

Hiking-LATTE: 登山者の歩行能力を考慮した登山の到着時刻推定手法の提案

浅古 瑞穂¹ 田原康之² 大須賀昭彦³
清雄一⁴

登山は人気のアクティビティである一方、山岳遭難は深刻な問題となっている。登山の所要時間を正確に予測することは、適切な登山計画の策定や遭難防止に貢献する重要な課題である。従来の研究では、地形の斜度と歩行速度の関係を基に所要時間を推定する手法や、深層学習を活用した到着時刻推定手法が提案されてきた。しかし、登山者の歩行能力や疲労の影響を適切に考慮することが課題として残されていた。本研究では、登山者の歩行能力を考慮するために、Attention 機構を組み込んだ新たな深層学習モデル Hiking-LATTE を提案する。本手法は、修正ローレンツ関数を用いた歩行能力推定に加え、Attention 機構を活用することで登山者の位置情報ログ全体を考慮し、時系列的関係を適切に捉えることが可能である。Kaggle の登山ログデータを用いた実験により、Hiking-LATTE は既存の登山到着時刻推定手法を上回る精度を達成し、従来手法と比較して平均絶対パーセント誤差、平均絶対値誤差、平均二乗誤差の指標において最良の性能を示した。これにより、Hiking-LATTE は登山計画支援や安全対策の向上に貢献する有望な手法であることが示された。

1 はじめに

登山は人気のアクティビティである一方で、山岳遭難は重大な問題である [1]。日本では、2023 年に 3126 件の山岳遭難が発生しており、統計を取り始めた 1961 年以降最多件数であった [2]。山岳遭難を未然に防ぐためにも、事前に無理のない計画を立て、登山中に予期せぬアクシデントや遅れが発生した場合は、引き返すかビバークするなど適切な対処を行うことが重要である。

登山計画を立案する際には、従来から登山地図に掲載されたコースタイムが広く活用されてきた。コースタイムとは、山中の各地点間の所要時間を示した地図であり、登山者が目的地に到着する時刻を見積もるための指標として用いられる。しかし、正確に所要時間を見積もるためには、登山者個人の特性によって補正が必要である。補正には、登山者の経験値や体力、装備、パーティ人数、天候、季節等、様々な要因が影響する。また、疲労によ

る速度低下や、休憩による所要時間増加も考慮する必要がある。さらに、登山中に状況は刻一刻と変わり、登山中に予測を見積もりなおすことも必要である。これらの要素を総合的に考慮し、正確な所要時間を見積もるには、豊富な経験と知識が求められる。登山経験者であれば、過去の自分自身の登山記録から、大まかに歩行ペースを推定することができる。一方で、登山初心者は過去の記録がほとんどないため、自分の歩行ペースを推定することが難しく、所要時間の正確な見積もりは困難である。

そのため、本研究では、登山における所要時間推定タスクに取り組む。登山の所要時間推定における既存研究では、斜度と速度の関係から到着時刻を推定するものが多く提案されている。特に、Campbell [3] らは、登山の位置情報ログの斜度と速度の関係を修正ローレンツ関数によりフィッティングすることにより、精度の高い所要時間推定を可能にした。しかし、これらの手法は、疲労による影響や天候や、個人の能力などの、斜度と速度の関係性以外の特徴を考慮することができない。特に、疲労による速度低下や登山者の歩行能力差は、到着時刻に大きな影響を与える要因であり、これらを正確に反映するには、時系列的関係を柔軟に捉えられるモデルが必要である。

また、Asako [4] らは、登山における到着時刻推定タスクのための深層学習モデルである HikingTTE を提案した。深層学習手法を用いることによって、単なる斜度と速度の関係のみならず、地形、標高、個人の歩行能力、歩行ペースの変化など、各特徴量間の複雑な相互関係を捉えることが可能になった。しかし、HikingTTE では、修正ローレンツ関数を介することで登山者の歩行能力を推定するにとどまっており、実際の登山者の位置情報ログに含まれる全ての情報を用いることができないという問題点があった。

そこで本研究では、近年時系列データ解析において精度向上に貢献している Attention 機構 [5–7] を活用し、登山者の歩行能力を考慮した新たな深層学習モデル Hiking-LATTE を提案する。本モデルは、修正ローレンツ関数を用いた斜度—速度関数に加え、Attention 機構を採用することで、登山者の位置情報ログに含まれる全ての情報をモデルの推論に活用可能にした。その結果、Hiking-LATTE は、HikingTTE を含む既存の深層学習モデルによる到着時刻推定手法と比較して、高い精度を達成した。

本論文の構成は以下の通りである。第 2 章では、既存の登山歩行速度推定手法と、交通分野での到着時刻推定手法、登山における深層学習を用いた到着時刻推定手法について述べる。第 3 章では、今回扱う登山ログと、登山の歩行速度推定タスクについて詳細を述べる。第 4 章では、本研究で提案する提案手法について述べる。第 5 章では、既存手法と提案手法の比較を行う。第 6 章では、提案手法について議論を行う。第 7 章では、本論文のまとめを述べる。なお本論文は、査読無し研究会発表論文 [8] を基にしている。

¹ 非会員 電気通信大学
asako.mizuho@ohsuga.lab.uec.ac.jp

² 非会員 電気通信大学
tahara@uec.ac.jp

³ 非会員 電気通信大学
ohsuga@uec.ac.jp

⁴ 正会員 電気通信大学
seiuny@uec.ac.jp

2 関連研究

2.1 登山における到着時刻推定手法

2.1.1 Naismith の法則

Naismith の法則 [9] は、登山の所要時間を見積もるための手法の一つである。Naismith は歩行ペースの推定方法として、「1 時間ごとに 3 マイル (約 4.8km)、ただし 2000 フィート (約 610m) 登るごとに所要時間は 1 時間増加する」という経験則を述べた。計算式が単純であり、登山中に容易に計算することができるという利点があるが、登り坂のみを考慮しており、下り坂を考慮していないという欠点がある。

2.1.2 Tobler の歩行モデル

Tobler のモデル [10] は、歩行速度と斜度の関係を指数関数で表したモデルである。具体的には、以下の数式 (1) で表される。

$$W = 6e^{-3.5|S+0.05|}, S = \frac{dx}{dh} \quad (1)$$

ここで、 W は歩行速度 (km/h)、 S は斜度 (%) である。このモデルは下り坂においても適用可能だが、指数関数モデルであるため、斜度が大きい場合に速度を過小評価してしまう。

2.1.3 Campbell の歩行モデル

Campbell の歩行モデル [3] は、歩行速度と斜度の関係をローレンツ関数で表したモデルである。具体的には、以下の式 2 で表される。

$$r = c \left(\frac{1}{\pi b \left(1 + \left(\frac{s+a}{b} \right)^2 \right)} \right) + d + es \quad (2)$$

ここで、 r は歩行速度 (m/s)、 s は斜度 (°) を表す。 a, b, c, d, e はモデルのパラメタである。ローレンツ関数による近似は、指数関数による近似と比較して、斜度が大きい場合でも速度を過小評価しすぎないという利点がある。また、Asako [4] らによると、Campbell の歩行モデルは、斜度-速度ベースの登山の到着時刻推定手法の中で、最も精度よく推定できる手法である。

2.1.4 Wood の歩行モデル

Wood のモデル [11] は、歩行方向の斜度 (歩行斜度) の他に、地形斜度と障害物レベルの影響を考慮したモデルである。歩行斜度は歩行者が歩く方向に対する斜度である。地形斜度は地形そのものの斜度で、地形の特定のある地点に対して一意に定まる。具体的には、以下の数式 (3) で表される。

$$v = \exp(a + b\phi + c\theta + d\theta^2) \quad (3)$$

ここで、 v は歩行速度 (km/h)、 ϕ は地形斜度 (°)、 θ は歩行斜度 (°) を表す。 a, b, c, d は障害物レベルによって異なる値を持つパラメタである。地形斜度と障害物を加えることによって、歩行速度の予測精度が向上したと報告している。一方で、Tobler のモデルと同様に、指数関数モデルであるため、斜度が大きい場合に速度を過小評価しすぎる欠点がある。

しかし、これらの既存手法は、すべて斜度から歩行速度を推定するモデルである。そのため、疲労による速度低下や、休憩による所要時間の増加を適切に考慮することができないという欠点がある。

2.2 交通分野における到着時刻予測モデル

交通分野において、到着時刻推定タスクは重要な課題であり、これまで様々な研究がされてきた。本研究でも、交通分野における深層学習手法を重要なベースラインとして位置づけ、提案手法との比較を行う。

2.2.1 DeepTTE

DeepTTE [12] は、位置情報ログと属性情報を入力として、部分区間の所要時間推定と全体の所要時間推定を同時に学習するモデルである。1 次元量み込みと LSTM を用いることで、時空間的な関係性を捉えることを可能にし、Attention を用いたプーリングを行うことで、各部分区間の重要度によって全体の所要時間推定の精度を向上させた。

2.2.2 Gct-TTE

Gct-TTE [13] は、位置情報ログ (GPS sequences)、気象データ、マップパッチ (map patches)、道路グラフ (Road Graph) を入力として利用するトランスフォーマーベースのモデルである。複数のデータモダリティを入力として受け取り、それぞれのモダリティに対して専用の特徴抽出アルゴリズムを用いた。結果として、Gct-TTE による交通予測は、MAE および RMSE の評価指標において、DeepTTE を含む複数の最先端モデルと比較して優れた結果を示した。

一方、Gct-TTE の入力として用いる道路グラフ (Road Graph) の情報は、登山のタスクで用いることは難しい。Gct-TTE は都市部の道路網など、複雑なネットワークを前提として構築されているが、登山道は通常、単純な一本道や狭いトレイルで構成されており、Gct-TTE が道路グラフを通じて捉えようとする複雑なネットワーク構造が存在しない。その結果、登山道ではグラフ情報をを用いた特徴抽出の恩恵が少ないと考えられる。

2.2.3 AttentionTTE

AttentionTTE [14] は、タクシーの所要時間推定タスクに対し、現時点において最先端の精度を誇る手法である。AttentionTTE は、DeepTTE と同様に、位置情報ログと属性情報を入力として、部分区間と全体の所要時間推定を同時に行うモデルである。DeepTTE と異なる点として、以下の 2 点が挙げられる。まず 1 点目として、部分区間の所要時間推定の際に、位置情報の各データ点間の関係性を捉えるために、工夫された自己注意機構を使用している。DeepTTE では LSTM を用いて時系列関係を捉えていたが、AttentionTTE では自己注意機構と LSTM を組み合わせたアーキテクチャを採用している。この機構により、位置情報のシーケンス間の関係性を効果的に捉え、影響力の大きい位置情報のデータ点を強調することを可能にした。また、2 点目として、全体区間の所要時間推定の際に、自己注意行列を活用することで、全体の位置情報ログに対する各局所的な軌道セグメントの貢献度を正確に特定することを可能にした。結果として、タクシーの到着時刻推定タスクにおいて、DeepTTE を含む最先端の手法と比較した結果、最もよい精度を達成し、本タスクにおいて、現在、SoTA を達成した。

そのため、本研究では、DeepTTE と AttentionTTE を登山における到着時刻推定タスクのベースラインとし、提案手法と比較する。

2.3 登山における深層学習を用いた到着時刻推定手法

HikingTTE [4] は、登山の到着時刻推定タスクにおいて、初めて深層学習を用いた手法である。HikingTTE は、修正ローレンツ関数による斜度—速度関数により、登山者の歩行能力を推定し、LSTM により時系列的な関係性を捉えるモデルである。斜度—速度関数は登山者個人の歩行能力を適切に考慮することができ、結果として既存の登山における到着時刻推定手法と比較して推定精度を大幅に改善した。一方で、登山者の位置情報ログから斜度—速度関数を介して登山者の能力を推定する制約により、登山者の位置情報ログのすべての情報を予測に反映することができないという問題点があった。

そのため、本研究では、HikingTTE の斜度—速度関数に加えて、Attention 機構により登山者の位置情報ログを適切に重みづけし、予測精度を向上させることを目指す。

3 登山の到着時刻推定に関連する前提知識

本研究では、登山の到着時刻を推定するために、「登山の位置情報ログ」と「登山の属性情報」を用いる。

3.1 登山の位置情報ログ

登山における位置情報は、GPX 形式で保存される。時系列に沿って登山者の移動軌跡を保存することができ、各ポイントで緯度、経度、標高、時間の情報を含んでいる。GPX 形式は登山記録に最も良く使われている形式であり、Yamap [15], Yamareco [16] などの登山アプリでは、登山中の位置情報ログを GPX 形式で保存することができる。

本研究では、GPX 形式で保存された登山ログを用いて、「記録された GPX ログを歩き切るのにどのくらいの時間がかかるのか？」を推定することを目的とする。ここで、登山ログを次のように定義する。

- P : 位置情報ログ
- $p_1, p_2, \dots, p_T$: 各データポイント

各登山ログは長さ T である。また、各データ点に、追加の情報として、スタート地点からの累積距離、歩行斜度、地形斜度を追加する。ここで、歩行斜度は歩行方向の斜度で、地形斜度は地形そのものの斜度である。累積距離は到着時刻推定に直接影響する要素であるため追加する。歩行斜度と地形斜度は、歩行速度に関係していることが示唆されている [17]。累積距離と歩行斜度は緯度経度、標高の差分から計算し、地形斜度は DEM データ [18, 19] から取得した。さらに、登山の歩行速度に影響を与えられられる気象情報について、Open-Meteo [20] から取得した。つまり、とある登山ログの i 番目のデータ点は、緯度 lat 、経度 lng 、標高 ele 、スタート地点からの累積距離 $dist$ 、歩行斜度 ws 、地形斜度 ts 、スタート地点からその地点までの所要時間 $time$ に加え、気象条件として地表 2 メートルの気温 ($^{\circ}\text{C}$) $temperature_2m$ 、降水量 (mm) $rain$ 、降雪量 (mm) $snow_fall$ 、積雪深 (mm) $snow_depth$ 、雲量 (%) $cloud_cover$ 、地表 10 メートル地点での風速 (m/s) $wind_speed_10m$ を有する。

3.2 登山の属性情報

登山の属性情報は、登山ログ全体の特徴を示す情報である。先ほどの登山の位置情報ログから抽出可能な属性情報（以下 Attr）として、最高標高点 $ele_{\max}$ 、最低標高点 $ele_{\min}$ 、累積標高登り D^+ 、累積標高下り D^- 、ログの総距離 $dist_{\text{total}}$ 、ログの総時間 $time_{\text{total}}$ のほか、気象条件として地表 2 メートルの平均気温 ($^{\circ}\text{C}$) $temperature_2m_ave$ 、平均降水量 (mm) $rain_ave$ 、平均降雪量 (mm) $snowfall_ave$ 、平均積雪深 (mm) $snow_depth_ave$ 、平均雲量 (%) $cloud_cover_ave$ 、地表 10 メートル地点での平均風速 (m/s) $wind_speed_10m_ave$ を取得した。

3.3 本研究で扱うタスクの定義

本研究では、登山の位置情報ログ P と属性情報 Attr を入力とし、与えられた位置情報ログの区間を歩き切るために必要な所要時間を推定するタスクを扱う（図 1）。

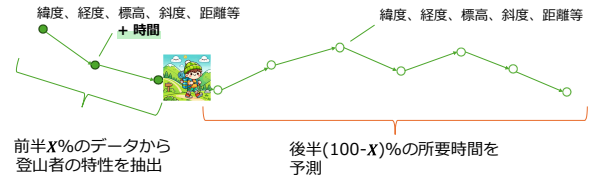


図 1 本研究が想定するシナリオ

ここで、位置情報ログを前半 $X\%$ と後半 $(100 - X)\%$ に分割する。前半 $X\%$ の位置情報ログは、その地点への到着時刻の情報 ($p_i.time$) を与え、登山者の特性を抽出するために用いる。そして、前半 $X\%$ の位置情報ログから抽出した登山者の特性と、後半 $(100 - X)\%$ の軌跡の情報を元に、後半 $(100 - X)\%$ を歩くために必要な所要時間を予測することが、本研究の目的となる。ここで、前半 $X\%$ の位置情報ログを $P_{\text{front}} = \{p_1, p_2, \dots, p_x\}$ とし、後半 $(100 - X)\%$ の位置情報ログを $P_{\text{back}} = \{p_{x+1}, p_{x+2}, \dots, p_T\}$ と表すこととする。この問題設定は、実際の登山において、「とある登山ルートを歩いている最中に、それまでの歩行実績を活用して、そのルートの残りを歩くのに必要な時間を推定する」という状況に該当する。登山では、仮に予定より遅れている場合には、登山の途中でも、引き返したり、エスケープルートを選択したりする必要がある。そのため、登山者の歩行実績を用いて、逐次到着時刻を修正し、登山者により正確な推定到着時刻を提供することは重要である。また、前半 $X\%$ から登山者の歩行特性を抽出する理由は、登山者の経験値、体力、装備、パーティ人数、季節といった歩行速度に影響を与える様々な要因を間接的に考慮するためである。これらの要因を直接データとして収集できない場合でも、前半 $X\%$ のその登山ログの歩行実績からそれらを推測し、予測に活用することができる。

本研究で提案するモデルは、登山者の歩行特性を予測に反映させるために、前半 $X\%$ の位置情報ログから登山者特性を抽出し、抽出した登山者特性に基づいて、後半 $(100 - X)\%$ のログの区間を歩き切るために必要な時間を推定する。次のセクションでは、具体的なモデルのアーキテクチャについて説明する。

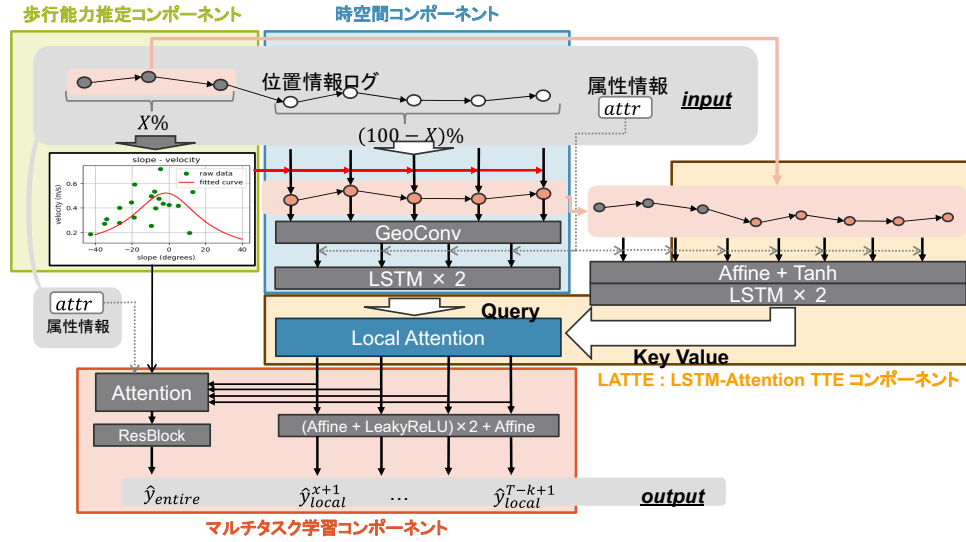


図2 Hiking-LATTE のモデル概要図

4 登山における到着時刻推定手法: Hiking-LATTE

本研究では、登山における到着時刻推定手法として、Hiking-LATTE を提案する。具体的なモデルアーキテクチャは図2の通りである。

このモデルは4つのコンポーネントから構成される。1つ目は登山者の歩行能力を推定する「歩行能力推定コンポーネント」、2つ目は時系列的・空間的な特徴を抽出する「時空間コンポーネント」、3つ目は Attention 機構により登山者の歩行能力を考慮する「LATTE コンポーネント」4つ目はマルチタスク学習を行う「マルチタスク学習コンポーネント」である。

4.1 歩行能力推定コンポーネント

歩行能力推定コンポーネントでは、各個人の歩行能力を推定する。本コンポーネントは著者らが提案した HikingTTE に基づいている。Campbell [3] らは、斜度と速度の関係を表現する際に、式2で示される修正ローレンツ関数を用いている。本研究においても、歩行速度と斜度の関係は修正ローレンツ関数でモデル化可能であると仮定し、前半 $X\%$ の位置情報ログ (P_{front}) から登山者ごとの歩行特性を推定する。ただし、Campbell の研究では、すべての登山者の歩行ログを統合して斜度一速度関数を作成しているため、大量のデータ点を用いたフィッティングが可能である。一方、本研究では各登山者の前半 $X\%$ のみを使用するため、データ点数が少なく、オーバーフィッティングの問題が生じる。そこで、修正ローレンツ関数のパラメータ a, b, c, d, e のうち、歩行能力に最も関連するパラメータ c のみを可変とし、その他のパラメータ (a, b, d, e) は固定する。このことにより、限られたデータでもオーバーフィッティングを防ぎつつ、個々の登山者の歩行特性を表現できるようにした。固定するパラメータ a, b, d, e は、Campbell の手法にて、50 パーセントモデルとして利用されていたパラメータの値である $a = -1.4579, b = 22.0787, d = 0.0525, e = 3.2002 \times 10^{-4}$ を利用

することとし、 c のみを変数としてデータ点にフィッティングを行う。

本コンポーネントで推定した「斜度一速度関数」は、次に説明する時空間コンポーネントに関数形式のまま渡されると同時に、斜度 $\{-30^\circ, -20^\circ, -10^\circ, 0^\circ, 10^\circ, 20^\circ, 30^\circ\}$ に対する推定速度 $\{\hat{v}_{-30^\circ}, \hat{v}_{-20^\circ}, \hat{v}_{-10^\circ}, \hat{v}_{0^\circ}, \hat{v}_{10^\circ}, \hat{v}_{20^\circ}, \hat{v}_{30^\circ}\}$ を計算する。この7種類の斜度における推定速度を属性情報 Attr に付与することで、「各登山者が、各斜度でどの程度の歩行速度であるか」という情報を、モデル全体に反映できるようになる。

4.2 時空間コンポーネント

時空間コンポーネントは、これから歩く後半 $(100 - X)\%$ の登山ルートに対応する位置情報ログを入力とし、位置情報ログの時空間的関係を学習するためのコンポーネントである。

まず、入力として後半の位置情報ログ $P_{\text{back}} = \{p_{x+1}, p_{x+2}, \dots, p_T\}$ を受け取る。続いて、各データ点 p_i に対して、先述の「斜度一速度」関数を用いて、追加の要素である「推定速度」の情報を追加する。この推定速度は登山者の歩行能力を反映したものであり、入力として与えることで、より正確な推定が可能になると考えられる。次に、各データ点 $p_i = \{p_i.\text{lat}, p_i.\text{lng}, p_i.\text{ele}, p_i.\text{dist}, p_i.\text{ws}, p_i.\text{ts}, p_i.\hat{v}\}$ を正規化したのち、非線形変換を行い、16次元の位置情報表現 loc_i を作成する。さらに、16次元の位置情報表現に対して、1次元畳み込みを行う(式4)。

$$\text{loc}_i^{\text{conv}} = \sigma_{\text{cnn}}(W_{\text{conv}} * \text{loc}_{i:i+k-1} + b) \quad (4)$$

ここで、 $*$ は畳み込み演算を行う演算子である。 k は畳み込み演算に用いる畳み込み演算に用いるフィルタ数は c であり、畳み込み後の位置情報表現 $\text{loc}_i^{\text{conv}}$ の次元数は c である。その後、属性情報を結合し、2層の LSTM レイヤに入力する(式5)。

$$h_{\text{back}} = \sigma_{\text{lstm}}(W_h \cdot [\text{loc}^{\text{conv}}, \text{Attr}]) \quad (5)$$

ここで得た LSTM の出力 h_{back} は、次の LATTE コンポーネント

に与えられる。

4.3 LATTE (LSTM-Attention TTE) コンポーネント

LATTE コンポーネントは、Attention 機構を用いて、後半の位置情報ログの各部分区間に対して、前半を含む全体の位置情報ログの各部分区間の重要度を計算するためのコンポーネントである。後半の位置情報ログの LSTM 層の出力 h_{back} を Query とし、全体の位置情報ログの LSTM 層の出力 h_{full} を Key, Value とする Attention 構造である。モデルのアーキテクチャの概要を図 3 に示す。

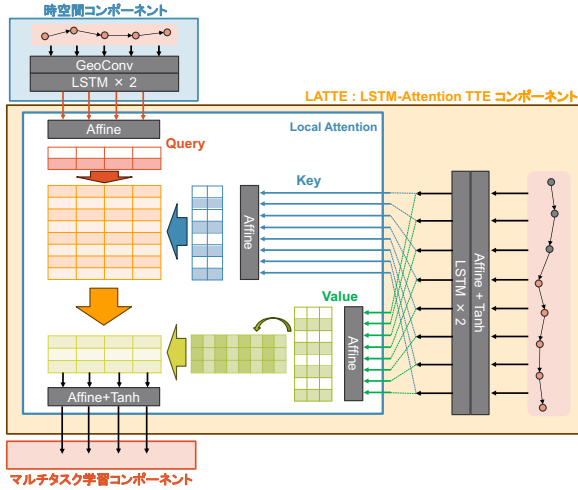


図3 LATTE コンポーネントの概要図

Hiking-LATTE では、部分区間の予測を行うために、後半の位置情報ログに対して、全体の位置情報ログの各部分区間の重要度を計算する。そのため、時空間コンポーネントの出力である、部分区間の位置情報表現 h_{back} が、Attention 機構における Query に相当する。以下の式 6 によって、Query を作成する。

$$\mathbf{Q} = W_q \cdot h_{back} \quad (6)$$

Key および Value には、前半を含む全体の位置情報ログを用いる。まず、前半 $X\%$ の実際の速度を含む歩行ログと、修正ローレンツ関数によって予測速度が追加された後半 $(100 - X)\%$ の歩行ログを結合する。これらの歩行ログの各データ点 p_i について、データ点間の差分を求め、正規化を行った後に非線形変換を適用して、16 次元の位置情報表現 loc^{full} を作成する。

そして、この位置情報表現 loc^{full} に、属性情報 Attr を正規化したものを結合し、LSTM レイヤに入力する (式 7)。

$$h_{full} = \sigma_{\text{LSTM}}(W_h \cdot [\text{loc}^{\text{full}}, \text{Attr}]) \quad (7)$$

なお、Key, Value の作成時には、GeoConv 層を挟まず、データ点間の差分を単純な LSTM に入力することにより位置情報表現を作成する。この設計の理由は、Key と Value は Query との重要度を計算する役割を担っており、位置情報の各データ点間の情報を細かく反映させる必要があるからである。

次に、全体区間の位置情報表現 h_{full} に対して線形変換を行い、

Key と Value を作成する (式 8)。

$$\mathbf{K} = W_k \cdot h_{full}, \mathbf{V} = W_v \cdot h_{full} \quad (8)$$

Key ($\mathbf{K}$) は Query ($\mathbf{Q}$) との類似度を計算するために利用され、Value ($\mathbf{V}$) は Attention スコアによって強調される具体的な情報を保持する。この設計により、Key と Value は前後半を結合した全体の位置情報ログの各データ点の特徴を保持しており、Query と Attention スコアの計算を行うことで、動的に重要な情報を抽出することが可能になる。

次に、Query ($\mathbf{Q}$), Key ($\mathbf{K}$), Value ($\mathbf{V}$) を用いて、後半の位置情報ログと全体の位置情報ログの関連性を計算し、重要度に基づいた位置情報表現の抽出を行う。Attention スコアを計算し、非線形層で変換することで、最終的な出力 h_{back}^{att} を得る (式 9)。

$$h_{back}^{\text{att}} = \tanh\left(W_o \cdot \text{softmax}\left(\frac{\mathbf{Q} \cdot \mathbf{K}^T}{\sqrt{d_k}}\right) \mathbf{V}\right) \quad (9)$$

最終的な出力 h_{back}^{att} は、後半の位置情報ログにおける重要な部分区間の情報を動的に考慮しており、前半の位置情報ログと後半の位置情報ログの両方を考慮した時空間的な特徴を効率的に抽出することが可能である。

4.4 マルチタスク学習コンポーネント

マルチタスク学習コンポーネントでは、部分区間の所要時間と、全体の所要時間の予測を同時に行う。マルチタスク学習は、DeepTTE [12] にて提案された手法であり、「部分区間の所要時間」と「全体の所要時間」を同時に学習し、モデルのトレーニング時にはその両方の予測精度を向上させるように学習させる手法である。

4.4.1 部分区間の所要時間推定部

部分区間の所要時間推定部では、先述の LATTE コンポーネントが出力する隠れ状態 h を入力とし、非線形変換を行う。図 2 に示すように、2 層の (Affine + LeakyReLU) 層を通した後、線形変換を行うことで、最終的な部分区間の所要時間予測 $\hat{y}_{\text{local}} = \{\hat{y}_{\text{local}}^{x+1}, \hat{y}_{\text{local}}^{x+2}, \dots, \hat{y}_{\text{local}}^{T-k+1}\}$ を得る。各予測 $\hat{y}_{\text{local}}^i$ は、位置情報ログ p_i に対応しており、 p_i の区間を歩くのに要する時間をモデルが予測した値となる。

4.4.2 全体の所要時間推定部

全体の所要時間推定部では、Attention 機構を用いて、属性情報 Attr と時空間コンポーネントから得られる隠れ状態 h_{back}^{att} を入力とし、全体の所要時間を予測する。

まず、属性情報を非線形変換し、その次元数を隠れ状態 h と一致させる。次に、非線形変換した属性情報を Query、隠れ状態 h を Key および Value として、Attention 機構により重み付き和 h_{att} を計算する。続いて、全結合層と残差接続層を通じて、最終的な所要時間 $\hat{y}_{\text{entire}}$ を出力する。この一連の機構により、属性情報と位置情報の両方を考慮した所要時間の予測が可能となる。

4.4.3 モデルのトレーニング

モデルの学習段階では、部分区間予測に対する MAPE Loss (L_{local}) と、全体の所要時間予測に対する MAPE Loss (L_{entire}) を重み付けした損失関数を用いる。部分区間予測に関する損失は、

式 10 で計算される。

$$L_{\text{local}} = \frac{1}{\text{length}(P_{\text{back}})} \sum_{i=x+1}^{T-k+1} \left| \frac{\hat{y}_{\text{local}}^i - y_{\text{local}}^i}{y_{\text{local}}^i + \epsilon} \right| \quad (10)$$

ここで、 ϵ は、MAPE Loss が大きな値になりすぎて学習が不安定になることを防ぐための定数である。また、全体の所要時間予測に関する損失は、式 11 で定義される。

$$L_{\text{entire}} = \left| \frac{\hat{y}_{\text{entire}} - y_{\text{entire}}}{y_{\text{entire}}} \right| \quad (11)$$

これらの損失を重みづけし、以下の値が最小になるように学習を進める。

$$\alpha \cdot L_{\text{local}} + (1 - \alpha) \cdot L_{\text{entire}} \quad (12)$$

ここで、 α は重み付けの係数であり、0 から 1 の値を取る。最終的な目標は L_{entire} を最小化することであるため、評価は全体の所要時間予測値と実際の値の差に基づく MAPE を用いて行う。

5 実験

本研究では、Kaggle に公開されている登山者の位置情報ログデータセット「GPS recorded hikes from hiker.org」[21] のデータを用いた。このデータセットは、hiker.org [22] に掲載されている登山者の位置情報ログを収集したものであり、約 12,000 件の登山ログからなるデータセットである。この登山ログは、各位置情報 p_i ごとに、緯度 (lat)、経度 (lng)、標高 (ele)、時刻 (time) の情報を有している。データセットには欠損値が含まれているため、欠損値を排除するために各登山ログに対して、

- 欠損値を含むポイントの削除
- データ点が 50 点未満の登山ログを除外
- 不正値（速度が 5 m/s = 18 km/h を超えるデータと、データ点同士の間隔が 1200 秒以上のデータ）の確認
- 連続 50 点以上正常値である区間が存在する場合、その区間を切り取り用いる
- 存在しない場合、その登山ログを除外する

の操作を行った。この操作の結果、新たに 8,538 件の登山ログが得られた。また、地形斜度は NASADEM [18] と ArcticDEM Mosaic [19]、気象情報は Open-Meteo [20] からそれぞれ取得し、登山ログに追加した。

5.1 モデルの学習

モデルを学習するために、データセットを train データ、validation データ、test データに分割した。全体の 80% を train データ、20% を test データとして分割し、train データの内 5% を validation データとした。損失関数は MAPE を用いて、学習は 4.3 章のように部分区間の所要時間推定と全体の所要時間推定の損失を重みづけして行った。Optimizer には Adam [23] を使い、学習率は 0.001 として、1000 エポック学習を行った。ハイパーパラメータとして、バッチサイズ 64、部分区間と全体の所要時間の損失の重みづけ係数 $\alpha = 0.5$ を選択した。

5.2 評価

評価指標として、MAPE (Mean Absolute Percentage Error)、MAE (Mean Absolute Error)、及び MSE (Mean Squared Error) を用いた。また、1000 エポックの学習の中で、Validation データに対する損失の推移を確認した。具体的には、1 エポックごとの Validation データに対する損失を記録し、フィルタサイズ 50 エポックで損失の移動平均を取り、その移動平均の値が最も小さくなった時点のモデルを最終的なモデルとして採用した。なお、学習重みを 10 エポックごとに保存したため、移動平均が最小になったエポックに最も近い重みを選択した。Test データに対して、MAPE、MAE、MSE を計算し、提案手法と既存手法との比較を行った。

5.3 比較手法

本研究では比較手法として、交通分野の到着時刻推定タスクで発展した **DeepTTE** と最新手法である **AttentionTTE**、及びこれらを本タスク向けに改良したモデル、**DeepTTE+Ave** 及び **AttentionTTE+Ave** を使用した。DeepTTE+Ave は、DeepTTE が入力とする属性情報 Attr に加えて、前半 X% の位置情報ログにおける平均歩行速度と、平均歩行斜度を追加で与えるモデルである。これにより、前半区間で観測された登山者の歩行能力に関する情報を、DeepTTE の学習および推定に反映できるようになる。AttentionTTE+Ave も同様に構築した。また、登山に特化した到着時刻推定モデルである **HikingTTE** も比較手法として使用した。DeepTTE+Ave 及び AttentionTTE+Ave は、3 章で述べた登山の属性情報 Attr に加えて、前半 X% の位置情報ログにおける平均歩行速度 $\bar{v}$ と、平均歩行斜度 $\bar{ws}$ を追加で与えるモデルである。これにより、前半区間で観測された登山者の歩行能力に関する情報を、DeepTTE・AttentionTTE の学習および推定に反映できるようになる。

5.4 実験結果

前半ログの分割率 X を 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% と変化させ、各分割率に対して提案手法と DeepTTE+Ave, AttentionTTE+Ave, HikingTTE を学習し、推論を行った。本実験の結果を表 1 及び 3 に示す。表 1 より、MAPE 及び MAE に関しては $X = 5\%$ を除くすべてのデータ分割率において、Hiking-LATTE は既存手法と比較して高精度であった。表 3 より、MSE に関しても $X = 5\%$ 及び $X = 90\%$ を除くデータ分割率において、Hiking-LATTE の精度が最も高かった。既存手法の中で最も精度の良い DeepTTE+Ave と比べても、Hiking-LATTE は平均して MAPE が 0.67 ポイント、MSE が 8.0% 改善した。

6 考察

交通分野における手法について、タクシーのタスクに対して、AttentionTTE は DeepTTE よりも高精度に予測時間を推定することができた。一方で、登山における到着時刻推定タスクに対しては、AttentionTTE より DeepTTE の方が高精度であった。これは、AttentionTTE の表現力が高すぎることによる過学習や、パラメータ数が多いことにより適切に収束しないことが問題であると考えられる。Hiking-LATTE はこれら交通分野にて発展した深層学習

表 1 分割率と MAPE の結果

データ分割率 (X) %	5	10	20	30	40	50	60	70	80	90
DeepTTE	13.55	14.20	13.85	13.80	14.97	15.93	16.46	17.83	18.24	21.34
DeepTTE+Ave	10.69	<u>10.37</u>	<u>10.50</u>	11.11	<u>10.84</u>	<u>11.39</u>	<u>11.94</u>	13.24	<u>14.17</u>	17.00
AttentionTTE	14.55	15.09	17.15	16.56	18.47	17.53	18.74	19.97	21.02	21.74
AttentionTTE+Ave	12.45	11.75	13.13	13.00	12.55	13.85	13.95	21.29	18.19	20.54
HikingTTE	<u>11.16</u>	10.73	10.79	<u>10.81</u>	11.11	11.74	12.17	<u>13.07</u>	14.26	<u>16.76</u>
Hiking-LATTE (ours)	11.35	10.04	10.18	10.09	9.92	10.45	11.33	12.65	12.84	15.71

表 2 分割率と MAE の結果

データ分割率 (X) %	5	10	20	30	40	50	60	70	80	90
DeepTTE	26.19	25.78	23.46	19.99	18.02	15.81	12.79	10.18	7.22	4.04
DeepTTE+Ave	22.07	<u>20.33</u>	<u>18.07</u>	16.93	<u>14.30</u>	<u>12.55</u>	<u>10.46</u>	8.50	<u>6.06</u>	3.55
AttentionTTE	27.68	27.59	27.52	22.64	21.48	17.24	13.70	11.38	7.91	4.13
AttentionTTE+Ave	24.88	22.58	22.58	18.89	16.38	14.37	11.57	12.26	7.31	3.97
HikingTTE	22.76	21.26	18.94	<u>16.86</u>	14.65	12.91	10.55	<u>8.44</u>	6.19	<u>3.47</u>
Hiking-LATTE (ours)	<u>22.64</u>	19.61	17.69	15.53	13.37	11.80	9.73	8.18	5.60	3.34

表 3 分割率と MSE の結果

データ分割率 (X) %	5	10	20	30	40	50	60	70	80	90
DeepTTE	1998.4	1865.9	1600.9	1150.2	944.3	667.4	470.5	304.6	160.6	53.6
DeepTTE+Ave	1306.1	<u>1182.6</u>	<u>892.8</u>	<u>765.4</u>	<u>576.7</u>	<u>434.2</u>	313.2	215.6	<u>115.2</u>	43.6
AttentionTTE	2210.0	2212.2	2072.5	1531.1	1190.5	837.4	537.5	388.4	186.5	65.5
AttentionTTE+Ave	1682.3	1357.9	1351.1	911.9	780.1	592.9	373.3	360.4	156.6	59.8
HikingTTE	<u>1401.8</u>	1302.1	973.4	777.2	618.1	469.1	<u>304.6</u>	<u>212.0</u>	119.9	40.7
Hiking-LATTE (ours)	1424.5	1089.7	828.0	651.6	519.1	406.4	262.1	190.8	100.4	<u>42.8</u>

手法である DeepTTE と AttentionTTE と比較して、高精度に登山の到着時刻を推定できることがわかった。さらに、HikingTTE と比較した場合、Hiking-LATTE は MAPE を 0.67 ポイント、MSE を 8.0% 程度改善した。これは、登山ログの前半 X% に含まれる実際の歩行速度を、LATTE コンポーネントによって効果的に抽出できているからだと考えられる。すなわち、Attention 機構を用いて部分区間の重要度を重みづけして考慮することは、所要時間をより正確に推定するうえで有用であると推察される。また、前半ログの実際の歩行速度をそのまま入力として活用できる点も、推定精度の向上に大きく貢献していると推察される。

本論文では、各登山道の整備状況や標準的なコースタイム等の情報は利用しなかった。これらの情報を入手できる場合は、属性情報 Attr に追加することで対応可能である。この場合、既存手法及び提案手法全ての予測精度の向上が見込まれる。しかし、Attention 機構を用いて部分区間の重要度を重みづけして考慮する提案手法の優位性は変わらないと考えられる。より多くの情報を収集し、それらの情報を考慮して実験を行うことは本研究の将来課題である。

7 おわりに

本研究は、登山における到着時刻推定タスクのための深層学習モデルである「Hiking-LATTE」を提案した。Hiking-LATTE は、深層学習ベースの到着時刻推定モデルに、修正ローレンツ関数を用いた歩行能力推定コンポーネントと、Attention 機構により部分区間の重要度を重みづけする LATTE コンポーネントを取り入れることで、個人の歩行能力、歩行ペースの変化、地形、標高等を包括的に考慮することができる。結果として、類似タスクであるタクシーの到着予測手法である DeepTTE、AttentionTTE を登山に適用した手法と比較して高精度で予測できた。さらに、既存の深層学習ベースの登山到着時刻推定手法である HikingTTE と比較しても、高精度で到着時刻を推定できることが分かった。今後の課題として、歩行能力に影響を与えと考えられる追加の特徴量である年齢、性別、パーティ人数等をデータに追加すること、その登山者の過去の登山ログをモデルに反映させることが挙げられる。

謝辞

本研究は JSPS 科研費 JP22K12157, JP23K28377, JP24H00714 の助成を受けたものです。

参考文献

- [1] S. Rauch, B. Wallner, M. Ströhle, T. Dal Cappello and M. Brodmann Maeder: "Climbing accidents - prospective data analysis from the international alpine trauma registry and systematic review of the literature", *International Journal of Environmental Research and Public Health*, **17**, 1, p. 203 (2019).
- [2] 警察庁生活安全局生活安全企画課: "令和 4 年における山岳遭難の概況", Accessed on 30.05.2023. URL: https://www.npa.go.jp/publications/statistics/safetylife/r04sangakusounan_gaikyou.pdf (2023).
- [3] M. J. Campbell, P. E. Dennison and M. P. Thompson: "Predicting the variability in pedestrian travel rates and times using crowdsourced gps data", *Computers, Environment and Urban Systems*, **97**, 101866, pp. 1–13 (2022).
- [4] M. Asako, Y. Tahara, A. Ohsuga and Y. Sei: "Deep learning-based travel time estimation in hiking with consideration of individual walking ability", *Cybernetics and Information Technologies*, **24**, 4, pp. 3–21 (2024).
- [5] D. Bahdanau: "Neural machine translation by jointly learning to align and translate", arXiv preprint arXiv:1409.0473 (2014).
- [6] A. Vaswani: "Attention is all you need", *Advances in Neural Information Processing Systems* (2017).
- [7] H. Song, D. Rajan, J. Thiagarajan and A. Spanias: "Attend and diagnose: Clinical time series analysis using attention models", *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32 (2018).
- [8] 浅古瑞穂, 田原康之, 大須賀昭彦, 清雄一: "登山者の歩行能力を考慮した Attention 機構による登山の到着時刻推定手法の提案", *SMASH25 Winter Symposium*, 信学技報, AI2024-12, 第 124 巻, pp. 19–24 (2025).
- [9] W. W. Naismith: "Cruach ardran, stobinian, and ben more", *The Scottish Mountaineering Club Journal*, pp. 135–136 (1892).
- [10] W. Tobler: "Three presentations on geographical analysis and modeling", Technical report, National Center for Geographic Information and Analysis (1993).
- [11] A. Wood, W. Mackaness, T. I. Simpson and J. D. Armstrong: "Improved prediction of hiking speeds using a data driven approach", *PLOS ONE*, **18**, 12, pp. e0295848:1–12 (2023).
- [12] D. Wang, J. Zhang, W. Cao, J. Li and Y. Zheng: "When will you arrive? estimating travel time based on deep neural networks", *Thirty-Second AAAI Conference on Artificial Intelligence*, Vol. 32, pp. 2500–2507 (2018).
- [13] V. Mashurov, V. Chopuryan, V. Porvatov, A. Ivanov and N. Semenova: "Gct-tte: graph convolutional transformer for travel time estimation", *Journal of Big Data*, **11**, 15, pp. 1–14 (2024).
- [14] M. Li, Y. Feng and X. Wu: "AttentionTTE: a deep learning model for estimated time of arrival", *Frontiers in Artificial Intelligence*, **7**, 1258086, pp. 1–13 (2024).
- [15] 株式会社ヤマップ: "Yamap", Accessed on 30.05.2024. URL: <https://yamap.com/> (2024).
- [16] 株式会社ヤマレコ: "Yamareco", Accessed on 30.05.2024. URL: <https://www.yamareco.com/> (2024).
- [17] F. Arnet: "Arithmetical route analysis with examples of the long final courses of the world orienteering championships 2003 in switzerland and 2005 in japan", *Scientific Journal of Orienteering*, **17**, 1, pp. 4–21 (2009).
- [18] NASA JPL: "Nasadem merged dem global 1 arc second v001 [data set]", https://doi.org/10.5067/MEASURES/NASADEM/NASADEM_HGT.001 (2020). Accessed: 2024-10-30.
- [19] C. Porter, P. Morin, I. Howat, M.-J. Noh, B. Bates, K. Peterman, S. Keesey, M. Schlenk, J. Gardiner, K. Tomko, M. Willis, C. Kelleher, M. Cloutier, E. Husby, S. Foga and H. Nakamura: "Arcticdem, version 3", <https://doi.org/10.7910/DVN/OHHUKH> (2024). Accessed: 2024-10-30.
- [20] P. Zippenfenig: "Open-meteo.com weather api" (2023). Creative Commons Attribution 4.0 International. Available at <https://open-meteo.com/>.
- [21] "Gps recorded hikes from hiker.org", <https://www.kaggle.com/datasets/roccoli/gpx-hike-tracks>. Accessed: 2024-10-30.
- [22] "Hiker.org", <https://www.hiker.org>. Accessed: 2024-10-30.
- [23] D. P. Kingma and J. L. Ba: "Adam: A method for stochastic optimization", *International Conference on Learning Representations (ICLR)* (2015).