

GTFS Realtime 形式を用いた 遅延予測情報を含む動的な定期船データの提供

東中村華帆¹ 浦上美佐子¹

概要：近年、携帯端末の普及により、複数の陸上公共交通機関の動的データが1つのサービスから検索される機会が増え、特に乗り物の現在位置や遅延情報に対する関心や需要が高まっている。しかし、国内の海上公共交通機関である定期船に着目すると、定期船の動的データのオープンデータ化や陸上公共機関との共有事例は少ない。そのため、海上公共交通機関の利用者が知りたいリアルタイムな情報を提供できていない現状がある。本研究では、遅延情報の予測を推定所要時間の導出として問題を捉え、国際的に普及が進んでいる GTFSRealtime 形式に基づき、瀬戸内海西部を航行する定期船の最新の動的な情報を一定間隔で提供することを目標とした。特に、遅延が予測される経由港を持つ航路に着目する。港から船の現在地までの直線距離と速度のみで求めた推定所要時間を比較基準とし、近年陸上交通機関での遅延予測に使用される複数の機械学習の手法を用いてより正確な推定所要時間を導出する方法を検証した。次に、検討結果を踏まえて GTFSRealtime 形式のデータを定期的に作成・公開するシステムを設計し、これを実装した。

キーワード：GTFS Realtime, 機械学習, 回帰モデル, 推定所要時間, 遅延予測, 公共交通機関連携, 定期船

1. はじめに

近年、一人一人の需要に合わせて複数の公共交通機関等に対して最適な検索・予約・決済等を一括で行えるサービス、通称 MaaS (Mobility-as-a-Service) の構築が官民連携で進められている。これに向けた取り組みの1つに、時刻表などの交通機関の運行情報のオープンデータ化及び整備・連携が挙げられる[1]。特に、各交通機関の円滑な連携のためには正確な遅延情報や乗り物の位置情報等のリアルタイムな共有が必要となる。実際に、陸上の公共交通機関である鉄道・路面電車・バスの利用者に対する地域公共交通サービスの評価に関する調査研究では、利用者の関心の高い評価項目として、運航時刻の正確さ、タクシーやバス同士の乗り継ぎが挙げられていた[2]。国内では、国際的に採用されている交通機関の動的データを表現する形式である GTFS(General Transit Feed Specification)Realtime [3]を軸とした動的データ共有の動きが広がっている。しかし、その多くは陸上交通機関の事業者によるものであり、海上交通機関の事業者による動的データ共有事例は現状無い[4]。また、定期船の現在位置に基づいて導出する次の港までの所要時間や到着時刻などの遅延情報の予測方法の検討、及びこれら遅延情報の共有も行われていない。通常、定期船の運行は時刻表から大きく外れることはないが、到着時刻の±5分程度の誤差範囲での運航はありうる[5]。この時刻表と実際の到着時刻との予測される誤差の情報は提示されている方が、交通機関の利用者にとっては望ましいと言える。

予測方法の検討に当たっては、2つの港間のみを航行する航路の場合、その航路が比較的直線に近いことが多いため、次の停泊港から船の現在地までの直線距離と航路内での速度の最頻値のみで十分な遅延予測を行うことができる

と考えられる。しかし、経由港を持つ航路の場合、経由のために島の外周をなぞるような航路となることが多く、直線距離では正確な距離が求められず、十分な遅延予測結果が得られない場合がある。さらに、先に述べた2つの港間だけの航路よりも速度の変化が生じやすく、従来の方法では±5分程度の誤差を解消することには繋がりにくい。

以上のことから、本稿では、瀬戸内海西部を航行する片道所要時間が30分以内の定期船を対象とし、航行時の船の位置情報に基づく遅延予測の検討と、動的データの提供方法の検討を行う。まず、短距離の3航路の内経由港を持つ1航路において、近年路線バス等の遅延予測において使用される機械学習の手法を用いることにより、単純な距離計算である従来の方法と比較しての遅延予測の精度の向上を試みた。機械学習を行う上では、手法の種類、学習データ量、及び停泊時間をデータ内に含めるか否かの点から予測モデルの良し悪しを検討した。また、経由港を持つ1航路と2つの港間のみを航行する2航路を合わせた3航路を航行する定期船の動的データを加工し、予想した次の港への到着時刻や次の停泊港の情報を含めて GTFSRealtime で共有するためのシステム構築を試みた。

本稿の構成は以下の通りである。2章では公共交通機関での遅延予測における機械学習の活用事例や GTFSRealtime を使用した国内での動的データ共有事例について述べる。3章では使用した3航路の動的データの詳細とその取得方法、及び次章以降で動的データを扱うため行った前処理について述べる。4章では経由港を持つ航路において遅延予測を行うモデルの作成、及び各予測モデルの検証結果を述べる。5章では3航路に対しての GTFSRealtime データを提供するシステムを4章で作成した予測モデルを内部に含めて作成し、一連の動作と動作に

¹ 徳山工業高等専門学校情報電子工学科
National Institute of Technology, Tokuyama College.

Department of Computer Science and Electronic Engineering,

かかる所要時間を確認した結果を述べる。最後に、6章で今後の展望を述べ、7章で本稿をまとめる。

2. 関連研究

本章では本研究の位置づけを明らかにする。まず、2.1節で交通機関の遅延予測に適用されている機械学習手法を取り上げ、海上の公共交通機関の遅延予測に適用する際の課題点を取り上げる。次に、2.2節では、既に陸上の公共交通機関でGTFSオープンデータを公開運用されている事例を取り上げ、海上の公共交通機関での適応可能性を述べる。

2.1 機械学習を使用した交通機関の遅延予測

文献[6]ではスペイン・トルコ間の長距離航路を対象とし、現在地から目的地に到着するまでの時間をニューラルネットワーク(NN)によって予測している。この際、NNの構造の違いに着目しており、最も良かったものはLSTM(Long Short-Term Memory)で、次点にグリッドサーチによりパラメータ調整を行った順伝播型ニューラルネットワークによる予測が良かったとされる。評価の際には、精度の差が僅かであることに対して、LSTMと順伝播型では順伝播型の構造が軽いことが示唆されている。機械学習とは異なるが、文献[7]ではAIS(Automatic Identification System)データ、LRIT(Long Range Identification and Tracking)データ、地形メッシュを使用してダイクストラ法を基に航路を正しく推定し、所要時間を予測する方法が提案されている。このように長距離航路の事例は多くあるものの、陸上公共交通機関との連携サービスを提供したい生活航路や観光航路となる片道所要時間が30分程度から1時間程度の短・中距離航路での事例は見当たらなかった。そこで、研究事例が比較的多い陸上の路線バスにおいて遅延予測を行った研究の調査を行った。

文献[8]は、バス停間を移動するためにかかった実際の所要時間と、時刻表上の所要時間との誤差を予測する上での外れ値検出方法や、予測モデル構築手法、学習データ量による差異を検証している。この予測モデル構築手法として、線形回帰、サポートベクトル回帰(SVR)、NN、ランダムフォレスト(RF)、XGBoostライブラリによる勾配ブースティング決定木(XGBoost)の5つの手法が用いられ、グリッドサーチによるパラメータ設定を行った状態で検証がなされた。その結果、5つの手法の中ではXGBoostの性能が全体的に良かったとされている。また、学習データ量による精度の違いではデータ量が2週間分以下と3週間分以上とで予測精度に有意な差が見られた。他にも、文献[9]は、12か所の停留所を持つバス1路線を対象として、便・停留所毎の到着時刻と運行ダイヤとの誤差を予測するモデルの構築方法を検討している。構築手法として、RF、エラスティックネット、SVRが選ばれている。その結果、2番目に性能が良かったSVRの指標値と大きく差をつけてRFの指

標値が最も良かったことを示している。文献[10]はNNに約1年間分の遅延時間の傾向等を学習させて遅延時間予測を行い、予測値的中率及び決定係数から評価を行っている。このNNは連続値を扱え、かつ、様々な要因から値が変動した場合も高い予測精度が維持できることが期待できる。推定した遅延時間と実際の遅延時間の誤差を±3分以内の範囲とした的中率では高い確率で予測ができていたものの、発生頻度が稀な状況に対しての予測には弱いことが示されている。

2.2 GTFSRealtimeデータの提供事例

路線バス等の陸上交通機関においてはGTFSRealtime及びこれを基として国土交通省が作成した動的バス情報フォーマット(GTFSリアルタイム)[11]による動的データ提供事例が増加している。このGTFSRealtime等で定められている形式では条件付き必須や任意の項目があり、各所から配信されるGTFSRealtimeに含まれる情報は異なる場合がある。文献[12]ではLPWA(Low Power Wide Area)を使用したIoTバスロケーションシステムにより収集した動的なバス情報をGTFSRealtimeにより整備を行い、30秒間隔での動的データ提供に取り組んでいる。このデータには時刻表と実際のバス停出発時刻を比較することで得られた遅延情報が含まれている。その他、自治体及び自治体と企業が連携しての取り組みが全国的にみられる[13][14]。

現在、国土交通省海事局において、GTFSに準拠した標準的なフェリー・旅客船航行情報フォーマットが公開され[15]、静的な航路情報の共有化が推し進められてきている一方で、動的バス情報フォーマットと同等の海上交通機関の動的データを共有するための形式は未だ作成されていない。しかし、これらと同等の形式で船の動的データ提供を行うことで、より利用者が船の動的データを確認し易い状況になると考えられる。よって、本稿では動的データを提供するための形式として、GTFSRealtimeを採用した。

2.3 本稿で扱う課題

本稿では海上の公共交通機関として、瀬戸内海西部の沿岸を航行する3隻の定期航路を対象とする。瀬戸内海では島に集落が点在していることから、1つの航路に複数の経由港を含む定期船航路が多く存在する。そのため、本稿で扱う3隻の定期船航路のうち、1つの航路は発着港の途中に経由港が存在する経由港を持つ定期船、残り2つの航路は発着港となる2港を往復する定期船を対象とした。

本稿で扱う課題は2つである。一つ目は、船の遅延予測を行う上での課題を取り上げ、検討することである。この課題では特に経由港を持つ航路における次の港までの所要時間を対象とし、機械学習による予測を行う上で最適な手法を検討する。検討する手法は、2.1節でも候補手法として取り上げられることの多かったSVR、RF、NN、そして、

文献[8]において全体的に優れた性能であった XGBoost による 4 手法での予測精度の比較を行う。さらに、経由港の有無での対応についても検討する。二つ目は、経由港を持つ 1 航路と経由港を持たない 2 航路それぞれに対応する GTFSRealtime データの作成を試み、データ取得からデータ提供までの一連の処理の確認を行う。

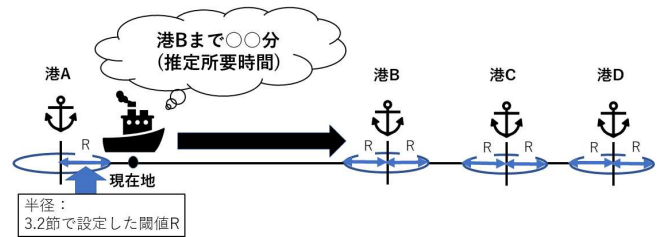


図 2 現在地からの「推定所要時間」の定義

3. 定期船データの概要

3.1 対象定期船から得られる動的データの概要

本稿で扱う定期船 3 隻の動的データを取得するシステムの概要を図 1 に示す[5][16]。このシステム構成は、定期船に設置した GPS センサから動的データとして、データ取得時間、データ取得時の緯度経度、対地速度 (knot)、対地方位 (度) を 10 秒間隔で取得する。これら取得した動的データ 6 件を 1 つのメールにまとめ、1 分間隔で携帯電話網を用いてクラウドメールサーバに送信し、蓄積される。サーバ上のプログラムによりメールから取得された動的データは、定期船識別コードを付与し、データベース (以下 DB) に自動格納するように設定している。なお、このシステム導入は、自治体が運航管理する定期船のリアルタイムな運航状況を確認する目的で導入されており、動的データをリアルタイムに活用する一方で、この DB には導入時から過去 7 年間程度のデータが蓄積されている。

本稿ではこの導入システムを利用し、機械学習の予測モデル構築に過去の蓄積データを用い、運航時の現在地での遅延予測にリアルタイムな運航データを用いることにした。

3.2 経由港の有無による予測方法の検討

1 章に述べたように、2 つの港間のみを航行する航路は比較的直線に近いことが多く、次の停泊港から船の現在地までの直線距離と航路内での速度の最頻値のみで十分な予測を行うことができる。しかし、経由港を持つために島の外周をなぞるような航路においては直線距離では正確な距離が求められず、十分な予測結果が得られない場合がある。さらに、上記 2 つの港間だけの航路よりも速度の変化が生じやすく、従来の方法では ± 5 分程度の誤差を解消することには繋がりにくいため、遅延予測を回帰問題として設定

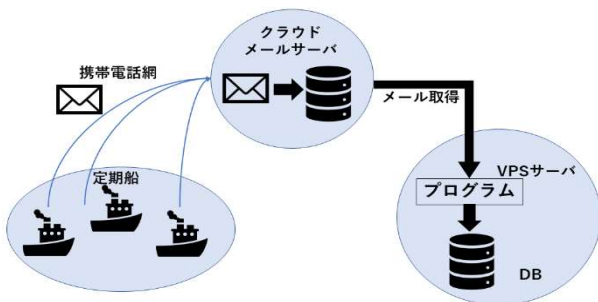


図 1 動的データ取得に使用したシステムの構成図

し、教師あり機械学習を検討した。

3.3 テストデータ及び学習データの作成

3.1 節で示した動的データには、今回予測対象とする「所要時間」の情報が存在しない。このため、各方法の予測結果の検証に使用するテストデータ、及び機械学習で使用する学習データの作成にあたっては、所要時間の情報を新たに組み込む必要がある。所要時間の例を図 2 に示す。図 2 内の直線は、往路として、港 A を出港し、港 B と港 C を経由して、港 D を終点とする航路を示す。この航路において、船が港 A から出港した後でありかつ港 B に着いていない場合、所要時間とは港 B まで後何分で到着するかを表す。また、所要時間を何らかの方法で予想した値である推定所要時間についても同様に定義する。過去のデータからこの所要時間の情報を整理する上で、本稿では定期船が次の港に到着した時点と、到着するまでの航行データの取得時刻から所要時間を求めた。停発着時での定期船の緯度経度は毎回同一ではないことから、本稿では港から一定の範囲に船が近づいた時を停泊したと見なすこととした。この一定の範囲を定める閾値を図 2 の R で示す。対象となる 3 航路のうち、経由港のない 2 航路は 80[m] に距離の閾値を設定することで停発着の正しい判断が可能であったが、経由港を持つ 1 航路は停発着の正しい判断ができなかった。この理由として、経由港の棧橋に乗船者がいないことが目視で確認できた場合、定期船が経由港に入港せずに、次の港へ行く場合が考えられた。そのため、経由港をもつ 1 航路の場合、この閾値を 250[m] に拡大した。また、経由港での停発着時の船の挙動は不規則である上に、所要時間の変化には影響しないと考えられた。そのため、データセット作成時には停発着時の船のデータと見なした閾値内を航行する除外する処理を加えた。

4. 推定所要時間導出方法の検討

本章では 3 章に示した方法により前処理を行ったデータを用い、経由港を持つ航路における従来手法による距離と速度から算出される予測結果 (4.1 節) と本稿で新たに検討する 4 種類の機械学習による予測モデル (回帰モデル) の導出方法による予測結果 (4.2 節) の評価を行う。なお、これら評価においては、3.3 節で示したテストデータを使用した。評価指標には、MAE (絶対平均誤差)、RMSE (平方

二乗平均誤差)を使用した。この2つの指標は、指標値が正解の値の単位と同じになり、直感的に分かりやすいことから、2.1節に示した先行研究の多くで使用されている。そのため、本稿でも採用している。式(1)、式(2)のそれぞれにおいて、 n がテストデータ数、 y_i が予測値、 \hat{y}_i が実測値を示す。これらは共に予測精度に使用される指標であり、値が0に近い程予測値と実測値の差が小さいことを示す。MAEと比較してRMSEはより誤差に反応し易い傾向があり、実測値と予測値が大きく外れる場合に値が大きくなる。

$$MAE = \frac{1}{n} \sum_{i=0}^{n-1} |y_i - \hat{y}_i| \quad (1)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2} \quad (2)$$

4.1 距離と速度による予測

従来手法での予測として、次の停泊港から船の現在地までの直線距離と速度のみで次の港までにかかると思われる所要時間(以下、図2に示す推定所要時間とする)の推定を行った。なお、直線距離の導出にあたっては、WGS84測地系に基づく2地点間の大円距離(Geopy2.2.0ライブラリ)を使用した。速度はテストデータ(表1)を基に航路全体での速度の最頻値を算出したものとする。その結果、MAEが0.638、RMSEが1.089となった。このMAEとRMSEの2つの値を4.2節以降で示す回帰モデルとで求めた所要時間の誤差の比較基準とする。図3は、横軸は運航時間、縦軸は距離と速度のみを用いた所要時間の予測値である推定所要時間(青色)と実測値(橙色)を示す。線が谷になっている部分が経由港で停泊判定が行われた時点、山の部分が出港した直後の時点を表す。つまり、1つの山が航行区間を示している。図3には3つの山が表れており、3つの航行区間での予測値と実測値の推移を確認することができた。この航路では3つ目の航行区間が最も運航距離が長い。そのため、3つ目の航行区間での速度の最頻値が、航路全体での速度として採用されている。本来、3つ目の航行区間(1番右の山)のように予測値と実測値が一致していることが望ましい状態である。しかし、航行区間ごとに速度が異なることから1つ目と2つ目の航行区間では予測値と実測値の所要時間の大きさに差が生じている。また、2つ目の航

表1 データセット毎のデータ件数
(経由港を持つ定期船のみ抜粋)

	動的データ件数 (3.1節, 未処理)	動的データ件数 (閾値R区間を除外)
テストデータ(2日分) 2021.9.21-22	4,002	1,733
学習データ(1週間分) 2021.9.13-20	13,215	6,763
学習データ(3週間分) 2021.8.13-30, 9.10-20	39,593	20,433

行区間においては地形の問題から直線距離と実際の航路上の距離が異なるため、値の変化の傾向が実測値と異なる。以上の課題から、速度と距離を用いて適切な所要時間の予測を行う方法は、限界があると言える。

4.2 回帰モデルによる予測

4.1節で提起した従来手法での課題を基に、本節では、最適な機械学習手法の検討、及びデータ量・前処理による予測性能の変化を調べる。機械学習の手法に関しては2.1節にあげた先行研究にて良い性能が見られたNN、RF、XGBoostを採用した。NNは、今回のように数字のみでは所要時間との相関が得られ辛い非線形なデータでも、ハイパーパラメータの調整次第で柔軟に対応できる特徴がある。NNには様々な種類があるが、今回は最も基本的な順伝播型ニューラルネットワークを用いる。RFは、異なる方法で生成した複数の決定木の多数決により予測結果を決定するため、過学習になりにくいとされる。XGBoostは、誤差を修正しながら一つの決定木を繰り返し作成し、学習を進める。これにより予測値と実測値の誤差を小さくでき、精度よく予測できるとされる。SVRは、2.1節で示した先行研究では良い性能を得られていないが、NNと同様の理由から採用した。また、手法の検討に加えて、扱うデータセットの比較も行った。表1に示す約1週間分の定期船から取得した動的データからなるデータセットと、約3週間分のデータセットの2種類を用いて、指標値を比較することで、学習データ量の違いによる精度向上が見込めるかについても着目する。各データセットには出港前の停泊時のデータが多く含まれたため、データセット内の速度分布を調査した。

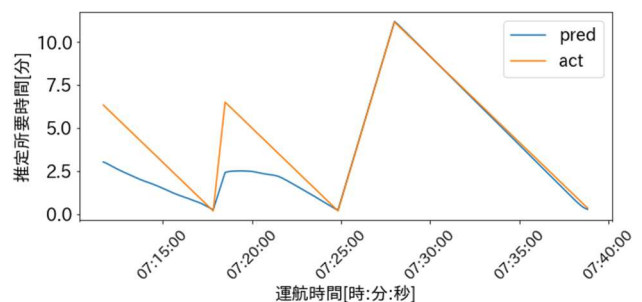


図3 距離と速度による推定所要時間

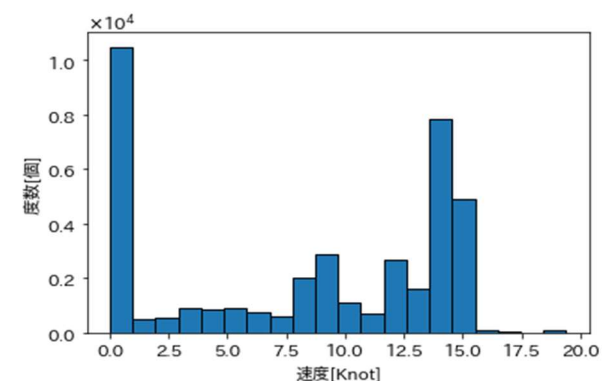


図4 表1の学習データ(3週間分)の速度分布

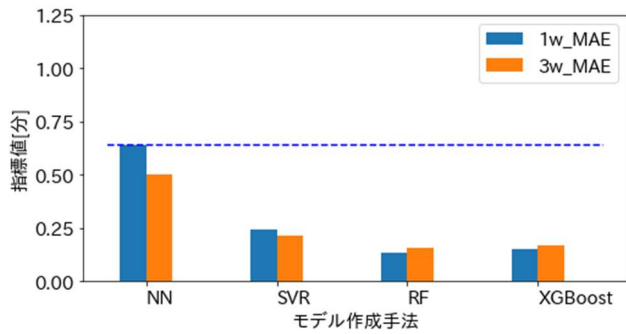


図 5 手法・学習データ量と指標値 (MAE) の比較

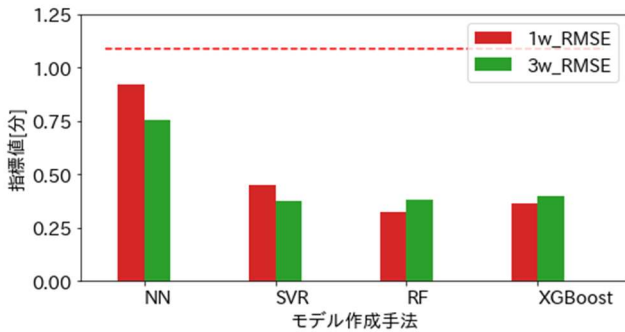


図 6 手法・学習データ量と指標値 (RMSE) の比較

表 1 のデータセットの内学習データ (3 週間分) の未処理データの速度分布を示す。図 4 より, 0[Knot]から 3[Knot]は, 3.2 節で示した図 2 の閾値 R 区間内の停泊中や接岸・離岸時のデータであり, 予測そのものへの影響は少ないと考えられたため, 除外処理を行った。その結果, 表 1 の動的データ件数 (3.1 節, 未処理) と動的データ件数 (閾値 R 区間の除外処理) に示す件数の推移を比較すると, 除外処理後ではデータ件数が半分になった。機械学習では, この除外処理後のデータセットを用いる。なお, 機械学習手法毎の予測モデル作成は, Python3.8.2 上で動作させた scikit-learn0.24.1[a]により行った。また, 各手法による予測モデルのハイパーパラメータは, 表 2 に示した範囲で, ランダムサーチを行うことにより設定した。この表 2 に記載されていないパラメータは, ライブラリのデフォルト値を使用した。各予測モデルのパラメータは, 試行回数 30 回でのランダムサーチの内, RMSE が最も低かったものを採用した。

4.2.1 モデル作成手法・学習データ量による精度の差

モデル作成手法・学習データ量による精度の差を図 5, 図 6 に示す。図 5 は, 指標値を MAE とし, 従来手法での指標値 (以下基準値と呼ぶ) を青破線で示した。図 6 は, 指標値を RMSE とし, 基準値を赤破線で示した。図 5, 図 6 とともに, 1 週間分 (1w) と 3 週間分 (3w) のデータセットによる比較も行った。はじめに, モデル作成手法の違いに着目する。図 5, 図 6 とともに, 各モデル作成手法の指標値が, 基準値である破線を下回っていることから, 機械学習を使用することによる予測精度の向上は概ね達成できたと考え

表 2 探索したハイパーパラメータの範囲

手法	パラメータ	範囲
SVR	C	0.01, 0.11, ..., 99.91
	gamma	scale (固定)
	epsilon	0, 0.1, ..., 0.9
NN	max_iter	2000 (固定)
	alpha	0.01, 0.11, ..., 99.91
	hidden_layer_sizes	1, 2, ..., 10
	activation	logistic, tanh, relu
	solver	lbfgs, sgd, adam
RF	n_estimators	1, 11, ..., 91
	min_samples_leaf	1, 2, ..., 10
	max_depth	1, 2, ..., 100
XGBoost	n_estimators	1, 11, ..., 91
	min_samples_leaf	1, 2, ..., 10
	max_depth	1, 2, ..., 100
	learning_rate	0.01, ..., 0.1

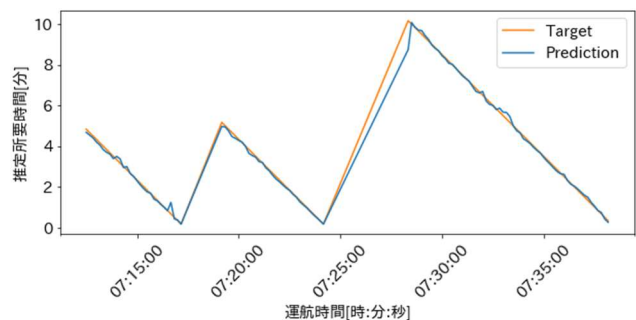


図 7 RF による予測値と実測値 (学習データ : 3 週間分)

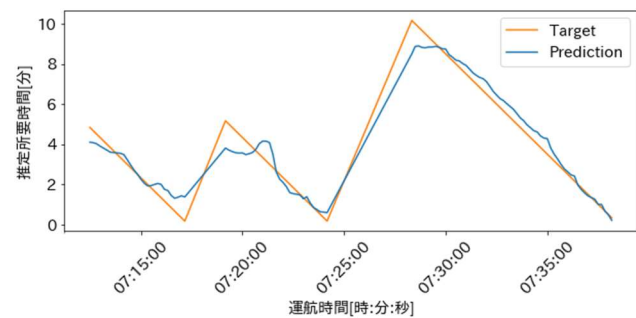


図 8 NN による予測値と実測値 (学習データ : 3 週間分)

られる。図 5, 図 6 とともに, RF が最も指標値が低いことから, 今回のパラメータ設定方法とデータでは, RF が最も予測に適した手法であったと言える。しかし, NN は基準値を下回ってはいるものの, 他の手法と比較すると指標値が高く, 悪い結果といえる。最も指標値の低かった RF と最も指標値の高かった NN による, 3 週間分のデータセットによる所要時間の予測値と実測値を示したグラフを図 7 と

a “scikit-learn”.<https://scikit-learn.org/stable/>, (参照 2022-2-8)

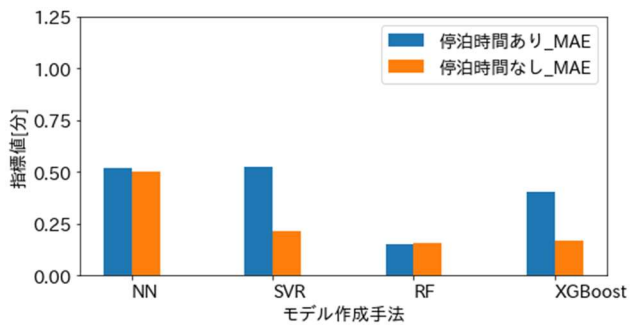


図 9 閾値 R 区間の除外 (有/無) と指標値 (MAE)

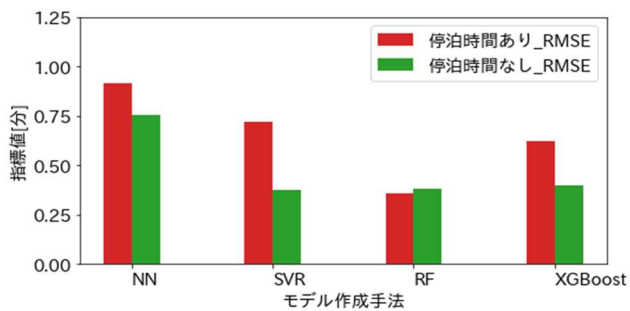


図 10 閾値 R 区間の除外 (有/無) と指標値 (RMSE)

図 8 に示す。縦軸は推定所要時間、横軸が運航時刻を示している。図 7 に示す RF による所要時間の推定値と実測値のグラフと比較すると、図 8 に示す NN による推定所要時間と実測値を示す線はうまく重なっておらず、予測ができていないと言える。次に、学習データ量による違いに着目する。図 5、図 6 より、NN、SVR は学習データ量の増加により指標値が下がり予測精度が向上したと言えることに対して、RF、XGBoost では指標値が若干上昇していた。このことから、RF、XGBoost では学習データの増加は、予測の改善に繋がらなかったと言える。

4.2.2 停泊時間データの有無による検証

4.2.1 より、もっとも予測精度が低かった NN では、閾値 R 区間近辺である出入港・停泊時に特に予測値と実測値の差が生じていた。この誤差の原因の 1 つに、停泊中のデータと見なす閾値 R 区間近辺の機械学習がうまくできていないことが考えられた。このため、検証用として、閾値 R 区間 (停泊時間) を除外せず、取得した全ての動的データを学習データとして学習させた予測モデルを作成し、再度予測の検証を行った。なお、同一の基準で評価するため、テストデータは閾値 R 区間を除外したもののままとした。評価の結果を図 9、図 10 に示す。図 9 にて橙、図 10 にて緑で示した停泊時間なし (閾値 R 区間航行時のデータを除外したもの) による予測は、図 5 にて橙、図 6 にて緑で示した 3w (3 週間分のデータセット) と同一のものであり、値は同じものを使用している。それぞれ停泊時間あり (閾値 R 区間航行時のデータを含めたもの) と比較すると、停泊時間なしの方が指標値は低く、停泊時間の追加はかえって予測精度の悪化につながったことが分かる。

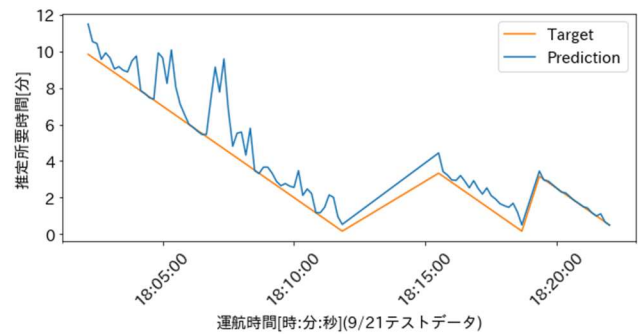


図 11 実測値と速度が通常よりも速い時の予測値 (閾値 R 除外無/3 週間分のデータセットで学習した RF 予測)

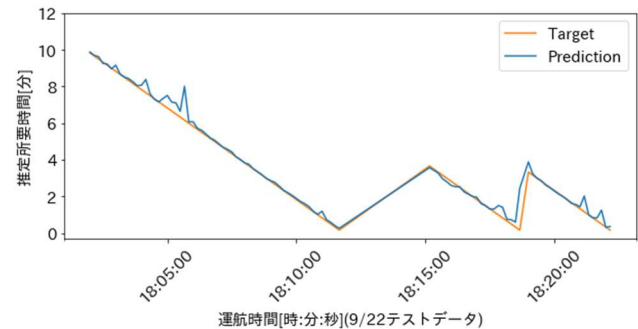


図 12 実測値と速度が通常時の予測値 (閾値 R 除外無/3 週間分のデータセットで学習した RF 予測)

4.3 考察

まず、4.2.1 の結果を考察する。テストデータの実測値に対する 3 週間分のデータで学習した時の RF によるモデルの予測値を確認したところ、図 11 に示すように一部予測値が大きく外れている箇所があった。テストデータを確認したところ、該当箇所では通常よりも速い速度で船が航行していたことを確認した。このことから、予測値と実測値のずれは、通常とは異なる運行をした場合に起きていると考えられる。実際に、通常で運行された翌日の同時刻では、図 12 に示す通り予測値と実測値に大きなずれがなく、予測が安定していると言える。

4.2.2 での全体的な指標値の悪化の原因には、停泊時間の追加による目的変数の複雑化が考えられる。ハイパーパラメータの設定によっては過学習し易い RF 以外の手法では、学習データに過剰に適合してしまい、停泊時間の無いテストデータに対する予測がうまく行えなかったと考えられる。また、RF でも停泊時間を追加したことによる精度の改善はほぼ無いことから、停泊時間の追加は予測精度の向上には寄与しないと言える。今回 4.2.1, 4.2.2 での予測精度悪化に繋がったモデルの過学習が起きた背景として、ハイパーパラメータの設定方法が適切でなかった可能性が高いと考えられる。今回は予測に当たって、ランダムサーチにより広範囲を探索することでハイパーパラメータを設定した。しかし、結果から、過学習を抑える方向で探索範囲を絞る必要があることが分かった。

5. GTFSRealtime を用いたデータ提供

5.1 提供するデータ

GTFSRealtime では動的データとして TripUpdate (ルート最新情報), VehiclePosition (車両位置情報), Alerts (運行情報) の3種類を定義している. この内, 本稿では TripUpdate と VehiclePosition の情報を自動更新の対象とする. 停留所等からの発着に関する情報を記載する TripUpdate 内の stop_time_update フィールドは, 経由する停留所を設定するフィールド(stop_sequence, stop_id), 及びその停留所に対応する到着時刻及び出発時刻を設定するフィールド(arrival, departure) を持つ. 本稿では stop_time_update が示す情報を次の停泊港に関する情報のみに絞り, 経由する停泊港を設定する両方のフィールド及び推定した未来の到着時刻を示す arrival フィールドのデータを整備することとした. また, 定期船からのデータに記載されたセンサデータ取得時刻から該当する便の推定を行い, これを提供データに含めた. timestamp については, 船での GPS センサデータ取得時刻に対応させた.

5.2 データ提供システムの概要

本研究で作成したデータ提供システムの概要を図 13 に示す. 本システムでは3章にて述べた既存システム内の DB から 1 分間隔で該当航路の最新データ 6 件を受け取り, GTFSRealtime データファイルの作成を行う. 経由港のある 1 航路に関しては, 4 章にて作成した 1 週間分のデータで学習した RF による予測モデルを使用して推定所要時間を導出する. 経由港の無い 2 航路に関しては速度がほぼ一定であることから, 予測モデルを使用せず, 距離を該当定期船の 0.5[Knot]未満を除く速度の最頻値で割る従来手法により推定所要時間を導出した. GTFSRealtime では推定所要時間の情報を UNIX 時間で記録する必要があるため, 求めた推定所要時間を秒数に変換し, 動的データの取得日時に追加して UNIX 時間に変換した. 次の停泊港を求める上で必要となる停発着の判定を行う際は 3 章で提示した方法により行い, 距離の閾値は経由港のある航路においては 250[m]未満, それ以外の航路では 80[m]未満を用いた. データは HTTP 通信可能な固定 URL にファイルを置くことで配信し, 利用者が HTTP リクエストを送信することで取得

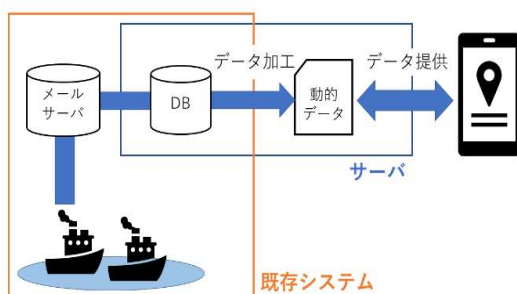


図 13 データ提供システムの概要

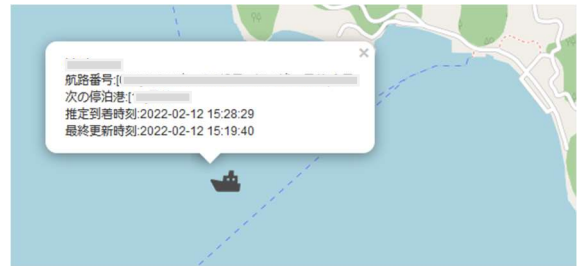


図 14 データ利用者による表示画面
(ページ更新時刻 同日15:23:31)

するものとした. また, DB からデータを取得し, 動的データに加工するまでを 30 秒以内で行えることを目標とした. これらを合わせて定期船のデータ送信から作成した GTFSRealtime データファイルの提供までを最大 1 分半程度で行うことを目標とした.

5.3 データ提供システムの動作検証結果

システムの動作確認は固定 URL にアクセスしてダウンロードした GTFSRealtime ファイルに記入された情報を地図にマーカーでプロットすることで行った. ファイル内には定期船の位置情報, 推定到着時刻, 該当航路の情報, 次の停泊港の情報が含まれている. 動作確認実行時の結果を図 14 に示す. 匿名性のため, 定期船を特定できる情報は隠している. 動作検証の結果, データ更新は 1 分おきに正常に行われていたが, GTFSRealtime に記載された最終更新時刻とページ更新時刻とで 3 分以上の遅延が生じていた.

5.4 データ表示の遅延原因への考察

現段階で考えられる遅延の原因は 2 つある. 1 つは各処理のタイミングが適切でないことである. データ処理部分において DB から渡されるデータは 1 分前に定期船から送信されたデータ 6 件になると想定していた. しかし, 実際には 2,3 分前に定期船から送信されたデータ 6 件が渡されていた. これは, 既存システム内でメールサーバにアクセスして DB にデータを格納し終わる前にデータ加工部分で DB にアクセスしていることが考えられる. この課題に関しては, データ格納時の処理とデータ加工処理が同時に行われないようにタイミングを調整することで対処は可能であると考えられる. 2 つめの理由としては, DB への問い合わせに約 35 秒を要していることが挙げられる. これは, DB に格納されているデータ件数が 800 万件以上であり, 処理に必要な最新データ 6 件の取得に時間を要するためである. 今回作成したシステムでは距離により停泊判定を行うことから最新データを漏れなく得る必要がある. この課題を解決するには, データに含まれる対地方位や速度から船の挙動を推定する方法を検討する必要がある. また, 並行して動的データを送信する間隔を見直すなど, 既存システムの改善を行うことで, 表示遅延はより短くできると考えている.

6. 今後の展望

今後は、既に整備済みである3航路のGTFSデータと合わせてオープンデータとしての提供を目指す。GTFSRealtimeと互換性があり、現在国内で普及しているGTFSリアルタイムではデータの更新間隔を30秒以下とすることが推奨されており、これに合わせたデータ更新を行うことでGTFSリアルタイムに対応済みの各交通機関との連携、及びオープンデータの二次利用が円滑に進むと考えられる。このために、まず、定期船が通常とは異なる運行をした場合でも安定した遅延状況の予測・データ提供ができるよう、ハイパーパラメータの調整等の予測モデルの改良を行う。また、5.4節で述べた課題を解決すべく、本稿で作成したデータ提供システムの調整を行う予定である。

7. まとめ

本稿ではMaaSに向けた取り組みの1つである、利用者にとって必要な動的データの共有を瀬戸内海西部の定期船3航路において行うため、GTFSRealtime形式を用いた遅延予測情報（推定到着時刻）を含む動的な定期船データの提供方法を検討した。また、遅延予測情報の提供にあたっては経由港を持つ複雑な航路にも対応すべく、近年路線バス等において取り入れられている機械学習による遅延予測を、手法や学習データによる指標値の変化を調べつつ試みた。その結果、機械学習の導入により従来手法よりも精度が改善されることを確認した。また、採用した手法の中でも、RFは全体的に精度が良かった。一方、今回のデータ前処理及びハイパーパラメータの設定方法による予測では、全体的に過学習の傾向があった。また、作成した予測モデルを一部航路の遅延予測に組み込んだGTFSRealtimeでの動的データ共有方法の検討、及びシステムの実装を行った。定期船からデータが送信されてから利用者にデータが表示されるまで遅延はあるが、システムの組み立て自体は行えたと言える。今後は6章で述べたように今回確認した課題を解決しつつ、3航路の動的データを提供するシステムの実用化に繋げていく。

謝辞 本研究の一部は、令和3年度徳山高専テクノ・アカデミア事業研究シーズブラッシュアップ課題「MaaS連携に向けて離島航路動的データの可視化を目指した機械学習手法の検討評価」の支援を受けています。

参考文献

- [1] “総務省 次世代の交通 MaaS “. https://www.soumu.go.jp/men_u_news/s-news/02tsushin02_04000045.html, (参照 2022-2-9)
- [2] 国土交通省 国土交通政策研究所. 地域公共交通サービスの評価に関する調査研究. 国土交通政策研究 第134号, 2016年9月. <https://www.mlit.go.jp/pri/houkoku/gaiyou/pdf/kkk134.pdf>,(参照 2022-2-1)

- [3] “Google Transit リアルタイム乗換案内データ “. <https://developers.google.com/transit/gtfs-realtime?hl=ja>,(参照 2022-1-18)
- [4] “GTFS・「標準的なバス情報フォーマット」オープンデータ一覧 “. <https://tshimada291.sakura.ne.jp/transport/gtfs-list.html> <https://tshimada291.sakura.ne.jp/transport/gtfs-list.html>, (参照 2022-2-8)
- [5] 金岡 拓哉, 浦上 美佐子, 本木 久也, 西下 葉津紀, 大内 清香, 松井 孝, 和田 雅昭. プローブシップデータに基づく定期船運航管理時の異常検知. マルチメディア通信と分散処理ワークショップ2015論文集, 2015, no5, p191-197.
- [6] Sara El Mekkaoui, Loubna Benabbou, and Abdelaziz Berrado. 2020. Predicting Ships Estimated Time of Arrival based on AIS Data . In 13th International Conference on Intelligent Systems: Theories and Applications (SITA'20), September 23–24, 2020, Rabat, Morocco. ACM, New York, NY, USA,
- [7] Alfredo Alessandrini, Fabio Mazzarella, and Michele Vespe. Estimated Time of Arrival Using Historical Vessel Tracking Data. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, Vol. 20, No. 1, JANUARY 2019.
- [8] Tsubasa Yamaguchi, Mansur A.S. and Tsunenori Mine. Prediction of Bus Delay over Intervals on Various Kinds of Routes Using Bus Probe Data. 2018 IEEE/ACM 5th International Conference on Big Data Computing Applications and Technologies (BDCAT)
- [9] 佐藤 孝大, 大場 みち子. バスの到着時刻予測モデルの開発と移動手段提案システムの検討. 研究報告情報システムと社会環境 (IS), 2018, no7, p1-7
- [10] 轟 朝幸, 川崎 智也, 野村 大智, 横関 敬裕. ニューラルネットワークを用いた路線バスの遅延時間予測. 交通工学論文集, 2017, 3 巻, 2 号, p. A_202-A_207
- [11] “はじめよう! 「標準的なバス情報フォーマット」 “. <https://www.gtfs.jp/get-started.html>, (参照 2022-2-14)
- [12] 竹内 未来, 保下 拓也, 田畑 俊貴, 鷗田 一博, 鈴木 秀和, 松本 幸正. GTFS Realtime を用いたくりんばすの遅延情報の提供. 第81回情報処理学会全国大会講演論文集, Vol.2019, No.1, 2Y-07, pp.303–304, 2019年3月.
- [13] “佐賀県におけるバス情報オープンデータ化の取組 “. <https://www.wtb.mlit.go.jp/kyushu/content/000090553.pdf>,(参照 2022-2-9)
- [14] “広島県バス協会 GTFS オープンデータサイト “. <https://www.bus-kyo.or.jp/gtfs-open-data>,(参照 2022-2-9)
- [15] “国土交通省 標準的なフェリー・旅客船航路情報フォーマット Ver.2 “. https://www.mlit.go.jp/maritime/maritime_tk3_000061.html,(参照 2022-2-9)
- [16] 和田雅昭, 畑中勝守, 上瀧實. ラビッドプロトタイプングのための汎用ボード"マイクローキューブ"の設計と活用, 情報処理学会研究報告, 2008-UBI-18 (6), 2008.