令和元年度　茨城大学大学院理工学研究科情報工学専攻　修士学位論文

敵対的学習を利用した
文書分類の領域適応

提出日　　令和２年２月５日
18NM724F　趙　紫名
指導教員　　新納　浩幸　教授

令和元年度　茨城大学大学院理工学研究科情報工学専攻　修士学位論文

敵対的学習を利用した
文書分類の領域適応

氏名：18NM724F 趙　紫名
指導教員：新納　浩幸　教授

論文要旨

本論文では文書分類の領域適応に対して、敵対的ネットワークを利用する手法を提案する。

文書分類とは、レビュー文書が肯定的なものか、否定的なものかを判定するタスクである。文書分類は、教師あり学習を用いて解決できる。しかし判定先の文書が学習データの領域とは異なる領域の文書（例えば音楽のレビュー）であった場合に、教師あり学習で得られた分類器の精度が下がってしまう。これが領域適応の問題である。領域適応の問題に対する手法は、素性ベースのものと事例ベースのものに大別できる。素性ベースの手法は学習データの素性に重みを付けた学習であり、 事例ベースの手法は学習データの事例に重みを付けた学習である。また深層学習を用いた領域適応の手法も基本的に素性ベースの手法と見なせる。

ここでは新たな素性ベースの手法を提案する。深層学習は画像の分野では大きな成功を収めているが、自然言語処理の分野での応用は必ずしもうまくいっていない。いくつか原因があるが、1 つは自然言語処理分野でサンプルは基本的に文であり、文が系列的な離散値であることから、 領域識別器の学習が困難となる点にある。ここでは、ソースドメインからのラベル付きデータとターゲットドメインからの大量のラベルなしデータを用いる敵対的学習により領域問題の解決を図る。トレーニングが進むにつれて、本手法は（i）ソースドメインの主要な学習タスクを識別し、（ii）領域間のシフトに関して不変である「ディープ」機能の出現を促進する。この適応動作は、ほとんどすべてのフィードフォワードモデルで、少数の標準レイヤーと単純な新しい勾配反転レイヤーを追加することで実現できる。

データセットとして Amazon Dataset を利用する。Amazon Dataset の領域 books,、DVD、 music の 3 つを利用し、計 6 通りの領域適応を試す。各領域適応で正解率を求め、それら平均値によって手法を評価する。提案手法の効果を確認したが、はいくつかの問題があることが判明した。

Master ' s thesis 2020

Domain Adaptation Using Domain Adversarial Neural Networks for Document Classification

Author : ZIMING ZHAO (18NM724F)
Adviser : Prof. Hiroyuki Shinnou

ABSTRACT

In this paper, we propose a method using a adversarial network for domain adaptation of document classification.

Document classification is a task for determining whether a review document is positive or negative.Document classification can be solved using supervised learning.However, if the document to be determined is a document in an area different from the area of the learning data (for example, a music review), the accuracy of the classifier obtained by the supervised learning decreases.This is the problem of domain adaptation.Methods for domain adaptation problems can be broadly classified into feature-based and case-based methods.The feature-based method is learning in which the features of the training data are weighted, and the case-based method is training in which the training data cases are weighted.A domain adaptation method using deep learning can also be considered as a feature-based method.

Here, this paper proposes a new feature-based method.Although deep learning has been very successful in the field of images, its application in the field of natural language processing has not always been successful.There are several causes, one of which is that in the field of natural language processing, samples are basically sentences, and the sentences are sequential discrete values, which makes learning a classifier difficult.Here, the domain problem is solved by adversarial learning using labeled data from the source domain and a large amount of unlabeled data from the target domain.As training progresses, the method (i) identifies key learning tasks in the source domain and (ii) facilitates the emergence of “deep” functions that are invariant with respect to shifts between domains.This adaptation can be achieved in almost all feedforward models by adding a few standard layers and a simple new gradient descent layer.

Use Amazon Dataset as a dataset,which includes three domains: books, DVD, and music.The correct answer rate is determined for each domain adaptation, and the method is evaluated based on the average value.We confirmed the effectiveness of the proposed method, but found that there were some problems.

目 次

第1章　序論

1.1　概要

本論文では文書分類の領域適応に対して、ネットワークとして敵対的ネットワークを利用する手法を提案する。

文書分類とは、レビュー文書が肯定的なものか、否定的なものかを判定するタスクである。文書分類は、教師あり学習を用いて解決できる。しかし判定先の文書が学習データの領域とは異なる領域の文書（例えば音楽のレビュー）であった場合に、教師あり学習で得られた分類器の精度が下がってしまう。これが領域適応の問題である。領域適応の問題に対する手法は、素性ベースのものと事例ベースのものに大別できる。素性ベースの手法は学習データの素性に重みを付けた学習であり、 事例ベースの手法は学習データの事例に重みを付けた学習である。また深層学習を用いた領域適応の手法も基本的に素性ベースの手法と見なせる。

領域適応の問題に対する手法は、素性ベースのものと事例ベースのものに大別できる。素性ベースの手法は学習データの素性に重みを付けた学習であり、 事例ベースの手法は学習データの事例に重みを付けた学習である。また深層学習を用いた領域適応の手法も基本的に素性ベースの手法と見なせる。

ここでは新たな素性ベースの手法を提案する。深層学習は画像の分野では大きな成功を収めているが、自然言語処理の分野での応用は必ずしもうまくいっていない。いくつか原因があるが、1つは自然言語処理分野でサンプルは基本的に文であり、文が系列的な離散値であることから、 領域識別器の学習が困難となる点にある。ここでは、ソースドメインからのラベル付きデータとターゲットドメインからの大量のラベルなしデータを用いる敵対的学習により領域問題の解決を図る。トレーニングが進むにつれて、本手法は（i）ソースドメインの主要な学習タスクを識別し、（ii）領域間のシフトに関して不変である「ディープ」機能の出現を促進する。この適応動作は、ほとんどすべてのフィードフォワードモデルで、少数の標準レイヤーと単純な新しい勾配反転レイヤーを追加することで実現できる。

データセットとして Amazon Dataset を利用する。Amazon Dataset の領域 books,、DVD、 music の3つを利用し、計 6 通りの領域適応を試す。各領域適応で正解率を求め、それら平均値によって手法を評価する。提案手法の効果を確認したが、はいくつかの問題があることが判明した。

1.2　本論文の構成

本論文では、はじめに理論とその手法について紹介する。第 2 章では文書分類の領域適応（Domain Adaptation;DA）の背景と関連研究を説明する。第 3 章では敵対的学習の先行研究と GAN につて述べる。第 4 章ではディープ領域適応の手法を説明する。第 5 章では敵対的学習を利用した実験の内容と結果を示す。第 6 章ではその実験結果について考察する。最後に第 7 章で結論を述べる。

第2章　文書分類の領域適応

2.1　文書分類

文書分類とはレビュー文書の内容が肯定的か、否定的かを判別するタスクである。ある文書分類のタスクは、教師あり学習を用いて解決可能である。このタスクの解決を目標とするデータセットは、各データが二値 (または多値) に分類やラベル付与されている。

しかし判定先の文書 (例えば音楽のレビュー) が学習データの領域 (例えば書類のレビュー) とは異なる領域の文書であった場合に、教師あり学習で得られた分類器の精度が下がってしまう。これが領域シフトの問題である。

2.2　領域適応

領域シフトの問題に対する手法は、領域適応と呼ばれる。領域適応は、ターゲット領域のラベル付きデータを用いる教師ありの手法と、それを用いない教師なしの手法に大別できる。教師ありの手法の場合、Daumé の手法 (Daumé III, Hal, 2007) が簡易でしかも能力が高いため、標準手法となっている。

本論文で扱うのは教師なしの手法である。教師なしの手法の場合、素性ベースのものと事例ベースのものに大別できる [18]。素性ベースの手法は学習データの素性に重みを付けた学習であり、事例ベースの手法は学習データの事例に重みを付けた学習である。(Domain Adaptation Using a Combination of Multiple Embeddings for Sentiment Analysis)[1] で扱うのは教師なしの手法である。事例ベースの手法は学習データの事例に重みを付けた学習であり、共変量シフトを仮定する。共変量シフトとは $P_S(c|x) = P_T(c|x)$ かつ $P_S(x) = P_T(x)$ という仮定である。共変量シフト下では、ソース領域のデ ータ x に対して確率密度比 $r = P_S(x)/P_S(x)$ を重みとした重み付き学習から、$P_S(c|x)$ を得ることができる (Sugiyama and Kawanabe, 2011)。素性ベースの手法は学習データの素 性に重みを付けた学習であり、古典的には SCL (Blitzer et al., 2006) [19] が有名である。あるいは素性ベースの手法はソース領域のデータとターゲット領域のデータを shared feature subspace W にマップする手法とも見なせ、MMD はその代表的な研究である (Borgwardt et al., 2006)。またこのタイプの研究として CORAL (Sun et al., 2016)[20] は簡易でしかも能 力が高いため近年注目されている。また深層学習を用いた領域適応の手

法も基本的に素 性ベースの手法と見なせる (Glorot et al., 2011)。CORAL を拡張した手法 (Sun and Saenko, 2016) や敵対性ネットワークを利用した手法 (Ganin and Lempitsky, 2015)(Tzeng et al., 2017) などがある。

本論文では文書分類の領域適応に対して、ネットワークとして敵対的ネットワークを利用する手法を提案する。

第3章　敵対的学習

3.1　概要

ディープフィードフォワードアーキテクチャは、さまざまな機械学習タスクおよびアプリケーション全体で、最先端技術に印象的な進歩をもたらしました。ただし、現時点では、これらのパフォーマンスの向上は、大量のラベル付きトレーニングデータが利用可能な場合にのみ発生する。同時に、ラベル付きデータが不足している問題については、大規模で深いモデルをトレーニングするのに十分なトレーニングセットを取得することも可能であるが、「テスト時間」で遭遇する実際のデータからのデータ分布のシフト 。

トレーニング分布とテスト分布の間にシフトがある場合に、識別分類器または他の予測子を学習することは、領域適応（DA）として知られている。例えば、データ表現/特徴が与えられて、固定される状況で、浅い学習の文脈で、ドメイン適応への多くのアプローチは提案されました。提案されたアプローチは、ソース（トレーニング時間）とターゲット（テスト時間）ドメインの間のマッピングを構築するので、ドメイン間の学習マッピングを使用して構成されると、ソースドメインのために学習された分類器はまた、ターゲットドメインに適用することができる。ドメイン適応アプローチのアピールは、対象ドメインデータが完全にラベル付けされていない（教師なしドメイン注釈）か、いくつかのラベル付きサンプル（半教師付きドメイン適応）を持つ場合の状況でのドメイン間のマッピングを学習できることである。以下において、我々は難しい教師なしの事例に焦点を当てている。しかし、提案されたアプローチはむしろ簡単に半監督されたケースに一般化されることができる。

そのため、本文は（i）識別性と（ii）領域不変性を結合する学習特徴に集中する。これは、これらの特徴の上で動作する 2 つの識別分類器と同様に基礎的な特徴を共同で最適化することによって達成され : (i) クラスラベルを予測し、トレーニング中とテスト時に使用されるラベル分類器（ii）トレーニングの間、ソースとターゲットドメインを区別するドメイン分類器をラベルする。分類器のパラメーターはトレーニングセットでのエラーを最小限に抑えるために最適化されるが、基になる深い特徴マッピングのパラメーターは、ラベル分類器の損失を最小限に抑え、ドメイン分類器の損失を最大化するために最適化される。後者は、最適化の過程でドメイン不変の機能が出現することを促進する。

基本的に、すべての 3 つの訓練プロセスは、標準的な層と損失関数を使用する適切に構成された深いフィードフォワードネットワークに埋め込まれることがで

き、確率勾配降下またはその修正（例えば、運動量を伴うSGD）に基づいて標準的な逆伝搬アルゴリズムを使用して訓練されることができることを示す。バックプロパゲーションによってトレーニング可能な既存のフィードフォワードアーキテクチャにドメイン適応を追加するために使用できるため、このアプローチは汎用的である。バックプロパゲーションによってトレーニング可能な既存のフィードフォワードアーキテクチャにドメイン適応を追加するために使用できるため、このアプローチは汎用的である。実際には、提案されたアーキテクチャの唯一の非標準コンポーネントは、順伝播中に入力を変更せず、逆伝播中に負のスカラーを乗算することで勾配を逆にするかなり些細な勾配反転層である。

3.2 先行研究

近年、多数のドメイン適応手法が提案されており、ここでは最も関連性の高い手法に焦点を当てている。複数の方法は、ソースとターゲット領域における特徴分布のマッチングにより教師なしドメイン適応を行う。ソースドメインからサンプルを再重み付けまたは選択することでこれを実行するアプローチもある（Borgwardt et al.,2006; Huang et al.,2006; Gong et al.,2013）[1][2][3]。他のものは、ソース分布をターゲット分布にマッピングする明示的な特徴空間変換を求めている（Pan et al.,2011; Gopalan et al.,2011; Baktashmotlagh et al.,2013）[4][5][6]。分布マッチングアプローチの重要な側面は、分布間の（非）類似性の測定方法である。ここで、人気のある選択肢の1つは、カーネルを再現するヒルベルト空間（Borgwardt et al.,2006; Huang et al.,2006）[7][2] の分布平均を一致させることだが、（Gong et al.,2012; Fernando et al.,2013）[3][8] 各ディストリビューションに関連する主軸をマップする。私たちのアプローチは、特徴空間分布のマッチングも試むが、これは、再重み付けや幾何学的変換ではなく、特徴表現自体を変更することで実現される。また、この方法では、（暗黙的に）かなり異なる方法を使用し、深い判別訓練された分類器による分離可能性に基づいて分布間の不均衡を測定する。

いくつかのアプローチでは、トレーニング分布を徐々に変更することで、ソースドメインからターゲットドメインへの段階的な移行を実行する（Gopalan et al.,2011; Gong et al.,2012）[5][3]。これらのメソッドの中では（S. Chopra & Gopalan、2013）[9]、ソースドメインのサンプルをターゲットドメインのサンプルに徐々に置き換えながら、ディープオートエンコーダーのシーケンスをレイヤーごとにトレーニングすることにより、「深い」方法でこれを行う。

これは、両方のドメインに対して単一のディープオートエンコーダーを単純にトレーニングする（Glorot et al.,2011）[10] の同様のアプローチを改善する。どちらのアプローチでも、実際の分類器/予測器は、オートエンコーダーによって学習された特徴表現を使用し、別のステップで学習される。Gloot et al .，2011 ; S . Chopra and amp; Gopalan，2013)[10][9] と対照的に、我々のアプローチは、統合学習において、機能学習、領域適応、分類器学習を統一的に行い、単一の学

習アルゴリズム（逆伝搬）を使用する。したがって、私たちのアプローチは（概念的にも実装に関しても）よりシンプルであると主張する。また、この方法は、人気のある OFFICE ベンチマークでかなり優れた結果を達成している。

上記のアプローチは教師なしドメイン適応を実行したが、ターゲットドメインからのラベル付きデータを活用することにより教師ありドメイン適応を実行するアプローチがある。ディープフィードフォワードアーキテクチャのコンテキストでは、このようなデータを使用し、ソースドメインでトレーニングされたネットワークを「微調整」できる（Zeiler & Fergus,2013; Oquab et al.,2014; Babenko et al.,2014）[11][12][13]。このアプローチでは、ラベル付きターゲットドメインデータは必要がない。同時に、そのようなデータが利用可能になったときに簡単に組み込むことができる。

私たちのアイデアに関連するアイデアは、 (Goodfellow et al.,2014)[14] で説明されている。彼らのゴールは全く異なっている（サンプルを合成することができる生成深いネットワークを構築する）。彼らがトレーニングデータの分配と合成データの分配の間の不一致を測定し、最小にする方法は、我々のアーキテクチャが測る方法と非常に類似し、2 つのドメインのための特徴配布の間の不一致を最小にする。

最後に、最近の同時報告（Tzeng et al.,2014）[15] も、フィードフォワードネットワークでのドメイン適応に焦点を当てている。それらの一連の技術は、ドメイン全体のデータ平均の距離を測定し、最小化する。このアプローチは、我々のアプローチの「一次」近似と見なすことができ、分布間のより緊密な alignment を求める。

3.3 GAN

3.3.1 概要

本節では、(Goodfellow et al.,2014)[14] を説明する。GAN は深層生成モデルの 1 つであり、基本的には画像生成に用いられる。当初、VAE よりも質の高い画像を生成できるが、学習が困難という問題があった。すぐに逆畳み込みを用いた DCGAN により、安定して学習できるようになった。その後、GAN は単純な画像生成だけでなく、画像変換や画像編集にも応用された。また GAN の生成する画像は基本的にランダムであり、どのような画像が生成されるかはわからない。Conditional-GAN はラベルと画像のペアをデータとすることで、生成された画像からそのラベルも得ることができる。このためラベル付きデータを大量に得ることができるので、識別器の精度を高めることができる。ただし Conditional-GAN の学習にはラベル付きデータしか利用していない。また Conditional-GAN の Discriminator が入力されたデータが偽物と判断した場合、画像の質が低くて偽物と判断したのか、画像とラベルの対応が悪くて偽物と判断したのかがわからないという問題がある。

画像分野で大きな成功を収めた GAN であるが、自然言語処理では画像分野ほ

ど成功しているとは言いがたい。いくつか問題があるが、1つは生成するサンプルが基本的には文であり、系列情報、つまり離散値であることである。系列的な離散値であるために Discriminator の設計や学習が困難になっている。SeqGAN では強化学習を用いることで Generator を学習し、Discriminator ではモンテカルロ探索を行っている。textGAN では Generator に潜在空間を考慮した LSTM、Discriminator では CNN、学習は MMD (Maximum Mean Discrepancy) の最小化により、SeqGAN よりも質のよい文を生成している。MaskGAN では文生成に使われる encoder-decoder モデルの学習に GAN を利用している。生成器は encoder でマスクされた単語を decoder で復元できるように学習している。また GAN の応用としては、基本的に、文を生成するタスクが対象となるので、また GAN の自然言語処理への応用としては、機械翻訳と対話の研究が活発である。機械翻訳では基本的に人間の訳か機械が作った訳かを敵対的に学習している。対話では強化学習が用いられている。

3.3.2 GAN の基本構造

一般的に、GAN で学習させる Generator (生成器) と Discriminator (識別器) にはそれぞれニューラルネットワークを使い、以下の図 3.1 のような構成で学習する。

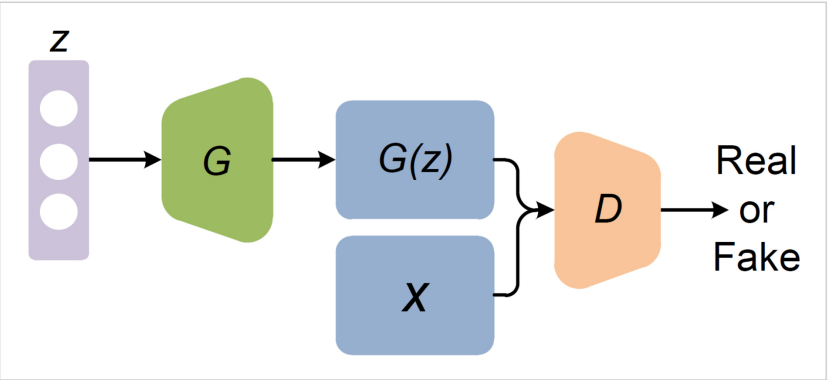

図 3.1: GAN の基本構造
z：ノイズベクトル　G:Generator　G(z)：G が生成した偽のデータ
x：本物のデータ (学習データ)　D：Discriminator

Generator はノイズベクトル z を入力とし、偽のデータ G(z) を生成する。Discriminator は、偽物のデータ G(z) と本物のデータ X を入力とし、それが本物か偽物かを判定する二値分類を行う。Generator は Discriminator が本物と判定してしまうような偽データを生成できるように学習し、Discriminator は Generator が

生成したデータが偽物だと見破れるように学習する。一般的に、Generator の入力となるノイズベクトル z には、-1〜1 の範囲で各値をランダム生成 (一様分布 or 正規分布) した 1 次元のベクトル (1 階のテンソル) が使われる。

Generator と Discriminator が影響し合って学習を進めるために、通常のニューラルネットワークとは違う損失関数が設計されている。交差エントロピーに似ているものの、Generator の損失関数に Discriminator が含まれているし、Discriminator の損失関数にも Generator が含まれており、互いに依存した式になっている。

Generator の損失関数は以下の式 (式 3.1) で定義される。

$$L_G = \frac{1}{m}\sum_{i=1}^{m} log(1 - D(G(z))) \tag{3.1}$$

つまり、Generator が生成した偽データ G(z) を Discriminator に識別させ、本物と判定される (1 が出力される) と最小になる式。D(G(z)) を最大化するという言い方もできる。

Discriminator の損失関数は以下の式 (式 3.2) のようになる。

$$L_D = \frac{1}{m}\sum_{i=1}[logD(x) + log(1 - D(G(z)))] \tag{3.2}$$

つまり、本物のデータ x を Discriminator で識別すると本物と判定 (1 が出力) され、偽物のデータ G(z) を識別すると偽物と判定 (0 が出力) されると最小になる式である。

Generator と Discriminator の 2 つの損失関数を合わせて、GAN 全体の学習目標として目的関数に定式化すると以下の式 (式 3.3) のようになる。

$$\min_G \max_D V(D,G) = E_{x\sim P_{data(x)}}[logD(x)] + E_{z\sim P(z)}[log(1 - D(G(z)))] \tag{3.3}$$

右辺の第 1 項は、Discriminator が本物データを本物と判別する期待値で、第 2 項は偽物データを偽物と判別する期待値である。Discriminator のネットワークは正しく判別したいので、上記の式を最大化しようとするが、逆に Generator のネットワークは誤認識させたいので、上記の式を最小化しようとする。

GAN の学習では Generator と Discriminator を交互に更新していく。GAN の学習プロセスは以下 2 つのステップの繰り返し。

1.Discriminator の更新

Generator のパラメータを固定した状態で Discriminator を学習する。

ランダム生成したノイズベクトル z を Generator に入力し、偽データ G(z) を生成する。そして、生成した偽データに対応する教師信号は「偽物」を表す 0 とし、本物のデータに対応する教師信号は「本物」を表す 1 として Discriminator

に入力して識別を行う。その識別誤差を逆伝搬して Discriminator のパラメータを更新する。

2.Generator の更新

Discriminator のパラメータを固定した状態で Generator を学習する。

ランダム生成したノイズベクトル z を Generator に入力し、偽データ G(z) を生成する。そして、この生成した偽データに対応する教師信号を今度は「本物」を表す 1 として Discriminator に入力して識別を行う。その識別誤差を逆伝搬して Generator のパラメータを更新する。

この 1 と 2 のステップを交互に繰り返すことで、Generator と Discriminator の性能が徐々に向上していく。両者の性能が向上することで、最終的に Generator は本物のデータと見分けがつかないほどリアルな偽データを生成できるようになる。

第4章　ディープ領域適応

4.1　モデル

ここで、ドメイン適応の提案モデルを詳しく説明する。モデルは入力サンプル x∈ X で機能すると仮定する。ここで、X は入力空間とラベル空間 Y からの特定のラベル（出力）y である。以下では、Y が有限集合（Y = 1,2...L）である分類問題を想定しているが、このアプローチは一般的であり、他のディープフィードフォワードモデルが処理できる出力ラベルスペースを処理できる。さらに、$X \bigotimes Y$ に S（x、y）と T（x、y）の 2 つの分布が存在すると仮定する。これは、ソース分布とターゲット分布（またはソースドメインとターゲットドメイン）と呼ばれる。両方の分布は複雑で未知であると想定され、さらに類似しているが異なる（言い換えれば、S は何らかのドメインシフトによって T から「シフト」されている）。

最終的な目標は、ターゲット分布の入力 x を与えられたラベル y を予測できるようにすることである。トレーニング時に、限界分布 S（x）および T（x）に従って配布されたソースドメインとターゲットドメインの両方からのトレーニングサンプル $\{x_1, x_2, ..., x_N\}$ 大きなセットにアクセスできる。i 番目の例では、d_i でバイナリ変数（ドメインラベル）を示します。これは、xi がソース分布 ($d_i = 0$ の場合は $x_i \sim S(x)$) かターゲット分布 ($d_i = 1$ の場合は $x_i \sim T(x)$) の場合。 ソース分布 ($d_i = 0$) の例では、対応するラベル y_i∈ Y はトレーニング時に既知である。ターゲットドメインの例では、トレーニング時にラベルがわからないため、テスト時にそのようなラベルを予測する。

ここで、各入力 x がそのラベル y∈ Y とそのドメインラベル $\{0, 1\}$ を予測するディープフィードフォワードアーキテクチャを定義する。入力 x が最初にマッピング G_f（特徴抽出器）によって D 次元の特徴ベクトル f∈ $\mathcal{R}^{\mathcal{D}}$ にマッピングされると仮定する。フィーチャマッピングにはいくつかのフィードフォワードレイヤーも含まれる場合があり、このマッピングのすべてのレイヤーのパラメーターのベクトルを θ_f すなわち f = G_f（x ; θ_f）として示す。次に、特徴ベクトル f はマッピング G_y（ラベル予測器）によってラベル y にマッピングされ、このマッピングのパラメーターは θ_y で示される。最後に、同じ特徴ベクトル f がドメインラベル d に、パラメーター θ_d を使用したマッピング G_d（ドメイン分類器）によってマッピングされる。（図 4.1）

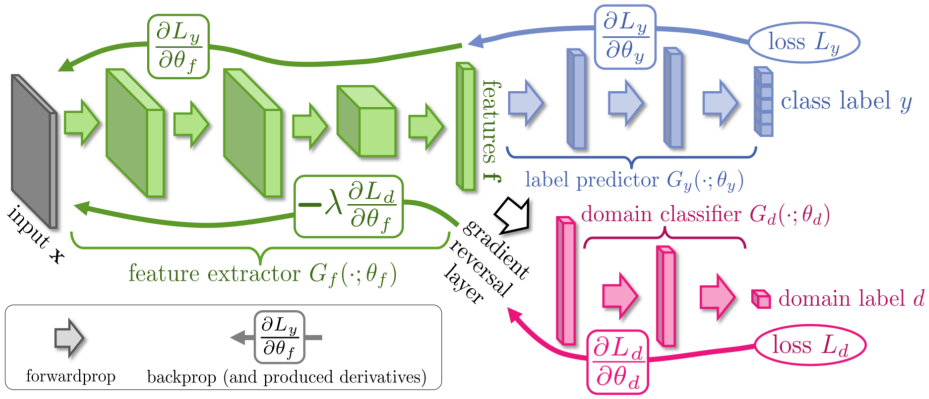

図 4.1：ディープ領域適応のモデル

学習段階では、学習セットの注釈部分（すなわち，ソース部分）上のラベル予測損失を最小化することを目的とし、ソース領域サンプルの経験的損失を最小にするために、特徴抽出器とラベル予測器のパラメータを最適化した。これにより、特徴 f の識別性と、ソースドメインでの特徴抽出とラベル予測の組み合わせの全体的な良好な予測パフォーマンスが保証される。

同時に、特徴 f のドメイン不変の機能を作りたい。すなわち、分布 $S(f) = \{G_f(s;)\theta_f|x \sim S(x)\}$ と $T(f) = \{G_f(s;)\theta_f|x \sim T(x)\}$ とを同様にしたい。共変量シフトの仮定の下では、これにより、ターゲットドメインのラベル予測精度がソースドメインと同じになる（Shimodaira,2000）[17]。ただし、f が高次元であり、学習が進むにつれて分布自体が常に変化していることを考えると、分布 S（f）と T（f）の非類似性の測定は重要である。非類似性を推定する一つの方法は、領域分類器のパラメー θ_d が最適な方法で 2 つの特徴分布を識別するために訓練されたことを前提として、ドメイン分類器 G_d の損失を調べることである。

この観察は私たちのアイデアにつながる。トレーニング時に、ドメイン不変の特徴を取得するために、ドメイン分類子の損失を最大化する特徴マッピングのパラメーター θ_f を探し (2 つの機能分布を可能な限り類似させることにより)、同時に、ドメイン分類器の損失を最小化するドメイン分類器のパラメーター θ_f を探す。さらに、ラベル予測子の損失を最小限に抑えるよう努めている。

$$E(\theta_f, \theta_y, \theta_d) = \sum_{\substack{i=1..N \\ d_i=0}} L_y(G_y(G_f(x_i;\theta_f);\theta_y), y_i) -$$

$$\lambda \sum_{i=1..N} L_d(G_d(G_f(x_i;\theta_f);\theta_d), y_i) = \sum_{\substack{i=1..N \\ d_i=0}} L_y^i(\theta_f, \theta_y) -$$

$$\lambda \sum_{i=1..N} L_d^i(\theta_f, \theta_d) \tag{4.1}$$

ここで、L_y（・,・）はラベル予測の損失（例：多項)、L_d（・,・）はドメイン分類の損失（例：ロジスティック)、L_y^i と L_y^i は、i 番目のトレーニングの例で評価された対応する損失関数を示す。私たちのアイデアに基づき、関数（4.1）のサドルポイントを提供するパラメーター $\widehat{\theta}_f$、$\widehat{\theta}_y$、$\widehat{\theta}_d$ を探している:

$$(\widehat{\theta}_f, \widehat{\theta}_y) = \arg\min_{\theta_f, \theta_y} E(\theta_f, \theta_y, \widehat{\theta}_d) \tag{4.2}$$

$$\widehat{\theta}_d = \arg\min_{\theta_d} E(\widehat{\theta}_f, \widehat{\theta}_y, \theta_d) \tag{4.3}$$

サドルポイントでは、ドメイン分類器のパラメータ θ_d は、ドメイン分類損失を最小にする（マイナス符号で（4.1）に入るので)、ラベル予測器のパラメータ θ_y はラベル予測損失を最小にする。特徴マッピングパラメーター θ_f は、ラベル予測損失を最小限に抑える（つまり、特徴が判別的である）一方で、ドメイン分類損失を最大限にする（つまり、特徴はドメイン不変である)。パラメーター λ は、学習中に特徴を形作る 2 つの目的の間のトレードオフを制御する。

以下では、標準的な確率的勾配ソルバー（SGD）をサドルポイント（4.2）-（4.3）の検索に適応できることを示す。

4.2 バックプロパゲーションによる最適化

サドルポイント (4.2)-(4.3) は、以下の確率的アップデートの固定点として見つかることができる：

$$\theta_f \leftarrow \theta_f - \mu(\frac{\partial L_y^i}{\partial \theta_f} - \lambda \frac{\partial L_d^i}{\partial \theta_f}) \tag{4.4}$$

$$\theta_y \leftarrow \theta_y - \mu \frac{\partial L_y^i}{\partial \theta_y} \tag{4.5}$$

$$\theta_d \leftarrow \theta_d - \mu \frac{\partial L_d^i}{\partial \theta_d} \tag{4.6}$$

ここで、μ は学習率である（時間とともに変化する)。更新（4.4）-（4.6）は、ラ

ベル予測器とドメイン分類器に供給される特徴抽出器を含むフィードフォワードディープモデルの確率的勾配降下（SGD）更新に非常に似ている。差別は、（4.4）-λ因子である（このような因子がなければ、確率的勾配降下は、ドメイン分類損失を最小化するために、ドメイン間で特徴が異なるようにすることを試みる）。SGD としての（4.4）-（4.6）の直接実現は可能ではないが、SGD（およびその変種）が深層学習のために大部分のパッケージで実行される主な学習アルゴリズムであるので、SGD のいくつかの形に更新（4.4）-（4.6）を減らすことは非常に望ましい。

幸いなことに、このような削減は、次のように定義された特別な勾配反転層（gradient reversal layer,GRL）を導入することで実現できる。勾配反転層には、関連するパラメーターがないである（メタパラメーターλは別として、λは逆伝播によって更新されない）。前方伝搬の間、GRL はアイデンティティ変換として機能する。ただし、バックプロパゲーション中に、GRL は後続のレベルから勾配を取得し、それに-λを乗算し、前のレイヤーに渡す。ディープラーニング用の既存のオブジェクト指向パッケージを使用したこのようなレイヤーを実装するのは簡単だが、forwardprop（恒等変換）、backprop（定数で乗算）、およびパラメーターの更新（なし）の手順を定義するのは簡単である。

上で定義された GRL は、特徴抽出器とドメイン分類器の間に挿入され、図 4.1 に示すアーキテクチャになる。バックプロパゲーションプロセスが GRL を通過すると、GRL の下流にある損失の偏微分（つまり、L_d） GRL の上流にあるレイヤーパラメーター（つまり、θ_f）に-λが乗算される。つまり、$\frac{\partial L_d}{\partial \theta_f}$ は、実質的に-$\lambda\frac{\partial L_d}{\partial \theta_f}$ に置き換えられる。結果として生じるモデルの SGD を実行することは、更新（4.4）-（4.6）を実行し、（4.1）の鞍点に収束する。数学的には、順方向および逆方向伝搬の振る舞いを記述する 2 つの（互換性のない）方程式で定義される「疑似関数」$R_\lambda(x)$ として勾配反転層を正式に扱うことができる：

$$R_\lambda(x) = x \tag{4.7}$$

$$\frac{dR_\lambda}{d_X} = -\lambda\mathcal{I} \tag{4.8}$$

ここで、$\mathcal{I}$ は単位行列である。次に、$\theta_f, \theta_y, \theta_d$ の客観的な「擬似関数」を定義できる。これは、メソッド内の確率的勾配降下法によって最適化されている：

$$\widetilde{E}(\theta_f, \theta_y, \theta_d) = \sum_{\substack{i=1..N \\ d_i=0}} L_y(G_y(G_f(x_i;\theta_f);\theta_y), y_i) + \sum_{i=1..N} L_d(G_d(G_f(x_i;\theta_f);\theta_d), y_i) \tag{4.9}$$

実行された更新（4.4）-（4.6）はそれから（4.9）のために SGD をするように実行されることができ、同時にドメイン不変で、特徴的である特徴の出現につながる。学習後、ラベル予測子 $y(x) = G_y(G_f(x;\theta_f);\theta_y)$ を使用し、ターゲットドメ

イン（およびソースドメイン）からのサンプルのラベルを予測できる。

上記で概説された単純な学習手順は、（Goodfelet et al.,2014）[14] で提案されている方針に沿って再導出/一般化されることができる。

第5章　実験

5.1　実験設定

実験で使用したデータセットは、以下のサイトで公開されている Amazon のレビュー文書である。

https://webis.de/data/webis-cls-10.html

このデータセットは books(B)、DVD(d) と music(M) の 3 つの領域を持ち、3 つの領域で実験を行った。データ数は教師データとテストデータがそれぞれ 2000 文書存在する。各領域に依存する文書数を表 5.1 に示す。評価の 4 と 5 を positive、評価の 1 と 2 を negative とした感情分析データとして利用できる。予め、各文書を Bert により 768 次元の埋め込み表現に変換した。

表 5.1: 実験データ

domain	training	test
books	2000	2000
DVD	2000	2000
music	2000	2000

5.2　実験結果

領域適応としては、B → D、B → M、D → B、D → M、 M → B、M → D の 6 通りが存在する。それぞれの領域適応に対して、前章で述べた敵対的ネットワークを利用した。実験結果を表 5.2 と図 5.1 に示す。

表 5.2: 実験結果

ソース領域	ターゲット領域	敵対的学習
books	DVD	0.7870
books	music	0.7944
DVD	books	0.7925
DVD	music	0.7990
music	books	0.7720
music	DVD	0.7850

この結果を利用し、第五章では領域適応を用いられない手法など、色々な手法を用いたの結果を比較する。

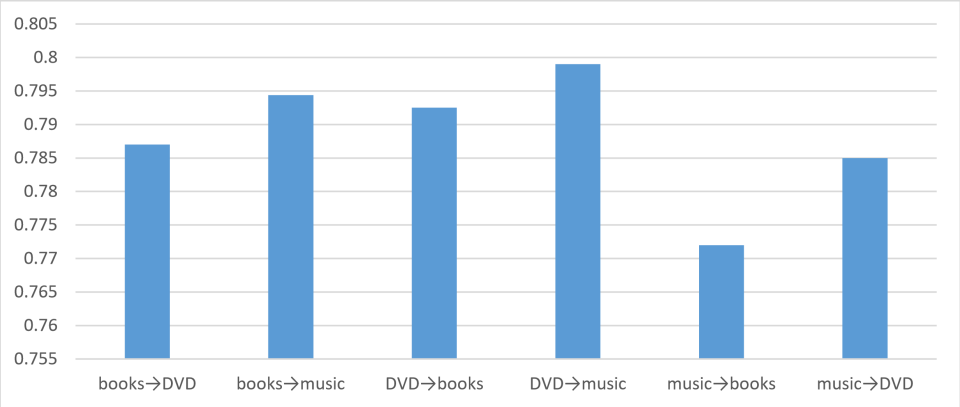

図 5.1: 実験結果

第6章　考察

領域適応としては、B → D、B → M、D → B、D → M、 M → B、M → D の 6 通りが存在する。それぞれの領域適応に対して、SVD 及び提案手法を用いた結果 (テストデータに対する正解率) を表 6.1 と図 6.1 に示す。SVD とは、次元縮約として特異値分解を用いて文書を 100 次元ベクトル) に変換し、文書の特徴ベクトルを用いた手法である。表 6.1 の SVD は [1] の表から取り出した。表 6.1 中の source only は領域適応の手法を用いず、単にソース領域の訓練データから構築した分類器をそのままターゲット領域のテストデータに適用した結果である。また train on target はターゲット領域の訓練データを用いて分類器を学習し、それをターゲット領域のテストデータに適用した結果である。

表 6.1:　領域適応手法の結果の比較

source target	books DVD	books music	DVD books	DVD music	music books	music DVD	平均
source only	0.7914	0.7904	0.7990	0.8080	0.7810	0.7845	0.7923
SVD	0.7360	0.7050	0.7260	0.7410	0.6835	0.7205	0.7186
提案手法	0.7870	0.7944	0.7925	0.7990	0.7720	0.7850	0.7883
train on target	0.8115	0.8255	0.8149	0.8255	0.8149	0.8115	0.8173

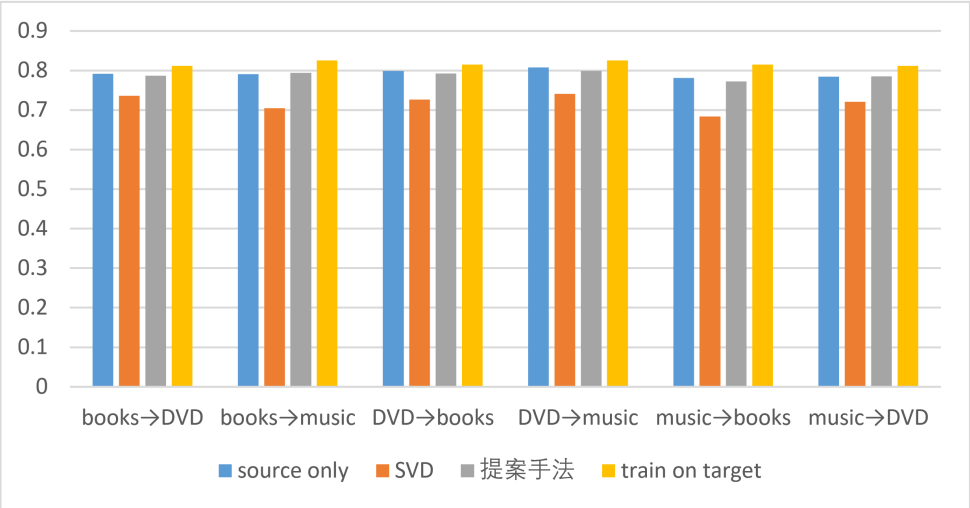

図 6.1: 領域適応手法の結果の比較

埋め込み表現を利用した手法である SVD および提案手法を比較すると、全部 6 個の領域適応の中で提案手法が最も高い正解率を出している。提案手法は深層学習の手法としては、優秀であると言える。

表 6.1 の source only と提案手法を比較すると、6 個の領域適応の中で 4 個について source only が最も高い正解率を出し、残りの 2 個は提案手法が最も高い正解率を出している。図 6.2 は四つの領域適応手法の平均値である。6 個の平均を取ると、明らかに source only の方が正解率が高い。本実験データに限れば、敵対的学習の手法は領域適応には効果がないと言える。

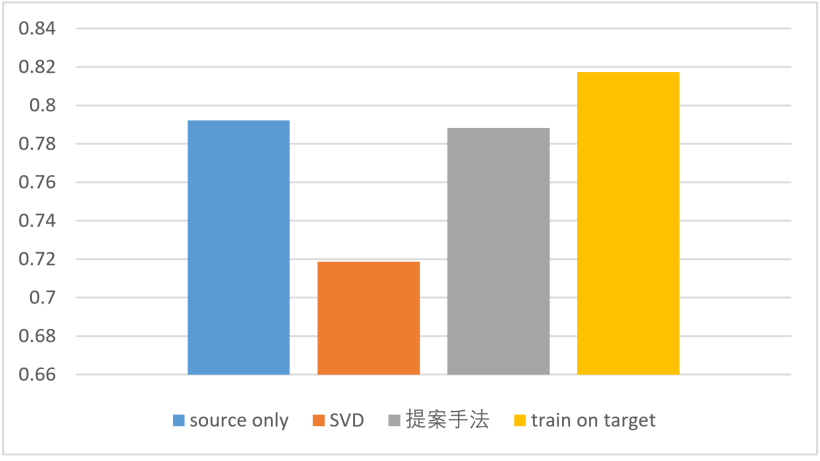

図 6.2: 四つの領域適応手法の平均値

第7章　結論

本論文では文書分類の領域適応に対して、敵対的ネットワークを利用する手法を提案した。データセットとして Amazon Dataset を利用した。Amazon Dataset の領域 books, DVD, music の 3 つを利用し、計 6 通りの領域適応を試す。各領域適応で正解率を求め、それら平均値によって手法を評価する。実験では確埋め込み表現を利用した手法である SVD と比較することで、提案手法の優位性を示した。ただし提案手法では領域適応に対して効果はすくなかった。事例ベースの手法と素性ベースの手法は簡単に組み合わせることができるので、今後は事例ベースの手法を組み合わせることを試していきたい。

謝辞

本研究を進めるにあたり、多くのご指導、ご協力を頂いた指導教員の新納浩幸教授に感謝致します。また、日常の議論を通して多くの知識や示唆を頂いた佐々木稔講師、古宮嘉那子講師と新納研究室の皆様に感謝します。

参考文献

[1]Hiroyuki Shinnou,Xinyu Zhao.Kanako Komiya .Domain Adaptation Using a Combination of Multiple Embeddings for Sentiment Analysis.In PACLIC32, 2018.

[2]Huang, Jiayuan, Smola, Alexander J., Gretton, Arthur,Borgwardt, Karsten M.,and Scholkopf, Bernhard. Correcting sample selection bias by unlabeled data.In NIPS,pp. 601–608, 2006.

[3]Gong, Boqing, Grauman, Kristen, and Sha, Fei. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In ICML, pp. 222–230, 2013.

[4]Pan, Sinno Jialin, Tsang, Ivor W., Kwok, James T., and Yang, Qiang. Domain adaptation via transfer component analysis. IEEE Transactions on Neural Networks, 22(2): 199–210, 2011.

[5]Gopalan, Raghuraman, Li, Ruonan, and Chellappa, Rama. Domain adaptation for object recognition: An unsupervised approach. In ICCV, pp. 999–1006, 2011.

[6]Baktashmotlagh, Mahsa, Harandi, Mehrtash Tafazzoli, Lovell, Brian C., and Salzmann, Mathieu. Unsupervised domain adaptation by domain invariant projection. In ICCV, pp. 769–776, 2013.

[7]Borgwardt, Karsten M., Gretton, Arthur, Rasch, Malte J., Kriegel, Hans-Peter, Scholkopf, Bernhard, and Smola, ¨ Alexander J. Integrating structured biological data by kernel maximum mean discrepancy. In ISMB, pp. 49– 57, 2006.

[8]Fernando, Basura, Habrard, Amaury, Sebban, Marc, and Tuytelaars, Tinne. Unsupervised visual domain adaptation using subspace alignment. In ICCV, 2013.

[9]S. Chopra, S. Balakrishnan and Gopalan, R. Dlid: Deep learning for domain adaptation by interpolating between domains. In ICML Workshop on Challenges in Representation Learning, 2013.

[10]Glorot, Xavier, Bordes, Antoine, and Bengio, Yoshua. Domain adaptation for large-scale sentiment classification: A deep learning approach. In ICML, pp. 513–520, 2011.

[11]Zeiler, Matthew D. and Fergus, Rob. Visualizing and understanding convolutional networks. CoRR, abs/1311.2901, 2013.

[12]Oquab, M., Bottou, L., Laptev, I., and Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In CVPR, 2014

[13]Babenko, Artem, Slesarev, Anton, Chigorin, Alexander, and Lempitsky, Victor S. Neural codes for image retrieval. In ECCV, pp. 584–599, 2014.

[14]Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative adversarial nets. In NIPS, 2014.

[15]Tzeng, Eric, Hoffman, Judy, Zhang, Ning, Saenko, Kate, and Darrell, Trevor. Deep domain confusion: Maximizing for domain invariance. CoRR, abs/1412.3474, 2014.

[16]Borgwardt, Karsten M., Gretton, Arthur, Rasch, Malte J.,Kriegel, Hans-Peter, Scholkopf, Bernhard, and Smola,Alexander J. Integrating structured biological data by kernel maximum mean discrepancy. In ISMB, pp. 49–57, 2006.

[17]Shimodaira, Hidetoshi. Improving predictive inference under covariate shift by weighting the log-likelihood function. Journal of Statistical Planning and Inference, 90(2):227–244, October 2000.

[18]Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. Knowledge and Data Engineering, IEEE Transactions on, Vol. 22, No. 10, pp. 1345–1359, 2010.

[19]John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In EMNLP-2006, pp. 120–128, 2006.

[20]Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In Computer Vision–ECCV 2016 Workshops, pp. 443– 450, 2016.