

圧延工程を考慮した温度予測 Transformer モデル A Temperature Prediction Transformer Model Considering Hot Rolling Process

竹川 修平*¹
Shuhei Takekawa

関本 真康*²
Masayasu Sekimoto

新居 稔大*²
Toshihiro Nii

今成 宏幸*²
Hiroyuki Imanari

小原 一浩*²
Kazuhiro Ohara

内田 真人*¹
Masato Uchida

概要

熱間圧延は、回転するロールを通して高温の鋼材を薄く伸ばす重要な加工工程である。この工程は粗圧延と仕上圧延の段階に分かれ、それぞれの段階で5つの異なる位置に設置された温度センサによって鋼板の温度が測定される。熱間圧延において管理すべき重要な項目の一つに温度があり、特に圧延材の加工後温度 (FDT) を予測し、目標値に制御することが高品質を得る上で重要である。従来の機械学習を用いた手法の研究では、静的特徴量に基づいた予測モデルが主に使用されてきたが、これらはプロセス中の変動をリアルタイムで反映することができない。本研究では、熱間圧延プロセス全体を通じた高精度な時系列予測モデルを構築することを目指し、段階的デコーダを組み込んだ新しい Transformer モデルを提案する。このモデルは、圧延前の初期データから予測を開始し、各温度センサから新しいデータが取得されるたびに予測を更新する。これにより、最新のデータを反映した逐次的な予測が可能となる。実験では、本研究で提案したモデルを MLP, Autoformer, Crossformer, LTSF-Linear などの従来のベースラインモデルと比較した。実験結果より、提案モデルがベースラインモデルに比べて予測精度に優れていることが明らかとなった。

1 序論

熱間圧延は、回転するロールに高温の鋼材を通して薄く伸ばす加工法であり、この工程で製造された鋼板は多様な産業で広く利用されるため、極めて重要な加工工程である。熱間圧延は、図1に示すように、段階的に行われる。一般に、粗圧延機は1~3基、仕上圧延機は6~7基の圧延スタンドで構成される。まず、圧延材は粗圧延機で、厚さ250mmから30~40mm程度に圧延され、その後、仕上圧延機で1.2~12mmほどの目標板厚まで圧延される。熱間圧延においては、圧延材の温度管理が重要であり、圧延ライン上に設置された温度センサによって、圧延材の温度が測定される。温度センサは、たとえば、粗圧延機入側温度 (RET)、粗圧延機出側温度 (R1DT, R2DT)、仕上圧延機入側温度 (FET)、仕上圧延機出側温度 (FDT) を測定するために、それぞれの場所に設置される。これらのデータは、圧延の各工程で、圧延材全長にわたり収集される。

仕上圧延においては、製品品質を保つために、FDTを目標値に制御することが要求され、FDT測定値を用いたフィードバック制御や、FDT予測値を用いたフィードフォワード制御が行われる。熱間圧延において、圧延材

が目標温度を達成するために実施される制御手法は主に二つである。一つは圧延速度の調整であり、これは圧延材が目標温度より高温になると予測される場合に、圧延速度を遅くして空気との接触時間を延ばし、流出する熱量を増やして温度を下げる。もう一つは、圧延スタンド間の冷却スプレーの水量調整であり、冷却スプレーの水量を増減させることで圧延材の冷却度合いを調整し、温度を制御する。

フィードフォワード制御では、モデルによる温度予測精度が、制御精度に大きな影響を及ぼすため、温度モデルの精度を向上させる取り組みが、従来より行われてきた。圧延プロセスは多くの変数に影響され、その関係性は線形および非線形と複雑多岐にわたるため [1]、多層パーセプトロン (Multilayer Perceptron, MLP) などの機械学習技術が用いられてきた [2]。機械学習技術を用いたフィードフォワード制御では、圧延プロセスの各段階で取得されるデータに基づいてモデルの FDT 予測値を更新することで、製造プロセス中の変化を事前に予測し、適切な制御を行うことができると期待される。

しかし、従来の研究では主に静的特徴量を用いた予測モデルが使用されてきた [3, 4]。静的特徴量とは、圧延プロセスの開始以前の段階で収集されるデータや人工的に作成された特徴量を指す。したがって、静的特徴量に基づく予測手法では、圧延プロセス中の変動をリアルタイムで反映することができない。圧延プロセス中の変動を反映した予測を行うには、各段階に設置された温度センサが重要な役割を果たす。これらの温度センサは、それぞれ独立に圧延材の先端から尾端に至るまでの温度を計測し、個別の時系列データを形成している。各センサから取得される時系列データによって、圧延材の状態変化を詳細に把握することができる。これらの時系列データを活用することで、圧延プロセス中の外的要因による温度変化を反映した精緻な予測が可能となる。

以上のように、各センサから取得されるデータは時系列データであるため、これを活用するには時系列予測手法を適用する必要がある。しかし、従来の時系列予測手法では、圧延プロセスの進行に伴いデータを逐次収集し、全データが得られた後に最終段階で予測を行うため、製造中の各段階で途中出力を確認することができない。各工程後に逐次予測値を確認できれば、それらを比較することで、問題の発生原因となった工程を迅速に特定し、適切な対策を講じることができる。

本研究では、これらの課題を克服し、圧延プロセス全体を通じた高精度な時系列予測モデルを構築することを目指す。このモデルでは、まず圧延前に得られる圧延材の初期状態や設備の設定値などの静的変数のみから圧延プロセスを考慮しない最初の FDT 予測を行う。次に、圧延工程が進み、最初の温度センサから時系列データが取得されると、そのデータを考慮して予測を更新する。同様にして圧延材が各センサを通過するたびに温度データが収集され、その都度予測モデルにフィードバックされ

*1 早稲田大学

Waseda University, Tokyo, Japan.

*2 株式会社 TMEIC

TMEIC Corporation, Tokyo, Japan.

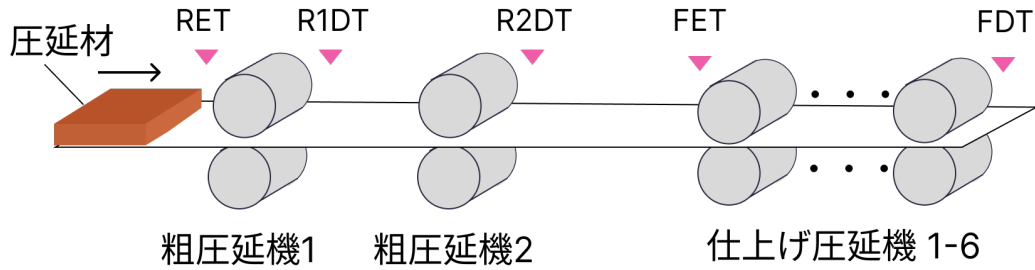


図1 圧延工程における温度センサの位置関係のイメージ図

る。これにより、予測値は逐次更新され、圧延プロセス全体を通じて最新のデータを反映した高精度な FDT 予測が可能となる。上記を実現するために、段階的デコーダを取り入れた新しい Transformer モデルを提案する。段階的デコーダは各センサからのデータを順次入力として取り込み、各段階で圧延材の隠れ状態を再計算する。この手法によって、各工程において最新の情報を考慮した FDT 予測が可能となる。

実験においては、本研究で提案したモデルと従来のベースラインモデルを比較した。ベースラインモデルとしては、静的特徴量に基づく予測モデルとして MLP、従来の時系列予測手法として多次元時系列予測 Transformer である Autoformer[5]、Crossformer[6] や線形モデルの LTSF-Linear[7] を用いた。これらのモデルと比較することで、提案モデルの有効性を検証した。実験結果より、提案モデルがベースラインモデルに比べて予測精度に優れていることを明らかにした。

2 関連研究

2.1 熱間圧延における機械学習の適用

熱間圧延において高品質を実現するためには、圧延プロセス前およびプロセス中に品質予測を行い、それに基づいてフィードバック制御およびフィードフォワード制御を行うことが重要である。特に、制御に用いる情報を機械学習を用いて予測する手法が多く提案されている [8]。既存の研究では、主に圧延装置の設定や圧延材の初期状態など、圧延前に得られる静的変数を用いて製品品質を予測する手法が提案されてきた [9, 3]。

Jiang らは、GAN と MLP を用いて、圧延装置の状態などの熱間圧延情報から品質指標の一つである板幅方向板厚差 (クラウン) を予測するモデルを提案している [9]。Hwang らは、鋼種や初期温度、設備仕様などの特徴量に加え、ドメイン知識を活用して作成した特徴量を用いて、XGBoost や MLP による温度予測を行っている [3]。しかし、これらの予測は圧延材の先端、中央、尾端の 3 点の代表的な予測にとどまり、時系列性を考慮していない。Zhang らは、目標板厚や鋼種などの静的変数および圧延プロセス中に測定された代表点から、粗圧延から仕上げ圧延までに必要な圧延材の冷却時間を予測する MLP を提案しているが、圧延プロセス中の時系列センサデータを考慮していない [4]。

これらの既存研究は、静的変数 n 個を $Z = \{Z_1, \dots, Z_n\}$ 、目的変数を \mathcal{Y} とした場合、 Z から \mathcal{Y} を予測する手法であると考えられる。しかし、同じ圧延材や圧延パラメータであっても、圧延工程における外的要因により仕上がり品質にはばらつきが生じるため、説明変数として、静的変数だけでなく圧延プロセス中に得られる時系列データを活用することが重要である。これに対

し本論文では、時系列データを $\mathcal{X} = (X_1^t, \dots, X_C^t)_{t=1}^L$ としたとき、 \mathcal{X}, Z から \mathcal{Y} を予測する手法を提案する。ここで、 C はセンサの数、 L は時系列データの長さを表す。具体的には、まず静的変数を MLP によって埋め込み表現に変換し、次に時系列データを用いて埋め込み表現を逐次更新する。この方法により、外的要因などによって変化した状態を反映した予測を行うことができる。

2.2 様々な時系列予測手法

時系列予測は多様な分野で用いられ、多くの手法が提案されてきた。統計的手法の代表例として、AutoRegressive Integrated Moving Average (ARIMA)[10] が挙げられる。ARIMA は、過去の値の線形結合と誤差項の和を用いて将来の値を予測する時系列モデルであり、短期的な株価予測など多岐にわたる応用がある [10]。しかし、本研究で扱う温度データは時系列が長く、複数ステップにわたる予測が必要であるため、ARIMA のような 1 ステップ予測を繰り返す手法では誤差が蓄積する問題が生じる。このため、本研究では複数ステップを直接予測する手法が適している。

深層学習を用いた時系列予測手法として、Long Short Term Memory (LSTM) [11]、Temporal Convolutional Network (TCN) [12]、および Transformer ベースの手法が挙げられる。Recurrent Neural Network (RNN) ベースの手法は、勾配消失や勾配爆発の問題があり学習が困難であることが知られている [13]。一方、Transformer ベースの時系列予測手法は、複数ステップの予測を直接行うことができ、時間的な距離に関係なく全ての時刻の情報にアクセス可能であるため、長期的な時間的依存関係の抽出に適している [14]。圧延における温度データは時系列が長いので、Transformer が有効である。

近年、Transformer は自然言語処理 [15] やコンピュータビジョン [16] などの系列データ処理分野で高い性能を示している。Autoformer は、既存の Transformer ベースの手法における自己注意機構が点ごとの表現を用いていることに着目し、部分時系列レベルでの依存関係のモデリングを自己相関機構を用いて実現している。Crossformer は、Autoformer と同様に部分時系列レベルでの依存関係を抽出するために、Dimension-Segment-Wise Embedding と Hierarchical Encoder を活用している。Long Time Series Forecasting-Linear (LTSF-Linear)[7] は、単層の非常にシンプルな線形モデルであり、データの周期性やトレンドが明確な場合に、既存の Transformer ベースの手法と比べて少ないパラメータ数で時間的依存関係を抽出できる。Electricity Transformer Temperature[17] や Exchange rate[18] などのいくつかのデータセットでは、Transformer ベースの手法より高精度を達成することが示されている。

これらの手法の説明変数は、多次元時系列データで

あり、目的変数はその続きとなる時系列である。本研究において時系列予測手法を適用する場合、予測時系列の長さを N とすると、 $\{(X_1^t, \dots, X_C^t)\}_{t=1}^L$ から $\{(X_1^t, \dots, X_C^t)\}_{t=L+1}^{L+N}$ を予測することになる。ただし、 C はセンサの数、 L は時系列データの長さとする。この場合、以下の 3 つの問題点がある。1 つ目は、2.1 節で述べた、予測に有効であると多くの研究で実験的に示されている静的変数を活用することができないことである。2 つ目は、本研究のタスクと異なることである。上記の時系列予測手法では、各センサの時系列の続きを出力するのに対して、本研究のタスクは説明変数とは別の時系列である FDT の時系列 $\mathcal{Y} = \{Y^t\}_{t=1}^N$ を予測することである。この問題に関しては、出力を線形変換して FDT の時系列とすることで、既存の時系列予測手法を適用することが可能である。5 節ではベースラインモデルとして時系列予測手法を用いるが、このように処理することで本研究の対象のタスクに適用している。3 つ目はモデルの入力が $\{(X_1^t, \dots, X_C^t)\}_{t=1}^L$ であり、 C 個のセンサを全て通過し、データが収集された後のみ予測が可能である点である。さらに、これらの手法は、予測を 1 度だけ出力する仕様であるため、各工程の後に予測を確認できない。また、複数のセンサーを扱う際に、多次元時系列として並列に扱うため、センサ間の順序関係が無視されてしまう。本研究の提案手法ではこれらの問題を解決するために、静的変数の埋め込みと段階的デコーダを取り入れている。詳細については 3 節で述べる。

3 提案手法

本節では、提案手法について説明する。まず、3.1 節では、提案手法の構造や、その意図について述べる。次に、3.2 節では、一般的な時系列予測と本研究で対象としている予測の違いについて確認する。続く 3.3 節では時系列データだけでなく、静的変数を活用するために静的変数の埋め込みを提案する。3.4 節では、予測結果を逐次的に確認できる構造にするために、段階的デコーダを提案する。

3.1 提案手法の構造

品質予測に用いるデータは、圧延工程前または圧延工程中に取得される。既存研究 [3, 4, 9] では、主に圧延工程前に得られる静的変数を用いて圧延材の温度などの品質予測を行っている。しかし、同じ圧延材や圧延パラメータでも、圧延中の様々な外的要因により仕上出側における品質にはばらつきが生じるため、時系列データを用いて圧延プロセス中の変化を反映した予測が重要である。1 節で述べたように、圧延工程の各段階には温度センサが設置されており、圧延材の温度を測定している。この各段階で取得されたセンサデータを基に、予測結果を逐次更新していくことが重要である。

予測結果を逐次更新する構造には、以下の三つの利点がある。第一に、予測結果の解釈性の向上である。温度異常が発生した際に、予測結果の推移を確認することで、どの工程で異常の原因が発生したかを特定しやすくなる。第二に、柔軟性である。一度のみ予測を出力するモデルでは、圧延の各工程の前後で予測結果を確認できず、すべてのセンサデータが取得された後にしか確認できない。そのため、すべてのセンサデータが取得される前には予測が得られず、フィードフォワード制御などの調整につなげることができない。第三に、予測精度の向上である。複数のセンサの時系列データを扱う際、既存の手法では多次元時系列データとして並列に扱われるため、各センサの順序関係が無視され、モデルが誤った依存関係を学

習する可能性がある。予測結果を逐次更新する仕組みにより、モデルに実際に取得されるデータの順序と同じ順序でデータを入力することができる。

提案手法のモデルの構造を図 2 に示す。まず、静的変数を埋め込み表現に変換する。詳細な説明は 3.3 節で述べる。次に、得られた埋め込み表現をエンコードする。エンコーダでは、マルチヘッドセルフアテンションとフィードフォワードネットワークによって埋め込み表現をエンコードする。エンコードされた埋め込み表現を線形変換することで暫定的な予測結果を得る。次に段階的デコーダによって、各センサデータに基づいた隠れ状態の更新と、逐次的な予測結果の出力を行う。提案手法の詳細を次節にて述べる。

3.2 定式化

C を変数の数、 L を入力時系列の長さ、 N を予測時系列の長さとするとき時系列予測は、 $\mathcal{X} = \{(X_1^t, \dots, X_C^t)\}_{t=1}^L$ から $\hat{\mathcal{X}} = \{(\hat{X}_1^t, \dots, \hat{X}_C^t)\}_{t=L+1}^{L+N}$ を予測する問題として定式化できる [7]。ただし、 X_i^t は t 番目の時間ステップにおける i 番目の変数の値である。本研究では、予測する時系列が、入力時系列の続きではなく、別時系列である。さらに、予測には時系列データだけではなく、圧延装置の設定値や、圧延材の初期状態、目標仕様などの静的変数を用いる。そのため本研究における最終的な時系列予測は $\{\mathcal{X}, \mathcal{Z}\} = \{(\{X_1^t, \dots, X_C^t\}_{t=1}^L, \{Z_1, \dots, Z_n\})\}$ から $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を予測する問題として定式化できる。ただし、 $\mathcal{Z} = \{Z_1, \dots, Z_n\}$ は静的変数を表し、静的変数は全てで n 種類あるとする。圧延において、センサデータの取得順が $\mathcal{X}_1 = \{X_1^t\}_{t=1}^L, \dots, \mathcal{X}_C = \{X_C^t\}_{t=1}^L$ であるとする。圧延の各工程の後に予測値を出力するために、圧延開始時には、 \mathcal{Z} から $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を出力し、1 つ目のセンサのデータ $\mathcal{X}_1 = \{X_1^t\}_{t=1}^L$ が取得された後に、 $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を更新する。つまり、 $\{\mathcal{X}_1, \mathcal{Z}\}$ から $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を出力する。同様に新しいデータが得られるたびに予測値 $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を更新し、最終的には、 $\{\mathcal{X}, \mathcal{Z}\}$ から $\hat{\mathcal{Y}} = \{Y^t\}_{t=1}^N$ を出力する。

時系列予測における予測の仕方は 2 つに分けられる。1 つ目は 1 ステップ予測を繰り返す方法 (iterated multi-step forecasting: IMS forecasting) である。2 つ目は直接複数ステップ予測する方法 (direct multi-step forecasting: DMS forecasting) である。 $N > 1$ の場合、IMS 予測では、1 ステップ予測を繰り返していく必要があり、 N が大きい場合に誤差の蓄積が問題となる。そのため、本研究で提案する手法は DMS 予測を採用している。DMS 予測を採用すると、実用上は各工程のセンサデータを取得し終わった後に、仕上出側のセンサデータを予測することになる。この予測の後、予測結果を受けて、仕上圧延中に 2.1 節で述べたような圧延速度やスタンド間冷却スプレーの調整による温度の制御が可能であるため、DMS 予測を用いることに問題はない。

3.3 静的変数の埋め込み

本節では、静的変数を活用するための手法について述べる。Crossformer [19] では、多次元時系列を時系列ごとに長さ L_{seg} のセグメントに分割した後に埋め込み表現を獲得している。

$$\mathbf{x}_{1:N} = \left\{ \mathbf{x}_{i,c}^{(s)} \mid 1 \leq i \leq \frac{N}{L_{seg}}, 1 \leq c \leq C \right\} \quad (1)$$

$$\mathbf{x}_{i,c}^{(s)} = \{x_{t,c} \mid (i-1) \times L_{seg} < t \leq i \times L_{seg}\} \quad (2)$$

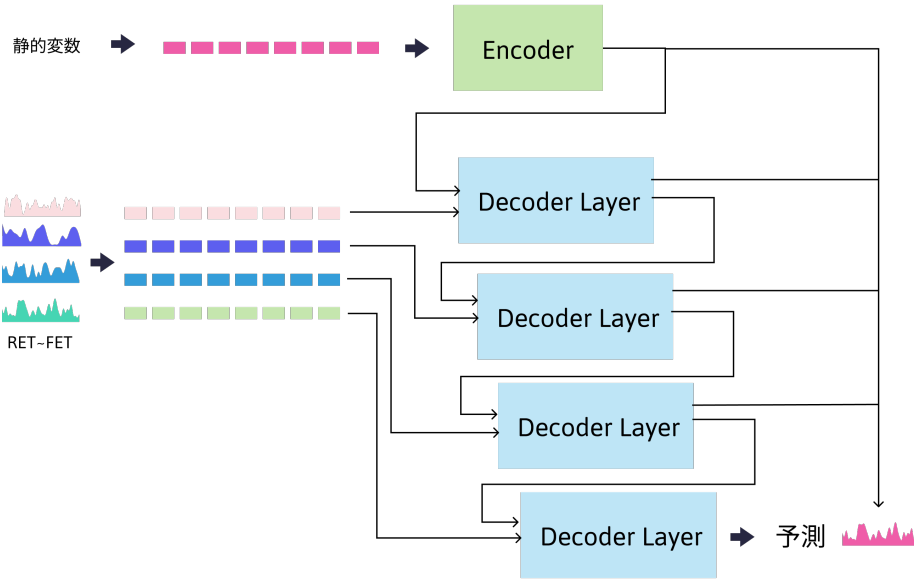


図 2 提案手法の構造

ただし, $\mathbf{x}_{i,c}^{(s)} \in \mathbb{R}^{L_{seg}}$ は c 番目のセンサにおける i 番目のセグメントを表す. N は時系列の長さであり, C はセンサの数を表す. 本研究においても, 時系列データをセグメント単位で扱う.

過去の研究から, MLP を用いて静的変数のみから品質を予測できることが実験的に確認されている [3]. 5 節でも, MLP によって静的変数のみから高い精度で温度を予測することができた. つまり, 静的変数から温度予測に有効な埋め込み表現を作成する方法として MLP は適切であると考えられる. そこで, MLP を用いて, n 個の静的変数を並べたベクトル $Z \in \mathbb{R}^n$ を d_{model} 次元の埋め込み表現 $\mathbf{h}_i \in \mathbb{R}^{d_{model}}$ に変換する.

$$\mathbf{h}_i = \text{MLP}(Z) + \mathbf{E}_i^{(pos)} \quad (3)$$

ただし, MLP は多層パーセプトロンを表す. 本研究において多層パーセプトロンの隠れ層は 2 層としている. $\mathbf{E}_i^{(pos)}$ は i 番目のセグメントの学習可能な位置エンコーディングを表す. 位置エンコーディングは Transformer において重要な要素の 1 つであり, 埋め込み表現に位置情報を付与する目的で用いられる. $\text{MLP}(Z) \in \mathbb{R}^{d_{model}}$ は全てのセグメントにわたって共通であるが, 学習可能な位置エンコーディング $\mathbf{E}_i^{(pos)}$ を足すことで, \mathbf{h}_i は位置ごとに異なった埋め込み表現となる. (3) 式によって作成された N/L_{seg} 個のベクトルをエンコーダの入力として用いる.

3.4 段階的デコーダ

本節では, 各センサ通過後に逐次的に予測を確認できる段階的デコーダについて説明する. 段階的デコーダの利点は 2 つある. 1 つ目は実際に取得されるデータの順序と同じ順序でデータをモデルに入力することができる点である. 従来のデコーダでは, 多次元時系列を同時にデコーダに入力するため, 圧延工程のように, 時系列間に順序関係がある場合に, 順序関係が無視されて並列に

扱われてしまう. 段階的デコーダでは, 各時系列データの順序関係に従ってそれら进行处理することで, 時系列間の順序関係を考慮する. 2 つ目は各センサの通過後に暫定的な予測結果を確認できる点である. 段階的デコーダでは, それぞれのセンサデータを反映した隠れ状態を線形変換することで, 暫定的な予測とする. 最終的には, すべての段階における予測を足し合わせることで, 最終予測とする.

段階的デコーダの各レイヤについて説明する. まず, 一つ目のセンサの埋め込み表現を作成する. ここでは, 長さ L のデータを長さ L_{seg} のセグメントごとに分割し, セグメントを線形変換することでそれぞれのセグメントで d_{model} 次元の埋め込み表現を得る. その際に, セグメントごとに学習可能な位置エンコーディングをたし合わせることで, セグメントの順番の情報を反映する. 次に, センサデータの埋め込み表現を, センサごとに式 (4) のようにマルチヘッドセルフアテンションによって隠れ状態にマッピングする.

$$\tilde{\mathbf{X}}_c = \text{MultiHead}(X_c, X_c, X_c) \quad (4)$$

ただし, $X_c \in \mathbb{R}^{\frac{L}{L_{seg}} \times d_{model}}$ は c 番目のセンサの埋め込み表現である. また, MultiHead はマルチヘッドアテンション [15] を表す. 次に, 静的変数の埋め込み表現がエンコードされて生成された値をキー, バリュースとして, 式 (4) で獲得した隠れ状態をクエリとしてマルチヘッドアテンション機構に通す.

$$\tilde{\mathbf{Z}}_c = \text{MultiHead}(\tilde{\mathbf{X}}_c, \mathbf{Z}_{c-1}, \mathbf{Z}_{c-1}) \quad (5)$$

ここで, $\mathbf{Z}_0 \in \mathbb{R}^{\frac{L}{L_{seg}} \times d_{model}}$ は式 (3) によって生成した埋め込み表現をエンコーダに通したものである. 式 (5) によって c 番目のセンサデータを用いて, それまでの隠れ状態を更新する. 式 (5) によって \mathbf{Z}_0 から順に,

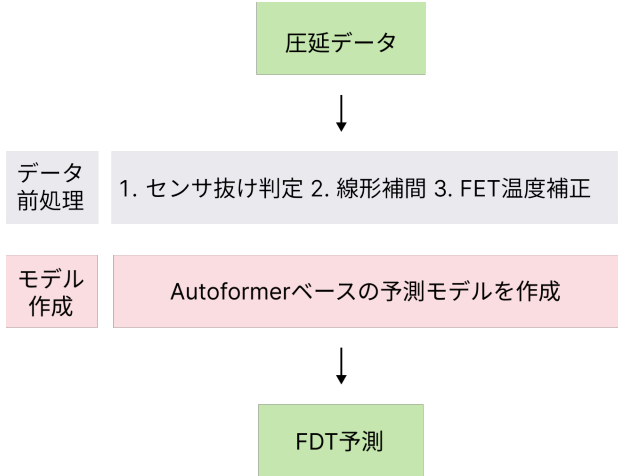


図 3 本研究の手順

$\mathbf{Z}_1, \dots, \mathbf{Z}_C$ と隠れ状態を更新していく。この段階的な更新プロセスにより、各時点のセンサデータが順序関係を保ちながら反映される。また、それぞれのデコーダレイヤは、上記のマルチヘッドアテンション機構に加えて、スキップ接続やフィードフォワードネットワークを含む。

$$\hat{\mathbf{Z}}_c = \text{LayerNorm}(\tilde{\mathbf{X}}_c + \tilde{\mathbf{Z}}_c) \quad (6)$$

$$\mathbf{Z}_c = \text{LayerNorm}(\hat{\mathbf{Z}}_c + \text{FFN}(\hat{\mathbf{Z}}_c)) \quad (7)$$

式 (7) で各センサ通過後に計算できる \mathbf{Z}_c を線形変換することで、途中出力とし、それらを足し合わせる。最終的に、全ての途中出力を足し合わせたものを最終予測とする。式 (7) で用いたフィードフォワードネットワークは以下のように表され、活性化関数として Rectified Linear Unit (ReLU) を用いている。

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (8)$$

$W_1 \in \mathbb{R}^{d_{model} \times d_{ff}}, W_2 \in \mathbb{R}^{d_{ff} \times d_{model}}$ は重み行列、 $b_1 \in \mathbb{R}^{d_{ff}}, b_2 \in \mathbb{R}^{d_{model}}$ はバイアス項である。中間層の次元数は d_{ff} であり、入力ベクトルはまず、 $W_1 \in \mathbb{R}^{d_{model} \times d_{ff}}$ によって、 d_{ff} 次元の中間表現に変換され、再び d_{model} 次元の出力ベクトルに変換される。提案手法では、 $d_{model} = 64, d_{ff} = 128$ としている。

4 データ

4.1 データの概要

過去の圧延で得られた生センサデータに前処理を加えて実験を行う。データは静的変数と時系列データの 2 種類に分類される。静的変数には、目標寸法仕様などの初期情報や圧延の設定のパラメータが含まれる。時系列データは RET, R1DT, R2DT, FET, FDT の 5 種類のデータが含まれ、それぞれのセンサにおいて圧延材が検知される間に一定間隔 (0.1 秒) で測定が行われている。本研究の目的は静的変数に加えて RET, R1DT, R2DT, FET のデータから FDT を予測することである。しかし、生センサデータはデータの長さにはばらつきがあったり、不要な部分を含んでいたりするため、そのままではモデルの入力として適していない。そこで、生

センサデータに前処理を施し、データの質を高めることでモデルの予測精度の向上を図る。前処理の手順を図 3 に示す。前処理の内容については次節で詳述する。

4.2 データの前処理

本節では生センサデータをモデルに入力できる形にするために時系列データに対して施した前処理について説明する。具体的には以下の 3 つの処理を行った。

1. センサ抜け判定
2. 線形補間
3. FET 温度補正

センサは基本的に圧延材を検知している間のみ測定を行うが、一部のデータには、圧延材がセンサを通過した後も測定を続けているものや、センサに入る前に測定を始めているものが含まれる。このようなデータをそのままモデルに入力すると、モデルが誤った時間的依存関係を捉えてしまう可能性がある。これを防ぐために、事前にセンサを通過した後のデータを削除した。センサを通過した後も測定を続けているデータの特徴として、測定温度が急激に変化していることが挙げられる。この特徴を基に、閾値ベースの簡易なセンサ抜け判定器を作成した。この判定器では一定の時間内 (0.3 秒) に 200°C 以上温度が変動した点を検知し、その点以降のデータをセンサ通過後のデータと判断して除去した。同様に、圧延材がセンサに入る前に測定を開始しているデータに対しても閾値ベースのセンサ入り前判定器を作成し、センサ入り前のデータを除去した。

次に、線形補間によるデータ長の調整を行った。圧延材は複数の圧延機を通過することで次第に長さが伸びるため、センサ間でセンサデータの長さが異なる。また、同じセンサのデータでも圧延材の長さなどに応じてデータの長さが異なる。時系列予測モデルを訓練する際には、入力と出力は固定長である必要があるため、データの長さを揃える必要がある。本実験では、時間的な値の変動を維持しながら長さを増やすことができる線形補間を用いて、各センサのデータの長さを 1,000 に統一した。

最後に、FET の温度データを補正した。本研究のデータを取得する際に使用しているセンサの位置は固定されており、圧延材がセンサを通過している間にデータを取得する仕組みになっている。そのため、圧延材の先端のデータ取得と尾端のデータ取得には数十秒の時間差がある。時間の経過によって圧延材の温度は低下するため、センサデータの測定値は前半から後半にかけて低下する傾向が見られる。この影響は特に FET の温度データにおいて顕著であり、その他のセンサデータでは影響が比較的小さい。モデルが温度低下のトレンドを誤って学習しないようにするため、あらかじめ線形回帰によって平均低下温度を計算し、元データに補正值を加えることで温度低下を補正した。

5 評価実験

本節では、4 節で説明したデータを用いて提案手法の評価実験を行った。全データを訓練データ、評価データ、テストデータの 3 つに分割した。全データの数は 6348 個であり、訓練データ、評価データ、テストデータをそれぞれ 3808 個、1270 個、1270 個に分割した。過学習を防ぐために、評価データを用いて Early Stopping を行い、最終的な評価指標の計算はテストデータを用いた。比較モデルとして、LTSF-Linear[7]、Autoformer[5]、Crossformer[19]、MLP、LTSF-Linear+MLP、Crossformer+MLP を訓練した。MLP は、圧延装置の設定

値や、目標仕様、圧延材の初期状態などを含む 51 種類の静的変数のデータを入力としている。LTSF-Linear や Autoformer, Crossformer は、静的変数を含まず、RET, R1DT, R2DT, FET のみを入力として、FDT を予測する。モデル + MLP としたものは、モデルと MLP を並列に繋げ、それぞれで予測結果を出力し、足し合わせたものを最終的な予測とする構造となっている。

実験結果を表 1 に示す。表 1 では、比較モデルと提案手法それぞれにおいて、平均二乗誤差 (MSE)、平均絶対誤差 (MAE)、二乗平均平方根誤差 (RMSE) を比較した。実験は 3 回行い、3 回の実験における評価指標の平均値と標準偏差を記載している。LTSF-Linear や Autoformer は圧延プロセス中の時系列センサデータを用いているにも関わらず、MLP を用いた静的変数のみからの予測よりも精度が低い。従来の Transformer ベースの手法の中では Crossformer が最も高い精度で予測できており、MLP を用いた静的変数のみからの予測を上回っている。比較手法の中では、Crossformer + MLP の精度が最も高く、温度予測には、静的変数のみだけではなく、圧延工程中に得られる時系列センサデータを活用することが重要であることが分かる。提案手法は、比較手法の精度を上回っており、予測を逐次的に確認できる利点があるだけでなく、精度の面でも優れていることが分かる。

表 2 に提案手法の各工程後の段階的な出力における評価指標を示す。また、図 4 に各工程後の段階的な出力を示す。図 4 において Ground Truth は、FDT 測定値を示し、その他については、各段階後の予測値を示す。予測値や、測定値は、FDT 目標値を基準とした差分を示している。表 2 や図 4 から最終出力に近づくにつれて予測が実際の値に近づいていくことが分かる。しかし、図 4 のように、途中出力では、セグメントの長さに依存したピークが観測された。今後の研究として、途中出力におけるピークを抑制し、安定した出力が得られるように、損失関数などを工夫することが考えられる。

6 結論

本研究では、圧延の工程を考慮した Transformer を提案した。提案手法では、時系列データだけでなく、静的変数を活用するために、MLP と学習可能位置エンコーディングを用いた、埋め込みを提案した。従来の時系列予測手法ではできなかった、予測結果を逐次的に出力することを実現するために、段階的デコーダを考案した。段階的デコーダを用いることで、圧延の工程に合わせて、データにモデルを与え、さらに、各センサ通過後に予測値を更新する構造を実現した。過去の圧延データを用いた実験より、提案手法が、既存の Transformer ベースモデルや、MLP などのモデルと比べて高い精度で予測できることを確認した。

参考文献

- [1] Qingquan Xu, Jie Dong, Kaixiang Peng, and Xuyan Yang. A novel method of neural network model predictive control integrated process monitoring and applications to hot rolling process. *Expert Systems with Applications*, 237, 2024. Cited by: 1.
- [2] Fengxiang Gao, Changsong Wang, Yubao Zhang, and Xiao Chen. Application of variable-metric chaos optimization neural network in predicting slab surface temperature of the continuous casting. page 2296 – 2299, 2009. Cited by: 5.
- [3] Rakhoon Hwang, Hyeontae Jo, Kyung Seok Kim, and Hyung Ju Hwang. Hybrid model of mathematical and neural network formulations for rolling force and temperature prediction in hot rolling processes. *IEEE Access*, 8:153123–153133, 2020.
- [4] Zhuolun Zhang, Bailin Wang, Shuaipeng Yuan, Yiren Li, Jiahui Yu, Tieke Li, and Xiqing Wang. Deep learning-based prediction framework of temperature control time for wide-thick slab hot rolling production. *Expert Systems with Applications*, 227, 2023. Cited by: 0.
- [5] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. volume 27, page 22419 – 22430, 2021. Cited by: 282.
- [6] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2023.
- [7] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? volume 37, page 11121 – 11128, 2023.
- [8] Dong Chen, Rui Zhang, Zhenlei Li, Yunjie Li, and Guo Yuan. Temperature distribution prediction in control cooling process with recurrent neural network for variable-velocity hot rolling strips. *International Journal of Advanced Manufacturing Technology*, 120:7533–7546, 6 2022.
- [9] Tianru Jiang, Kai Zhao, Wei Zhao, and Zhimin Lv. A joint model for hot-rolled strip quality in advance prediction. *Ironmaking and Steelmaking*, 50(9):1224 – 1233, 2023. Cited by: 0.
- [10] Adebisi A. Ariyo, Adewumi O. Adewumi, and Charles K. Ayo. Stock price prediction using the arima model. In *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, pages 106–112, 2014.
- [11] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. page 95 – 104, 2018. Cited by: 849; All Open Access, Green Open Access.
- [12] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *CoRR*, abs/1803.01271, 2018.
- [13] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1310–1318, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.
- [14] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyu Zhou, Wenhua Chen, Yu-Xiang Wang, and

表 1 FDT 温度予測結果

	MSE	MAE	RMSE
LTSF-Linear	10.416 (0.031)	2.092 (0.009)	3.227 (0.005)
Autoformer	15.347 (0.518)	2.622 (0.049)	3.917 (0.066)
Crossformer	8.299 (0.125)	1.823 (0.019)	2.881 (0.022)
MLP	8.413 (0.019)	1.845 (0.008)	2.901 (0.003)
LTSF-Linear + MLP	7.654 (0.092)	1.740 (0.013)	2.766 (0.017)
Crossformer+MLP	7.205 (0.056)	1.700 (0.011)	2.684 (0.010)
提案手法	7.059 (0.141)	1.700 (0.015)	2.657 (0.027)

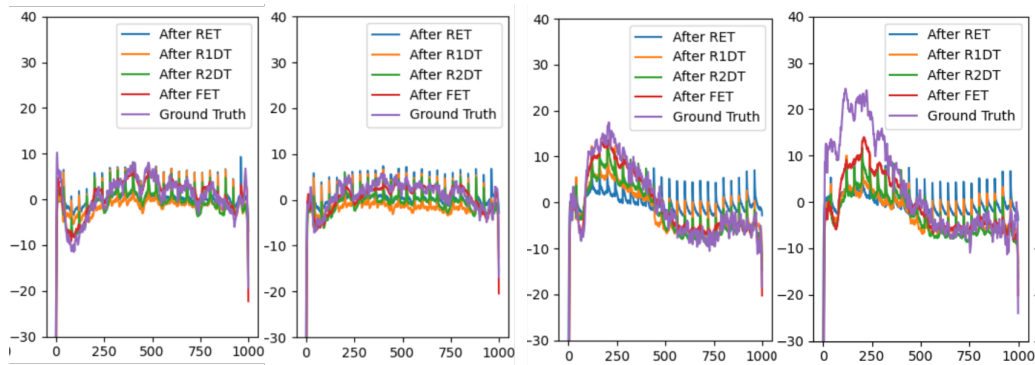


図 4 段階的予測の結果

表 2 FDT 温度予測結果

説明変数	MSE	MAE
RET まで	21.814	3.065
R1DT まで	14.660	2.599
R2DT まで	10.935	2.171
FET まで	7.066	1.700

Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

- [15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for

image recognition at scale. 2021. Cited by: 3780.

- [17] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. volume 12B, page 11106 – 11115, 2021. Cited by: 911.
- [18] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. SIGIR '18, page 95–104, New York, NY, USA, 2018. Association for Computing Machinery.
- [19] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2023.