学習データと異なる識別タスクに適応するデータ選択型敵対的訓練法の提案

1X19C045-1　木村恵悟
指導教員　後藤正幸

1. はじめに

今日，社会の様々な場面で機械学習が活用されており，その 1 つに予測タスクがある．一般に，機械学習では学習データに基づいた予測を行うため，学習データと予測対象データの間で扱う特徴量が同じでも，それらの統計的構造が異なる場合には適切な予測は保証されない．しかし，現実問題ではモデルの学習段階から構造が変化してしまう対象に対して予測モデルを適用しなければならない状況も多い．そこで近年，この問題に対処する技術（以下，ドメイン適応）に関する研究が盛んに行われており，その代表的な手法として Adversarial Discriminative Domain Adaptation[1]（以下，ADDA）が提案されている．

ADDA は，Generative Adversarial Networks[2]（以下，GAN）を応用した敵対的訓練を用いることで，データセット間の分布を近づけながらモデルを推定する手法であり，ドメイン適応を用いたクラス分類タスクにおいて優れた予測精度を示すとされている．しかし，ADDA は学習の際に，データを分割したミニバッチを全て用いており，これらの中に学習を悪化させるデータが含まれる可能性がある．

一方，GAN に対しては Top-k Training of GANs[3]（以下，Top-k Training）が提案されており，学習に使用するデータをモデル内の出力値に基づいて選択することで，モデルの性能向上が可能とされている．そこで本研究では，ADDA の学習に Top-k Training を援用した改善手法を提案する．これにより，ADDA の学習に有用なデータのみが用いられ，モデルの予測精度が向上することが期待される．本研究では，実データを用いて推定精度評価を行い，提案手法の有効性を示す．

2. 準備

2.1. 問題設定

本研究では，学習対象（以下，ソースドメイン）から得られたデータによって学習したモデルを用いて，予測対象（以下，ターゲットドメイン）のデータのクラス分類予測を行う問題を考える．特徴量ベクトル $\boldsymbol{x}_n^{(s)}$ の集合 $\boldsymbol{x}^{(s)} = \{\boldsymbol{x}_n^{(s)}\}_{n=1}^N$ と対応するラベル $y_n^{(s)}$ の集合 $\boldsymbol{y}^{(s)} = \{y_n^{(s)}\}_{n=1}^N$ で構成されるソースドメイン内のデータ集合 $\boldsymbol{D}^{(s)} = \{\boldsymbol{x}_n^{(s)}, y_n^{(s)}\}_{n=1}^N$，および特徴量ベクトル $\boldsymbol{x}_m^{(t)}$ の集合 $\boldsymbol{x}^{(t)} = \{\boldsymbol{x}_m^{(t)}\}_{m=1}^M$ と対応するラベル $y_m^{(t)*}$ の集合 $\boldsymbol{y}^{(t)*} = \{y_m^{(t)*}\}_{m=1}^M$ で構成されるターゲットドメイン内のデータ集合 $\boldsymbol{D}^{(t)} = \{\boldsymbol{x}_m^{(t)}, y_m^{(t)*}\}_{m=1}^M$ を定義する．なお，$\boldsymbol{y}^{(t)*}$ は予測対象であるため，実際にはデータとして与えられない．なお，$\boldsymbol{x}_n^{(s)}$ と $\boldsymbol{x}_m^{(t)}$ の特徴量空間は同次元であるか統計的特徴が異なってもよいものとし，$y_n^{(s)}$ と $y_m^{(t)*}$ のラベル空間は同じであるとする．ここで，$\boldsymbol{x}^{(s)}$ から $\boldsymbol{y}^{(s)}$ を予測する際のドメイン知識を活用し，$\boldsymbol{x}^{(t)}$ から $\boldsymbol{y}^{(t)*}$ を予測するタスクを設定する．このとき，ドメイン間で異なる性質の特徴量空間を扱うため，ドメイン適応を用いたモデルを構築する．

2.2. Adversarial Discriminative Domain Adaptation

ADDA はドメイン適応の手法の 1 つであり，ソースエンコーダー $M^{(s)}$, ターゲットエンコーダー $M^{(t)}$，クラス分類器 C，ドメイン識別器 D の 4 つのネットワークが用いられる．$M^{(s)}$ はソースデータ $\boldsymbol{x}_n^{(s)}$ を，$M^{(t)}$ はターゲットデータ $\boldsymbol{x}_m^{(t)}$ をそれぞれ同次元の中間表現 $M^{(s)}(\boldsymbol{x}_n^{(s)})$，$M^{(t)}(\boldsymbol{x}_m^{(t)})$ に変換し，両ドメイン共通のクラス分類器 C はこれらを入力として分類予測を行う．まず，事前にラベル付きのソースデータ $\boldsymbol{x}_n^{(s)}$ を正しくクラス分類できるように，ソースエンコーダー $M^{(s)}$ とクラス分類器 C を学習する．次に，ドメイン識別器 D が中間表現のドメインを見分けられなくなるように，ターゲットエンコーダー $M^{(t)}$ の学習を行う．これにより，両ドメインの中間表現が同様の分布となり，共通のクラス分類器器 C を用いた分類予測が可能となる．

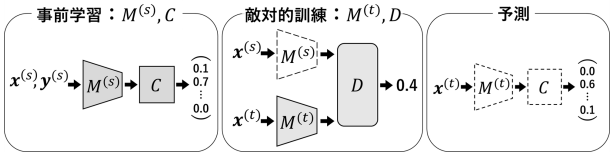

図 1: ADDA の学習のイメージ図

2.2.1. $M^{(s)}$，C の事前学習

ソースデータ $\boldsymbol{x}_n^{(s)}$ のラベルは入手可能であるため，教師あり学習が可能である．そのため，式 (1) で定義される交差エントロピー損失 L_{cls} を最小化することで，$M^{(s)}$，C のパラメータを学習する．

$$L_{cls} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K}\delta_{nk}\log C(M^{(s)}(\boldsymbol{x}_n^{(s)})) \quad (1)$$

ここで，N はソースデータ $\boldsymbol{x}_n^{(s)}$ の数，K は分類するクラス数，δ_{nk} は n 番目のデータが k 番目のクラスに所属している場合は 1，そうでない場合は 0 を返す指示関数，$C(\cdot)$ はクラス分類器により出力される各クラスへの所属確率である．

2.2.2. $M^{(t)}$ の敵対的訓練による学習

ターゲットデータ $\boldsymbol{x}_m^{(t)}$ のラベルは入手できないため，教師あり学習によるクラス分類モデルの学習が行えない．そこで，ソースデータ $\boldsymbol{x}_n^{(s)}$ を用いて事前学習した C による分類予測を可能にするために，敵対的訓練を用いて両ドメインの中間表現の分布を近づける．敵対的訓練のためにドメイン識別器 D を導入し，ドメイン識別器 D は中間表現のドメインを正しく識別できるように学習する．一方，$M^{(t)}$ の学習は，$M^{(t)}(\boldsymbol{x}_m^{(t)})$ を $M^{(s)}(\boldsymbol{x}_n^{(s)})$ であると D に誤識別させるように行う．具体的には，$M^{(t)}$ を $M^{(s)}$ のパラメータで初期化し，$M^{(t)}$ を固定した状態で式 (2) の L_D を最小化するように D を学習する過程と，D を固定した状態で式 (3) の $L_{M^{(t)}}$ を最小化するように $M^{(t)}$ を学習する過程を交互に繰

り返す．

$$L_D = -\frac{1}{B}\sum_{n=1}^{B}[\log D(M^{(s)}(\boldsymbol{x}_n^{(s)}))] - \frac{1}{B}\sum_{m=1}^{B}[\log(1 - D(M^{(t)}(\boldsymbol{x}_m^{(t)})))] \quad (2)$$

$$L_{M^{(t)}} = -\frac{1}{B}\sum_{m=1}^{B}[\log D(M^{(t)}(\boldsymbol{x}_m^{(t)}))] \quad (3)$$

ここで，B はバッチサイズ，$D(\cdot)$ は入力した中間表現がソースドメインものである予測確率である．

これにより，$M^{(t)}$ と C を組み合わせることでターゲットデータ $\boldsymbol{x}_m^{(t)}$ のクラス分類が可能となる．

2.3. Top-k Training of GANs

Top-k Training は，従来の GAN がミニバッチのデータを全て用いて学習するのに対し，GAN の敵対的訓練における生成器の学習の際に識別器の出力値に基づく有用なデータのみを使用する改良手法である．これにより，計算コストを増やすことなく GAN の性能を向上させることができる．

3. 提案手法：Top-k Training を援用した ADDA

従来の ADDA はミニバッチのデータを全て用いて学習しているが，ミニバッチの中には性能を悪化させる方向に学習を進めるデータが含まれている可能性がある．そこで本研究では，ターゲットエンコーダー $M^{(t)}$ の学習に Top-k Training を援用することで，ADDA の性能向上に貢献するデータを選択して学習に用いる手法を提案する．具体的には，$M^{(t)}(\boldsymbol{x}_m^{(t)})$ のうち，ドメイン識別器 D を上手く欺き，ソースドメインである確率 $D(M^{(t)}(\boldsymbol{x}_m^{(t)}))$ の値が上位 k 件のデータを選択し，$M^{(t)}$ を更新する際の勾配の計算に用いる (図 2)．ここで，学習初期は D の性能が低く，その出力に基づくデータ選択が信用できない．そのため，k の初期値を大きく設定し，学習の経過とともに徐々に k を減少させる．

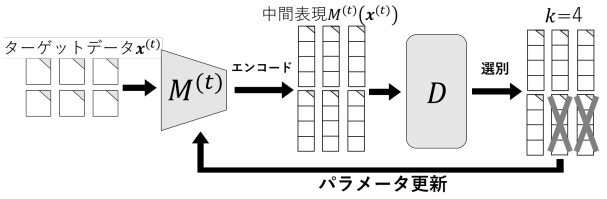

図 2: 提案手法のイメージ図

4. 実験

4.1. 実験条件

本節では，提案手法と従来の ADDA の予測精度を比較するため，2 種類の数字画像データセットを用いたドメイン適応の実験を行う．ソースデータセット $\boldsymbol{x}^{(s)}$，ターゲットデータセット $\boldsymbol{x}^{(t)}$ としてそれぞれ MNIST1,860 件と USPS1,860 件，評価指標として正解率を用いる．また，$M^{(s)}$ と $M^{(t)}$ の各層の次元数は入力層から順に (784,2880,800,500)，C も同様に (500,10) とした．Top-k Training については，$M^{(t)}$ の学習の際にミニバッチから選択するデータ数 k に関するパラメータとして，初期値 k_{init}，エポック毎の減少倍率 γ，バッチサイズに対する下限の割合 ν を決める必要がある．本実験では，k_{init}=100（バッチサイズ）で統一し，γ，ν の設定についてはそれぞれ 5 通り，6 通りの計 30 通りの組み合わせで 30 回ずつ実験を行った．従来手法との平均正解率の比較および各パラメータ設定での平均正解率の比較を行う．

4.2. 実験結果と考察

従来手法の平均正解率を表 1 に，(γ, ν) の各設定における提案手法の平均正解率を表 2 に示す．

表 1: 従来手法の平均正解率 [%]

手法	平均正解率 [%]
ドメイン適応なし	85.33
従来手法	92.46

表 2: (γ，ν) の各組み合わせにおける平均正解率 [%]
(濃色ほど平均正解率が高い，白色は悪化)

		減少倍率γ				
		0.99	0.95	0.90	0.85	0.80
下限の割合ν	0.50	92.72	92.33	93.10	93.22	93.55
	0.40	92.73	93.50	93.48	93.51	92.67
	0.30	92.77	93.16	93.05	93.32	93.39
	0.20	93.08	93.36	93.27	93.27	93.33
	0.10	93.34	92.72	92.94	92.90	92.97
	0.01	92.64	91.44	91.25	91.35	90.84

表 1，表 2 より，多くのパラメータ設定において提案手法の平均正解率が従来手法を上回ることが確認できる．特に改善幅が 1%を超える設定もあり，従来手法の正解率が 92.46%であることを考慮すると十分大きな改善といえる．また，表 1 より，(γ 大,ν 大) の設定の場合は平均正解率は悪化しにくいが改善度も小さい傾向があり，(γ 小,ν 小) の設定の場合は平均正解率が悪化しやすい傾向があることが確認できる．一方で，(γ 大,ν 小)，(γ 中,ν 中)，(γ 小,ν 大) の設定の場合は平均正解率が特に高いことが多く，これらの設定が精度を悪化させるリスクを適切に制御できていると考えられる．

5. 結論と今後の課題

本研究では，ADDA の学習の際に Top-k Training を援用した手法を提案した．提案手法は，従来手法と比較して平均正解率が高いことが示され，有効な手法であるといえる．また，Top-k Training に関する各パラメータについて 30 通りの組み合わせで実験を行い，予測精度を比較することで適切な設定の検討を行った．今後の課題としては，実問題では本稿のような複数通りのパラメータ設定での予測精度の比較ができないため，実用的なパラメータの選択方法の検討を行うことが挙げられる．

参考文献

[1] Tzeng, Eric, et al, "Adversarial discriminative domain adaptation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, 63(11), p.7167-7176, 2017

[2] Goodfellow, Ian, et al, "Generative adversarial networks," *Communications of the ACM*, 63(11), 139-144, 2020

[3] Sinha, Samarth, et al, "Top-k training of gans: Improving gan performance by throwing away bad samples," *Advances in Neural Information Processing Systems*, 33, 14638-14649, 2020