

1. はじめに

継続学習は、学習中にクラスやタスクが増えることを想定した学習法である。継続学習では、学習中に学習用データが変化するため、モデルが新しいクラスやタスクのデータに過度に適合してしまい、過去に学習したクラスやタスクを忘れてしまう破滅的忘却が発生する。Lifelong Unsupervised Mixup (LUMP) [1] は、データ拡張法の mixup [2] と過去のタスクの学習に利用したデータの一部を保存するリプレイバッファを活用することで、破滅的忘却を抑制する自己教師あり継続学習手法である。LUMP はランダムにデータを選択してリプレイバッファに保存するため、バッファ内のデータに偏りが発生する。また、全ての学習ステップで mixup を使用するため、クラス間やタスク間の境界が曖昧な特徴空間となることが考えられる。

本研究では、これらの問題を改善するために、リプレイバッファ内のデータの偏りを抑制するデータ選択方法と旧タスクの効果的な学習方法を提案する。評価実験では、クラス分類問題において提案手法が有効であることを示す。

2. 継続学習

継続学習は、学習中にクラスやタスクが増えることを想定し、過去に学習したクラスやタスクを忘れてしまう破滅的忘却に対処する学習法である。継続学習では、モデル構造やパラメータの更新方法、学習するデータを工夫することで破滅的忘却を抑制する。特にシンプルな破滅的忘却の抑制方法として、新しいクラスやタスクの学習時に過去に学習したデータを再利用するリプレイベースの手法がある。

LUMP [1] は、リプレイベースの自己教師あり継続学習法である。LUMP では、過去に学習したタスクのデータの一部をランダムにリプレイバッファというメモリに保存する。新しいタスクの学習時には、データ拡張手法の mixup [2] を用いて新しいタスクのデータとリプレイバッファの過去タスクのデータを合成し、モデルに入力して学習することで破滅的忘却を抑制する。

LUMP には2つ問題がある。1つ目は、ランダムにデータを選択していることである。学習するデータに偏りがある場合、ランダムにデータを選択するとリプレイバッファ内で元のデータ分布を再現することが困難である。2つ目は、2タスク目以降の全ての学習ステップで mixup を使用することである。この時、新旧タスクのデータの中間的なデータを学習し、タスク間の特徴表現が近くなり、特徴空間が曖昧になる。また、タスクの切り替わり時に学習するデータの分布が大幅に変わるため学習が不安定になる。

3. 提案手法

本研究では、自己教師あり継続学習における分類精度の向上と破滅的忘却の抑制を目的として、リプレイバッファに保存するデータの選択方法と旧タスクの学習方法を改善する。以下では、提案手法を継続学習におけるリプレイバッファに保存するデータの選択方法と旧タスクの効果的な学習方法に分けて述べる。

3.1. リプレイバッファに保存するデータ選択

提案手法では、多様性と学習不足を考慮してリプレイバッファに保存するデータを選択する。本手法のデータ選択方法を図1に示す。本手法では、まず k-means 法によりクラスターリングを行う。この時、指定するクラス数 k は、保存するデータ数の2倍とする。次に、各クラスターのセントロイドから一番近いデータを選択候補データとする。そして、選択候補データのうち、Contrast Scoring (CS) [3] により非類似度が高い上位半数のデータをリプレイバッファに保存する。CS では、入力データと入力データを左右反転した2つのデータの特徴抽出器であるエンコーダに入力し、エンコーダが出力する2つの特徴量の非類似度が高いデータを学習不足データとして、リプレイバッファに保存する。CS で求める非類似度 S を式 (1) に示す。

$$S = 1 - \text{cossim}(z_i, z'_i) \quad (1)$$

ここで、 z_i はデータ x_i をエンコーダに入力した時の特徴ベクトル、 z'_i はデータ x_i を左右反転した x'_i をエンコーダに入力した時の特徴ベクトル、 $\text{cossim}()$ はコサイン類似度関数である。

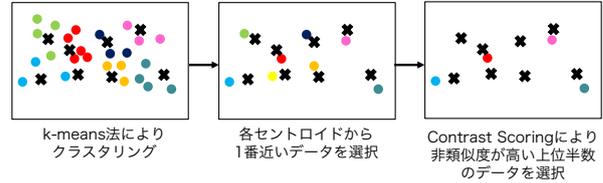


図1: k-means 法と CS を用いたデータ選択方法

3.2. 旧タスクデータの効果的な学習

提案手法では、旧タスクのデータをバランスよく学習するために、旧タスクと新タスクを mixup により混合した中間タスクを作成する。そして、旧タスクと現在タスクを含めた3つのタスクを動的に選択して学習するリプレイラーニングを導入する。本手法の学習方法を図2に示す。

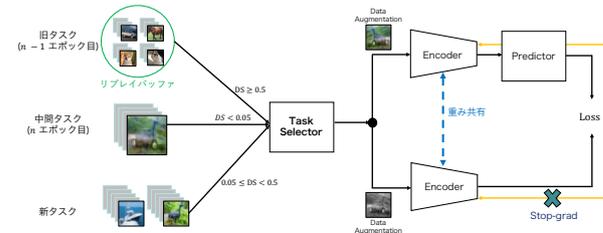


図2: Task Selector と DS を用いた学習方法

提案手法のリプレイラーニングでは、損失値の増減幅による Task Selector を用いて、学習するタスクを選択する。選択基準として、式 (2) に示す Diffcult Score (DS) を用いる。

$$DS = \text{Loss}_n - \text{Loss}_{n-1} \quad (2)$$

ここで、 n は現在のエポックである。DS は損失値の増減幅を表し、学習の安定性を測る指標である。

リプレイラーニングでは、過去の知識を引き継ぐために、1エポック目で旧タスク、2エポック目で中間タスクのデータを学習する。そして、3エポック目以降は、Task Selector で選択したタスクのデータを学習する。Task Selector では、DS が 0.05 以上 0.5 未満の時にあらかじめ決めた順序でタスクを選択して学習する。DS が 0.5 以上の時は、学習が不安定であると判断し、旧タスクを学習することで、パラメータの大幅な変動を抑える。DS が 0.05 未満の時は、学習難易度が高い中間タスクを学習することで、過学習を抑え、新旧タスクの両方に適したパラメータに変動させる。このように、DS に基づいた Task Selector を用いて、次の学習タスクを選択することで、学習の安定性を高めるとともに、多様なデータを学習して特徴空間を維持可能にする。

4. 評価実験

本研究で提案したリプレイバッファに保存するデータの選択方法と旧タスクデータの学習方法の有効性をクラス分類問題を用いて定量的に評価する。4.3 節と 4.4 節の比較実験では、評価方法として k-NN 法を用い、4.5 節の従来手法との比較実験では、評価方法として線形分類を用いる。

4.1. 評価指標

本実験では、評価指標として式 (3) に示す平均分類精度 A_τ と式 (4) に示す平均忘却率 F を用いる。

$$A_\tau = \frac{1}{\tau} \sum_{i=1}^{\tau} a_{\tau,i} \quad (3)$$

$$F = \frac{1}{T-1} \sum_{i=1}^{T-1} \max_{\tau \in (1, \dots, T)} (a_{\tau, i} - a_{T, i}) \quad (4)$$

ここで、 T はタスク数、 $a_{\tau, i}$ はタスク τ について事前学習をしてタスク i について評価した精度、 $a_{T, i}$ は最後のタスク T について事前学習をしてタスク i について評価した精度である。本実験では、式 (3) において $\tau = T$ として、すべてのタスクを学習し終えた時の平均精度を確認する。

4.2. 実験条件

本実験では、データセットは CIFAR-10 を 2 クラスずつ 5 タスクに分割した Split CIFAR-10 と CIFAR-100 を 20 クラスずつ 5 タスクに分割した Split CIFAR-100, Tiny-ImageNet の半分のデータを用いて 10 クラスずつ 10 タスクに分割した Split Tiny-ImageNet を使用する。エンコーダの学習条件は、LUMP の学習設定に基づき、ネットワークは ResNet-18, 自己教師あり学習法は SimSiam [4], 学習回数は 200 エポック, バッチサイズとバッファサイズは Split CIFAR-10 の実験で 256, Split CIFAR-100 と Split Tiny-ImageNet の実験で 512 とする。また、線形分類時の線形層の学習条件は、CroMo-Mixup [5] の設定に基づき、学習回数は 200 エポック, 最適化手法は MomentumSGD, バッチサイズは 256 である。

4.3. データ選択方法の比較

この評価実験では、従来手法 LUMP のランダム、k-means 法のみ、k-means 法 + CS の 3 通りのデータ選択方法を比較する。3 通りのデータ選択方法を用いた場合の平均分類精度と平均忘却率を表 1 に示す。表 1 より、ランダムにデータを選択した場合と比較して、k-means 法のみや k-means 法と CS を組み合わせた提案手法を用いることで従来手法と比較して、精度の改善と忘却の抑制が確認できる。

表 1: データ選択方法の比較結果

Method	平均分類精度 [%] (\uparrow)		平均忘却率 [pt] (\downarrow)	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
LUMP (Random)	89.91	60.46	7.33	2.93
k-means	90.89	62.70	1.73	2.25
k-means + CS	90.91	63.91	2.49	1.03

4.4. Task Selector の有効性

この評価実験では、学習時に Task Selector を使用した場合と使用しない場合を比較する。この実験では、リプレイバッファに保存するデータの選択に k-means 法のみ、もしくは k-means 法と CS を 9 : 1 で使用する。

Task Selector を使用した場合と使用しない場合の平均分類精度と平均忘却率を表 2 に示す。表 2 では、カッコ内にリプレイバッファに保存するデータの選択方法を示している。表 2 より、データ選択に k-means 法を用いた提案手法は、Task Selector を使用しない場合と比較して忘却率が少し悪化したものの、精度が改善していることが確認できる。さらに、提案手法のデータ選択に CS を加えることで、精度を大幅に改善したことが確認できる。

Split CIFAR-10 における各タスクの精度について Task Selector を使用しない場合を表 3, Task Selector を使用する提案手法 (k-means) で学習した場合を表 4 に示す。表 3, 表 4 より、提案手法では Task Selector を使用しない場合と比較して、各タスクを学習後の最新タスクに対する精度が向上し、最終タスクまで学習した場合のすべてのタスクの精度が高くなっている。これより、提案した学習方法が多様なデータを学習することにより、最新タスクと過去タスクの両方に対処できたと考えられる。

表 2: Task Selector (TS) の有無による比較結果

Method	平均分類精度 [%] (\uparrow)		平均忘却率 [pt] (\downarrow)	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
w/o TS (k-means)	90.89	62.70	1.73	2.25
w/ TS(k-means)	92.30	66.32	2.36	2.54
w/ TS(k-means+CS)	92.30	66.89	2.45	1.03

表 3: w/o TS (k-means) の各タスクの精度推移 [%]

		評価タスク					平均
		1	2	3	4	5	
学 習 タ ス ク	1	96.95	-	-	-	-	96.95
	2	95.50	83.20	-	-	-	89.35
	3	95.15	80.25	86.15	-	-	87.18
	4	95.20	80.50	88.45	96.90	-	90.26
	5	94.90	81.70	87.60	94.35	95.90	<u>90.89</u>
忘却率		2.05	1.50	0.85	2.55	-	<u>1.74</u>

表 4: w/ TS (k-means) の各タスクの精度推移 [%]

		評価タスク					平均
		1	2	3	4	5	
学 習 タ ス ク	1	96.80	-	-	-	-	96.80
	2	95.15	85.95	-	-	-	90.55
	3	96.55	84.70	93.85	-	-	91.70
	4	96.30	83.65	90.95	97.60	-	92.13
	5	96.20	82.90	90.10	95.55	96.75	<u>92.30</u>
忘却率		0.60	3.05	3.75	2.05	-	<u>2.36</u>

4.5. 従来手法との比較

この評価実験では、自己教師あり継続学習の従来手法である CaSSLe [6], CroMo-Mixup [5] と提案手法を比較する。従来手法と提案手法の線形分類による評価結果を表 5 に示す。表 5 より、2 つのデータセットにおいて提案手法が従来手法より分類精度が高いことが確認できる。

表 5: 従来手法と提案手法の比較結果

Method	分類精度 [%] (\uparrow)	
	Split CIFAR-100	Split Tiny-ImageNet
CaSSLe	62.20	43.40
CroMo-Mixup	65.48	47.32
提案手法	67.26	47.85

5. おわりに

本研究では、継続学習におけるリプレイバッファに保存するデータの選択方法として、多様性と学習不足を考慮したデータ選択方法、旧タスクデータの学習方法として、Task Selector を用いた特徴空間をより維持可能な学習方法を提案した。評価実験により、それぞれの手法が有効であり、これらの手法を組み合わせることで最も有効な手法になることを確認した。今後は、Task Selector を最適化し、旧タスクのデータの学習方法を改善する予定である。

参考文献

- [1] D. Madaan, *et al.*, “Representational Continuity for Unsupervised Continual Learning”, ICLR, 2022.
- [2] H. Zhang, *et al.*, “Mixup: Beyond empirical risk minimization”, ICLR, 2018.
- [3] Y. Wu, *et al.*, “Enabling On-Device Self-Supervised Contrastive Learning With Selective Data Contrast”, DAC, 2021.
- [4] X. Chen, *et al.*, “Exploring Simple Siamese Representation Learning”, CVPR, 2021.
- [5] E. Mushtaq, *et al.*, “CroMo-Mixup: Augmenting Cross-Model Representations for Continual Self-Supervised Learning”, ECCV, 2024.
- [6] E. Fini, *et al.*, “Self-supervised models are continual learners”, CVPR, 2022.

研究業績

- [1] N. Hayashi, *et al.*, “Diverse Data Selection Considering Data Distribution for Unsupervised Continual Learning”, VISAPP, 2024
(他 学会発表 3 件)