

系列データにおける特徴空間上での疑似訓練データ生成

Generate pseudo training data in the feature space of sequential data

水野 充大[†]岡留 剛[†]

Mitsuhiro Mizuno

Takeshi Okadome

概要

Long Short-Term Memory (LSTM) や Recurrent Neural Network (RNN) などの深層学習を用いた時系列データを始めとする系列データの分類問題は盛んに研究がなされている。系列データの収集や前処理は他のデータに比べ、コストが高く大規模なデータセットの収集、利用は困難である。そのため、深層学習に対して十分な学習用データを準備することができず、少数の学習用データを取り扱うアプローチの1つにデータ拡張 (Data Augmentation) を用いることがある。しかし、系列データに対してのデータ拡張は収集したデータの特徴性に左右され困難である。本研究では、系列データに対しての特徴量抽出とデータ拡張のための疑似訓練データ生成手法をあわせて提案する。特徴量抽出は、系列データからの特徴量抽出は LSTM AutoEncoder を用いて系列データから固定長の特徴量ベクトルを抽出する。そして、抽出された特徴量ベクトルに対して分類器の出力を加味しながら逐次的に疑似訓練データを生成する。

Many researches deal with the classification problem of series data including time series data using deep learning such as Long Short-Term Memory (LSTM) and Recurrent Neural Network (RNN). Collecting and preprocessing series data is much more expensive than other data, and it is difficult to collect and use large datasets. It is, therefore, not possible to have enough training data for deep learning, and one approach to handle small numbers of training data is to use data augmentation. Extending the data to series data is, however, very difficult because it depends on the characteristics of the collected data. In this research, we also propose a pseudo training data generation method for feature extraction and data expansion of sequence data. Feature extraction uses the LSTM AutoEncoder to extract features from the series data and a fixed-length feature vector from the series data. Then, the pseudo training data is sequentially generated while adding the output of the classifier to the extracted feature amount vector.

1. はじめに

時系列データを始めとする系列データの分類問題は盛んに研究がなされており、IoT 分野やヘルスケア分野、金融分野など様々な分野への応用が期待されている。

特に近年、Long short-term memory (LSTM) [1] や Recurrent Neural Network (RNN) [2] といった系列データを扱うことが得意なアーキテクチャをもった深層学習モデルが登場し、LSTM をベースとした手法が大きく成

果を出している。しかし、これらの深層学習モデルは非常の多くのパラメータを持ち、それらを学習するには多くの学習データが必要である。大規模な系列データを収集、学習用データとしてラベル付けや前処理を行うことは高いコストを有する。そのため、十分に実用的な精度をもつ分類器の構築は困難である。さらに、IoT 機器やウェアラブル端末などに搭載されたセンサーから取得される変動の激しい複雑な系列データなどでは、少数の学習用データをつかった深層学習だけでなく他の機械学習

[†] 関西学院大学, Kwansai Gakuin University

技術でも高精度な分類が困難である。

機械学習において、少数の学習用データを取り扱うアプローチの1つにデータ拡張(Data Augmentation)がある。データ拡張は、学習用データに微小な変換を加え、データ量を擬似的なデータで水増しをすることで分類器の過学習が起きにくくすることで、少数の学習用データでの学習を可能とする手法である。データ拡張は画像データなどに対してよく用いられている[3, 4]。しかし、系列データにおいてのデータ拡張は処理によってデータの特性を破壊することがあり、データの特性にあわせた処理を行う必要で困難である[5, 6]。

また、系列データに対する特徴量抽出に関しても同様にデータの特性への理解が必要不可欠である。手動での特徴量抽出は処理のパラメータ調整がセンシティブであり、最適なパラメータを見つけ出すのが困難である。そのため、データの特性に精通した専門家が人が特徴抽出を行うことが必要となり系列データを扱う難しさの1つである。

本研究では、系列分類問題において特徴量抽出手法と抽出特徴に対してのデータ拡張手法を提案する。この手法では、特徴量抽出にLSTM Autoencoder[7]を用いて系列データから固定長の特徴量ベクトルを抽出する。そして、特徴量ベクトルに対してデータ拡張を行う。この特徴量ベクトルはLSTM Autoencoderによって特徴空間上に写像したものであり、この特徴空間上で疑似訓練データ生成を行う。

本論文の貢献は、系列データに対して特徴量抽出と疑似訓練データの生成手法を提案し、人の知識に頼らずに分類器の分類精度向上を実現したことである。

2. 関連研究と研究背景

2.1 特徴量抽出

系列データに対する分類問題を解く上で特徴量は重要となってくる。ここでは、系列データに対する分類問題を解くために用いられている特徴量のうち、特徴量の自動抽出のに関して取り上げる。

松原らは大規模時系列データの中から、異なるトレンドを発見し、すべての時系列パターンを表現する手法として、AutoPlaitを提案しました[8]。AutoPlaitは、様々な時系列パターンを含む複雑なシーケンスが与えられた

ときに、そのシーケンスデータの中から重要な特徴を発見し、それらの情報を統計的に要約、表現する。この手法ではパラメータに異存しないため、事前情報の付与またはパラメータのチューニングを行うことなく特徴抽出を自動で行うことができる。しかし、特徴量抽出器には大規模時系列データを必要とするため、少数のデータ化での利用は難しい。

Mehdiyevらは、深層学習モデルを用いた多変量時系列分類の特徴量抽出にLSTM Autoencoder(LSTM-AE)を用いた[9]。LSTM-AEを用いることでデータの特性を知らずとも教師なし学習によって特徴量を抽出することができる。LSTM-AEによって抽出した特徴量を順伝播型ニューラルネットワークに用いること高い分類性能を発揮した。

他にも系列データに対する特徴量の自動抽出に関する研究が盛んに行われている。本研究では、特徴量抽出にLSTM-AEをベースとして用いる。

2.2 データ拡張

データ拡張は、画像データに対して頻繁に用いられている。画像データのデータ拡張手法として、一般的なものに、幾何学的変換や部分マスク、ノイズ付与など様々な物がある。

Zhunらは、画像データに対するデータ拡張手法としてRandom Erasing Data Augmentationを提案した[3]。この手法は教師データとなる画像のランダムな一部矩形領域をマスクすることでデータ拡張をおこなう。この手法は画像データに対する手法であるが、系列データに対して同様のデータ拡張を行うと周波数特性などのデータ特性が大きく変化し本来のデータとかけ離れたものになってしまう。そのため、系列データに適応することは難しい。

Unらは、ウェアラブル端末から取得された加速度データに対して画像データに対する様々なデータ拡張手法を用い、比較を行った[5]。切り抜きやノイズ、スケール変換、回転変換などを試した。その結果、回転変換及び回転変換と他の変換を組み合わせたときに分類性能向上がみられた。これは、ウェアラブル端末の特性上センサーの装着角度が時々刻々と変化することを回転変換で再現したことによる。実際のデータが取得される環境を想定したデータ拡張手法によって実際のデータに近い

データを生成されたことによる結果である。

Zhangらは、脳波センサ値 (EEG) による系列分類に対してデータ拡張を行った [6]. EEG の時間空間でノイズ与えるだけでは低周期信号対雑音比や非正常性を破壊してしまい、性能を悪化させてしまう。そのため、単純に系列データに対してガウスノイズを与えるのではなく、周波数空間上に対してガウスノイズを与えることで低周期信号対雑音比や非正常性の破壊しないデータ拡張を実現し、分類性能向上を実現した。

これらの関連研究のように系列データに対するデータ拡張手法は、それぞれのタスクへの依存が大きく、タスクによって向き不向き顕著に現れる。そのため、系列データに対して一般的な手法を提案することは困難である。そのため、研究では系列データに対して直接データ拡張を施すのではなく、系列データから抽出された特徴量に対してデータ拡張を行う。

3. 提案手法

ここでは、提案手法である特徴量抽出手法とデータ拡張手法の全体を説明した後に、それぞれに関して述べていく。

図 1 に提案手法の概要を示す。提案手法では LSTM-AE によって系列データから特徴量を抽出し、その特徴量に対して特徴空間上においてデータ拡張を施し複数の疑似特徴量を生成する。その後、疑似特徴量を任意の分類器に入力した後、学習に寄与すると見込まれるものだけを疑似訓練データとして選択し学習に利用する。

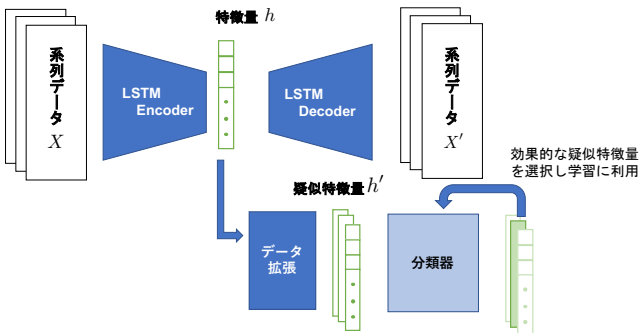


図 1: 系列データ X を入力として特徴ベクトル h を抽出し、 h に対してデータ拡張を行った疑似特徴量 h' を任意の分類機に入力する

3.1 特徴量抽出

本手法では、特徴量抽出に LSTM-AE を用いる。しかし、LSTM は構造が複雑でパラメータ数が多く、少数のデータで用いる場合は過学習が起きやすい。そのため、LSTM に比べて構造の簡略化された Gated Recurrent Unit (GRU) [10] を用いる。GRU は、一般に LSTM と同等の性能を持つとされており、学習データが少数の場合は LSTM より優れている場合がある。本研究は、データ拡張が必要とされる学習データが少数の場合を考慮するため、GRU を用いた GRU Autoencoder (GRU-AE) を用いる。

3.2 データ拡張

データ拡張は井手らの手法 [11] を系列データに拡張する。井手らの手法の手法は、画像データに対してデータ拡張で複数の疑似データ集合を生成し、対象とするモデルに疑似データを入力したときの予測出力を元に学習に用いる疑似訓練データを選択する。

系列データに対して、画像データと同様のデータ拡張処理を適応すると、系列データに含まれる周波数特性などの分類に強く寄与する特徴を欠落させてしまう。これに対して、本研究では系列データに直接データ拡張を行うのではなく、GRU-AE によって抽出された特徴量に対して特徴空間でデータ拡張を行う。

まず、特徴量抽出器として用いる GRU-AE によって抽出された特徴量に対して mixup [12] を用いて疑似データを生成する。mixup は Beta 分布からサンプリングした混合比率をもとに、入力・出力ともに混合させてデータ拡張を行う手法で決定境界が滑らかになるため汎化性能が向上するとされている。

$$\lambda \sim \text{Beta}(\alpha, \alpha), \text{ for } \alpha \in (0, \infty) \quad (1)$$

$$\hat{X} = \lambda X_1 + (1 - \lambda) X_2 \quad (2)$$

$$\hat{y} = \lambda y_1 + (1 - \lambda) y_2 \quad (3)$$

mixup は式 (1), 式 (2), 式 (3) で表される。ベータ分布よりサンプルされる混合比 λ を元に 2 つの入力データ及び one-hot ベクトルで表されるラベルに対して混合する。

mixup によって同一の学習データの組から疑似特徴量を複数生成した後、分類器に入力し効果的な疑似特徴

量を選択する。本研究では、分類問題を扱うため疑似特徴量の選択には分類器で計算される交差エントロピー誤差が大きく期待勾配を得やすい疑似訓練データのみを選択する。

その後、選択された疑似訓練データを用いて分類器を学習する。ここでの分類器の学習にはミニバッチ学習を用い、バッチデータからの疑似特徴量の生成及び選択と選択された疑似訓練データでの学習を交互に行う。

4. 評価方法

提案手法を評価するため、複数の系列データのデータセットを用いて特徴抽出を行った。

4.1 データセット

本稿では、時系列分類の評価用データセット集である UCR 時系列分類アーカイブ (UCR Time Series Classification Archive) [13] を用いた。本データセット集はセンサ値や心電図といった時系列データや画像の輪郭線から抽出された系列データなどの 128 個のデータセットで構成されている。各データセットの入力データは 1 次元系列データであり、クラス数、系列長はデータセットでそれぞれで異なっている。

4.2 設定

本実験において、GRU-AE のエンコーダ及びデコーダは両者とも 1 層 10 ユニットの GRU を使用する。GRU-AE のエンコーダに関して、入力が 1 次元の系列データ $\mathbf{x}_t (t = 1, \dots, T)$ 、出力が 10 次元の単一のベクトル \mathbf{h} である。GRU-AE のデコーダに関して、入力が 10 次元の単一のベクトル $\hat{\mathbf{h}}$ 、出力が 1 次元の系列データ $\hat{\mathbf{x}}_t (t = 1, \dots, T)$ である。それぞれのデータセットにおける系列の長さ T は、各データサンプルにより異なる。

提案手法によって得られた特徴量を入力とし、クラス分類に使用する分類器にはニューラルネットワークを使う。10 次元の特徴量を入力として受け付け、各データセットに含まれるクラスのいずれかに分類を行う。このニューラルネットワークは 3 層の全結合層で構成され、第 1 層は 64 次元、第 2 層は 128 次元、第 3 層は次元数はクラス数と同一である。

特徴抽出モデルの GRU-AE 及びクラス分類に使用するニューラルネットワークの双方において、学習時には、最適化関数として Adam[14]、バッチサイズは 48 とする。

特徴抽出モデルの GRU-AE に対しては再構成誤差を最小化、クラス分類に使用するニューラルネットワークに対しては、交差エントロピーの最小化を行う。

提案手法との比較には、GRU-AE による特徴量抽出を行わず系列データで有用とされているエネルギーや周波数領域エントロピーなどを含む 794 個の特徴から特徴量選択した 10 特徴を用いた場合、GRU-AE による特徴量抽出のみを行いデータ拡張を行わない場合の 2 つを比較対象とする。

5. 結果と議論

UCR 時系列分類アーカイブには多数のデータセットが含まれているため、本稿では一部のデータセットである系列データである ArrowHead, Beef と時系列データである CinCECGTorso, InsectWingbeatSound の結果を取り上げる。ArrowHead は画像データから抽出された系列データであり、系列長 251, クラス数 3 クラス, 学習データ数 36 系列。Beef は赤外線分光によって取得されたスペクトルであり、系列長 251, クラス数 5 クラス, 学習データ数 30 系列。CinCECGTorso は胸部で取得された ECG センサ値であり、系列長 1639, クラス数 4 クラス, 学習データ数 40 系列。InsectWingbeatSound は昆虫の羽音の音声データであり、系列長 256, クラス数 11 クラス, 学習データ数 220 系列となっている。

図 2, 3 に、それぞれのデータセットの従来手法と提案手法を用いて分類を行った際の F 値を示す。有用とされている特徴量から特徴量選択を行った場合に比べ、多くの場合において提案手法による性能向上がみられた。一方で、ArrowHead と CinCECGTorso データセットに関しては GRU-AE による特徴量抽出のみを行った場合はベースラインに比べ性能が劣る。これは、学習データが少なく GRU-AE が過学習を起こしているため適切な特徴量抽出ができていなかったのではないかと考えられる。そのため、GRU-AE に対して過学習を防ぐ何らかのアプローチを検討する必要がある。しかしながら、系列データに対してデータ拡張は難しく本手法で目指すことではない。そのため、疑似訓練データの生成の枠組を取り入れて GRU-AE を学習できる用に検討したい。

データ拡張に関しては、扱ったすべてのデータセットに対して、データ拡張を施していないものに比べ性能向

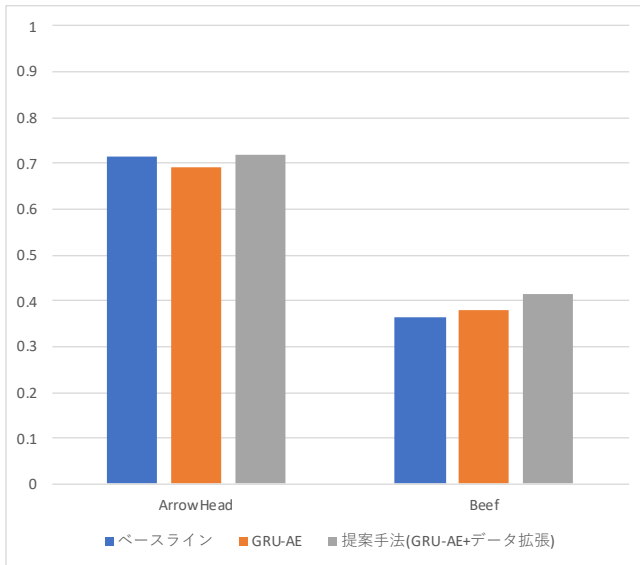


図 2: ArrowHead, Beef の提案手法と比較手法での分類の F 値. ベースラインが有用とされている特徴量を用いた場合, GRU-AE が特徴量抽出を GRU-AE 行った場合, 提案手法が GRU-AE による特徴量抽出に加えデータ拡張を行った場合である.

上がみられた. 特に ArrowHead に関しては画像処理によって生成された系列データであるため, データの特性を掴むことは難しい. 本手法を用いることで容易に有用な疑似特徴データを得ることができた. そのため, 系列データに対して特徴空間上でデータ拡張をするのは有用であり, 学習に有用でない水増しされた疑似特徴データを学習から除外し, 汎化性能を向上させた.

Mixup において 2 データの中間的な疑似訓練データを生成するとき他のクラスに属するデータ分布上に生成される場合がある [15] 特徴空間上でデータ拡張を行うことで, 生成される疑似訓練データが他のクラスに属するデータ分布上に生成されることを抑制したとことの汎化性能を向上につながったと考えられる.

しかしながら, 性能向上が見られたといえあまり向上しなかったデータセットも存在する. 特徴空間でのデータ拡張においても, データ特性や特徴量抽出に用いる GRU-AE の学習状況によって影響を受ける. そのため, 本手法で扱った Mixup 以外に井出ら [11] が用いた cutmix[16] やその他の処理の検討も必要である.

本研究では, UCR 時系列分類アーカイブに含まれている一部のデータセットに対して評価を行った. 本研究

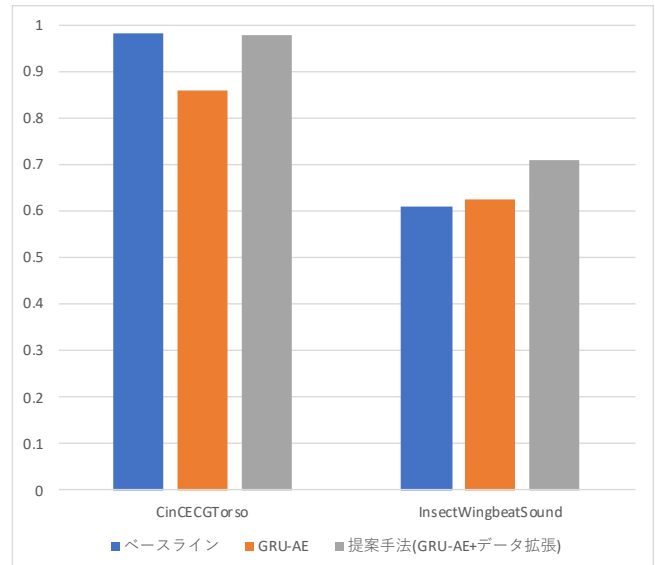


図 3: CinCECGTorso, InsectWingbeatSound の提案手法と比較手法での分類の F 値. 図 2 と同様, ベースラインが有用とされている特徴量を用いた場合, GRU-AE が特徴量抽出を GRU-AE 行った場合, 提案手法が GRU-AE による特徴量抽出に加えデータ拡張を行った場合である.

では取り上げたデータセット以外にも多数の様々な特性を持った系列データが存在する. そのため, より慎重に他のデータセットでの本手法の評価, 議論を行う必要がある.

6. まとめ

本研究では, 系列データに対しての特徴量抽出と特徴空間上でのデータ拡張のための疑似訓練データ生成手法をあわせて提案した. 特徴量抽出には GRU-AE を, データ拡張は特徴空間において mixup を行い, その疑似特徴量から学習に有用でない水増しされた疑似特徴量を学習から除外し, 汎化性能を向上させた. 一方で, 特徴量抽出において GRU-AE での特徴量抽出は学習データが少なく適切な困難であった. そのため, 今後の課題として, 特徴量抽出に用いる GRU-AE に対して過学習を防ぐアプローチやデータ拡張に用いる処理の検討が必要がある.

参考文献

- [1] Hochreiter, S. and Schmidhuber, J.: Long short-term memory, *Neural computation*, Vol. 9, No. 8, pp. 1735–1780 (1997).
- [2] Rumelhart, D. E., Hinton, G. E. and Williams, R. J.: Learning internal representations by error propagation, Technical report, California Univ San Diego La Jolla Inst for Cognitive Science (1985).
- [3] Zhong, Z., Zheng, L., Kang, G., Li, S. and Yang, Y.: Random Erasing Data Augmentation., *AAAI*, pp. 13001–13008 (2020).
- [4] Perez, L. and Wang, J.: The effectiveness of data augmentation in image classification using deep learning, *arXiv preprint arXiv:1712.04621* (2017).
- [5] Um, T. T., Pfister, F. M., Pichler, D., Endo, S., Lang, M., Hirche, S., Fietzek, U. and Kulić, D.: Data augmentation of wearable sensor data for parkinson ’ s disease monitoring using convolutional neural networks, *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 216–220 (2017).
- [6] Zhang, X.-R., Lei, M.-Y. and Li, Y.: An amplitudes-perturbation data augmentation method in convolutional neural networks for EEG decoding, *2018 5th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS)*, IEEE, pp. 231–235 (2018).
- [7] Srivastava, N., Mansimov, E. and Salakhudinov, R.: Unsupervised learning of video representations using lstms, *International conference on machine learning*, pp. 843–852 (2015).
- [8] Matsubara, Y., Sakurai, Y. and Faloutsos, C.: Autoplait: Automatic mining of co-evolving time sequences, *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pp. 193–204 (2014).
- [9] Mehdiyev, N., Lahann, J., Emrich, A., Enke, D., Fettke, P. and Loos, P.: Time series classification using deep learning for process planning: A case from the process industry, *Procedia Computer Science*, Vol. 114, pp. 242–249 (2017).
- [10] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation, *arXiv preprint arXiv:1406.1078* (2014).
- [11] 井手敦也, 岡留 剛学習用入力の複数サンプルに対する予測出力の調和による学習法, 第 22 回情報論的学習理論ワークショップ (2019).
- [12] Zhang, H., Cisse, M., Dauphin, Y. N. and Lopez-Paz, D.: mixup: Beyond empirical risk minimization, *arXiv preprint arXiv:1710.09412* (2017).
- [13] Bagnall, A., Dau, H. A., Lines, J., Flynn, M., Large, J., Bostrom, A., Southam, P. and Keogh, E.: The UEA multivariate time series classification archive, 2018, *arXiv preprint arXiv:1811.00075* (2018).
- [14] Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
- [15] Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D. and Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states, *International Conference on Machine Learning*, pp. 6438–6447 (2019).
- [16] Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J. and Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6023–6032 (2019).