{\mathbf{W}} ||Y_s - \mathbf{W}X_s||_F^2 + \beta ||\mathbf{W}||_F^2
\]

where \( Y_s \in \mathbb{R}^{C_n \times N_s} \) denotes the truth one-hot label matrix of support data, \( C_n \) denotes the number of novel categories; \( \mathbf{W} \in \mathbb{R}^{C_n \times \text{dim}} \) represents the to-be-learned classifier. We directly optimize the objective function and obtain the \( \mathbf{W} \) as:

\[
\mathbf{W} = Y_sX_s^T \left(X_sX_s^T + \beta I\right)^{-1}
\]

where \( I \) denotes the diagonal matrix.

Then, we use the \( \mathbf{W} \) to classify the category of \( X_q \) by:

\[
Y_q = \mathbf{W}X_q
\]

IV. EXPERIMENTS

This section introduces the benchmark datasets briefly and demonstrates the experimental implementation in detail. Then, we demonstrate the experimental results and discuss them. Finally, we design ablation experiments to analyze the performance influence factors. All experiments are conducted on a Tesla-V100 GPU with 32G memory.

A. Dataset

We evaluate our methods on three FSC benchmark datasets, i.e., mini-ImageNet, tiered-ImageNet, and FC100. Both of mini-ImageNet and tiered-ImageNet are the subset of ImageNet dataset. For the mini-ImageNet, it consists of 100 classes and each class has 600 images with the size of 84 × 84. In experiments, we split the datasets as [12], and use 64, 16, 20 classes as the base set, validation set, novel set for mini-ImageNet, respectively. For the tiered-ImageNet, it has 608 classes and each class contains 1, 281 images on average. Same as mini-ImageNet, the images are resized to 84 × 84. In the experiments, 351, 97, and 160 classes are selected as the base set, validation set, and novel set, respectively. Besides, researchers grouped the tiered-ImageNet dataset into 34 high-level categories, and each one has unbalanced classes, and the base, validation, and novel set come from different high-level categories. The FC100 is the subsets of the CIFAR-100 dataset [29], which includes 100 classes. We divided it into 60 classes as the base set, the validation set contains 20 classes, and the novel set includes 20 classes. The image size of FC100 dataset is set to 32 × 32.
TABLE I: Comparison results with state-of-the-art methods in mini-ImageNet and tiered-ImageNet. The reported accuracies are in 95% confidence intervals over 600 episodes with inductive setting. The top two results are shown in **bold** and **underline**, respectively.

| Method      | Backbone | mini-ImageNet       | tiered-ImageNet     |
|-------------|----------|---------------------|---------------------|
|             |          | 5-way 1-shot        | 5-way 5-shot        | 5-way 1-shot | 5-way 5-shot |
| MTL [20]    | ResNet12 | 61.20 ± 1.80        | 75.50 ± 0.80        | -           | -            |
| TADAM [21]  | ResNet12 | 58.50 ± 0.30        | 76.70 ± 0.30        | -           | -            |
| SLA-AG [22] | ResNet12 | 62.93 ± 0.63        | 79.63 ± 0.47        | -           | -            |
| Meta-Base [23] | ResNet12 | 63.17 ± 0.23        | 79.26 ± 0.17        | 68.62 ± 0.27 | 83.74 ± 0.18 |
| MetaOpt [24] | ResNet12 | 62.64 ± 0.61        | 78.63 ± 0.46        | 65.99 ± 0.72 | 81.56 ± 0.53 |
| NCA [25]    | ResNet12 | 62.55 ± 0.12        | 78.27 ± 0.09        | 68.35 ± 0.13 | 83.20 ± 0.10 |
| DeepEMD [26] | ResNet12 | 65.91 ± 0.82        | 82.41 ± 0.56        | 71.16 ± 0.87 | 86.03 ± 0.58 |
| CTM [27]    | ResNet18 | 62.05 ± 0.55        | 78.63 ± 0.06        | 64.78 ± 0.11 | 81.05 ± 0.52 |
| EBM [28]    | ResNet12 | 63.80 ± 0.40        | 80.10 ± 0.30        | 71.20 ± 0.40 | 85.30 ± 0.30 |
| **DICS-Net** | ResNet12 | **68.57 ± 1.26**    | **84.30 ± 0.60**    | **74.03 ± 1.12** | **88.96 ± 0.61** |

TABLE II: Comparison results with state-of-the-art methods in FC100. The reported accuracies are in 95% confidence intervals over 600 episodes with inductive setting. The top two results are shown in **bold** and **underline**, respectively.

| Method      | Backbone | FC100       |
|-------------|----------|-------------|
|             |          | 5-way 1-shot | 5-way 5-shot |
| TADAM [21]  | ResNet12 | 40.10 ± 0.40 | 56.10 ± 0.40 |
| SLA-AG [22] | ResNet12 | 42.20 ± 0.60 | 59.20 ± 0.50 |
| MTL [30]    | ResNet12 | 45.10 ± 1.80 | 57.60 ± 0.90 |
| MetaOpt [24] | ResNet12 | 41.10 ± 0.60 | 55.50 ± 0.60 |
| **DICS-Net** | ResNet12 | **47.67 ± 0.70** | **60.05 ± 0.57** |

TABLE III: Comparison results with other methods which also use label information. The results are conducted in transductive setting. The top two results are shown in **bold** and **underline**, respectively.

| Method      | Backbone | mini-ImageNet       |
|-------------|----------|---------------------|
|             |          | 5-way 1-shot        | 5-way 5-shot        |
| COMMET [13] | ResNet12 | 69.11 ± 0.67        | 81.21 ± 0.80        |
| HSIC [14]   | ResNet12 | 64.43 ± 1.02        | 76.50 ± 0.64        |
| GPN [15]    | ResNet12 | 70.13 ± 0.17        | 80.00 ± 0.45        |
| S2M2 with MAGR [16] | ResNet12 | 66.93 ± 0.65        | 83.35 ± 0.53        |
| LEO with MAGR [16] | ResNet12 | 60.93 ± 0.19        | 76.33 ± 0.17        |
| **DICS-Net** | ResNet12 | **73.57 ± 1.26**    | **84.30 ± 0.60**    |

TABLE IV: Comparison results of incorporating implicit-component-supervised auxiliary task.

| Dataset        | Method | Base | Novel |
|----------------|--------|------|-------|
|                |        |      | 1-shot | 5-shot |
| mini-ImageNet  | Baseline | 95.62 | 63.21 | 81.69 |
| DICS-Net       | Baseline | 95.69 | 68.57 | 84.30 |
| increase       | 0.07   | **5.36** | **2.61** |
| tiered-ImageNet| Baseline | 70.22 | 70.02 | 86.37 |
| DICS-Net       | Baseline | 70.31 | 74.03 | 88.96 |
| increase       | 0.09   | **4.01** | **2.59** |

C. Experimental Results

1) Comparison Results with Inductive State-of-The-Art Methods: In this section, we first compare our DICS-Net with other state-of-the-art methods in the inductive case, which is shown in Table I, II. Obviously, whatever in 5-way 1-shot case or 5-way 5-shot case, our DICS-Net achieves outstanding performances compared with others. Specifically, in mini-ImageNet, DICS-Net achieves significant improvements of 2.7%-10.1% in 5-way 1-shot case, and 1.9%-8.8% in 5-way 5-shot case. In tiered-ImageNet, DICS-Net can exceed others 2.8%-8.4% in 5-way 1-shot case, and 2.9%-7.9% in 5-way 5-shot case. In FC100, DICS-Net can exceed others 2.6%-6.1% in 5-way 1-shot case, and 0.9%-4.6% in 5-way 5-shot case.

2) Comparison Results with Methods also Use Label Semantics: Our DICS-Net employs the label semantics to generate hierarchical component information, so that it is necessary to compare it with recently proposed methods which also use this kind of extra information. For fair comparison, all the experiments are conducted in the mini-ImageNet with transductive setting. The comparison results are presented in the Table III. We can see that, our DICS-Net outperforms others at least 3.4% in the 1-shot case and 1.0% in the 5-shot case.

D. Ablation Studies

The ultimate goal of designing the implicit-component-supervised auxiliary task is to enable the pre-trained model...
(based on base data) to be more applicable to data in the novel set. It is interesting to find out if the model has achieved the purpose. Here, we research this question from two views.

1. We compare the impact of implicit-component-supervised auxiliary task on the base set and the impact on the novel set, which is shown in Table IV. Baseline denotes the network without the implicit-component-supervised auxiliary task, and DICS-Net represents the Baseline + Implicit-Component-Supervision. From this table, we find that whatever in mini-ImageNet or tiered-ImageNet, the implicit-component-supervised auxiliary task has little influence on classifying the base data, but contributes a lot when recognizing the novel data on both 1-shot case and 5-shot case. This phenomenon verifies that our DICS-Net is available for the cross-category task.

2. While, why DICS-Net based feature distribution is more discriminative? We guess that the implicit-component-supervised auxiliary task allows the pre-trained network to pay more attention on the important information of the sample when extracting novel class features. In order to verify this conjecture, we refer to [30] to visualize the performance of the network. Observation results are listed in Figure 4. Obviously, our DICS-Net outperforms the baseline method a lot.

3. Furthermore, we will discuss the influence of implicit-component-number, i.e., the size of dictionary in Equation (1). We conduct the experiments on mini-ImageNet and show the comparison results in Figure 5. \(K = 0\) means that we do not use the implicit-component-supervision strategy. We observe that the experimental results first rise and then descend with the increase of \(K\) in both 5-way 1 shot and 5-way 5-shot case.

4. From Equation (9), we know \(\alpha\) is a key parameter to our method, which is used to control the influence of the component-supervised auxiliary task. We list the results of mini-ImageNet with different \(\alpha\) in Figure 6. Look at the DICS-Net based line (red line). The result first rises then declines with the increase of \(\alpha\), achieving the best performance when \(\alpha = 0.4\). How to explain this phenomenon?

As described above, the \(\alpha\) is an empirical value. It is determined by the accuracy of implicit-component-based multi-label, comes from our DICG. However, multi-labels obtained in this way are sometimes inaccurate. Indeed, this is a challenge of our method in application, but if we think from another angle, if we can solve this problem, it will be beneficial to the performance improvement for our method. We believe that adjusting the \(\alpha\) can solve this problem to a certain extent. Specifically, to balance the impact, we should assert a large
Fig. 6: The comparison results with different $\alpha$ (x-axis) on mini-ImageNet. DICS denotes the multi-label constructed from DICG. DICS with noised multi-label indicates that we introduce some noise to the multi-label matrix.

value when the multi-label is accurate, and it is the opposite if the multi-label is confused. To evaluate it, we deliberately add some noise to the multi-label of samples and show the final results in Figure 6. Observe the yellow line, the performances of the red line are higher than the yellow one, but the $\alpha$ corresponded to the best result on the yellow line is lower than that on the red one (i.e., $\alpha_{\text{yellow}} = 0.2 < \alpha_{\text{red}} = 0.4$). It is demonstrated that the $\alpha$ is capable of influence the component-supervised auxiliary task.

V. CONCLUSION

Few-shot classification (FSC) exists a fundamental problem, i.e., Feature-Extractor-Maladaptive (FEM) problem. To solve it, this paper proposes a novel Dictionary-guided Implicit-Component-Supervision Network (DICS-Net). We believe that all different category targets have similar implicit component information. Introducing this information is helpful to improve the adaptability of the feature extractor. Therefore, we first attempt to mine the information through our designed Dictionary-based Implicit-Component Generator (DICG), then use them by designing an implicit-component-supervised auxiliary task. The outstanding performances have evaluated the efficiency of our method. For future work, we are committed to find more reasonable and efficient ways to mine the implicit-components to make the to-be-trained network more robust.

REFERENCES

[1] L. Wen, Y. Wang, and X. Li, “A new cycle-consistent adversarial networks with attention mechanism for surface defect classification with small samples,” TII, 2022.

[2] C. Luo, Z. Tan, G. Min, J. Gan, W. Shi, and Z. Tian, “A novel web attack detection system for internet of things via ensemble classification,” TII, vol. 17, no. 8, pp. 5810–5818, 2020.

[3] M. Liu, K. Yang, N. Zhao, Y. Chen, H. Song, and F. Gong, “Intelligent signal classification in industrial distributed wireless sensor networks based industrial internet of things,” TII, vol. 17, no. 7, pp. 4946–4956, 2020.

[4] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in ICML, 2017, pp. 1126–1135.

[5] H. Dong, K. Song, Q. Wang, Y. Yan, and P. Jiang, “Deep metric learning-based for multi-target few-shot pavement distress classification,” TII, vol. 18, no. 3, pp. 1801–1810, 2021.

[6] P. Rodríguez, I. Laradji, A. Drouin, and A. Lacoste, “Embedding propagation: Smoother manifold for few-shot classification,” in ECCV, 2020.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.

[8] S. G. Mallat and Z. Zhang, “Matching pursuit with time-frequency dictionaries,” TSP, vol. 41, no. 12, pp. 3397–3415, 1993.

[9] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, “Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems,” TII, vol. 17, no. 8, pp. 5790–5798, 2020.

[10] P. Mangla, N. Kumari, A. Sinha, M. Singh, B. Krishnamurthy, and V. N. Balasubramanian, “Charting the right manifold: Manifold mixup for few-shot learning,” in WACV, 2020, pp. 2218–2227.

[11] S. Yang, L. Liu, and M. Xu, “Free lunch for few-shot learning: Distribution calibration,” in ICLR, 2021.

[12] Y. Wang, C. Xu, C. Liu, L. Zhang, and Y. Fu, “Instance credibility inference for few-shot learning,” in CVPR, 2020, pp. 12 836–12 845.

[13] K. Cao, M. Bribic, and J. Leskovec, “Concept learners for few-shot learning,” in ICLR, 2021.

[14] Y.-H. H. Tsai and R. Salahudden, “Improving one-shot learning through fusing side information,” arXiv preprint arXiv:1710.08347, 2017.

[15] L. Liu, T. Zhou, G. Long, J. Jiang, and C. Zhang, “Learning to propagate for graph meta-learning,” in NeurIPS, 2019.

[16] Y. Z. Liu, D. Wang, L. Zhao, N. Gu, and S. Maybank, “Self-taught semisupervised dictionary learning with nonnegative constraint,” TII, vol. 16, no. 1, pp. 532–543, 2019.

[17] M. Yang, L. Zhang, X. Feng, and D. Zhang, “Fisher discrimination dictionary learning for sparse representation,” in ICCV. IEEE, 2011, pp. 543–550.

[18] B.-D. Liu, Y.-X. Wang, B. Shen, Y.-J. Zhang, and Y.-J. Wang, “Block-wise coordinate descent schemes for sparse representation,” in ICASSP, 2014, pp. 5267–5271.

[19] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, “Meta-transfer learning for few-shot learning,” in CVPR, 2019, pp. 403–412.

[20] B. Oreshkin, P. R. López, and A. Lacoste, “Tadam: Task dependent adaptive metric for improved few-shot learning,” in NeurIPS, 2018, pp. 721–731.

[21] H. Lee, S. J. Hwang, and J. Shin, “Self-supervised label augmentation via input transformations,” in ICML, 2020, pp. 5714–5724.

[22] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, “Meta-baseline: exploring simple meta-learning for few-shot learning,” in ICCV, 2021, pp. 9062–9071.

[23] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, “Meta-learning with differentiable convex optimization,” in CVPR, 2019, pp. 10 657–10 665.

[24] S. Laenen and L. Bertinetto, “On episodes, prototypical networks, and few-shot learning,” in NeurIPS, vol. 34, 2021, pp. 24 581–24 592.

[25] C. Zhang, Y. Cai, G. Lin, and C. Shen, “Deepemd: Few-shot image classification with differentiable earth mover’s distance and structured classifiers,” in CVPR, 2020, pp. 12 203–12 213.

[26] H. Li, D. Eigen, S. Dodge, M. Zeiler, and X. Wang, “Finding task-relevant features for few-shot learning by category traversal,” in CVPR, 2019, pp. 1–10.

[27] Y. Liu, B. Schiele, and Q. Sun, “An ensemble of epoch-wise empirical bayes for few-shot learning,” in ECCV, 2020, pp. 404–421.

[28] A. Krizhevsky, G. Hinton et al., “Learning multiple layers of features from tiny images,” Computer Science Department, University of Toronto, 2009.

[29] S. Shao, L. Xing, C. Yixin, W. Yanjiang, L. Baodi, and Z. Yicong, “Cim: Class-irrelevant mapping for few-shot classification,” arXiv preprint arXiv:2109.02840, 2021.