Supplementary Material for
Few-shot Continual Infomax Learning

Ziqi Gu#, Chunyan Xu#, Jian Yang, Zhen Cui*
PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional
Information of Ministry of Education, School of Computer Science and Engineering,
Nanjing University of Science and Technology, Nanjing, China.
{ziqigu, cyx, csjyang, zhen.cui}@njust.edu.cn

1. Background Knowledge

Entropy: Entropy is a basic concept in information theory that represents a random variable as a measure of uncertainty [17]. For example, $H(X)$ denotes the entropy of a random distribution $X$.

Transfer Entropy: Transfer entropy is a measure of the amount of information transferred by two stochastic processes. The transfer of stochastic process $X$ to stochastic process $Y$ is achieved by knowing the past of $X$ to reduce the uncertainty of the future of $Y$, where the information is measured by entropy [2], formally:

$$T_{X \rightarrow Y} = H(Y_t|Y_{t-1:t-L}) - H(Y_t|Y_{t-1:t-L}, X_{t-1:t-L}).$$

Eqn. 1 is equivalently transformed to conditional mutual information, formally:

$$T_{X \rightarrow Y} = I(Y_t; X_{t-1:t-L}|Y_{t-1:t-L}).$$

Mutual Information Estimation: Mutual information (MI) is the reduction of uncertainty in one random variable due to the knowledge of another random variable [1, 7]. Specifically, it is the information obtained from one random variable through another random variable. For two random variables $X$ and $Y$, the joint probability distribution is $p(X,Y)$. The mutual information between $X$ and $Y$ is given by,

$$I(X;Y) = \int dx dy \ p(X,Y) \ \log \left( \frac{p(X,Y)}{p(X)p(Y)} \right).$$

Mutual information is equivalently represented as,

$$I(X;Y) = \sum_{x,y} p(x,y) \log \left( \frac{p(x,y)}{p(x)p(y)} \right)$$

$$= \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} - \sum_{x,y} p(x,y) \log p(y)$$

$$= \sum_{x,y} p(x)p(y|x) \log(p(y|x)) - \sum_{x,y} p(x,y) \log p(y)$$

$$= -\sum_{x} p(x)H(Y|X = x) - \sum_{y} \log p(y)p(y)$$

$$= H(Y) - H(Y|X).$$

where, $H(X)$ is the marginal entropy, $H(X|Y)$ is the conditional entropy, $H(X,Y)$ is the joint entropy of $X$ and $Y$. Thus, $I(X;Y)$ equals $H(X) - H(X|Y)$ and $H(X) - H(X|Y)$ [1, 7].

Due to the high-dimensional feature vector, it is difficult to accurately compute the mutual information between two variables. Thus, the mutual information of two random variables $X$ and $Y$ can be represented by Kullback-Leibler divergence [5], formally:

$$I(X;Y) = D_{KL}(p(X,Y)||p(X) \otimes p(Y))$$

$$= \mathbb{E}_{p(x,y)}[F] - \log \mathbb{E}_{p_X \otimes p_Y}[e^F],$$

where, $p(X,Y)$ is the joint probability distribution of $X$ and $Y$, all functions $F$ such that both expectations are finite. Since the mutual information of high-dimensional vectors is difficult to compute, to solve the Eqn. 5, we use neural network to estimate the maximum lower bound [1, 7].

$$\mathbb{E}[F((X;Y), \vartheta^{\text{MI}})] - \log \mathbb{E}[e^{F((X;Y), \vartheta^{\text{MI}})}].$$

Here, $\vartheta^{\text{MI}}$ refers to a neural network to estimate mutual information between $X$ and $Y$.

# Equal Contribution.
* Corresponding Author.
2. Datasets and other results

2.1. Datasets

**CIFAR100**: CIFAR100 [9] contains 100 classes with a total of 60,000 RGB images with the size $32 \times 32$, and each class contains 500 training images and 100 test images.

**CUB200**: CaltechUCSD Birds-200-2011 (CUB200) [16] is a fine-grained classification dataset, which contains 11,788 images in 200 classes, each image size is $224 \times 224$.

**miniImageNet**: miniImageNet [13] is a subset of ImageNet, which contains 100 classes with 60,000 images, and each image size is $84 \times 84$.

2.2. Results of other datasets

In the main paper, we report the detailed performance of the miniImageNet. We report the performance of the CUB200 [16] and CIFAR100 [10] in Table 1 and Table 2. We can infer that our proposed FCIL has better final accuracy, Avg and KR, indicating FCIL better than state-of-the-art methods.

Algorithm 1 Few-shot Continual Infomax Learning (FCIL)

**Require**: the training sets $\{(X_t, Y_t) | t = 1, \ldots, T\}$, the number of previous sessions $K$.

**Ensure**: the final model $\Theta$ and $\vartheta$.

1. while $t = 1, 2, \ldots, T$ do
2. \hspace{1em} if $t = 1$ then
3. \hspace{2em} Optimize the base network $\Theta^{base}$ and the MI network $\Theta^{M}$ on the training set $(X_1, Y_1)$;
4. \hspace{2em} Construct base class structure $S(A', R')$;
5. \hspace{1em} else
6. \hspace{2em} # for the $t$-th session data
7. \hspace{3em} Learn new-class model $\Theta_{fc}'$ by feature embedding infomax $\mathcal{L}_{FEI}$;
8. \hspace{3em} Update class structure $S'(A', R')$;
9. \hspace{3em} Update the classifier $\Theta_{fc}$ by performing continual structure infomax $\mathcal{L}_{CSI}$;
10. end if
11. end while
References


