

2D-1

バックプロパゲーションにおける
高速アルゴリズムの研究 — 補習学習 —益岡 竜介 渡部 信雄 川村 旭 大和田 有理 浅川 和雄
(株)富士通研究所

1. 序

近年、階層型ニューラルネットワークの学習方式として、バックプロパゲーション法(以下BPと省略)が注目を集めているが、その実行には膨大な計算時間を要し、その高速化が求められている。我々は、ニューラルネットワークの並列処理の特徴を生かしながら、柔軟性に富む高速な学習アルゴリズムの開発を進めた。我々は高速化に対する1つの解答として、追記学習アルゴリズム^{1,2}を既に開発した。今回は、学習中の無駄な計算を低減することに着目した補習学習法を報告する。更に、それらのアルゴリズムをアルファベットフォント認識の学習に適用した結果を示す。

2. いままでのBP学習の問題点と補習学習

まずBP学習のどこで計算量を減らすことができるかを観察した。学習中のネットワークの状態を観察すると、学習の後半では、ほとんどのパターンに関して出力は収束している。しかし、学習は最後まで全てのパターンに関して繰り返されている。このことは、2つの意味で学習の進行を阻害している。

1つは、学習が済んでいるパターンによる、未学習パターンの学習の進行に対する阻害である。主に未学習のパターンのエラーにより重みが変わり始めると、今まで小さかった既学習パターンによるエラーが増加する。すると組織だっていない未学習パターンによるエラーは増加した既学習パターンによるエラーに負けてしまい、重みの変化が妨げられてしまう。

もう1つは、阻害と言うよりは計算量に関する無駄である。既に学習が済んでいるパターンは、エラーが十分小さいので、それらのパターンに関するエラーの逆伝播量は非常に小さくなる。それらのエラーにより重みはほとんど変わらないので、計算する

だけ無駄である。

これらのBPにおける問題点を回避するには、エラーの大きなパターンに関してだけエラーを逆伝播すればよい。これが補習学習の基本的な考え方である。

3. 補習学習のアルゴリズム

補習学習のアルゴリズムを説明する。各学習パターンに対して、ネットワークを入力パターンに関して実行して出力を出させ、その入力パターンに対する出力パターンが学習できているかを判定する。学習できていれば、その出力に対するエラーを逆伝播しない。学習できていないと判定された場合のみ、エラーを逆伝播する。以上のことを全学習パターンに対して繰り返してから、重みを1回変更する。全てのパターンに対しての学習が終わるまで、以上の重み変更の手続きを繰り返すのが補習学習のアルゴリズムである。

4. 追記学習アルゴリズムとの関係

補習学習のアルゴリズムは他の学習アルゴリズムとも簡単に組み合わせることができる。特に追記学習のアルゴリズムとは相性が良い。

補習学習を追記学習に組み合わせるには、追記学習の各段階の学習に対して、補習学習のアルゴリズムを適用すればよい。特に追記学習の復習フェーズでは、全てのパターンを一度学習した後なので、ほとんどのパターンが直ぐに再学習されてしまい、補習学習が効果的である。

5. 補習学習の実験

この補習学習アルゴリズムの効果を見るためにアルファベットフォント認識の学習を例に取った。学習定数の設定は、通常のBPがほぼ最適に学習するように選んだ。

実験では、ランダムに初期値を設定した同じ状態のネットワークからそれぞれ通常のBPによる学習、追記学習、補習学習、追記学習と補習学習の各アルゴリズムでアルファベットフォント認識を学習させ

A study on supplementary learning algorithm in back propagation

Ryusuke Masuoka, Nobuo Watanabe, Akira Kawamura, Yuri Owada, Kazuo Asakawa
FUJITSU LABORATORIES LIMITED

て、学習の速さを比較した。

フォントには、8×8ドットの英大文字フォントを使った。使ったネットワークのサイズは、入力層64ユニット、中間層15ユニット、出力層26ユニットである。閾値は中間層と出力層に入れた。

エラーを逆伝播するパターン数を p としたときに学習レート ϵ は、 $\epsilon = 5.2/p$ とし、モーメントム α を $\alpha = 0.4$ とした。

学習の収束判定は、教師信号からの差が各出力ユニットで0.4未満になったときとした。

6. 実験の結果と今後の課題

これらの学習アルゴリズムの効果を実験した結果が第1図のグラフである。補習学習単独では、約1.2倍の高速化である。追記学習（9分割）では通常のBPに比べて約1.4倍の高速化であるが、追記学習（9分割）と補習学習を組み合わせた場合は通常のBPに比べて約4倍の高速化という結果を得た。このグラフにあるインストラクション数とは、実行、エラーの逆伝播、重み更新に要する加減乗除の回数を合計したもので、ほぼ学習終了までの時間に比例する。

第2図は、同じ初期状態からのネットワークの学習についてグラフにしたものである。学習中か否かに関わらず全パターンのエラーの平均を縦軸に、インストラクション数を横軸にプロットしたものである。この図では、分かり易いように4分割の場合で示している。特徴的なのは、補習学習を入れたアルゴリズムでは、エラーの平均が高い所で学習終了条件を満足して学習を完了していることである。これは、学習が平均的に進んでいることを示している。

追記学習、補習学習の各アルゴリズムではパターン数が動的に変化するので、学習定数の設定が難しい。各学習段階、パターン数に応じて学習定数を最適に調整することが課題である。

また、この補習学習のアルゴリズムをアナログパターンの学習に適用する実験も進めている。

7. 謝辞

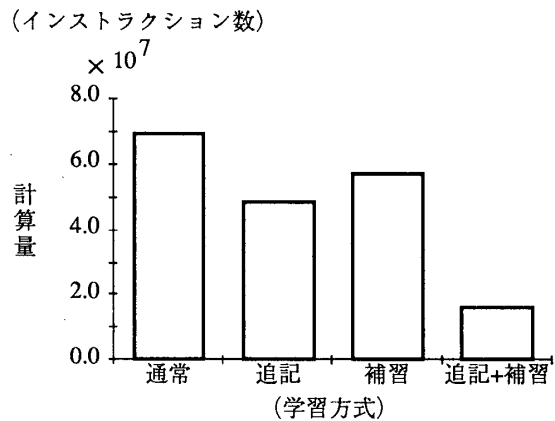
日頃から御指導頂く情報処理研究部門棚橋純一部門長ならびに人工知能研究部林弘部長に感謝します。

References

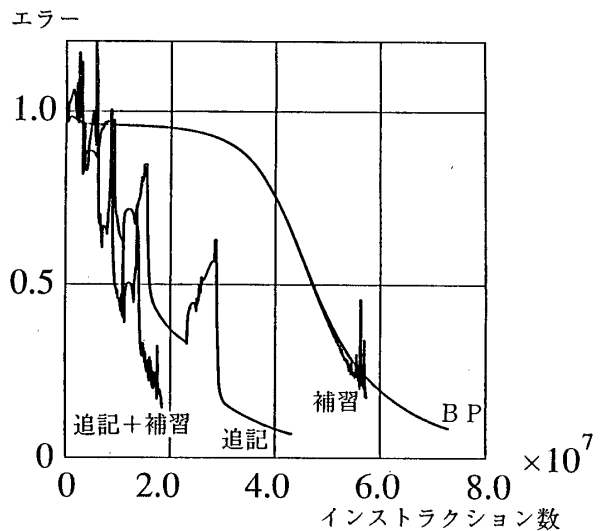
1. 益岡, 渡部, 木本, 川村, 浅川, “バックプロパゲーションの高速学習アルゴリズム—追記学習—,” 情報処理学会第38回全国大会6F-5, pp. 492-493,

17 Mar. 1989.

2. 益岡, “バックプロパゲーションの高速学習アルゴリズム—追記学習—,” 信学技報, vol. 88, no. 466, pp. 121-126, 15 Mar. 1989.



第1図 アルファベットフォントの認識学習実験結果



第2図 学習中のエラーの変化