ISSN 2186-5647

MLP-Mixer の重みの共有

日大生産工　○色川 善哉　　日大生産工　山内 ゆかり

1. まえがき

近年、画像認識領域では、畳み込みニューラルネットワークやアテンション機構を用いたネットワークが優れた性能を発揮している。Google Researchは畳み込みやアテンション機構を使わないMLP-Mixer[1]を提案し、空間ごととチャンネルごとに学習する層を追加することで、多層パーセプトロンのみでも優れた性能を発揮していることが報告がされている。しかし、アルゴリズムに全結合層が多く、重みなどのパラメータ数が多いため、メモリを多く消費するという問題がある。

本研究では，パラメータ数が多いという問題において、畳み込みニューラルネットワークのカーネルのように重みを層ごとに共有することを提案し、従来研究と提案手法をパラメータ数及び認識精度を比較し、重みを共有したときに認識精度がどう変化するかについて報告する。

2. 従来研究

2.1 MLP-Mixer

MLP-Mixer[1]は多層パーセプトロン（MLP）のみを用いたアーキテクチャである。画像を大きさ $P \times P$ のパッチに分割し、パッチごとに独立して扱う層（Token Mixing MLP層）とパッチの位置ごとに独立して扱う層（Channel Mixing MLP層）の2種類の層からなるMixer層を主な層として構成している。さらに、Mixer層に突入する前にパッチごとに全結合をする層、Mixer層から出た後にチャンネルごとに平均を出して要素数を $P \times P$ にする層（Global Average Poolingをする層）、それをもとに全結合をする層、画像認識をする層からなる。

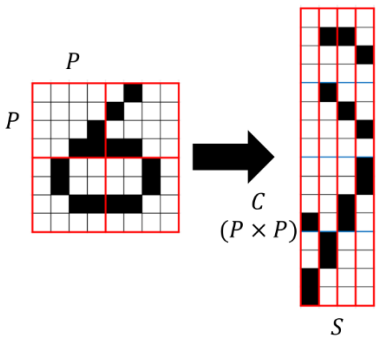

Figure 1　画像の分割方法

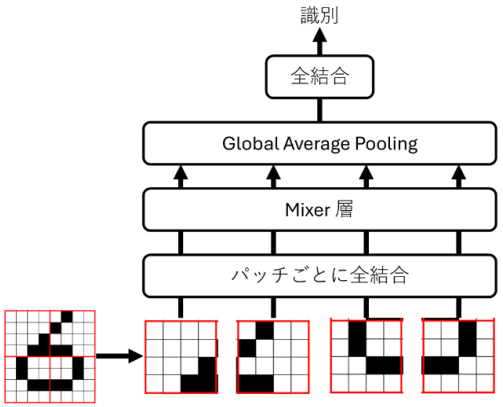

Figure 2　MLP-Mixer のアルゴリズム

Figure 1は、画像の分割方法を図に表したものである。C はチャンネル数、S はパッチ数である。また、Figure 2は、MLP-Mixerのアルゴリズムを示したものである。

2.2 Mixer 層

Mixer層では、主にToken Mixing MLP層とChannel Mixing MLP層で構成されている。さらに、転置をする層、レイヤー正規化をする層からなる。また、スキップ接続、ドロップアウトも含まれている。Figure 3にMixer層のアルゴリズムを示す。また、式に表すと以下のようになる。

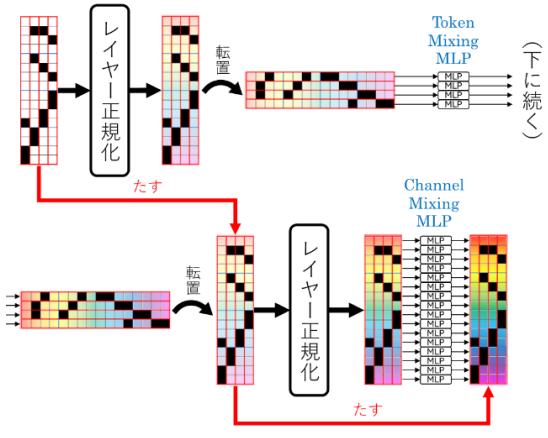

Figure 3　Mixer 層のアルゴリズム

$$\boldsymbol{U}_{*,i} = \boldsymbol{X}_{*,i} + \boldsymbol{W}_2\sigma\left(\boldsymbol{W}_1 \mathrm{LayerNorm}(\boldsymbol{X})_{*,i}\right), \quad \text{for } i = 1 \ldots C \tag{1}$$

MLP-Mixer Weight Sharing

Yoshiya IROKAWA and Yukari YAMAUCHI

$$\boldsymbol{Y}_{j,*} = \boldsymbol{U}_{j,*} + \boldsymbol{W}_4\sigma\left(\boldsymbol{W}_3\mathrm{LayerNorm}(\boldsymbol{U})_{j,*}\right), \quad \text{for } j = 1 \ldots S \tag{2}$$

式(1)は、入力層から受け取ったデータ $\boldsymbol{X}$ をレイヤー正規化してパッチ（Figure 1 の S 側）を基準にデータを分け、重み $\boldsymbol{W}_1$ と掛け合わせて活性化関数 σ を適用し、重み $\boldsymbol{W}_2$ と掛け合わせたものをさきほどの $\boldsymbol{X}$ と足し合わせて $\boldsymbol{U}$ を出力しているのを表している。また式(2)は先ほどのデータ $\boldsymbol{U}$ を、チャンネル（Figure 1 の C 側）を基準にデータを分けているところを除いて式(1)と同じことを行っていることを表している。

塊で学習するところと離れた場所同士で学習するところがあることによって、画像の特徴を学習している。Token Mixing MLP 層と Channel Mixing MLP 層に使われている多層パーセプトロンは、2 つの全結合層と活性化関数で構成されている。式(1)と(2)の $\boldsymbol{W}\sigma(\boldsymbol{W}\boldsymbol{Y})$ に対応しており、活性化関数 σ は GELU を使用している。

2.3 レイヤー正規化

レイヤー正規化[2]は、画像ごとに独立して行う正規化である。レイヤー正規化をすることで、値の発散を防ぎ、訓練時間を短縮し精度を向上させることができるという特徴がある。

2.4 GELU

GELU[3]は活性化関数であり、以下のように定義される。

$$GELU(x) = x \cdot \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right) \tag{3}$$

$\mathrm{erf}(x)$は誤差関数と呼ばれるもので、以下のように定義される。

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt \tag{4}$$

活性化関数にGELUを用いると、ReLUを用いた時よりも精度が上がるという特徴がある。

3. 提案手法

MLP-Mixer は優れた性能を発揮していることが報告がされているが、アルゴリズムに全結合層が多く、重みなどのパラメータ数が多いため、メモリを多く消費するという問題がある。そこで、Figure 4 で示すように、同じ層にある多層パーセプトロンの重みを使い回すことで、パラメータ数を減らすことを提案する。

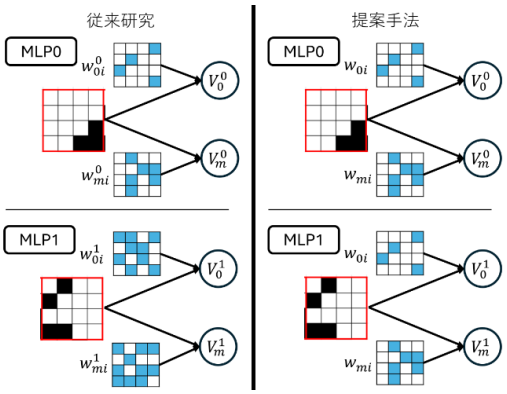

Figure 4　提案手法のイメージ図

4. 実験及び検討

提案手法と従来研究MLP-Mixerとのパラメータ数及び認識精度を比べる。実験にはMNISTという画像サイズが28×28の手書き数字データを使用する。パッチ数 S を7×7の49、チャンネル数 C を16にして実験を行う。重みを共有したことによりパラメータ数を減らしたとき、認識精度がどのように変化するかを記録し、検討をする。全体のMLPを共有化するだけではなく、Token Mixing MLP層やChannel Mixing MLP層のみ、入力層～中間層間や中間層～出力層間のみ、特定のMixer層のみなどに適用するなど、様々なパターンにおいて実験、検討をする。

5. まとめ

本研究では、重みを層ごとに共有することによりパラメータ数を減らす提案を行った。提案手法の導入によって、パラメータ数を削減しつつ精度を維持することができる層が有ることが予想される。今後、バイアスについても層ごとに共有をした時に精度がどのように変化するか実験をしていきたい。

参考文献

[1] Google Research Brain Team, “MLP-Mixer: An all-MLP Architecture for Vision”, v4: Updated the x label in Figure 2(right) (2021) pp.1-10.
[2] Jimmy Lei Ba, Jamie Ryan Kiros and Geoffrey E. Hinton, “Layer Normalization”, v1 (2016) pp.1.
[3] Dan Hendrycks and Kevin Gimpel, “Gaussian Error Linear Units (GELUs)”, v5: Trimmed version of 2016 draft (2020) pp.1.