

複数周期に着目した畳み込みネットワークによる 小売時系列データの予測手法

保住 純^{1,a)} 岩澤 有祐^{1,b)} 松尾 豊^{1,c)}

受付日 2021年1月6日, 採録日 2021年8月6日

概要: 小売業において、将来の商品の需要を正確に予測することは売上を増加させるうえで重要であるため、これまでに計量経済学や統計学といった様々な分野で数々の時系列予測手法が研究されている。近年では、さらに予測性能を上げるために、深層学習を活用する試みもある。ただし、これまでの深層学習を用いた手法は小売データに特化して設計された手法ではないため、小売データにおける有用性は不明瞭である。ゆえに、その特徴をとらえる機構をモデルに組み込むことができれば、これまでの手法を超える予測精度を達成できる蓋然性は高い。このことをふまえ、本研究では小売データが持つ周期性に関するドメイン知識を取り入れた深層学習による予測モデルを提案する。この周期性に関するドメイン知識とは、曜日や月（季節）のような複数の商品に共通する売上の周期変動の他に、商品ごとに特有の異なる消費の周期があり、それぞれ商品の売上データごとにそれらの異なる複数の周期が反映されていると考えられることである。本研究では、既存手法を時系列ごとに異なる複数の周期をとらえられるように拡張することで周期性に関するドメイン知識を考慮して予測を行う手法を提案する。そして、複数種類の実際の小売データや人工的に作成したデータを用いた予測実験を行い、その精度の向上をもって提案手法の有効性を確認する。

キーワード: 予測, 深層学習, 時系列, マーケティング

Retail Time-Series Prediction Model by Multi-Window Convolution

JUN HOZUMI^{1,a)} YUSUKE IWASAWA^{1,b)} YUTAKA MATSUO^{1,c)}

Received: January 6, 2021, Accepted: August 6, 2021

Abstract: Prediction of future demand for products is important for increasing sales in the retail industry and a number of time series forecasting methods have been studied in various fields. In recent years, some attempts have been made to utilize deep learning to improve the prediction performance. However, since previous methods using deep learning were not designed specifically for retail data, the validity for retail data is unclear. If we can incorporate a mechanism for capturing the characteristics of the data, we can develop models that can make more accurate predictions. Based on the above, this study proposes a predictive model based on deep learning that incorporates domain knowledge about the periodicity of retail data. This domain knowledge of periodicity is based on the fact that in addition to cyclical sales fluctuations common to multiple products, such as days of the week and months, there are different consumption cycles specific to each product, and each product's sales data is expected to reflect these different cycles. In this study, we extend the previous method to capture multiple periods that are different for each time series, and propose a method for making predictions that takes into account the domain knowledge about periodicity. We conducted experiments using several types of actual retail data and artificially generated data and the effectiveness of the proposed method is confirmed by improving its accuracy.

Keywords: forecasting, deep learning, time-series, marketing

¹ 東京大学
The University of Tokyo, Bunkyo, Tokyo 113-8656, Japan

a) hozumi@weblab.t.u-tokyo.ac.jp
b) iwasawa@weblab.t.u-tokyo.ac.jp
c) matsuo@weblab.t.u-tokyo.ac.jp

1. はじめに

小売業において、将来の商品の需要を正確に予測することは売上を増加させるうえで重要であり、企業が下す意思

決定においても、需要予測は重要な位置を占めている [18]. 企業は、将来の各商品需要を予測したうえで商品の発注を行うが、その際に、需要を高く見積もりすぎると過剰在庫が発生し、廃棄などによる経済的損害が生じる要因となる. 一方で、需要を低く見積もりすぎると、売り切れが発生してしまい、商品販売の機会損失が発生してしまう. そのため、売上をより正確に予測できるようにすることが、最適な発注や、ひいては利益の最大化につながる.

近年、このような需要予測において、属人性を排除することや、より正確な予測を行うことを目的として、計量経済学や統計学といった様々な分野で数々の時系列予測手法が研究されている. たとえば、主に計量経済学分野で使用される自己回帰 (AR; Autoregressive) モデルがある. 他にも、機械学習による予測モデルとして XGBoost [4] や LightGBM [10] といった手法も提案され、ライブラリとして公開されている. 近年では、さらに予測性を上げるために、深層学習を活用しようとする試みもある [3]. たとえば、Long- and Short- term Time-series Network (LSTNet) [13] は線形モデルに深層学習による予測モデルを組み合わせている. また、Significance-Offset Convolutional Neural Network (SOCNN) [2] は線形モデルのパラメータや入力の補正項を深層モデルによって学習させて予測を行っている.

ただし、これらの深層学習を用いた手法は小売データに特化して設計された手法ではなく、異なるドメインの系列データに向けて提案されたものであり、小売データにおける有用性は不明瞭である. たとえば、LSTNet [13] は交通量や電力といったドメインでの検証を行っている. 一方で、SOCNN [2] は金融というドメインに特化してモデルを構築し、検証を行っている. 文献 [2] のように、小売データの特性をふまえ、それをとらえる機構をモデルに組み込むことができれば、これまでの手法を超えるような予測精度を達成できる蓋然性は高い.

以上の内容をふまえ、本研究では図 1 で示されるような小売データが持つ周期性に関するドメイン知識を取り入れた深層学習による予測モデル (Consume Prediction Net; CPNet) を提案する. この周期性に関するドメイン知識とは、それぞれ商品の売上データごとに異なる複数の周期が含まれているということである. たとえば、消費財の売上データの時系列には、曜日や月 (季節) のような複数の商

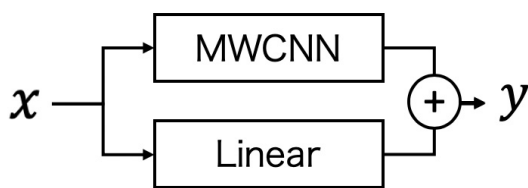


図 1 CPNet の概念図

Fig. 1 Overview of CPNet model.

品に共通する売上の周期変動の他に、商品ごとに特有の異なる消費の周期があり、それら複数の周期が反映されていると考えられる. 予測モデルにこのような知識を取り入れることができれば、予測精度を向上させる可能性は高い.

本研究では、既存手法である LSTNet を基に、時系列ごとに異なる複数の周期をとらえられるように拡張することで周期性に関するドメイン知識を考慮して予測を行う手法を提案する. そして、複数種類の実際の小売データや人工的に作成したデータを用いた予測実験を行い、その精度の向上をもって提案手法の有効性を確認する.

本研究で提案するモデルは、売上を把握する粒度が商品別、カテゴリ別、店舗別と異なっている場合や、時間方向の粒度が日時や月次のように異なっている場合であっても、同じ方法で売上予測ができる汎用性のあるモデルとなるようにすることを目標としている. 一方で、売上は商品の値段変動による効果や在庫による数量制限の効果の影響を受けると考えられ、売上を予測する際にはこれらの要因についても考慮すべきである. しかし、需要予測をする際につねにそれに関するデータが使用できるとは限らず、今回実験に使用したデータセットからもこれらの情報が把握できないため、今回提案する予測モデルでは商品の値段や在庫の影響については考慮しないこととする.

2. 関連研究

2.1 古典的な時系列予測モデル

これまでに研究されてきた古典的な時系列モデルの代表的な例として、自己回帰モデルがあげられる. 参照期間を M とした $AR(M)$ モデルにおける時点 t ($t \in \mathbf{N}$) での出力 x_t は、以下の式で示される.

$$x_t = \sum_{k=t-M}^{t-1} a_{(k-t+M+1)} x_k + c_t + \epsilon_t \quad (1)$$

ここでの a_1, a_2, \dots, a_M はモデルのパラメータ、 c_t は定数項、 ϵ_t は正規分布に従うノイズである. これはスカラー時系列の予測モデルであるが、ベクトル時系列 \mathbf{x}_t を予測する VAR モデルも存在する. また、これらのモデルは予測する時系列にマルコフ性を仮定している. これはいい換えれば、現在の値によって将来の値が定まることを意味する. ただし、実際は厳密にはマルコフ性が成立しない状況においてもマルコフ性が成立するように特徴量を設計することがある. たとえば、金融分野で用いられる移動平均線がその一例である. このようなモデルを実際に小売業に関連した時系列予測タスクに活用した研究として、文献 [1] があげられる.

他にも、実データに特化させた時系列予測手法として、Prophet [17] があげられる. このモデルは、入力データに対して 1 日や 1 週、1 年といった既定の周期によるフーリエ級数展開を行いつつ、それとは異なるトレンド項を推定

し、さらに別途与えられた外部の影響も反映させることで時系列のモデリングを行う。

2.2 機械学習による時系列予測モデル

有名な機械学習を使用した予測モデルとして、XGBoost [4] がある。これは勾配ブースティングと決定木を組み合わせた、時系列にとどまらない予測タスクに用いられる手法である。また、決定木の刈り取りを内部で自動で行うことによって、学習の高速化を実現している。その後、この研究をふまえ、より大きなデータセットに対応して高速な学習を実現する LightGBM [10] という派生手法も提案された。この手法は決定木の枝分かれの基準の探索を適切に省くことによって計算量を抑えることで、より大規模なデータに対して適用しやすくなる工夫がなされている。これらの手法はデータ予測コンペティションにおいてたびたび利用されているものであり、実際に Kaggle において好成績を出したモデルにも多く採用されている。

2.3 深層学習による時系列予測モデル

主に画像認識や翻訳タスクに用いられ、既存の機械学習による手法と比べて高い精度をあげてきた深層学習モデルは、近年では時系列の予測においても用いられるようになった。深層学習を用いた単純な例として、再帰的ニューラルネットワーク (RNN; Recurrent Neural Network) や学習実行時に過去の情報を失わないよう新たに内部にメモリセルを設定した Long-short Term Memory (LSTM) や Gated Recurrent Unit (GRU) の使用があげられる。ただし、RNN による予測は逐次的に入力を行い更新を行うため、入力に長期の時系列を入力した際には、機械学習による手法に比べて非常に多くの計算時間やメモリが必要になる。この問題をふまえ、入力を時間方向の 1 次元ベクトル (テンソル) ととらえ、それを学習して将来を予測する畳み込みニューラルネットワーク (CNN; Convolutional Neural Network) による時系列予測も行われるようになりつつある。深層学習による時系列予測の例には、音声認識分野では Wavenet [15] があり、それを時系列データ予測に応用した研究として [3] があげられる。

深層学習による実世界データの時系列予測の研究として、LSTNet [13] と SOCNN [2] があげられる。LSTNet は深層ネットワークによるモデルと AR モデルの学習を並行して実行し、最終的に両方を予測に用いる手法である。深層ネットワークのみを予測に用いると、周期性のある情報が失われる可能性があるが、AR モデル側に適切な参照期間を設計すればそれを保持することができ、最終的な予測性能が高まるということを示している。一方、SOCNN は時系列方向とベクトル内での畳み込みという二方向の畳み込みを行うベクトル時系列予測モデルである。RNN ではなく CNN を用いる理由としては、前述した計算時間の削

減のほか、畳み込みを重ねることでノイズが除去できる効果への期待がある。他にも、入力の一部観測できずに入力が離散的になってしまう場合にも、そのサンプリング間隔を同時に入力することでうまく予測を行える工夫もなされている。その結果、時系列の形成に様々な要素が考えられる金融時系列データの予測においても、他の時系列予測モデルに比べて高い精度を出している。本論文で提案する手法は主に文献 [2] や [13] から着想を得ている。これらの研究は深層モデルと線形 AR モデルの併用を指向している。

また、現状での小売業における時系列データの予測は、深層学習ではなく先述した XGBoost や LightGBM などが使用される場合が多く、深層学習を用いている研究事例はまだ数が少ない [18]。深層学習による研究の例としては、POS データから日本のスーパーマーケットにおける翌日の売上変動予測を試みた文献 [8] や、天気に関する情報を活用し日本のスーパーマーケットにおいて 1 週間後の売上が何倍になるかの分類問題を解いた文献 [14]、トルコのスーパーマーケットにおける売上予測をアンサンブル法によって試みた文献 [11]、商品売上だけではなくそれが属するカテゴリ全体の売上を入力することでエクアドルの小売チェーンの需要予測を試みた文献 [19]、深層学習を用いて EC サイトの確率的な需要予測を行う文献 [16] などがあげられる。これらの研究は単一のデータセットを用いて、精度を高めるために商品の特性や日ごとの天気、プロモーションの有無といった付加条件を可能な限り付与することによって精度を高めることを目標としているが、本研究で提案するモデルは、売上を把握する粒度が商品別、カテゴリ別、店舗別と異なっている場合や、時間方向の粒度が日時や月次のように異なっている場合であっても同一の方法で売上予測ができることを目標に設計され、複数のデータセットで検証されていることから、他の研究手法に比べると汎用性が高く適用範囲が広がっているといえる。また、直後の売上変動ではなく、ラグ期間を挟んだうえでの未来の売上を直接予測していることから、より実務への応用に即した手法であると考えられる。

3. 提案手法

3.1 問題設定

本論文で提案する手法は、実際の小売業分野での使用を想定した手法である。また、学習時にはなかった新商品や新店舗の売上予測を行いたい際にも学習したモデルを適用することができるようにするため、本研究では固定された次元数の多次元時系列 $\boldsymbol{x}(t)$ ではなく、1 次元時系列 $x(t)$ を入力とし、その予測を行う ($t \in \mathbf{N}$)。一般に、複数次元のベクトル時系列を扱う予測モデルのほうが入力される情報量が多くなり、特に深層学習を使用したモデルではその次元間の相互関係を学習することで予測精度が高まることが考えられる [2]。一方で、このような手法は入力する時系列

の次元を予測する商材の数に固定する必要があるため、たとえば、学習時にはなかった新商品や新店舗の売上予測を行いたい際に、その予測したい時系列を追加すると入力時系列の次元数が増えてしまうので、過去に学習した予測モデルをそのまま使用することができない。ゆえにコールドスタート問題に対応できず、予測したいデータを用いてモデルを一から学習し直さなければならない。これは小売業における実用性を失ってしまうことにつながるため、今回は1次元時系列の予測モデルを取り扱う。同様の理由で、曜日や天気、プロモーションの有無といった小売業時系列における追加の条件は入力しない。このため、本提案手法は周期性のないイベント効果については予測の対象としない。

そして、本手法は予測までのラグ L のある時系列予測タスクを行う、すなわち x_{k+L} ($L > 0$) を、それより L 前の長さ k の系列である $[x_1, x_2, \dots, x_k]$ より予測する。これは、本研究による予測結果を参考にした発注を行うと、その商品が納品されるまでにある程度の期間を要するということを前提としているからである。また、直後 x_{k+1} の予測を繰り返すことで逐次的に将来を予想するものではなく、ラグ L 後の値を直接予測する。そのほうが逐次的に予測する手法と比べて誤差が累積されず、最終的な予測精度が高くなるからである [5]。

発注の後に即時に商品が販売店に納品されるケースは少ないためラグ期間自体の設定は考慮しなければならないが、商品ごとに発注後に倉庫から販売店に輸送されるか、それとも在庫を持ってない商材であるため発注を受けてから商品の製造を始めなければならないか、など商品によってそのリードタイムは大きく異なるため、設定すべき適切なラグ期間が商材ごとに大きく異なる。また、発注する部署から直接自社工場に発注できるか、中間部署で注文を集約してから別の販売業者に発注を出すかなどで、商材だけではなく企業の実態に応じても予測までの適切なラグ期間の長さは大きく異なる。以上の理由で本研究では具体的な L の値についての詳細な検討は行わないが、その代わりに、複数種類のラグの場合による実験を行い、異なるラグ期間に対する予測モデルの汎用性を確認する。

3.2 周期性に関するドメイン知識

今回対象とする時系列の特徴の1つに、複数の周期が存在しているという点があげられる。人間の行動は1週間や1カ月、1年といった複数の異なる周期の影響を強く受けており、小売データなど人間の行動が反映されたデータにも、その周期が反映されている。その長さは時系列データのサンプリング周期によって異なるが、たとえば日ごとにサンプリングされた日時データの場合、週(7日)や月(約30日)といった周期が反映されていると考えられる。また、このような周期は、季節のような多くの商品に共通し

ているものもあるが、商品ごとに特定の周期で消費されて再購入されるものもあり、その消費周期は商材によって異なる。人の手によって将来の売上予測を行うときは、これら複数の時間幅で時系列をとらえ、その周期性などの傾向をとらえたうえで予測が行われる [9]。それぞれのデータに反映された周期をすべてとらえて明示的にモデリングするには実務上限界があるため、その周期性をとらえることができるモデルを設計し、それを用いて予測させるということが有効であると考えられる [17]。

従来の小売データ予測に関する研究では、時系列の周期性の知識を取り入れる試みとして、予測対象が Decomposable Time Series Model [7] で表すことができると仮定されていた。ここでは、予測できる時系列 y_t を以下の形式で書き表している。

$$y_t = g_t + s_t + h_t + \epsilon_t \quad (2)$$

この式における g_t はトレンド、 s_t は季節性、 h_t は休日効果、 ϵ_t はノイズである。これは、予測対象とする時系列の周期性を s_t や h_t としてモデリングしてとらえる手法であるが、実際のデータに含まれる周期性はより複雑であると考えられるので、できるだけ柔軟に複数の周期性をとらえられるようにしたいということが、本手法を提案する動機の一つである。

次に、上記のような複数周期性に関する仮定を深層モデルでとらえる方法として、カレンダー状の畳み込み、および複数ウィンドウによる畳み込みの2種類を検討する。

3.3 Cross CNN (CrCNN)

はじめに、予測対象の時系列が単調なトレンドと単純な周期を持つことを想定して設計した Cross CNN (CrCNN) を提案する。この手法は、折り返し周期を事前に設定したうえで、直近から折り返し周期分の間隔を開けたサンプリングを行った畳み込みと、最後から折り返し周期分のみ注目した畳み込みの2種類の畳み込みを行い、それらの結果を結合して予測結果を出力する。この処理を直感的に示すと、カレンダーのように入力データの折返しを行い、その縦方向と横方向に畳み込み処理をかける手法である。この処理の実例として、入力長が30で折り返し周期7に設定した場合の畳み込みの実例を図2に表す。数式で表すと、折返し周期を T として、以下の式(3)と表現できる。式の一部に Python における記法を使用しており、 $[-T:]$ は入力の最後尾から T 個分、 $[:, -1][::T]$ は入力の最後尾から順に T 個飛ばしでサンプリングしたものを表す。

$$c = (1 - u) \cdot \sigma(W_h \mathbf{x}_{[-T:]} + \mathbf{b}_h) + u \cdot \sigma(W_v \mathbf{x}_{[:, -1][::T]} + \mathbf{b}_v) \quad (3)$$

ただし、入力ベクトルを \mathbf{x} 、重み行列を W 、バイアスベクトルを \mathbf{b} 、畳み込みの影響の比率を $u \in [0, 1]$ とする。添

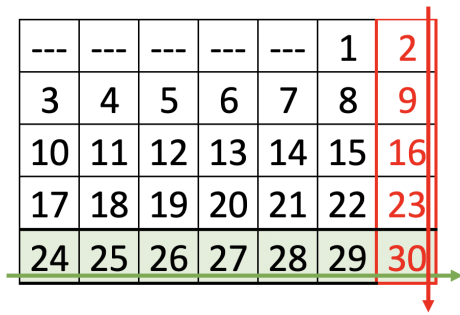


図 2 Cross CNN (CrCNN) の畳み込み例 (入力長 30, 周期 7). 横の緑色の矢印は式 (3) における $\mathbf{x}_{[-T:]}$ の参照部分, 縦の赤色の矢印は式 (3) における $\mathbf{x}_{[:-1][:T]}$ の参照部分を表す

Fig. 2 Example of Cross CNN (CrCNN) convolution (*input-length* = 30, *period* = 7). The horizontal green arrow represents the reference part of $\mathbf{x}_{[-T:]}$ and the vertical red arrow represents the reference part of $\mathbf{x}_{[:-1][:T]}$ in Eq. (3).

字 h は横方向, v は縦方向を表す. また, 活性化関数 σ はシグモイド関数とする. 日次にサンプリングしたデータに対しては週に相当する $T = 7$ で, 週次でサンプリングしたデータに対しては月に相当する $T = 4$ で折り返しを行うことを想定する.

この手法により, 単調なトレンドについては縦方向の畳み込みで, 周期については横方向の畳み込みによって学習できることが期待される. この二方向の畳み込みにより, 週や月といった明確な周期に関する知識を予測結果に取り入れられると考えられるが, それ以外の商品ごとに異なる周期をとらえることができないという欠点がある. このことをふまえ, 次に, 異なるサイズのウィンドウを用いて畳み込みを行う手法を提案する.

3.4 Multi-Window CNN (MWCNN)

先述したように, 小売データには複数の周期があることを前提とし, それらなるべく多くとらえて予測に反映させることが予測精度の向上につながると考えられる. これをふまえて, この人間が行うような着眼を予測モデルに組み込み直接実装する Multi-Window CNN (MWCNN) を提案する. 具体的には, 以下の演算を行う. この概念は図 3 のように描ける.

$$c = \sum_{k=1}^K w_k \cdot \sigma(W_k * \mathbf{x} + b_k) \quad (4)$$

ただし, W_k のウィンドウサイズは 2^k ($k \in \mathbf{N}$) とし, そのウィンドウサイズごとの重みを w (ただし $\sum_{k=1}^K w_k = 1$) で示す. また, $*$ は畳み込み演算を表す. 本論文の実験では k の最大値 K の値を 5 に設定する. これは, 小売データに想定される日足で記録されたデータに対して 1 カ月に相当する 30 日分をカバーする 32 をフィルタサイズに設定すべきと考えられるからである.

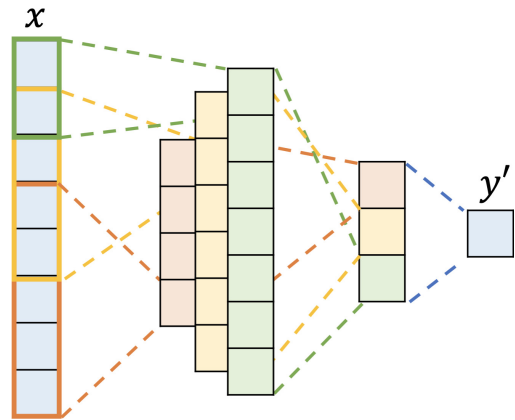


図 3 Multi-Window CNN の概念図. 1 層目では複数種類のウィンドウサイズによる畳み込みを行い, 2 層目と 3 層目ではそれぞれの畳み込み結果を全結合していく

Fig. 3 Overview of Multi-Window CNN. In the first layer, convolution with multiple window sizes is performed, and in the second and third layers, the outputs of each convolution are fully connected.

この手法は先述した CrCNN に比べ, 予測に活用する周期やトレンドを明示的に定義したものではないが, モデルによってそれらも学習できると考えられる. たとえば, $k = 5$ (ウィンドウサイズ 32) のフィルタが 7 日おきに 0 になるような学習が行われると, それは式 (3) のトレンド成分を学習させたことになる. 本手法はこのような週ごとの周期だけでなく, 商品ごとに異なる周期もとらえて予測に反映させられると考えられる.

3.5 線形スキップコネクション

ここまで周期性を学習できるような CNN 構造を提案したが, 文献 [2] や [13] でも述べられているように, 予測対象に周期性や線形のトレンドが存在することが明らかである場合には, 線形 AR モデルの式を直接使うことが予測精度を担保するうえで望ましいと考えられる. それをふまえ, 前述の CrCNN や MWCNN を用いた予測と同時に AR モデルによる予測を並行して行う. AR モデルも採用することによって, 予測する時系列に強い周期性やトレンドがある場合に, より早期にその特徴をとらえられるようになると思われる. このような深い深層学習モデルと浅い AR モデルを組み合わせる考え方は, 特に文献 [13] で明示的に導入されているが, 異なるタスクにおいても, たとえば推薦アルゴリズムである Wide and Deep Learning [6] にも取り入れられている.

文献 [13] では線形 AR モデルと深層モデル両方の出力を単純に合算したものを最終出力とし, それが実際の値と近くなるように学習を進めるが, 本手法では AR モデルを学習させると同時に, 深層モデルと線形 AR モデルの重みを合算する際の比率 $u \in [0, 1]$ も学習させ, それらを合算する. 線形スキップコネクションの重みベクトルおよびバイ

アス (定数項) をそれぞれ w_l , b_l と置くと, 式 (4) の c に引き続き, 最終的な出力 y を以下のように書ける.

$$y = (1 - u)(w_l \cdot x + b_l) + u \cdot c \quad (5)$$

提案手法である CPNet は Multi-Window CNN と線形スキップコネクションによって構成され, その概形は図 1 で表される.

4. 実験

本論文における手法が, 実際の小売業の時系列予測に対して有効であるかどうかを, 複数の小売業のデータに対して予測実験を行うことで検証した. これらのデータには店舗単位や部門単位, 週次や日次という性質や粒度が異なるデータを採用し, 本手法が小売業の時系列に対し汎用的なものであるかを検証した. さらに, ここではラグ期間を複数設定し, ラグの長さの変化による予測結果の変化も検証した. 比較対象とするモデルの中には曜日や天気, プロモーションの有無といった追加の条件を付与することで精度を高めることができるモデルがあるが, 本研究ではそのようなデータが使用できない場合にも精度を担保することを目標とし, 実験に使用するデータセットに含まれていないものもあるため, これら追加の条件は使用しないこととした.

各モデルの学習は, 訓練データを用いて行う. ハイパーパラメータが設定できるモデルについては, 異なるハイパーパラメータの組合せごとに学習を行い, バリレーションデータによる予測結果を比較することで, 最適なパラメータを選択する. 最後に, モデルの最終的な予測精度を, テストデータを用いた予測精度によって検証する.

4.1 データセット

本研究では小売業に関する公開されている実データとして Rossmann と Walmart と Favorita, そして人工的に作成したデータを用いて実験を行う. また, 人工データ以外のデータセットについては, 系列の一部に欠損が含まれていたため, その前後の値を用いて線形補完を行っている.

4.1.1 Rossmann

ヨーロッパに展開するドラッグストアである Rossmann による, 店舗ごとの日次の売上総額を記録したデータセット^{*1}の訓練用データセット (train.csv) における, Sales カラム時系列データである. 店舗数は 1,155 である. この売上データは, 日曜日が定休日であるため 7 日ごとに売上が必ず 0 になるという特徴があるため, 事前に日曜日に相当する分の売上データは取り除き, 予測の対象としないこととする. 例としてそのデータの概形を図 4 に表す. 訓練データ期間は 2013/1/1~2013/12/31, バリレーションデータ期間は 2014/1/1~2014/12/31, テストデータ期間

^{*1} <https://www.kaggle.com/c/rossmann-store-sales>

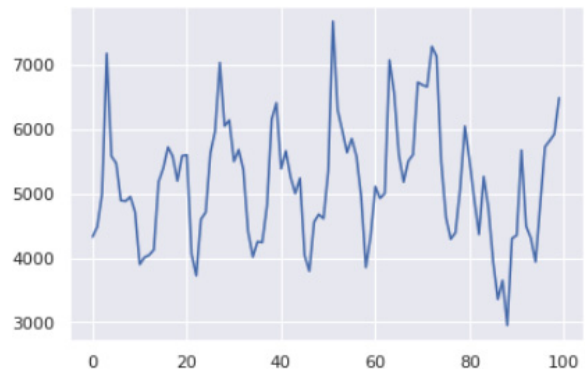


図 4 Rossmann データの一部概形

Fig. 4 Example time series of Rossmann dataset.

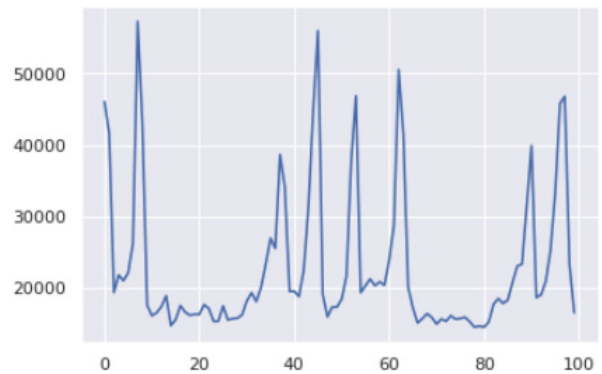


図 5 Walmart データの一部概形

Fig. 5 Example time series of Walmart dataset.

は 2015/1/4~2015/7/31 である. 説明変数となるモデルへの入力長は 90 (3 カ月相当) とし, 目的変数としてラグ期間を挟んだ後の売上を予測する. 予測までのラグは 0, 30 (1 カ月相当) および 60 (2 カ月相当) に設定する.

4.1.2 Walmart

アメリカに展開するスーパーマーケットである Walmart の, 店舗ごとの週次の売上総額を記録したデータセット^{*2}の訓練用データセット (train.csv) における, Weekly_Sales カラム時系列データである. その一部の概形を図 5 に表す. このデータセットには店舗 (Store) 45 店分のデータセットが含まれるが, それにさらに 81 種類の部門 (Dept) コードが別に付与されているので, 店舗ごとではなく各店舗の部門ごとに, 週次での売上予測を行う. 訓練データ期間は 2010/2/5~2011/9/30, バリレーションデータ期間は 2011/10/07~2012/3/30, テストデータ期間は 2012/4/6~2012/10/26 である. 説明変数となるモデルへの入力長は 24 (6 カ月相当) とし, 目的変数としてラグ期間を挟んだ後の売上を予測する. 予測までのラグは 0, 8 (2 カ月相当) および 12 (3 カ月相当) に設定する.

4.1.3 Favorita

エクアドルに展開する大手食料品小売企業である Corpo-

^{*2} <https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting>

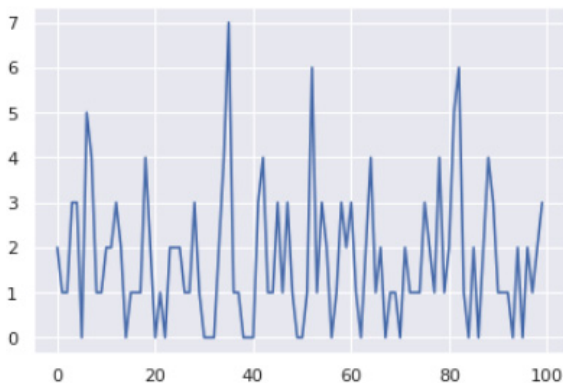


図 6 Favorita データの一部概形

Fig. 6 Example time series of Favorita dataset.

ración Favorita による、店舗ごとの日時の商品別販売数を記録したデータセット*3の訓練用データセット (train.csv) における、unit_sales カラム時系列データである。対象となる店舗別販売商品数のレコードは全部で 105,635 種類あるが、それらをすべて予測に用いるとメモリ不足が発生するモデルがあるため、そのうちの 20%に相当する 21,127 商品のレコードを無作為に選出し、それらを使用する。例としてそのデータの概形を図 6 に表す。訓練データ期間は 2013/1/1~2013/12/31、バリデーションデータ期間は 2014/1/1~2014/6/30、テストデータ期間は 2014/7/1~2014/12/31 である。説明変数となるモデルへの入力長は 90 (3 カ月相当) とし、目的変数としてラグ期間を挟んだ後の売上を予測する。予測までのラグは 0, 30 (1 カ月相当) および 60 (2 カ月相当) に設定する。

4.1.4 人工データ

以上の公表されている実世界での時系列データの他に、以下の式で表される時系列を作成した ($t \in \mathbf{N}$)。

$$x_t = \sin\left(\frac{2\pi}{7}t\right) + 2\sin\left(\frac{2\pi}{30}t\right) + 4\sin\left(\frac{2\pi}{365}t\right) + 0.0005t \quad (6)$$

これは、人間の活動によって生み出される時系列データが週次 (7 日)、月次 (30 日)、年次 (365 日) の周期と右肩上がりのトレンドが含まれる時系列であることを仮定して、この特徴が単純に反映された時系列データを直接正弦波を用いて実装したものである。その一部の概形を図 7 に表す。訓練データ期間は $t = 0 \sim 4,379$ (12 年相当)、バリデーションデータ期間は $t = 4,380 \sim 5,839$ (4 年相当)、テストデータ期間は $t = 5,840 \sim 7,299$ (4 年相当) をとる。説明変数となるモデルへの入力長は 90 (3 カ月相当) とし、目的変数として入力からラグ期間を挟んだ後の売上を予測する。予測までのラグは 0, 30 (1 カ月相当) および 60 (2 カ月相当) に設定する。

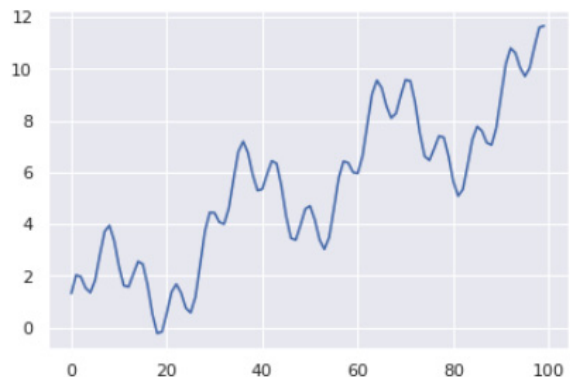


図 7 人工データの一部概形

Fig. 7 Example time series of artificial dataset.

4.2 実験設定

本研究では提案手法との比較対象として、時系列予測にたびたび用いられる深層学習を用いないモデルである線形自己回帰モデル (AR), Prophet [17] および XGBoost [4], 基本的な深層学習を用いた予測モデルである多層パーセプトロン (MLP), CNN および RNN, そして様々な時系列予測を目的として設計された深層学習を用いた予測モデルである LSTNet [13], SOCNN [2] および DeepAR [16] を設定し、それらとの比較実験を行った。DeepAR は RNN をベースとした、1 次元時系列を確率的に予測するための予測モデルである。また、MWCNN と線形スキップコネクションそれぞれの予測性能を評価するために、提案手法から線形スキップコネクションを省略したモデルである CrCNN および MWCNN のみによる予測実験も行った。

各モデルの損失関数には二乗平均平方根誤差 (RMSE) を採用し、それが最小化されるよう学習を進める。いずれのモデルも最適化関数は Adam [12], 活性化関数は LeakyRelu を採用し、勾配が最大 1.0 になるよう Gradient Clipping を課している。バッチサイズはいずれも 1,028 としたうえで、Rossmann は 2,000, Walmart は 200,000, Favorita および人工データは 20,000 エポック学習させた。これらのモデルは、AR モデルは $N = 7$, 深層モデルはいずれも入力層、隠れ層、出力層の三層で構築している。中間層のユニット数や CNN のチャンネル数は、ハイパーパラメータ自動最適化フレームワークである Optuna *4 を用いて [20, 200] より最適な値を探索した結果、いずれのモデルも 120 前後が最適であったため、一律で 120 に設定した。ウィンドウサイズ (MWCNN では k の最大値 K) は $k = 3, 5, 7$ より最も予測精度が高くなるものを選択する。XGBoost および Prophet, DeepAR において設定可能な各種ハイパーパラメータについても、Optuna によりそれぞれ 50 回の試行により最適な値を探索し、その値を設定した。

*3 <https://www.kaggle.com/c/favorita-grocery-sales-forecasting>

*4 <https://optuna.org/>

4.3 評価指標

各モデルの性能の特徴を多面的に評価するために、評価指標は以下の3種類を採用した。ここでの y は正解値、 \hat{y} は予測値、 n はデータセット内のデータ数、 i はデータセット内の各データに振られた id を意味する。

4.3.1 二乗平均平方根誤差 (RMSE)

先述のとおり、RMSE (Root Mean Squared Error) は学習時の目的関数でもあり、各予測手法は訓練データにおいてそれが最小となるように学習を進行させる。そして、テストデータにおいてこの値が小さいほど精度の高い予測であるといえる。

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

4.3.2 平均平方二乗誤差率 (RMSPE)

先述の RMSE は、売上のスパイクなど急激な変化における誤差を、そうでない時点と比較して過剰に考慮してしまう傾向がある。このため、正解値と予測値との誤差を正解値で割ることでスケールした値の二乗誤差を見る RMSPE (Root Mean Square Percentage Error) も評価指標として用いる。RMSPE も、テストデータにおいてこの値が小さいほど精度の高い予測であるといえる。

$$RMSPE = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{y_i - \hat{y}_i}{y_i} \right)^2} \quad (8)$$

4.3.3 相関係数 (CORR)

テスト結果における予測結果と実際の値の相関係数を算出し、実際の値と予測結果の関係性を確認する。この値が1に近いほど、予測値と実際の値の大小関係の一致具合が高く、よりよい予測であるといえる。

$$CORR = \frac{\sum_{i=1}^n (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2 \sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})^2}} \quad (9)$$

4.4 実験結果

実験結果を表1に記す。記載されている結果は、テストデータセットに含まれるそれぞれのテストデータごとに予測を行い各評価指標を計算した後、それらの平均値を記している。いくつかの結果を除き、提案手法である CPNet が高いスコアを示していることが確認された。ラグ期間を広げた場合には、全体として予測精度が低下する傾向があることも確認された。

CrCNN および MWCNN は通常の CNN と比べ、特に人工データの場合に大きく RMSE が改善されていることから、予測対象データに含まれる周期性が明確な場合に、より有効な手法であると考えられる。そして、線形スキップコネクションを用いた場合、CrCNN と MWCNN のどちらの場合においてもそれを使用しなかったモデルに比べて予測精度が向上していることから、その有効性が確認でき

る。一方で、RNN ではどの実験においても良い精度が出ずに他のモデルに比べて精度が悪くなっている。特に、AR モデルや RNN を用いたモデルについてラグ期間が0である場合とそうでない場合の結果を比較すると、他のモデルに比べて実験結果に大きな差があることも確認される。これより、RNN は予測対象までのスキップ期間が空いてしまう時系列予測タスクにおいては予測に不向きであることから、小売データの予測において CNN に比べて不向きであるとも考えられる。

CPNet における Rossmann (ラグ期間 30) データ予測時のウィンドウサイズごとの重み w を表2に記す。 w の総和は1であり、ウィンドウサイズが8および32における重みが他のウィンドウサイズと比較して大きいことから、週単位 (ウィンドウサイズ8) および月単位 (ウィンドウサイズ32) の長さにおける時系列の積み込みの重要性が示唆される。また、このモデルにおける線形スキップコネクションの MWCNN 側の重み u は 0.7551 であった。CPNet の出力における MWCNN の結果が占める比率は大きいものの、線形モデル側の重み $1-u = 0.2449$ も無視できるほど小さくはないため、本研究における AR モデルの精度自体は高くないものの、MWCNN と相互に補完し合うことによって CPNet の出力結果に貢献していると考えられる。

CrCNN, CrCNN+AR, MWCNN および CPNet (MWCNN+AR) における Walmart (ラグ期間8) データ予測時のエポックごとの RMSE の変化を図8に記す。線形スキップコネクション (AR) を併用するモデルは、そうでないモデルに比べて早急に収束することが分かる。また、学習したモデルで実際に時系列予測を行い正解データと比較したものを図9に記す。一部のモデルは大きく予測を外した部分があり、それを同一のグラフに可視化すると他のグラフが潰れてしまい予測の特徴がとらえられなくなってしまうので、比較的予測を大きく外していないモデルのみを抜粋している。CNN や LSTNet, SOCNN は他のモデルに比べて大きく予測値を外してはいないものの、予測対象とする時系列の周期の特徴をあまりとらえられていない結果を出力している。一方で、CrCNN では予測対象の周期のピークの一部をとらえて予測ができていたものの、他の部分でも周期性に沿って大きく増減してしまっており、結果として全体の予測精度が低くなってしまっている。提案手法である CPNet は、他のモデルに比べて大きく値自体を外すことなく周期的な増減もとらえられていると考えられる。

5. 考察

本研究では、小売時系列予測に特化した予測手法である CPNet を提案した。複数周期に関するドメイン知識をふまえ、CrCNN および MWCNN を用いた予測手法を提案し、4章の結果より、本提案手法の優位性を確認すること

表 1 テスト結果のまとめ (最も良い結果のものを太字で表記)

Table 1 Results summary of our experiment. The best results are shown in bold.

データセット		Rossmann			Walmart			Favorita			人工データ		
		ラグ期間			ラグ期間			ラグ期間			ラグ期間		
モデル	評価指標	0	60	90	0	8	12	0	60	90	0	30	60
AR	RMSE	1619.4	2117.8	2414.4	7207.5	8705.6	9284.5	0.6861	0.9293	0.9731	0.118	0.133	0.142
	RMSPE	0.249	0.277	0.347	0.815	0.865	0.876	1.414	1.426	2.912	22.81	26.78	25.94
	CORR	0.087	-0.131	-0.147	0.091	0.052	-0.101	0.014	0.025	0.014	0.663	0.673	0.694
Prophet	RMSE	1803.9	1861.5	1876.4	16.112	17.998	18.065	0.0920	0.1412	0.1773	0.068	0.041	0.046
	RMSPE	0.225	0.223	0.281	0.181	0.166	0.264	1.110	1.122	1.561	4.214	4.422	5.782
	CORR	0.571	0.554	0.511	0.450	0.446	0.390	0.420	0.392	0.262	0.712	0.710	0.695
XGBoost	RMSE	1949.3	1980.6	2012.4	22.105	25.881	26.229	0.1795	0.1759	0.1561	0.056	0.066	0.052
	RMSPE	0.249	0.272	0.336	0.182	0.171	0.229	2.128	2.285	2.447	4.388	5.886	6.211
	CORR	0.494	0.456	0.441	0.416	0.420	0.363	0.247	0.282	0.231	0.577	0.588	0.624
DNN	RMSE	1974.2	2144.4	2331.3	425.27	442.15	621.96	0.0072	0.0075	0.0076	0.544	0.512	0.528
	RMSPE	0.214	0.272	0.282	1.561	1.480	1.210	1.260	1.167	1.771	9.215	9.238	9.927
	CORR	0.198	0.144	0.161	0.255	0.221	0.171	0.457	0.401	0.382	0.469	0.337	0.227
CNN	RMSE	2431.2	2666.1	2641.3	472.42	521.97	518.85	0.5785	0.7023	0.7043	0.274	0.372	0.536
	RMSPE	0.385	0.411	0.458	0.285	0.239	0.290	3.672	3.556	4.741	8.244	10.466	8.161
	CORR	0.319	0.216	0.149	0.179	0.126	0.121	0.386	0.365	0.298	0.685	0.789	0.689
RNN	RMSE	2546.4	3038.8	3680.8	1979.7	2219.8	5462.5	0.0256	0.0556	0.0605	0.395	0.538	0.759
	RMSPE	0.385	0.691	0.882	0.662	0.977	0.953	2.281	4.628	6.925	8.095	8.877	8.796
	CORR	0.348	-0.053	-0.043	0.184	-0.009	-0.018	0.098	0.022	0.017	0.196	0.126	0.112
LSTNet	RMSE	2268.7	2735.6	2868.0	17.535	18.526	17.758	0.0684	0.0575	0.0578	0.242	0.209	0.250
	RMSPE	0.329	0.291	0.288	0.193	0.147	0.299	0.865	0.923	1.112	24.56	26.33	29.42
	CORR	0.465	0.267	0.354	0.106	0.226	0.243	0.453	0.342	0.388	0.789	0.512	0.551
SOCNN	RMSE	1719.9	2036.7	2325.5	18.067	18.046	16.251	0.8251	0.7196	0.7156	0.028	0.039	0.040
	RMSPE	0.218	0.276	0.286	0.381	0.387	0.408	1.138	1.183	0.997	23.16	26.18	26.64
	CORR	0.408	0.373	0.364	0.437	0.399	0.416	0.376	0.352	0.382	0.719	0.694	0.621
DeepAR	RMSE	1975.7	2140.4	2405.5	16.35	20.09	21.05	0.0385	0.0480	0.0492	0.031	0.033	0.039
	RMSPE	0.308	0.327	0.462	0.194	0.236	0.258	1.484	1.266	1.678	4.609	4.727	4.998
	CORR	0.476	0.511	0.488	0.479	0.486	0.456	0.268	0.254	0.250	0.619	0.661	0.622
CrCNN	RMSE	2480.5	2549.7	2633.1	179.35	180.92	184.61	0.192	0.0168	0.0198	0.049	0.051	0.053
	RMSPE	0.511	0.489	0.494	0.349	0.372	0.381	0.714	0.781	0.664	23.85	25.33	26.72
	CORR	0.419	0.332	0.343	0.392	0.388	0.376	0.373	0.414	0.355	0.484	0.409	0.442
CrCNN+AR	RMSE	1792.7	1809.8	2132.2	17.039	18.597	16.942	0.0168	0.0187	0.0205	0.047	0.044	0.058
	RMSPE	0.228	0.258	0.269	0.194	0.189	0.212	0.528	0.536	0.628	4.651	5.445	6.778
	CORR	0.489	0.470	0.443	0.456	0.446	0.479	0.579	0.454	0.506	0.765	0.737	0.643
MWCNN	RMSE	1848.9	2208.5	2368.2	16.175	17.986	17.755	0.0011	0.0012	0.0017	0.028	0.038	0.039
	RMSPE	0.376	0.400	0.391	0.528	0.572	0.596	0.569	0.698	0.776	27.71	26.13	28.32
	CORR	0.491	0.496	0.449	0.509	0.395	0.381	0.408	0.389	0.360	0.142	0.179	0.085
CPNet	RMSE	1259.5	1486.9	1512.9	15.563	17.672	16.519	0.0004	0.0006	0.0007	0.027	0.031	0.036
	RMSPE	0.175	0.237	0.268	0.179	0.149	0.199	0.479	0.542	0.556	4.175	4.569	4.824
	CORR	0.602	0.578	0.532	0.628	0.478	0.481	0.581	0.516	0.487	0.814	0.795	0.696

表 2 ウィンドウサイズごとの重み w (Rossmann, ラグ期間 30)

Table 2 Weights w for each window size (Rossmann dataset with lag = 30).

k	ウィンドウサイズ	w
1	2	0.0765
2	4	0.1364
3	8	0.3576
4	16	0.1626
5	32	0.2669

ができた。古典的な時系列予測手法を用いる場合は、予測対象とする時系列データの特徴をとらえたうえでそのモデリングを行うが、それを行うためには、専門知識や試行錯誤のための工数が必要になる。本手法は小売時系列の特徴を前提として設計された手法であり、予測方法に深層学習を検討する際にベースラインとして比較的容易に採用できる手法であるといえる。

今回の実験では、予測対象が複数の周期性を持つことを

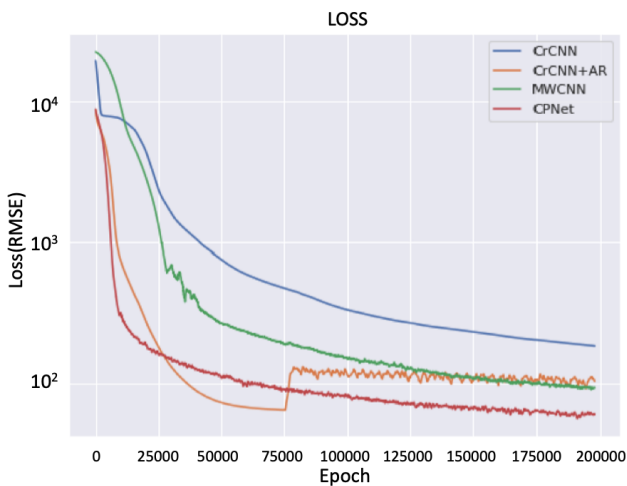


図 8 線形スキップコネクションを用いたモデルにおける RMSE の変化 (Walmart, ラグ期間 8)

Fig. 8 Changes in RMSE by models with linear skip-connections (Walmart dataset with lag = 8).

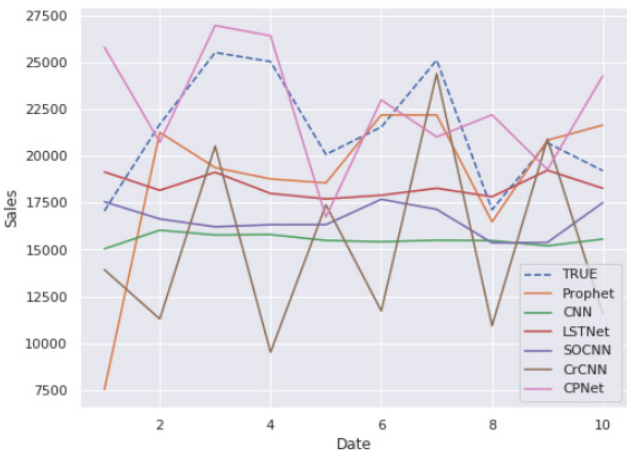


図 9 実際の売上 (TRUE) と予測モデルを用いた売上予測結果の例 (Walmart, ラグ期間 8)

Fig. 9 The true sales time series (TRUE) and the predicted time series with prediction models (Walmart dataset with lag = 8).

考慮している MWCNN を用いたモデルのほうがより精度の高い結果となったが、データセット内の商材のバリエーションが少ない場合や、学習に用いることができるデータセットが小さくなってしまふような場合には、MWCNN の代わりに CrCNN による畳み込む周期を固定したモデルを用いるほうがより高い予測性能を示す可能性があるとも考えられる。

提案手法である CPNet における予測結果に注目すると、Rossmann および Walmart の RMSPE が 0.2 前後であることから予測結果は平均でおよそ 2 割程度の誤差にとどまっていることが分かり、実務でも予測結果を活用することができる可能性があることが示唆される。一方で、人工データによる予測結果を見ると RMSPE が 4 を超えており、平均で 5 倍近く予測結果に誤差があることが分か

る。Favorita や人工データは予測する値自体が Rossmann や Walmart に比べて小さいことから、RMSE が小さく RMSPE が大きくなっていると考えられる。このため、大型で高単価な商品など予測対象の絶対数が数個程度と小さく、予測精度をより高めていく必要がある場合には、本提案手法は実務に耐えるものとはいえないため、今後の研究課題となる。

ただし、本手法は、大量のデータを用いる場合に大規模なマシンリソースや学習時間を要する。CPNet の線形スキップコネクションは文献 [13] で示されるように学習時間を短縮するための工夫の 1 つであるが、精度を大きく下げずにより高速に学習をすすめるようにすることが、より大規模な小売業での実務応用に向けた今後の課題である。また、予測精度をさらに上げていくためには、タスク自体を文献 [2] で示されているように入出力時系列がベクトルである多変量時系列回帰問題として設定することや、曜日や天気、プロモーションの有無といった追加条件の付与が有効であると考えられる。本手法は入力 scaler 時系列であることによる制約はないので、ベクトル時系列を予測する場合でもある程度の精度が出ることが期待されるが、具体的なモデルの設計方法や、新商品や新店舗の売上予測のようなコールドスタート問題に対してどのように対応して推定するかは、今後の研究課題である。

6. まとめ

本論文では、小売業での実務利用のための、深層学習と AR モデルを活用した時系列予測モデル (CPNet) を提案した。これは複数周期をとらえる Multi-Window CNN と AR モデルを並行させる線形スキップコネクションによる時系列予測モデルであり、実際にその精度が従来手法よりも高いことを実験により確認した。本論文の提案手法は、予測対象とする時系列の周期の把握をする必要がなく、予測までのラグ期間によらずに一定の精度が期待できるため、特徴量の設計を行わずに深層学習を用いて直接将来予測する場合におけるベースライン手法になると考えられる。本研究で提案した手法が学术界にとどまらず本当に実務において使用されることで、深層学習が実際に売上予測などに活用できるということが認知され、さらには売上予測以外の場面においても使用が検討されていくようになると嬉しい。

参考文献

- [1] Arunraj, N.S. and Ahrens, D.: A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting, *International Journal of Production Economics*, Vol.170, pp.321–335 (2015).
- [2] Bińkowski, M., Marti, G. and Donnat, P.: Autoregressive Convolutional Neural Networks for Asynchronous

- Time Series, *Proc. International Conference on Machine Learning*, pp.580–589, PMLR (2018).
- [3] Borovykh, A., Bohte, S. and Oosterlee, C.W.: Conditional Time Series Forecasting with Convolutional Neural Networks, arXiv preprint arXiv:1703.04691 (2017).
- [4] Chen, T. and Guestrin, C.: XGBoost: A Scalable Tree Boosting System, *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.785–794, ACM (2016).
- [5] Chevillon, G.: Direct Multi-Step Estimation and Forecasting, *Journal of Economic Surveys*, Vol.21, No.4, pp.746–785 (2007).
- [6] Cheng, H.T., Koc, L., Harmsen, J., et al.: Wide & Deep Learning for Recommender Systems, *Proc. 1st Workshop on Deep Learning for Recommender Systems*, pp.7–10, ACM (2016).
- [7] Harvey, A.C. and Peters, S.: Estimation Procedures for Structural Time Series Models, *Journal of Forecasting*, Vol.9, No.2, pp.89–108 (1990).
- [8] Kaneko, Y. and Yada, K.: A Deep Learning Approach for the Prediction of Retail Store Sales, *Proc. 2016 IEEE 16th International Conference on Data Mining Workshops*, pp.531–537, IEEE (2016).
- [9] Ilya, K., 株式会社クイープ (訳): AI アルゴリズムマーケティング 自動化のための機械学習/経済モデル, ベストプラクティス, アーキテクチャ, インプレス (2018).
- [10] Ke, G., Meng, Q., Finley, T., et al.: LightGBM: A Highly Efficient Gradient Boosting Decision Tree, *Proc. Advances in Neural Information Processing Systems*, pp.3146–3154 (2017).
- [11] Kilimci, Z.H., Akyuz, A.O., Uysal, M., et al.: An Improved Demand Forecasting Model Using Deep Learning Approach and Proposed Decision Integration Strategy for Supply Chain, *Complexity*, Vol.2019 (2019).
- [12] Kingma, D.P. and Ba, J.: Adam: A Method for Stochastic Optimization, arXiv preprint arXiv:1412.6980 (2014).
- [13] Lai, G., Chang, W.C., Yang, Y. and Liu, H.: Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks, *Proc. 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp.95–104, ACM (2018).
- [14] Liu, X. and Ichise, R.: Food Sales Prediction with Meteorological Data – A Case Study of a Japanese Chain Supermarket, *Proc. International Conference on Data Mining and Big Data*, pp.93–104, Springer (2017).
- [15] Oord, A., Dieleman, S., Zen, H., et al.: WaveNet: A Generative Model for Raw Audio, arXiv preprint arXiv:1609.03499 (2016).
- [16] Salinas, D., Flunkert, V., Gasthaus, J. and Januschowski, T.: DeepAR: Probabilistic forecasting with autoregressive recurrent networks, *International Journal of Forecasting*, Vol.36, No.3, pp.1181–1191 (2020).
- [17] Taylor, S.J. and Letham, B.: Forecasting at Scale, *The American Statistician*, Vol.72, No.1, pp.37–45 (2018).
- [18] Tsoumakas, G.: A survey of machine learning techniques for food sales prediction, *Artificial Intelligence Review*, Vol.52, pp.1–7 (2019).
- [19] 岡留有哉, 魏 文鵬, 相菌敏子: 複数の再帰型ニューラルネットワークを用いた需要予測アーキテクチャの開発, 電子情報通信学会論文誌 D, Vol.103, No.1, pp.24–33 (2020).



保住 純 (正会員)

2013年東京大学工学部システム創成学科知能社会システムコース卒業。2015年同大学院工学系研究科技術経営戦略学専攻修士課程修了。広告代理店勤務を経て、2021年同大学院工学系研究科技術経営戦略学専攻博士課程修了。

現在、同大学院工学系研究科システム創成学専攻特任研究員。博士(工学)。専門は機械学習・深層学習応用。



岩澤 有祐

2012年に上智大学(情報理工学科, 矢入研究室)を卒業。2014年に上智大学大学院博士前期課程(情報学領域, 矢入研究室)を修了。2017年に東京大学大学院博士後期課程(技術経営戦略学専攻, 松尾研究室)を修了。2017

年より東京大学松尾研究室で特任研究員, 特任助教を経て2020年12月より特任講師(現職)。博士(工学)。専門は深層学習, 特にウェアラブルセンシングへの応用と, 知識転移に関する技術。



松尾 豊 (正会員)

1997年東京大学工学部卒業。2002年同大学院博士課程修了。博士(工学)。産業技術総合研究所, スタンフォード大学を経て、2007年東京大学大学院工学系研究科技術経営戦略学専攻准教授。2019年より同大学院人工物工学

研究センター/技術経営戦略学専攻教授。2014~2018年まで人工知能学会倫理委員会委員長。2017年より日本ディープラーニング協会理事長。人工知能学会論文賞, 情報処理学会会長尾真記念特別賞, ドコモモバイルサイエンス賞受賞。専門は, 人工知能, 深層学習, Web工学。