

## 時系列予測モデル学習時間削減手法の検討

高橋 佑里子† 鈴木 成人‡  
†お茶の水女子大学

田原 司睦‡ 小口 正人†  
‡株式会社富士通研究所

### 1 はじめに

仮想環境において、計算資源のオーバーコミットに由来する仮想マシン (Virtual Machine: VM) の性能低下を防ぐことを目的として、VM の CPU 使用率を予測し、その結果に基づいて制御を行う技術が知られている。VM とそこで実行されるアプリケーションは時々刻々と変化するため、環境の変化に合わせて予測モデルを継続的に学習し、モデルを更新することで予測精度を担保する。しかし、大規模仮想環境における予測モデルの継続的な学習には多大な学習時間を必要するため、適応が困難であった。そこで本研究では、環境の変化を監視・評価し、精度担保に必要と判断された場合のみ予測モデルの学習・更新を行うことで、学習時間の削減を実現する手法を検討する。

### 2 関連研究

クラウドサービスにおける VM の CPU 使用率の予測や制御に関する研究は、以前から多く行われている。[1] では、ラックユニットの消費電力を分析し、API 連携を用いて VM の管理ソフトウェアと連携したシステムの提案を行っている。

### 3 提案手法

提案手法のフローチャートを図 1 に示す。準備として、過去のデータを細分化し、クラスタリングの基準を作成を行う。次に、事前に数種類の傾向それぞれに対してクラスタリングの基準を当てはめ、クラスタ番号の分布を保存し、学習を行い予測モデルを作成し保存する。運用段階では、最初に直近のデータを取得し、細分化を行った後、準備段階で作成した基準に当てはめ、得られたクラスタ番号の分布を過去の複数の分布と比較する。もし、過去に似た分布があった場合、学習を行う必要がないため、その時点で使用されたモデルを流用する。一方、過去に似た分布がなかった場合、

最も似た分布の時点で使用されていたモデルに対して学習を行い、作成したモデルを保存して使用する。この一連の流れを一定間隔で繰り返すというものである。

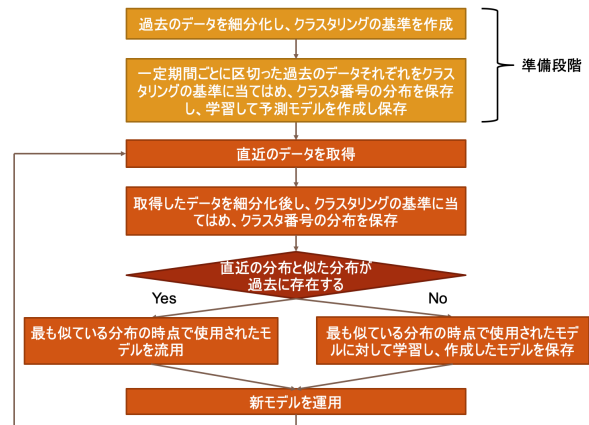


図 1: 提案手法のフローチャート

#### 3.1 細分化

本研究では、図 2 のように、データを 1 点ずつらしながら学習元データ数と正解データ数の合計値ごとに区切る処理を細分化と呼ぶ。本研究では、学習元データ数を 200、正解データ数を 1 と設定しているため、201 点ごとに区切る処理を行っていることになる。



図 2: 細分化のイメージ

#### 3.2 クラスタリングの基準

本研究では、準備段階において、過去のデータを細分化したものを元にクラスタ数を 30 としてクラスタリングの基準を作成し、運用段階においてもここで作成した基準を使い続けることとする。

Reducing the Learning Time for Time Series Prediction Model  
†Yuriko Takahashi  
‡Shigeto Suzuki  
†Masato Oguchi  
†Ochanomizu University  
‡FUJITSU LABORATORIES LTD.

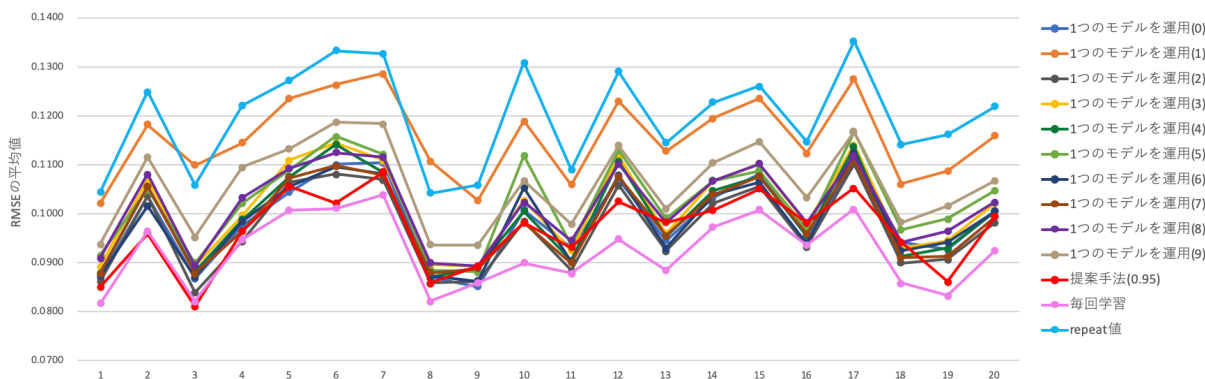


図 3: 実験結果

### 3.3 分布の類似度

分布の類似度の基準は、細分化後のデータをクラスタリングの基準に当てはめた際のクラスタ番号分布のコサイン類似度とした。

### 3.4 学習

本研究では、GRU を 2 層繋げた深層学習ネットワークを構築した。ライブラリは、TFLearn[2] を使用し、モデルの精度評価指標として RMSE を使用した。

## 4 実験

### 4.1 実験方法

本研究では、Microsoft 社が提供している Azure の VM トレースデータセット [3][4] の一部の、"avg cpu" 項目を使用した。異なる傾向を持つデータセットを作成する目的で、データセットを正規化した後 360 点に細分化し、その中から類似度の低い 100 種類の波形を抽出し、これらを異なる割合で含む傾向を 30 種類作成した。そして、準備段階として 10 種類の傾向で学習モデルを作成し、20 種類の傾向を運用段階とした。

提案手法での類似度の基準は、過去の分布の中で最も類似度の高い分布とのコサイン類似度が 0.95 以上ならば学習は行わず、0.95 未満であれば学習するということとした。この結果、学習時間は毎回学習する場合と比較して 75% 減少した。

そして、提案手法以外に、継続学習を行わずに 1 つのモデルを使用し続けるパターン 10 種類・各区間で毎回学習を行ったパターン・1 つ前の値を繰り返しただけのデータ (以下、repeat 値と呼ぶ) との比較も行った。

### 4.2 結果

上記のような条件で実験を行ったところ、結果は図 3 のようになった。毎回学習を行った場合 (ピンク色)

が最も精度が良いが、提案手法 (赤色) の結果は、repeat 値 (水色) や 1 つのモデルを使用し続けるパターン 10 種類と比較して、おおよその区間で精度が上回っていることが読み取れる。

## 5 まとめと今後の予定

精度担保に必要と判断された場合のみ予測モデルの学習・更新を行うことで、学習時間を削減する手法の検討を行った。その結果、学習時間を大幅に削減しながら、予測モデルの精度を保つことが可能であることが確認できた。今後は、類似度判定の基準を変化させて実験を行うことや、より良いフローチャートの作成に取り組みたいと考えている。

## 謝辞

本研究の一部はお茶の水女子大学と富士通研究所との共同研究契約に基づくものであり、JST CREST JPMJCR1503 の支援を受けたものである。

## 参考文献

- [1] Hiroyoshi Kodama, Hiroshi Endo, Shigeto Suzuki, and Hiroyuki Fukuda. High efficiency cloud data center management system using live migration. In *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*, pages 733–738. IEEE, 2017.
- [2] Tflern — tensorflow deep learning library. <http://tflern.org/>.
- [3] Mohammad Shahrads, Rodrigo Fonseca, Ínigo Goiri, Gohar Chaudhry, Paul Batum, Jason Cooke, Eduardo Laureano, Colby Tresness, Mark Russinovich, and Ricardo Bianchini. Serverless in the wild: Characterizing and optimizing the serverless workload at a large cloud provider. *arXiv preprint arXiv:2003.03423*, 2020.
- [4] Azure/azurepublicdataset: Microsoft azure traces. <https://github.com/Azure/AzurePublicDataset>.