

知的プログラミング演習 I 第8回レポート

2025年7月27日 学籍番号 35714121 氏名福富隆大

1 レポートのテーマ

本レポートでは、課題選択肢の中から「複雑なニューラルネットワークの実装」をテーマとして選択する。LSTM (Long Short-Term Memory) を用いた再帰型ニューラルネットワークを実装し、時系列データの分類問題に適用した結果について報告する。

2 実装したモデル

今回実装したのは、LSTM を用いた再帰型ニューラルネットワークである。LSTM は通常の RNN (再帰型ニューラルネットワーク) の欠点である勾配消失問題を解決するために開発された手法で、長期の依存関係を学習できる特徴がある。

LSTM の主要な構成要素は以下の通り：

- 忘却ゲート: 過去の情報をどれだけ忘れるかを制御
- 入力ゲート: 新しい情報をどれだけ取り入れるかを制御
- 出力ゲート: 出力をどれだけ制御するか
- セル状態: 長期記憶を保持する内部状態

今回の実装では、時系列データから特定のパターン ($[1,0,1]$ の並び) を検出する 2 クラス分類問題を解くことを目標とした。

3 ネットワークの詳細設定

実装した LSTM ネットワークの詳細設定は以下の通り：

- 入力層: 2 次元 (0 と 1 のワンホットエンコーディング)
- 隠れ層: 10 ユニット (LSTM セル)
- 出力層: 2 次元 (パターンあり/なしの 2 クラス)
- 系列長: 15 ステップ
- 活性化関数: シグモイド関数 (ゲート)、tanh 関数 (セル状態)
- 重み初期化: 標準偏差 0.1 の正規分布
- サンプル数: 200 個のランダム生成データ

プログラムは lec8.py に保存されており、完全な LSTM の順伝播計算を含んでいる。

4 実験結果

実装したLSTMモデルでいくつかのサンプルデータに対して予測を実行した。ただし、今回は学習処理（逆伝播による重み更新）は実装していないため、ランダム初期化状態での予測結果となっている。

実験では以下のような結果が得られた：

- モデルは正常に動作し、各時系列に対して確率値を出力
- 初期状態では予測精度は約50%（ランダム予測レベル）
- LSTMの内部状態（隠れ状態とセル状態）が正しく更新されることを確認
- 異なる入力系列に対して異なる出力を生成することを確認

実際の学習を行うためには、損失関数の計算、逆伝播アルゴリズム、最適化手法の実装が必要である。

5 考察

LSTMの実装を通じて、再帰型ニューラルネットワークの複雑さと有用性を理解できた。特に、ゲート機構による情報の制御が時系列データの処理に重要な役割を果たすことが分かった。

実装上の課題として、以下の点が挙げられる：

- 勾配の計算が複雑になる（時間方向への逆伝播が必要）
- パラメータ数が多く、適切な初期化が重要
- 長い系列では計算コストが高くなる

また、実用的な応用を考えると、適切な学習率、正則化、バッチ処理などの技術が必要になる。今回の実装は基本的な構造の理解に重点を置いたが、実際のプロジェクトではより洗練された実装が求められるだろう。

LSTMのような複雑なモデルを一から実装することで、深層学習フレームワークの便利さも再認識できた。

6 参考文献

1. LSTMの実装（RNN・自然言語処理）https://qiita.com/hara_tatsu/items/c3ba100e95e600846125
2. 機械学習の学習法<https://qiita.com/hokkey621/items/404e04d6057b98128971>
3. ニューラルネットワークの基礎https://tutorials.chainer.org/ja/13_Basics_of_Neural_Networks.html
4. LSTM and RNN Tutorialhttps://github.com/omerbsezer/LSTM_RNN_Tutorials_with_Demo

PyTorch RNNを使った時系列予測https://qiita.com/kuk_a_i_ai/items/0ea4b93d767ce7c83145