

11章 11.4まで

11.0 冒頭文

11.1 はじめに

前提条件1 データが少ない

そのための主な手法

前提条件2 ラベルなしデータ

それ以外の手法

11.2 データ拡張

11.2.1 基本的な考え方

前提条件 データ拡張

11.2.2 主に画像に対して有効な方法

11.2.3 主に言語に対して有効な方法

11.2.4 その他の方法

11.2.5 データ拡張の自動探索

11.3 転移学習

11.3.1 概要

11.3.2 特徴抽出器としての利用

11.3.3 ファインチューニング

11.3.4 ImageNet事前学習

11.3.5 事前学習の有効性

11.4 半教師あり学習

11.4.1 概要

11.4.2 一貫性正則化

11.0 冒頭文

1. 本書で紹介している手法は、多くの訓練データが必要という制限がある
 - a. これを用意するのは難しい
 - b. 少数データで分析する方法を模索する
2. 以下の手法で、そのような制限下の分析を可能にする
 - a. データ拡張
 - b. 転移学習
 - c. 半教師あり学習
 - d. 自己教師学習

- e. マルチタスク学習
- f. ドメイン適応
- g. 少数事例学習
- h. 能動学習

11.1 はじめに

前提条件1 データが少ない

1. 手元のデータからタスク \mathcal{T} のために学習したい
 - a. 主にクラス分類を想定する
 - b. しかし、大部分はクラス分類以外でも使える
2. 手元のデータは $\mathcal{D} = \{(\mathbf{x}_n, d_n)\}_{n=1, \dots, N}$
 - a. N は小さい
 - b. だが、これを用いて学習用の準備を行う

そのための主な手法

1. データ拡張(11.2節)
 - a. 後述の手法(前述のc.からh.の手法)のもとになる
 - b. 単体でもよくつかわれる
2. 転移学習(11.3節)
 - a. 広い意味ではタスク \mathcal{T}' の学習経験を利用することを指す
 - i. 日経平均株価の予測で使ったモデルをダウ平均株価の予測に活かす、のような話？
 - b. 狭い意味だと特定の手法を指す
 - i. ここでは具体的な言及なし

前提条件2 ラベルなしデータ

1. 手元のデータは $\mathcal{D}_{\text{UL}} = \{\mathbf{x}_n\}_{n=1, \dots, N}$

- a. 正解ラベルがない
- b. 半教師あり学習や自己教師あり学習は、このようなデータを対象としている

それ以外の手法

その他の手法の概要は、以下の通り。

マルチタスク学習	異なる複数タスクを1つのネットワークで学習する
ドメイン適応	訓練データとテストデータの間で統計的な分布にずれがある場合に適用する
少数事例学習	訓練データの数が極端に小さい場合を対象にした学習
能動学習	ラベルなしデータから限られたサンプルを選んで正解ラベルをつける(よいサンプルを選ぶ方法)

11.2 データ拡張

11.2.1 基本的な考え方

前提条件 データ拡張

1. $\mathcal{D} = \{(\mathbf{x}_n, d_n)\}_{n=1, \dots, N}$ をもっている
 - a. \mathbf{x}_n を適当な変換 φ で $\mathbf{x}'_n \equiv \varphi(\mathbf{x}_n)$ とする
 - b. 新たなサンプル (\mathbf{x}'_n, d_n) として利用する
2. 変換は自由度が高いが、以下の制約などがある
 - a. 変換によってラベルが変わらない
 - b. テストデータに出てきそうなデータにしないといけない
3. 数だけは際限なく増やすことができる

11.2.2 主に画像に対して有効な方法

- そもそもデータ拡張は画像データに対して有用
- 画像データを変化させても正解ラベルが変化しないような変換をする必要がある
- 図11.1にあるような変換を加える

- 図11.2にあるよう、正解ラベルが変わってしまいそうな加工等はよくない
 - また、タスクによって有効な加工が違う点にも注意
 - カットアウト、ランダム消去は物体認識に効果が高い
- 音声でも同様の有効な加工が存在する

11.2.3 主に言語に対して有効な方法

- 言語の加工は、意味が同じ言い換えを作る方法が考えられる
- 具体的には以下の方法がある
 - 文中の単語を類義語に入れ替える
 - 機械翻訳で別言語にしたものを再翻訳する
- 文の埋め込みベクトルにノイズを足す方法も考えられる
- 多様な言い換えができるかどうか重要

11.2.4 その他の方法

- クラス分類を対象とするmixupという方法がある
- 2つのサンプルを足し合わせる方法
- 特徴量、正解ラベルをそれぞれ適当な加重和で表現する
 - (\mathbf{x}_i, d_i) と (\mathbf{x}_j, d_j) について適当な λ で以下のようにサンプルを作る
 - $(\lambda \mathbf{x}_i + (1 - \lambda) \mathbf{x}_j, \lambda d_i + (1 - \lambda) d_j)$
- λ は適当なベータ分布からサンプルしたもの

11.2.5 データ拡張の自動探索

- 変換の候補が多数あるとき、その中から選択するのは重要
- バリデーションデータへの予測精度が最大化されるように変換パラメータを選択するような手法がある
 - パラメータ空間の探索方法もよく提案されている

11.3 転移学習

11.3.1 概要

1. $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1, \dots, N}$ と \mathcal{D}' をもっている
 - a. 前者については量が十分でなく、後者のデータは十分にある
 - i. 新規に上場した銘柄の株価を既存の銘柄の株価から推定するような話？
 - b. \mathcal{D}' で学習したタスク \mathcal{T}' と \mathcal{D} で学習したタスク \mathcal{T} には類似性が必要
 - c. 出力の空間は違っていてもかまわないが、特徴量・入力は同じ必要がある
2. 以下のような方法で転移学習を行う
 - a. あるネットワークで \mathcal{D}' を使いタスク \mathcal{T}' の学習をし、その結果 \mathbf{f}' を利用する
 - i. このタスク \mathcal{T}' の学習を \mathcal{T} の事前学習と呼ぶ
 - ii. 逆に、タスク \mathcal{T} は $\mathcal{T}\mathcal{T}'$ を下流タスクと呼ぶ
 - iii. タスク \mathcal{T}' を十分学習できたものとする
 - b. 結果 \mathbf{f}' を下流タスクのための特徴抽出器として使うことができる
 - c. 結果 \mathbf{f}' の構造、一部重みをコピーして別のネットワークを作る方法もある
 - i. この方法はファインチューニングと呼ばれる
3. 難しい点もある
 - a. タスクの類似度を測るのは難しい
 - b. 事前に転移学習の成否を予測するのも難しい

11.3.2 特徴抽出器としての利用

1. \mathbf{f}' の中間層を1つ選び、入力に対する中間出力とする
 - a. 入力 \mathbf{x}_i に対して選んだ層の出力 $\mathbf{z}(\mathbf{x}_i)$ を中間出力とする
 - b. $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1, \dots, N}$ から $\tilde{\mathcal{D}} = \{(\mathbf{z}(\mathbf{x}_n), \mathbf{y}_n)\}_{n=1, \dots, N}$ をつくる
 - c. 新たに作ったデータがよいものであれば、学習アルゴリズムはシンプルでも性能がでると想定できる
 - d. 層の選び方、
 - e.
 - f.
 - g.

h.
i.
j.
k.
l.
m.
n.
o.
p.
q.
r.
s.
t.
u.
v.
w.
x.
y.
z.
aa.
ab.
ac.
ad.
ae.
af.
ag.
ah.
ai.

aj.
ak.
al.
am.
an.
ao.
ap.
aq.
ar.
as.
at.
au.
av.
aw.
ax.
ay.
az.
ba.
bb.
bc.
bd.
be.
bf.
bg.
bh.
bi.
bj.
bk.

bl.
bm.
bn.
bo.
bp.
bq.
br.
bs.
bt.
bu.
bv.
bw.
bx.
by.
bz.
ca.
cb.
cc.
cd.
ce.
cf.
cg.
ch.
ci.
cj.
ck.
cl.
cm.

- cn.
- co.
- cp.
- cq. 複数の
- cr. (\mathbf{x}_i, d_i) を入力として選んだ中間層の出力を利用して新たなデータ $(\mathbf{z}(\mathbf{x}_i), d_i)$ を作って学習する
- cs. シンプルな予測器でも、選択した中間層がよければ性能がでると想定できる
- ct. 選択する中間層、および複数の層の出力を組み合わせるバリエーションに自由度がある
 - i. 選び方は、経験的な検証に依存する

11.3.3 ファインチューニング

1. \mathbf{f}' の一部をコピーして、新たなネットワーク \mathbf{f} をつくる
 - a. \mathcal{T}' と \mathcal{T} が近ければネットワークの変化はごくわずかだと想定され、それが名前の由来となっている
 - b. 単純に出力層以外をコピーする方法などが典型
 - i. 出力層のみ、クラス数の増減などの影響から作り直す必要がある
 - ii. コピーした層に追加で中間層を追加、あるいは部分的にコピーから削減するなどをしてよい
 - c. コピー、加工して作ったネットワーク \mathbf{f} を \mathcal{D} を用いて学習する
 - i. 追加した層の重みのみランダムで、そのほかはコピー元の値で初期化する
 - ii. この状態の \mathbf{f} を事前学習済みであると呼ぶ
 - d. 学習するパラメータ数は調整できる
 - i. 訓練は一部の層の重みだけ更新することもできる
 - ii. 少数データによる学習となるので、すべて更新する方法が裁量ではないと思われる
 1. 過剰適合のおそれがあるため
 - iii. 典型的には、出力層に近い一部の層のみを訓練する傾向にある

1. 入力に近い層よりも、出力に近い層のほうが具体的なタスク依存度が高いとされているため

11.3. 4 ImageNet事前学習

1. 物体認識をするための大きなデータセットを用いた事前学習
 - a. 学習結果のネットワークを前節、前々節の方法で学習できる
 - b. 手持ちの訓練データが小さくても、学習に利用できる
 - i. 物体認識から、タスクとしては遠い単眼深度推定までそれが可能

11.3. 5 事前学習の有効性

1. 事前学習に3つの有効性が示唆されている
 - a. 予測精度の向上
 - b. 学習の収束速度の向上
 - c. 推論時の頑健性の向上
2. 予測精度の向上について
 - a. 事前学習を通じて獲得した特徴抽出の能力が、下流タスクでも再利用されるという理解
 - b. 下流タスクのデータ量が十分な場合にはそれほどの効果がないことも報告されている(179より)
3. 学習の収束速度の向上
 - a. 少ない重みの更新で精度が向上すること
 - i. 下流タスクのデータセットが大きくても期待できる能力とされる(180、236より)
 - ii. 事前学習がランダムな初期値よりも統計的によい初期値に重みを重みに与えているためという理解
4. 推論時の頑健性の向上
 - a. 以下について性能が向上することを指す(181、182より)
 - i. GAN
 - ii. 分布外入力検出
 - iii. 確信度の校正精度

- b. 事前学習なしだと最低限の特徴空間しか得られないからとされている
5. 残された課題
- a. 以下の解明されていない部分がある(180、326より)
 - i. 統計的な性質が異なる、特徴の再利用がほぼないタスク間でも性能が向上する
 - 1. 事前学習用のデータセットが大きいから平気...というような理解

11.4 半教師あり学習

11.4.1 概要

1. $\mathcal{D}_L = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1, \dots, N}$ と $\mathcal{D}_{UL} = \{(\mathbf{x}'_n)\}_{n=1, \dots, N'}$ をもっている
 - a. 前者についてはラベルがあるが少数で、後者についてはラベルがないが多い、といった状態
 - i. アノテーションコストが高いが、データ自体は結構ある...といった状態
 - b. \mathcal{D}_L は普通の教師あり学習で、それ以外はいろいろ損失関数などを準備して訓練に利用する流れ
 - i. 各手法の特徴はラベルなしデータの利用方法にある
 - c. 最小化したいのは2つの損失関数 $E'(\mathbf{w}; \mathcal{D}_{UL})$ と $E(\mathbf{w}; \mathcal{D}_L)$ の和
 - i. 後者は普通の教師あり学習として考える
 - ii. 前者の特徴づけの異なる3つの手法を紹介する
 1. 一致性正則化
 2. 疑似ラベル
 3. エントロピー最小化

11.4.2 一致性正則化

1. $E'(\mathbf{w}; \mathcal{D}_{UL})$ を以下のように特徴づける場合