

پیش‌بینی حالت‌های حمل‌ونقل از نقاط خط سیر با استفاده از روش‌های تقویت‌کننده و یادگیری عمیق در حمل‌ونقل هوشمند

مقاله علمی - پژوهشی

سجاد صولتی، دانشکده مهندسی نقشه‌برداری و اطلاعات مکانی، دانشگاه تهران، تهران، ایران
رحیم علی‌عباسپور*، دانشکده مهندسی نقشه‌برداری و اطلاعات مکانی، دانشگاه تهران، تهران، ایران
علیرضا چهرقان، دانشکده مهندسی معدن، دانشگاه صنعتی سهند، تبریز، ایران
*پست الکترونیکی نویسنده مسئول: abaspour@ut.ac.ir

دریافت: ۱۴۰۱/۱۰/۲۰ - پذیرش: ۱۴۰۲/۰۴/۲۸

صفحه ۱۷۰-۱۴۹

چکیده

امروزه با گسترش شهرنشینی نیاز به حمل‌ونقل هوشمند به منظور تسهیل رفت و آمد شهروندان بیش از پیش مورد توجه قرار گرفته است. شناسایی و پیش‌بینی استفاده از حالت‌های حمل‌ونقلی یکی از اساسی‌ترین پیش‌نیازها در راه‌اندازی و استفاده از خدمات حمل‌ونقل هوشمند به شمار می‌آید. با پیشرفت فناوری‌های مکانی، ابزار و تلفن‌های هوشمند، اطلاعات زیادی با استفاده از سیستم‌های تعیین موقعیت ماهواره ای (GNSS) توسط بسیاری از دستگاه‌ها تولید می‌شود. در این پژوهش، چهار ویژگی نقطه‌ای، ۵۶ ویژگی سفر و سه ویژگی پیشرفته استخراج شده، چهار مدل کلاسه‌بندی GB، XGBoost، LightGBM و CatBoost زیر مجموعه روش تقویت‌کننده (Boosting) پس از انتخاب ویژگی ترکیبی به همراه سه مدل کلاسه‌بندی LSTM، CNN و ConvLSTM زیر مجموعه روش یادگیری عمیق پیاده‌سازی و بررسی شده تا بتوان حالت‌های حمل‌ونقلی شامل: پیاده‌روی، استفاده از دوچرخه، استفاده از اتوبوس، استفاده از اتومبیل و استفاده از قطار را با استفاده از مجموعه داده‌های GeoLife پیش‌بینی کرد. نتایج نشان داد مدل LightGBM با کسب F_{score} بالاتر (۹۵/۴۹ درصد) و پیچیدگی زمانی کمتر، بهترین مدل نسبت به مدل‌های دیگر است.

واژه‌های کلیدی: حمل‌ونقل هوشمند، حالت‌های حمل‌ونقل، خط سیر، یادگیری عمیق

۱-مقدمه

Chamoso, González-Briones et al. 2018, Tamane, Dey et al. 2020 که شامل استفاده از فناوری‌های اطلاعاتی و ارتباطی به منظور مدیریت زیرساخت‌ها، وسایل نقلیه، کاربران و جریان‌های ترافیکی و حرکتی است که در ارتباط با سایر حالت‌های حمل‌ونقلی می‌باشند (Payne 2015). عادت‌های حرکتی افراد و نحوه استفاده از حالت‌های حمل‌ونقلی و جابجایی بین دو نقطه مطابق با یک چرخه زمانی معین، دارای نظم و الگوی خاصی است.

امروزه با توجه به توسعه شهرها، ساخت، توسعه، مدیریت و استفاده از سیستم‌های حمل‌ونقلی امری ضروری برای کشورهای پیشرفته بوده و نقش کلیدی در زندگی روزمره شهروندان دارد. استفاده از وسایل حمل‌ونقلی در صورت برنامه‌ریزی صحیح می‌تواند به عنوان ابزاری مفید برای نظارت هوشمند مورد استفاده قرار گیرند (Quessada, Pereira et al. 2020). سیستم حمل‌ونقل هوشمند یکی از ارکان موثر شهر هوشمند شمرده می‌شود (Jarašūniene 2007,)

ساختار مقاله حاضر به این شرح است: پس از مقدمه در بخش اول، در بخش دوم به بیان پیشینه تحقیق پرداخته می‌شود. بخش سوم به تعاریف و رویکرد کلی مسئله اختصاص یافته، بخش چهارم پیاده‌سازی و ارزیابی نتایج صورت گرفته است و بخش آخر نیز به نتیجه‌گیری و ارائه پیشنهادها می‌پردازد.

۲- پیشینه تحقیق

مطالعاتی مختلفی در جهت پیش‌بینی حالت‌های حمل‌ونقل انجام پذیرفته و تفاوت این مطالعات در داده‌های مورد استفاده، ویژگی‌های استخراج شده و مدل‌های کلاسه‌بندی می‌باشد. برخی از مطالعات در کنار داده‌های GNSS از اطلاعات مکانی استفاده می‌کنند، به عنوان مثال در (Chung and Shalaby 2005) هر مسیر ثبت شده به وسیله نقاط GNSS با شبکه راه مطابقت داده می‌شود و سپس حالت‌های حمل‌ونقلی بر اساس مشخصات مسیر پیش‌بینی می‌شوند. در (Stenneth, Wolfson et al. 2011) با در نظر گرفتن ایستگاه اتوبوس، خط راه‌آهن و نقاط برخورد ثبت شده داخل اتوبوس، با استفاده از مدل جنگل تصادفی به دقت ۹۳/۵ دست یافته‌اند. استفاده از داده‌های مکانی منجر به بالارفتن دقت نهایی خواهد شد اما این داده‌ها در شهرهای پیشرفته، همواره در حال تغییر بوده و در برخی از مناطق در دسترس نمی‌باشند. تعدادی از مطالعات اطلاعات شخصی مسافران را در کنار نقاط GNSS در نظر می‌گیرند (Stopher, FitzGerald et al. 2008, Bantis and Haworth 2017). این اطلاعات برای پیش‌بینی حالت‌های حمل‌ونقلی مفید هستند اما دستیابی به آنها دشوار و موجب به خطر افتادن حریم خصوصی افراد خواهد شد.

برخی از مطالعات از حسگرهای شتاب در کنار داده‌های GNSS استفاده می‌کنند (Bedogni, Di Felice et al. 2016). در برخی دیگر از ترکیب سنسورهای دیگر تلفن‌های هوشمند (به جز GNSS) استفاده شده است (Su, Caceres et al. 2016). استفاده از این داده‌ها موجب دقت پیش‌بینی بالاتری خواهند شد اما وابستگی به سنسورهای بیشتری وجود خواهد داشت همچنین استفاده از این داده‌ها به محاسبات زیادی نیاز دارد و پردازش برخی از آنها دشوار خواهد بود. ویژگی‌های متفاوتی در مقالات استخراج شده است، به عنوان مثال در (Wang, Wang et al. 2018) با استخراج ۱۱ ویژگی به دقت ۹۰/۶ درصد دست یافته است. در (Dabiri

شناخت این الگوهای پنهان یکی از مهمترین مراحل توسعه و اداره شهرها در حمل‌ونقل هوشمند می‌باشد (Adler and Scheiner and Holz-Rau 2007, Blue 1998). پیشرفت فناوری‌های موقعیت‌یابی و توسعه خدمات مکان‌مبنا^۱ موجب گشته دستگاه‌های هوشمند به بخش جدانشدنی و ضروری در فعالیت‌ها و تعاملات روزمره افراد تبدیل شوند (Chon and Cha 2011, Chatzimilioudis, Konstantinidis et al. 2012, Kitchin 2014). استفاده از داده‌های سیستم‌های تعیین موقعیت ماهواره‌ای GNSS^۲ تولید شده از دستگاه‌های هوشمند، زمینه را برای بسیاری از تحقیقات مبتنی بر مکان فراهم می‌کند، از جمله این موارد شامل: استخراج الگوی فعالیت روزانه افراد (Phithakkitnukoon, Horanont et al. 2010)، ساخت و بروزرسانی شبکه جاده‌ها و مسیر عابر پیاده (Tang, Yang et al. 2018, Yang, Tang et al. 2016, et al.)، نظارت بر حمل‌ونقل (Tang, Kan et al. 2016)، تشخیص کاربری زمین^۳ (Pan, Qi et al. 2012) و برنامه‌ریزی مسیر و توصیه احداث نقاط مورد علاقه (Bao, Zheng et al. 2015, Cui, Luo et al. 2018) می‌باشد. گام اولیه و مهم برای دستیابی به این موارد شناسایی حالت (mode) حمل‌ونقلی افراد خواهد بود. هدف این پژوهش پیاده‌سازی کلاسه‌بندی‌های تقویت‌کننده^۴ و یادگیری عمیق و همچنین مقایسه دقت و زمان پاسخ‌گویی این دو مدل به منظور پیش‌بینی حالت‌های حمل‌ونقلی می‌باشد. در این پژوهش از مجموعه داده GeoLife به عنوان داده‌های GNSS استفاده شده (Zheng, Fu et al. 2011) که حالت‌های حمل‌ونقلی شامل: پیاده‌روی، استفاده از دوچرخه، استفاده از اتومبیل، استفاده از اتوبوس و استفاده از قطار را پیش‌بینی می‌کند. تعداد چهار ویژگی نقطه‌ای کینماتیک شامل: شتاب، سرعت، حرکت^۵، تغییر جهت و چهارده ویژگی آماری سفر برای هر یک از ویژگی‌های نقطه‌ای به همراه سه ویژگی پیشرفته استخراج شده است. به منظور افزایش دقت، ویژگی‌های سفر با اجرای روش ترکیبی انتخاب شده‌اند. پیش‌بینی حالت‌ها بر اساس مدل‌های GB^۶، XGBoost^۷، LightGBM^۸ و CatBoost^۹ که زیرمجموعه‌ای از روش تقویت‌کننده در یادگیری ماشین هستند، به همراه مدل‌های CNN^{۱۰}، LSTM^{۱۱} و ConvLSTM^{۱۱} یادگیری عمیق پیاده‌سازی شده است.

مدل‌های کلاسه‌بندی پیاده‌سازی شده و در پایان مدل‌ها ارزیابی می‌شوند. در ادامه این مراحل تحلیل و بررسی می‌شوند.

۳-۱- پیش‌پردازش داده‌ها

به دلیل امکان اندازه‌گیری و ثبت نادرست ناشی از محدودیت تکنیک‌های تعیین موقعیت، خطای سیستماتیک دستگاه‌های ثبت نقاط و خطاهای انسانی، برای انجام محاسبات صحیح، نیاز به پیش‌پردازش داده خواهد بود (Langley 1997). در فرآیند پیش‌پردازش، ابتدا داده‌های تکراری با زمان ثبت یکسان و مختصات‌های خارج از محدوده حذف می‌شوند سپس حد‌آستانه سرعت و شتاب برای هر حالت حمل‌ونقلی تعیین و تمامی داده‌ها براساس این حد‌آستانه فیلتر می‌شوند. داده‌هایی که فاصله زمانی آن با نقطه بعدی بیش از ۱۵ دقیقه باشد به عنوان یک سفر جداگانه بررسی می‌شوند (Guo, Liang et al. 2020). به دلیل محاسبه جرک، سفرهایی که کمتر از ۳ نقطه داده دارند، حذف می‌شوند و در نهایت با توجه به رویکرد کلاسه‌بندی، سفرها به یک حالت حمل‌ونقلی تقسیم‌بندی خواهند شد. جدول (۱) براساس ویژگی‌های طبیعی موجود در هر یک از حالت‌های حمل‌ونقلی، حد‌آستانه سرعت و شتاب را نمایش می‌دهد (Dabiri and Heaslip 2018).

جدول ۱. مقادیر حد‌آستانه سرعت و شتاب بر اساس حالت‌های

حمل‌ونقل

حالت حمل‌ونقلی	حد‌آستانه سرعت (متر بر ثانیه)	حد‌آستانه شتاب (متر بر مجذور ثانیه)
پیاده‌روی	۷	۳
استفاده از دوچرخه	۱۲	۳
استفاده از اتوبوس	۳۴	۲
استفاده از اتومبیل	۵۰	۱۰
استفاده از قطار	۳۴	۳

۳-۲- استخراج ویژگی

شناسایی حالت‌های حمل‌ونقلی نیازمند استخراج ویژگی از نقاط GNSS خواهد بود. استخراج ویژگی مناسب، رابطه مستقیمی در دقت پیش‌بینی خواهد داشت. ویژگی‌های استخراج شده شامل: ویژگی‌های نقطه‌ای، که با توجه به مختصات نقاط ثبت شده در سری نقاط، محاسبه می‌گردد (Dabiri and Heaslip 2018)؛ ویژگی‌های سفر که شامل عملگرهای آماری می‌باشند تا ویژگی‌های موثر را در طول کل

(and Heaslip 2018) از ۴ ویژگی کینماتیک استفاده شده است. در (Nawaz, Zhiqiu et al. 2020) ویژگی‌های حرکتی سرعت، شتاب، جرک و تغییر جهت و دو ویژگی روزهفته و زمان برش خورده به عنوان ورودی در الگوریتم ConvLSTM استفاده شده است که توانسته با اضافه کردن ویژگی‌ها، دقت را ۳ درصد بهبود بخشد. مورد دیگر پرداخته شده در مقالات استفاده از کلاسه‌بندی‌های متفاوت است. مدل‌هایی شامل: درخت تصمیم (Zheng, Liu et al. 2008)، ماشین بردار پشتیبان (Bolbol, Cheng et al. 2012)، مدل‌های فازی (Jahangiri and Rakha 2014, Das and Winter 2016)، مدل‌های تقویت‌کننده (Wang et al. 2018, Li, Pei et al. 2021) و روش‌های یادگیری عمیق (Endo, Toda et al. 2016, Song, Kanasugi et al. 2016, Vu, Dung et al. 2016, Dabiri and Heaslip 2018, Nawaz, Zhiqiu et al. 2020) به منظور افزایش توانایی شناسایی حالت حمل‌ونقل استفاده شده است. شناسایی حالت‌های حمل‌ونقلی افراد تنها با استفاده از داده‌های GNSS موضوع مهم و ارزشمندی می‌باشد که در این پژوهش از آن استفاده شده است. در بیشتر تحقیقات از داده‌های Geolife به دلیل کیفیت و کمیت بالا استفاده شده (Zheng, Liu et al. 2008, Dabiri and Heaslip 2018, Li, Pei et al. 2021) برخی از مطالعات نیز داده‌ها به صورت مستقل جمع‌آوری شده (Bantis and Haworth 2017) و یا توسط سازمان‌های دولتی منتشر شده‌اند (Biljecki, Ledoux et al. 2013).

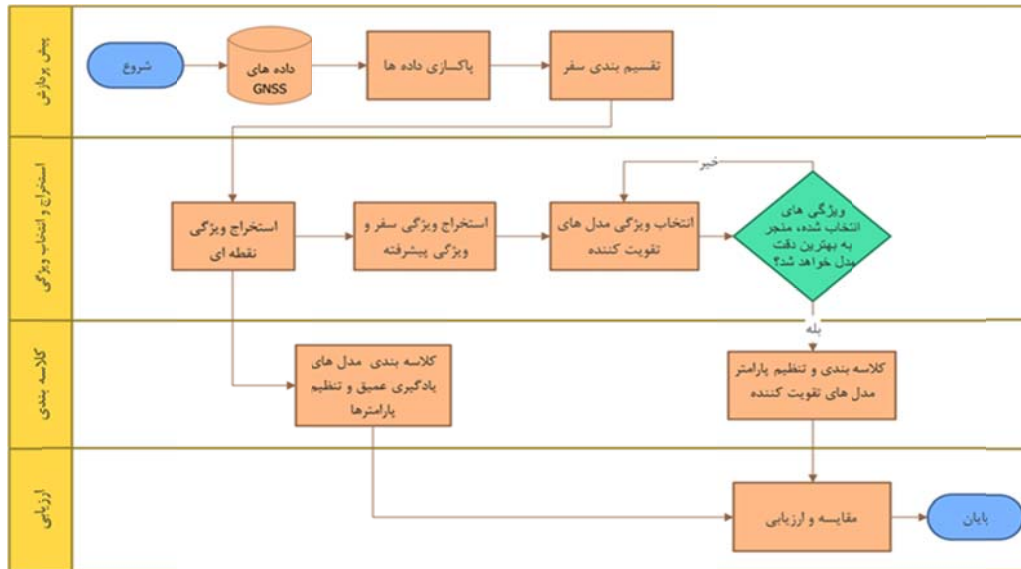
۳-۳- رویکرد مورد استفاده

کلاسه‌بندی‌های تقویت‌کننده نیازمند استخراج ویژگی‌های سفر و پیشرفته می‌باشند، این ویژگی‌ها توسط کارشناس به صورت دستی انتخاب می‌شوند. کلاسه‌بندی‌های یادگیری عمیق، نیاز به نظر کارشناس نداشته و ویژگی‌های عمیق را از داده‌های ورودی استخراج می‌کند. در این پژوهش، هر دو روش پیاده‌سازی شده تا بتوان دقت‌های پیش‌بینی بدست آمده را مقایسه کرد. رویکرد شناسایی حالت‌های حمل‌ونقلی شامل چهار گام پردازشی می‌باشد. مطابق با فلوچارت شکل (۱) داده‌های GNSS پیش‌پردازش شده و ویژگی‌های مؤثر استخراج و در مدل‌های تقویت‌کننده انتخاب می‌شوند. در ادامه

۳-۳- انتخاب ویژگی

انتخاب ویژگی به معنای استفاده از آزمون‌های آماری به منظور تعیین ویژگی‌هایی است که موجب آموزش سریع، کاهش پیچیدگی مدل، جلوگیری از بیش‌برازش و بهترین پیش‌بینی گردد. در این پژوهش انتخاب ویژگی در مدل‌های کلاسه‌بندی تقویت‌کننده پیاده‌سازی شده که از یک روش ترکیبی شامل روش برمبنای فیلتر^{۱۲} و روش برمبنای بسته-بندی^{۱۳} استفاده شده است (Venkatesh and Anuradha, 2019).

سفر با استفاده از ویژگی‌های نقطه‌ای استخراج می‌کند (Xiao, Wang et al. 2017) و ویژگی‌های پیشرفته که به منظور مقابله با شرایط ترافیکی و آب‌وهوایی با در نظر گرفتن حدآستانه از ویژگی‌های نقطه‌ای استخراج می‌شود (Zheng, Li et al. 2008)، می‌باشند. جدول (۲) ویژگی‌های استخراج شده در این پژوهش را همراه با معادله و توضیح هر یک از پارامترها نمایش می‌دهد. مقادیر حدآستانه ویژگی‌های پیشرفته، نرخ تغییر جهت، نرخ توقف و نرخ تغییر سرعت، به ترتیب ۱۹، ۳/۴ و ۰/۲۶ در نظر گرفته شده‌اند (Zheng, Li et al., 2008).



شکل ۱. فلوچارت رویکرد حل مسئله

جدول ۲. ویژگی‌های استخراج شده

نوع	نام	معادله	توضیحات
ویژگی های نقطه‌ای (Dabiri and Heaslip 2018)	سرعت	$V_{p_1} = \frac{D(P_1, P_2)}{\Delta T}$	V_{p_1} بیانگر سرعت در نقطه P_1 ، $D(P_1, P_2)$ معادل فاصله ژئودتیک بین دو نقطه P_1 و P_2 و ΔT زمان طی شده بین دو نقطه بر حسب ثانیه می‌باشد.
	شتاب	$A_{p_1} = \frac{V_{p_2} - V_{p_1}}{\Delta T}$	A_{p_1} بیانگر شتاب در نقطه P_1 ، V_{p_1} سرعت در نقطه P_1 ، V_{p_2} سرعت در نقطه P_2 و ΔT زمان طی شده بین دو نقطه می‌باشد.
	جرک	$J_{p_1} = \frac{A_{p_2} - A_{p_1}}{\Delta T}$	J_{p_1} بیانگر جرک در نقطه P_1 ، V_{p_1} سرعت در نقطه P_1 ، V_{p_2} سرعت در نقطه P_2 و ΔT زمان طی شده بین دو نقطه می‌باشد.
	تغییر جهت	$B_{(P_i)} = \text{Arctan}(\frac{y_i}{x_i})$ $B_Rate_{(P_i)} = B_{P_2} - B_{P_1} $	$B_{(P_i)}$ بیانگر تغییر جهت در نقطه P_1 و $B_{(P_i)}$ جهت نقطه P_1 را نشان می‌دهد.

تعداد نمونه‌ها و X_i مقدار نمونه‌ها می‌باشد.	$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}$	میانگین	ویژگی‌های سفر (Xiao, Wang et al. 2017)
تعداد نمونه‌ها و i شماره ایندکس نمونه می‌باشد.	$X_i \rightarrow \begin{cases} i = \frac{n}{2} & \text{if } n \text{ is even} \\ i = \frac{(n+1)}{2} & \text{if } n \text{ is odd} \end{cases}$	مد	
تعداد نمونه‌ها و μ میانگین داده‌ها می‌باشد.	$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \mu)^2}{n}}$	انحراف معیار	
بیشترین مقداری که در داده‌ها یافت می‌شود.	---	بیشینه مقدار	
کمترین مقداری که در داده‌ها یافت می‌شود.	---	کمینه مقدار	
بیانگر تفاضل بیشینه مقدار و کمینه مقدار می‌باشد.	---	محدوده مقدار	
تعداد نمونه‌ها، i شماره ایندکس، Q_{10} بیانگر ۱۰ امین صدک، Q_{25} ۲۵ امین صدک، Q_{75} ۷۵ امین صدک و Q_{90} ۹۰ امین صدک می‌باشد.	$X_i \rightarrow \begin{cases} i = \frac{1}{10}(n+1) \rightarrow Q_{10} \\ i = \frac{1}{4}(n+1) \rightarrow Q_{25} \\ i = \frac{3}{4}(n+1) \rightarrow Q_{75} \\ i = \frac{9}{10}(n+1) \rightarrow Q_{90} \end{cases}$	صدک	
Q_{25} بیانگر ۲۵ امین صدک و Q_{75} ، ۷۵ امین صدک می‌باشد.	$Q_{3,1} = Q_{75} - Q_{25}$	محدوده بین چارکی	
μ بیانگر میانگین و σ انحراف معیار داده‌ها می‌باشد.	$S = E \left[\left(\frac{X - \mu}{\sigma} \right)^3 \right]$	چولگی	
μ بیانگر میانگین و σ انحراف معیار داده‌ها می‌باشد.	$K = E \left[\left(\frac{X - \mu}{\sigma} \right)^4 \right] - 3$	کشیدگی	
μ بیانگر میانگین و σ انحراف معیار داده‌ها می‌باشد.	$V = \frac{\sigma}{\mu}$	ضریب تغییرات	ویژگی‌های پیشرفته (Zheng, Li et al. 2008)
$ P_c $ بیانگر تعداد دفعاتی است که مقدار جهت نقاط GNSS از حد آستانه تعیین شده بیشتر باشد.	$HCR = \frac{ P_c }{Distance}$	نرخ تغییر جهت	
$P_s = [p_i p_i \in P, V(p_i) < V_s]$ و V_s حد آستانه تعیین شده این ویژگی می‌باشد.	$SR = \frac{ P_s }{Distance}$	نرخ توقف	
$P_v = [p_i p_i \in P, VR(p_i) > V_r]$ و V_r حد آستانه تعیین شده این ویژگی می‌باشد.	$VCR = \frac{ P_v }{Distance}$ $VR(p_i) = \frac{ V_{p_2} - V_{p_1} }{V_{p_1}}$	نرخ تغییر سرعت	

۳-۳-۱- انتخاب ویژگی بر مبنای فیلتر

همچنین برای حذف ویژگی‌های نیمه ثابت مقدار حد آستانه واریانس ۰/۰۱ در نظر گرفته شده است. ویژگی‌هایی که دارای همبستگی زیادی با ویژگی‌های دیگر هستند علاوه بر تاثیر نداشتن در دقت مدل نهایی، موجب افزایش پیچیدگی مدل یادگیری خواهند شد به همین دلیل پس از تشکیل ماتریس همبستگی، ویژگی‌هایی که مقدار ضریب همبستگی آن‌ها در

در این روش، ویژگی‌هایی که دارای اهمیت کمتری هستند از مجموعه ویژگی‌ها فیلتر و حذف می‌شوند. هنگامی که داده‌های یک ویژگی بیش از حد به هم شبیه باشند، دقت پیش‌بینی کاهش پیدا می‌کند. به همین منظور ویژگی‌های ثابت و نیمه ثابت با تعیین حد آستانه واریانس، از داده‌ها حذف خواهند شد. با توجه به معادله (۱) اگر مقادیر مجموعه داده‌های یک ویژگی باهم برابر باشند، واریانس آن صفر محاسبه می‌گردد.

همچنین از step decay schedule طبق معادله (۲) به منظور محاسبه نرخ یادگیری (Ge, Kakade et al. 2019) به همراه روش توقف زودهنگام با استفاده از داده‌های ارزیابی برای جلوگیری از بیش‌برازش استفاده شده است (Caruana, Lawrence et al. 2001).

$$LR = LR_0 \times DF^{\text{epoch}/S} \quad (2)$$

در معادله (۲)، LR_0 بیانگر نرخ اولیه یادگیری، DF مقدار کاهش، $epoch$ تعداد تکرار و S تعداد گام کاهشی می‌باشد. در ادامه تعریف مختصری از مدل‌های کلاسه‌بندی ارائه خواهد شد.

۳-۴-۱- مدل GB

در این روش مدل‌های ضعیف با استفاده از درختان تصمیم ساخته می‌شوند و به صورت تقویت‌کننده و محاسبه تابع هزینه به شکل تکراری موجب پیش‌بینی می‌گردد. این مدل سعی در کاهش خطا (بایاس) دارد. معادلات (۳) نشان‌دهنده نحوه محاسبه این مدل می‌باشد (Friedman 2001).

$$F_0 = \operatorname{argmin} \sum_{i=1}^n L(y_i, \gamma) \quad (3)$$

$$L(y_i, \gamma) = -\log(\text{likelihood}) = -[y_i \log(P) + (1 - y_i) \log(1 - P)]$$

$$\frac{d}{d \log(\text{odds})} = -y_i + \frac{e^{\log(\text{odds})}}{1 + e^{\log(\text{odds})}} = -\text{observed} + \text{Predicted}$$

$$F_m(x) = F_{m-1}(x) + v \sum_{j=1}^m \gamma_{jm} I(x \in R_{jm})$$

$$\gamma = \frac{\text{Residual}}{P(1-P)}$$

در معادلات (۳)، L بیانگر تابع هزینه، y_i برچسب مشاهدات، x بیانگر ویژگی‌ها، P احتمال رخداد، v نرخ یادگیری، γ تبدیل چند احتمال یک برگ به یک احتمال، F_0 برآوردگر احتمال مرحله اول و $F_m(x)$ برآوردگر احتمال بقیه مراحل به جز مرحله اول می‌باشد.

۳-۴-۲- مدل XGBoost

این مدل یک روش بهبود یافته مدل GB می‌باشد که می‌تواند با سرعت و دقت بالایی پیش‌بینی را انجام دهد. این روش با اضافه کردن مقدار منظم‌سازی^{۱۷} موجب کاهش بیش‌برازش خواهد شد. معادله (۴) نحوه محاسبه تابع هزینه را نشان می‌دهد (Chen and Guestrin 2016). ترم اول این معادله بیانگر تابع هزینه و ترم دوم منظم‌سازی می‌باشد. T نشان‌دهنده تعداد برگ‌های درخت تصمیم و گاما پهنالتی

ماتریس بیش از ۰/۹ می‌باشند نشان‌دهنده همبستگی بسیار زیاد خواهد بود و از مجموعه ویژگی‌ها حذف می‌شوند (Jović, Brkić et al. 2015).

$$\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu)^2}{N} \quad (1)$$

در معادله (۱) N تعداد نمونه‌ها و μ میانگین داده‌ها می‌باشد.

۳-۳-۲- انتخاب ویژگی بر مبنای بسته‌بندی

این فرآیند یک روش جستجوی روبه‌جلو^{۱۴} می‌باشد که ویژگی‌ها را با توجه به برچسب داده‌ها تنظیم کرده و ویژگی‌هایی که موجب دقت بهینه می‌شوند، انتخاب می‌کند (Jović, Brkić et al. 2015). در این پژوهش پس از انتخاب هر ویژگی، دقت آن با توجه به مدل کلاسه‌بندی انتخابی بررسی شده و ویژگی که بیشترین دقت را کسب کرده، انتخاب می‌شود. ویژگی‌های بعدی شامل ویژگی‌های انتخاب شده قبلی نیز می‌باشد. به منظور بالا بردن دقت محاسبات، از روش K-Fold Cross Validation استفاده شده است، در این روش داده‌های آموزشی به K تعداد تقسیم و بسته‌بندی می‌شوند و فرآیند مورد محاسبه K بار تکرار می‌شود، در هر تکرار یک بسته به عنوان داده اعتبارسنجی انتخاب شده و محاسبه دقت انجام می‌پذیرد، در انتها دقت نهایی برابر میانگین K تکرار می‌باشد (Han, Kamber et al. 2011).

۳-۴-۳- کلاسه‌بندی

کلاسه‌بندی یکی از مراحل اساسی پیش‌بینی حالت‌های حمل‌ونقلی می‌باشد که به معنای آموزش مدلی برای جداسازی داده‌ها بر اساس برچسب آنها در کلاس‌هایی است که قابلیت پیش‌بینی داده‌های بدون برچسب را داشته باشد. کلاس‌های خروجی در این پژوهش شامل: پیاده‌روی، استفاده از دوچرخه، استفاده از اتومبیل، استفاده از قطار و استفاده از اتوبوس می‌باشد. چهار مدل کلاسه‌بندی تقویت‌کننده زیرمجموعه الگوریتم یادگیری ماشین شامل GB, XGBoost, CatBoost, LightGBM و سه مدل یادگیری عمیق شامل CNN, LSTM, ConvLSTM پیاده‌سازی شده است. در کلاسه‌بندی یادگیری عمیق از آنتروپی متقابل رسته‌ای^{۱۵} به عنوان تابع هزینه^{۱۶} و بهینه‌ساز Adam به منظور بروزرسانی پارامترها استفاده شده است (Kingma and Ba 2014)،

در معادله (۶) بردار داده‌ها، g_i مقدار گرادیان منفی محاسبه شده از تابع هزینه و $A_l = [x_i \in A : x_{ij} \leq d]$ و $B_l = [x_i \in B : x_{ij} \leq d]$ ، $A_r = [x_i \in A : x_{ij} > d]$ و $B_r = [x_i \in B : x_{ij} > d]$ می‌باشند.

با استفاده از الگوریتم حریم‌ناهن EFB می‌توان تعداد ویژگی‌ها را بدون آسیب رساندن به دقت کاهش داد. داده‌های با ابعاد بالا معمولاً به شکل پراکنده می‌باشند و مقادیر غیرصفر را در طول سطرها با هم در نظر نمی‌گیرند، به همین منظور از متد EFB استفاده می‌شود تا بتوان ابعاد ویژگی‌ها را کاهش داد و باندهایی از تلفیق ویژگی‌ها ساخت. در EFB از گراف‌های رنگی برای یافتن ویژگی‌هایی که قابلیت تلفیق دارند استفاده می‌شود و با توجه به هیستوگرام ویژگی‌ها، باندها تشکیل می‌شود (Ke, Meng et al. 2017).

۳-۴-۴- مدل CatBoost

یک الگوریتم ابتکاری که در آن ویژگی‌های رسته‌ای^{۲۰} به طور خودکار به جای پیش‌پردازش، در طول آموزش به عدد تبدیل می‌شوند. در این مدل درختان تصمیم به شکل متقارن ساخته می‌شود که منجر به پیش‌بینی سریع خواهد شد (Prokhorenkova, Gusev et al. 2017).

یکی از روش‌های تبدیل ویژگی‌های رسته‌ای، استفاده از Greedy TBS^{۲۱} می‌باشد؛ در این روش از میانگین مقدار برچسب و استفاده از آن در یافتن نقطه تقسیم درخت تصمیم استفاده می‌شود. در مدل CatBoost یک مقدار اولیه^{۲۲} به Greedy TBS افزوده و با استفاده از یک جایگشت تصادفی در داده‌ها، میانگین مقدار برچسب برای هر داده p مطابق معادله (۷) جایگزین می‌شود.

$$x_{\sigma_p, k} = \frac{\sum_{j=1}^{p-1} (x_{\sigma_j, k} = x_{\sigma_p, k}) \times y_{\sigma_j} + a \times p}{\sum_{j=1}^{p-1} (x_{\sigma_j, k} = x_{\sigma_p, k}) + a} \quad (7)$$

داده‌ها به صورت $D = [(X_i, Y_i)]_{i=1 \dots n}$ می‌باشند که در آن $X_i = (x_{i,1}, \dots, x_{i,m})$ یک بردار با m ویژگی است و $Y_i \in R$ برچسب داده‌ها می‌باشد. جایگشت‌ها به شکل $\sigma = (\sigma_1, \dots, \sigma_n)$ ویژگی‌های رسته‌ای با K مقدار اولیه با P نشان داده شده‌اند و پارامتر $a > 0$ وزن مقدار اولیه می‌باشد. برای اینکه بتوان نتیجه بهینه‌ای یافت می‌توان در هر مرحله P را تغییر داد. در (Dorogush, Ershov et al. 2018) از Ordered TBS برای CatBoost استفاده شده است. اگر مدل‌ها از داده‌آموزشی یکسان برای پیش‌بینی استفاده

تعریف شده توسط کاربر برای هرس درخت است. با حل بسط تیلور تا مرحله دوم، تابع هزینه مطابق معادله (۵) محاسبه می‌شود.

$$L(y_i, P_i) = [\sum_{i=1}^n L(y_i, P_i^m)] + \sum_k \Omega(f_k) \quad (8)$$

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|w\|^2$$

$$L^t(q) = -\frac{1}{2} \sum_{j=1}^T \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \quad (9)$$

در معادله (۵)، g_i بیانگر مشتق اول تابع هزینه و h_i مشتق دوم تابع هزینه می‌باشد.

۳-۴-۳- مدل LightGBM

این مدل یکی از انواع GB است که با استفاده از هیستوگرام داده‌ها می‌تواند سرعت پیش‌بینی را ۲۰ برابر به نسبت مدل سنتی افزایش دهد. هزینه اصلی در مدل‌های GB آموزش درخت تصمیم می‌باشد که بیشترین زمان صرف یافتن بهترین نقطه تقسیم می‌شود. زمانی که ابعاد ویژگی‌ها و داده‌ها افزایش می‌یابند، محاسبه *information gain* برای یافتن نقطه تقسیم مشکل است و موجب می‌گردد تاثیرگذاری و مقیاس پذیری کاهش یابد. استفاده از دو تکنیک GOSS^{۱۸} و EFB^{۱۹} در کنار مدل GB، LightGBM نامیده می‌شود. در این مدل، داده‌هایی که گرادیان بیشتری دارند نقش مهمی در ساخت درخت تصمیم ایفا می‌کنند. در GOSS بخش قابل توجهی از نمونه‌ها که گرادیان بالایی دارند به همراه نمونه‌برداری تصادفی از گرادیان‌های کوچک به تخمین *information gain* می‌پردازد. این مدل به دلیل استفاده کم از حافظه برای اجرا می‌تواند حجم زیادی از داده‌ها را کنترل کند. به این منظور ابتدا داده‌ها بر اساس قدرمطلق گرادیان محاسبه شده، مرتب می‌شوند؛ $100\% \times a$ داده‌هایی که بیشترین مقدار را دارند نمونه‌برداری می‌شوند و در زیرمجموعه A نگهداری می‌شوند؛ به شکل تصادفی $100\% \times b$ داده‌های باقی‌مانده نمونه‌برداری می‌شوند و در زیرمجموعه B نگهداری می‌شوند و در نهایت نمونه‌هایی که به شکل تصادفی نمونه‌برداری شده‌اند با ثابت $\frac{1-a}{b}$ تقویت می‌شود. مقدار تخمینی *variance gain* مطابق معادله (۶) محاسبه می‌شود (Ke, Meng et al. 2017).

$$\tilde{V}_j(d) = \frac{1}{n} \left(\frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b} \sum_{x_i \in B_l} g_i)^2}{n_l^j(d)} + \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b} \sum_{x_i \in B_r} g_i)^2}{n_r^j(d)} \right) \quad (6)$$

لایه تابع فعال‌سازی: نورون‌ها با یک تابع فعال‌سازی همراه هستند. در این مقاله از تابع فعال‌سازی ReLU^{۲۴} استفاده شده است (Krizhevsky, Sutskever et al. 2017).

۳. لایه Pooling: هدف از ساخت این لایه، کنترل بیش‌برازش و محاسبات کمتر می‌باشد (Scherer, Müller et al. 2010). در این پژوهش از Max pooling استفاده شده است که با توجه به اندازه فیلتر و گام مشخص شده، بیشینه مقدار فیلتر را در نظر می‌گیرد.

۴. لایه Fully-Connected: این لایه از نورون‌های کاملاً متصل تشکیل شده‌اند که نقش استخراج ویژگی را خواهند داشت. به دلیل چندکلاسه بودن مسئله، تابع فعال‌سازی لایه آخر Softmax می‌باشد که تعداد نورون‌ها در این لایه به اندازه تعداد کلاس‌ها تعریف می‌شود (Nwankpa, Ijomah et al. 2018).

۳-۶-۴- LSTM مدل

این مدل یک شبکه عصبی بازگشتی^{۲۵} به شمار می‌آید که در پردازش داده‌های زمانی و ترتیبی استفاده می‌شوند. مدل‌های RNN می‌تواند ورودی قبلی را در حافظه داخلی به خاطر سپرده و از آن برای پردازش دنباله ورودی‌ها استفاده نماید. به دلیل دو مشکل Vanishing و Exploding Gradient در مدل شبکه عصبی بازگشتی، توانایی ثبت وابستگی زمانی برای مدت طولانی وجود نخواهد داشت (Bengio, Frasconi et al. 1993, Hochreiter 1998). برای مقابله با این موضوع از معماری LSTM استفاده می‌شود. در معماری LSTM، سه دروازه شامل: دروازه فراموشی^{۲۶} f_t ، دروازه ورودی^{۲۷} یا بروزسانی i_t و دروازه خروجی^{۲۸} o_t به کار گرفته می‌شود. شکل (۳) و معادلات (۸) نشان‌دهنده مراحل انجام گرفته در مدل می‌باشد (Gers, Schmidhuber et al. 1999).

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (۸)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f)$$

$$\tilde{C}_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o)$$

$$h_t = o_t \times \tanh(C_t)$$

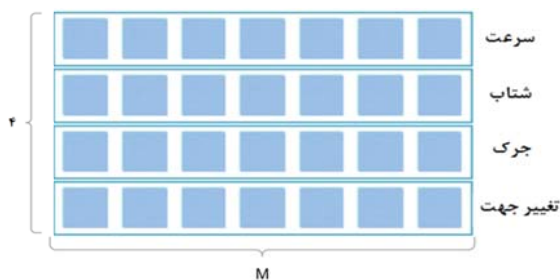
در این معادله h_{t-1} لایه پنهان مرحله قبل، x_t ورودی، W وزن، b بایاس، σ و \tanh تابع فعال‌سازی سیگموئید و هایپربولیک، C_{t-1} ورودی از مرحله قبلی، f_t دروازه

کنند، موجب Prediction Shift در پیش‌بینی‌ها می‌گردد که برای مقابله با این موضوع از Ordered Boosting استفاده شده است. در این روش داده‌های هر درخت تصمیم، مخلوط می‌شود. این مدل توانایی برخورد با داده‌های غیررسته‌ای را نیز داراست.

۳-۴-۵- مدل CNN

مدل CNN یک شبکه پیش‌خور^{۲۹} می‌باشد و از پرسپترون‌های چندلایه ساخته شده که توانایی ساخت ویژگی‌های عمیق را خواهند داشت (Simonyan and Zisserman 2014). این مدل توانایی آموختن و استخراج ویژگی‌های مکانی را داراست. در این پژوهش از ۴ لایه بر اساس (Dabiri and Heaslip 2018) استفاده شده است که در ادامه کار هر یک توضیح داده شده‌است.

۱. لایه ورودی: شامل مجموعه‌ای از نمونه‌های مستقل می‌باشد که هر نمونه یک بخش GNSS با چهار کانال شامل: سرعت، شتاب، جبرک و تغییر جهت می‌باشد. ابعاد ورودی، یک چندتایی با اندازه $M \times 4$ می‌باشد. هر سفر تعداد نقاط متفاوتی را در خود ذخیره کرده، به همین دلیل مقدار M متفاوت خواهد بود. ابعاد ورودی در یادگیری عمیق برای همه ورودی‌ها یکسان بوده و M ثابت در نظر گرفته می‌شود به این منظور سفرهایی که نقاط زیادی را دارا هستند، به تعداد M محدود می‌شوند و آنهایی که نقاط کمتر از M دارند، با مقدار صفر پر می‌شوند. شکل (۲) ابعاد لایه ورودی را نشان می‌دهد. مقدار M برابر با میانه تعداد نقاط موجود در سفرها انتخاب می‌شود (Dabiri and Heaslip 2018).



شکل ۲. ابعاد ورودی داده‌ها

۲. لایه Convolutional: این لایه شامل مجموعه‌ای از فیلترهای قابل یادگیری می‌باشد که با گام مشخصی در طول داده‌ها حرکت می‌کنند. در این پژوهش حالت Same Padding فعال است تا ابعاد فضای ورودی لایه تغییر نکند.

معادله (۱۲) استفاده شده است (Han, Kamber et al. 2017, Novaković, Veljović et al. 2011).

$$P = \frac{TP}{TP + FP} \quad (10)$$

$$R = \frac{TP}{TP + FN} \quad (11)$$

$$F_{Score} = \frac{2PR}{P + R} \quad (12)$$

در ماتریس پیچیدگی رسم شده در جدول (۳)، مثبت درست نشان‌دهنده تعداد مواردی است که نمونه عضو کلاس مثبت باشد و به درستی عضو همین کلاس تشخیص داده شود، منفی نادرست تعداد مواردی است که نمونه عضو کلاس مثبت باشد و به اشتباه عضو کلاس منفی تشخیص داده شود، مثبت نادرست تعداد مواردی است که نمونه عضو کلاس منفی باشد و به اشتباه عضو کلاس مثبت تشخیص داده شود و در نهایت منفی درست تعداد مواردی است که نمونه عضو کلاس منفی باشد و به درستی عضو همین کلاس تشخیص داده شود.

جدول ۳. ماتریس پیچیدگی

کلاس پیش‌بینی شده			کلاس واقعی
پیش‌بینی منفی	پیش‌بینی مثبت		
منفی نادرست (FN)	مثبت درست (TP)	واقعا مثبت	
منفی درست (TN)	مثبت نادرست (FP)	واقعا منفی	

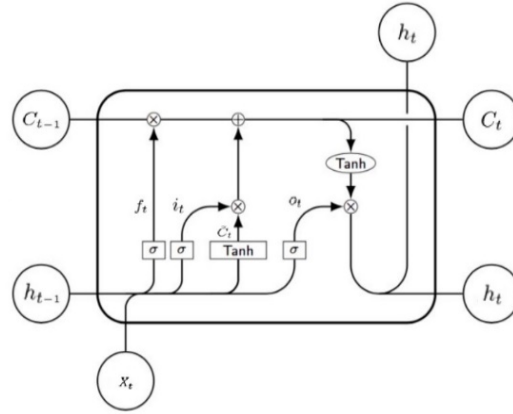
از آنجایی که تعداد کلاس‌های مورد پیش‌بینی بیش از دو کلاس می‌باشد، می‌بایست نتایج هر کلاس میانگین‌گیری شود. به همین دلیل از میانگین خرد (micro) طبق معادلات (۱۳) استفاده شده است (Lipton, Elkan et al. 2014).

$$P_{micro} = \frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n TP_i + \sum_{i=1}^n FP_i} \quad (13)$$

$$R_{micro} = \frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n TP_i + \sum_{i=1}^n FN_i}$$

$$F_{Score}(micro) = \frac{2P_{micro}R_{micro}}{P_{micro} + R_{micro}}$$

فراموشی، i_t دروازه ورودی، \tilde{C}_t خروجی دو دروازه فراموشی و ورودی، C_t آپدیت حالت سلول o_t ، o_t دروازه خروجی و h_t خروجی لایه پنهان می‌باشد.



شکل ۳. معماری مدل LSTM

۳-۴-۷- مدل ConvLSTM

معماری این مدل به جای ضرب ماتریسی موجود در LSTM، از عملیات Convolution در هر دو حالت، سلول به سلول و ورودی به سلول استفاده می‌کند که قادر است ویژگی‌های مکانی-زمانی را استخراج کند. معادلات (۹) نمایانگر مراحل انجام این مدل می‌باشد (Shi, Chen et al. 2015).

$$i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + W_{ci} * C_{t-1} + b_i) \quad (9)$$

$$f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + W_{cf} * C_{t-1} + b_f)$$

$$\tilde{C}_t = \tanh(W_{xc} * x_t + W_{hc} * h_{t-1} + b_c)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + W_{co} * C_t + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

پارامترهای این معادلات به مانند مدل LSTM می‌باشد با این تفاوت که (*) نشان دهنده عملیات Convolution و (x) ضرب درایه‌ای را نشان می‌دهد.

۳-۵- ارزیابی

به منظور ارزیابی مدل‌های کلاسه‌بندی از روش‌هایی شامل ماتریس پیچیدگی^{۳۱} مطابق با جدول (۳)، Precision مطابق معادله (۱۰)، Recall مطابق معادله (۱۱) و FScore مطابق

۴- پیاده‌سازی

داده‌های مورد استفاده در این مقاله مجموعه داده GeoLife می‌باشد، این مجموعه داده مسیره‌های GNSS را توسط ۱۸۲ کاربر به مدت بیش از ۵ سال اندازه‌گیری و ذخیره کرده است. این داده‌ها حاوی اطلاعات طول و عرض جغرافیایی، ارتفاع و زمان ثبت می‌باشند. این مجموعه داده توسط دستگاه‌های متنوع ثبت مختصات و تلفن‌های همراه هوشمند ذخیره شده‌اند. ۹۱/۵ درصد از مسیره‌ها شامل نقاط مترامی هستند که فاصله ثبت ۱ تا ۵ ثانیه یا ۵ تا ۱۰ متر به ازای هر نقطه را دارا می‌باشند. این مجموعه داده در بیش از ۳۰ شهر چین و در برخی از شهرهای واقع در ایالات متحده آمریکا و اروپا توزیع شده‌اند. حجم بیشتر این داده‌ها در پکن چین ثبت شده‌اند. در این مجموعه داده از بین تمامی کاربران، ۶۹ کاربر داده‌های خود را برچسب‌گذاری کرده‌اند.

۴-۱- پیش‌پردازش داده‌ها

پس از انجام پیش‌پردازش داده‌ها، ۹۹۷۰ سفر بر چسب‌گذاری به صورت نامتوازن^{۳۱} ثبت شده است که شامل: ۴۵۶۷ سفر پیاده‌روی، ۱۶۸۳ سفر استفاده از دوچرخه، ۲۰۱۱ سفر استفاده از اتوبوس، ۱۴۶۴ سفر استفاده از اتومبیل و ۲۴۵ سفر استفاده از قطار می‌باشد.

۴-۲- استخراج ویژگی

تعداد چهار ویژگی نقطه‌ای کینماتیک و چهارده ویژگی سفر برای هر یک از ویژگی‌های نقطه‌ای استخراج شده است که در مجموع موجب استخراج پنجاه‌وشش ویژگی سفر شده است. سه ویژگی پیشرفته نیز به منظور بالابردن دقت پیش‌بینی در شرایط ترافیکی و آب‌وهوای بارانی پیاده‌سازی و استخراج شده است. جدول (۴) نمونه ویژگی‌های نقطه‌ای استخراج شده برای حالت استفاده از دوچرخه را نمایش می‌دهد.

۴-۳- انتخاب ویژگی

برای هر روش کلاسه‌بندی تقویت‌کننده ذکر شده، انتخاب ویژگی به صورت جداگانه انجام پذیرفته است و تعداد ویژگی‌های ورودی مدل آمده است. مقدار ورودی پارامترهای کلاسه‌بندی در این مرحله، مقدار پیش فرض کتابخانه Scikit-learn در نظر گرفته شده است. مقادیر دقت در این مرحله با روش 10_Fold Cross Validation محاسبه شده‌اند. پس از پیاده‌سازی این روش تعداد ۲۶ ویژگی برای مدل GB، ۲۱ ویژگی برای مدل XGBoost، ۲۳ ویژگی برای مدل LightGBM و ۳۰ ویژگی برای مدل CatBoost انتخاب شده‌اند.

جدول ۴. نمونه ویژگی‌های نقطه‌ای

ID	زمان ثبت	عرض جغرافیایی	طول جغرافیایی	سرعت (m/s)	شتاب (m/s ²)	جرک (m/s ³)	تغییر جهت (درجه)
۱	۱۰-۰۶-۲۰۰۸ ۱۷:۲۸:۳۸	۳۹/۹۷۶۵۸۲	۱۱۶/۳۲۹۴۹۱	۲/۳۳۷	-۰/۸۹۴۵	۱/۸۲۴۷۴	۱۷۵/۸۲۵
۲	۱۰-۰۶-۲۰۰۸ ۱۷:۲۸:۳۹	۳۹/۹۷۶۵۶۱	۱۱۶/۳۲۹۴۸۹	۱/۴۴۳	۰/۹۳۰۲	۰/۶۳۱۷۹	۴۸/۹۷۷
۳	۱۰-۰۶-۲۰۰۸ ۱۷:۲۸:۴۰	۳۹/۹۷۶۵۷۴	۱۱۶/۳۲۹۴۸۹	۲/۳۷۳	۱/۵۶۲۰	-۳/۰۵۲۶۵	۴۴/۲۶۹
۴	۱۰-۰۶-۲۰۰۸ ۱۷:۲۸:۴۱	۳۹/۹۷۶۵۸۸	۱۱۶/۳۲۹۴۶۸	۳/۹۳۵	-۱/۴۹۰۶	۱/۵۸۲۱۱	۵۷/۴۴۷

۴-۴-۴- کلاسه‌بندی مدل‌های تقویت‌کننده

اعتبارسنجی و واریانس هریک از پارامترها محاسبه می‌شوند. مقادیری که دارای دقت آموزشی و اعتبارسنجی بالا و واریانس کمتری هستند انتخاب و با استفاده از یک پنجره جستجو گزیده تمامی پارامترها با هم در نظر گرفته و بهترین مقادیر پیاده‌سازی استفاده می‌شوند. مقادیر ابرپارامترهای شامل تعداد ۹۰۰ درخت، نرخ یادگیری برابر ۰/۱، کمترین تعداد تقسیم درخت برابر ۲، کمترین تعداد برگ درخت برابر ۳ و بیشترین عمق برابر ۱۰ انتخاب شدند که با توجه به ماتریس پیچیدگی جدول (۵)، مقدار میانگین F_{Score} ۹۴/۸۸ درصد محاسبه شد.

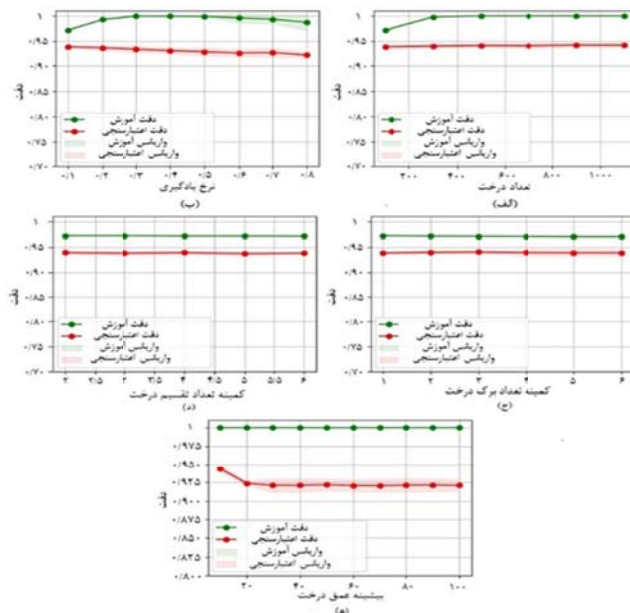
۴-۴-۲- مدل XGBoost

در این مدل، مقادیر پارامترهای تعداد درخت، نرخ یادگیری، کمترین وزن برگ درخت، بیشینه عمق درخت، γ و مقدار منظم سازی λ تنظیم شده‌اند. مقادیر دقت آموزشی، دقت اعتبارسنجی و واریانس هر یک از پارامترها به مانند مدل GB محاسبه و با استفاده از یک پنجره جستجو گزیده مقادیر ۲۰۰ درخت، نرخ یادگیری برابر ۰/۱، کمترین وزن برگ برابر ۲، بیشترین عمق برابر ۲۵، گاما برابر ۰/۵ و منظم‌سازی لامبدا برابر ۰/۹ بالاترین دقت را کسب کرده‌اند. با توجه به ماتریس پیچیدگی جدول (۶) مقدار میانگین F_{Score} برابر با ۹۵/۱۸ درصد محاسبه شده است.

در مدل‌های پیاده‌سازی شده، ۸۰ درصد داده‌ها تحت عنوان داده‌های آموزشی و ۲۰ درصد باقی‌مانده به عنوان داده‌های تست در نظر گرفته شده‌اند. به دلیل نامتعادل بودن برچسب‌ها از یک نمونه‌برداری طبقه‌ای استفاده شده تا بتوان از همه برچسب‌ها در آموزش و تست استفاده کرد. در مدل‌های تقویت‌کننده، بازه‌ای به جهت تنظیم هر یک از پارامترها در نظر گرفته شده است، با ثابت در نظر گرفتن پارامترهای دیگر و استفاده از روش 10-Fold Cross Validation، مقادیر مناسب پارامترها انتخاب می‌شوند سپس با استفاده از یک پنجره جستجو گزیده تمامی مقادیر پارامترها با هم در نظر گرفته شده و با استفاده از روش 10-Fold Cross Validation، بهترین مقادیر پارامترها که دقت بهینه را منجر می‌شود برای پیش‌بینی در نظر گرفته خواهد شد. کلاسه‌بندی‌ها با توجه به کتابخانه‌های منبع باز xgboost، Scikit-learn، CatBoost و LightGBM در زبان برنامه‌نویسی پایتون پیاده‌سازی شده‌اند. نتایج در ادامه بررسی شده‌اند:

۴-۴-۱- مدل GB

در مدل GB، مقادیر پارامترهای تعداد درخت، نرخ یادگیری، کمینه تعداد تقسیم‌بندی درخت، کمینه تعداد برگ درخت و بیشینه عمق درخت به صورت جدا با انتخاب بازه مقادیر تنظیم می‌شوند. مطابق شکل (۴) مقادیر دقت داده‌های آموزشی، دقت



شکل ۴. تنظیم پارامتر مدل Gradient Boosting

۴-۳-۴ مدل LightGBM

در این مدل، پارامترهای تعداد درخت، نرخ یادگیری، نمونه‌برداری ستون‌ها، بیشینه عمق درخت و کمینه تعداد فرزند تنظیم شده‌اند. ابتدا بازه‌ای برای هر یک از پارامترها انتخاب و سپس دقت آموزشی، دقت اعتبارسنجی و واریانس هر یک مطابق مدل GB محاسبه و مقادیر با دقت بالا و واریانس کم انتخاب می‌شوند. با استفاده از پنجره جستجو گریدی مقادیر ۲۰۰ درخت، نرخ یادگیری ۰/۰۵، نرخ نمونه‌برداری ستون‌ها ۰/۷، بیشترین عمق درخت ۱۱ و کمینه تعداد فرزند ۲۰، بالاترین دقت را کسب کرده‌اند. باتوجه به ماتریس پیچیدگی جدول (۷) مقدار میانگین FScore در این مدل ۹۵/۴۹ درصد برآورد شد.

۴-۴-۴ مدل CatBoost

در این مدل، پارامترهای تعداد درخت، نرخ یادگیری، عمق درخت، مقدار منظم‌سازی L2 و تعداد تقسیم تنظیم شده‌اند. بازه مقادیر دقت و واریانس مطابق مدل GB محاسبه و مقادیر با دقت آموزشی و اعتبارسنجی بالا و واریانس کمتر انتخاب با استفاده از پنجره جستجو گریدی تمامی مقادیر انتخابی پارامترها، باهم بررسی شد. با میانگین‌گیری از این مقادیر به صورت خرد، مقادیر ۱۰۰۰ درخت، نرخ یادگیری ۰/۱۵، پارامتر منظم‌سازی ۱، عمق درخت ۸ و تعداد تقسیم ۲۰۰، بالاترین دقت را کسب کرده‌اند. باتوجه به ماتریس پیچیدگی جدول (۸) میانگین FScore برابر با ۹۵/۰۸ درصد محاسبه شد.

جدول ۵. ماتریس پیچیدگی مدل GB

کلاس پیش‌بینی شده							
پیاپی	پیاپی	دوچرخه	اتوبوس	اتومبیل	قطار	مجموع	Recall
۹۰۵	۶	۲	۰	۰	۰	۹۱۳	۰/۹۹۱
۱۶	۳۱۹	۲	۰	۰	۰	۳۳۷	۰/۹۴۷
۱۱	۸	۳۶۲	۱۹	۱۹	۲	۴۰۲	۰/۹۰۰
۵	۱	۱۸	۲۶۸	۲۶۸	۱	۲۹۳	۰/۹۱۵
۳	۰	۳	۵	۵	۳۸	۴۹	۰/۷۷۵
۰/۹۶۳	۰/۹۵۵	۰/۹۳۵	۰/۹۱۸	۰/۹۲۷	-	-	Precision
۰/۹۷۷	۰/۹۵۱	۰/۹۱۸	۰/۹۱۶	۰/۸۴۴	-	-	F_Score

جدول ۶. ماتریس پیچیدگی مدل XGBoost

کلاس پیش‌بینی شده							
پیاپی	پیاپی	دوچرخه	اتوبوس	اتومبیل	قطار	مجموع	Recall
۹۰۶	۷	۰	۰	۰	۰	۹۱۳	۰/۹۹۲
۱۵	۳۲۰	۲	۰	۰	۰	۳۳۷	۰/۹۵۰
۱۱	۸	۳۶۵	۱۶	۱۶	۲	۴۰۲	۰/۹۰۸
۵	۲	۱۸	۲۶۸	۲۶۸	۰	۲۹۳	۰/۹۱۵
۳	۰	۲	۵	۵	۳۹	۴۹	۰/۷۹۶
۰/۹۶۴	۰/۹۵۰	۰/۹۴۳	۰/۹۲۷	۰/۹۵۱	-	-	Precision
۰/۹۷۸	۰/۹۵۰	۰/۹۲۵	۰/۹۲۱	۰/۸۶۷	-	-	F_Score

جدول ۷. ماتریس پیچیدگی مدل LightGBM

کلاس پیش‌بینی شده							
پیاپی	پیاپی	دوچرخه	اتوبوس	اتومبیل	قطار	مجموع	Recall
۹۰۷	۶	۰	۰	۰	۰	۹۱۳	۰/۹۹۳
۱۵	۳۱۹	۳	۰	۰	۰	۳۳۷	۰/۹۴۶
۱۰	۷	۳۶۷	۱۷	۱۷	۱	۴۰۲	۰/۹۱۳
۵	۱	۱۷	۲۷۰	۲۷۰	۰	۲۹۳	۰/۹۲۱
۳	۰	۱	۴	۴	۴۱	۴۹	۰/۸۳۷
۰/۹۶۵	۰/۹۵۸	۰/۹۴۶	۰/۹۲۸	۰/۹۷۶	-	-	Precision
۰/۹۷۹	۰/۹۵۲	۰/۹۲۹	۰/۹۲۵	۰/۹۰۱	-	-	F_Score

جدول ۸. ماتریس پیچیدگی مدل CatBoost

کلاس پیش‌بینی شده								کلاس واقعی
Recall	مجموع	قطار	اتوبیل	اتوبوس	دوچرخه	پیاده‌روی		
۰/۹۹۶	۹۱۳	۰	۰	۱	۳	۹۰۹	پیاده‌روی	
۰/۹۴۱	۳۳۷	۰	۱	۵	۳۱۷	۱۴	دوچرخه	
۰/۹۰۳	۴۰۲	۲	۲۱	۳۶۳	۵	۱۱	اتوبوس	
۰/۹۱۱	۲۹۳	۱	۲۶۷	۱۹	۱	۵	اتوبیل	
۰/۸۱۶	۴۹	۴۰	۳	۲	۱	۳	قطار	
-	-	۰/۹۳۰	۰/۹۱۴	۰/۹۳۱	۰/۹۷۰	۰/۹۶۴	Precision	
-	-	۰/۸۶۹	۰/۹۱۳	۰/۹۱۷	۰/۹۵۵	۰/۹۸۰	F_Score	

و $S = 5$ و $DF = 0.5$ ، $LR_0 = 0.001$ نیز Schedule مقدار دلتای توقف زودهنگام 0.001 است.

۴-۱-۵-۱- مدل CNN

ابتدا لایه‌های منجر به بهترین دقت طبق جدول (۹) انتخاب شدند که تعداد لایه‌های Convolution در ابتدا کم و سپس افزایش می‌یابد. با فرض تعداد نورون‌های لایه FC برابر ۱۲۸، معماری (و) بهترین دقت را کسب کرد. برای اجرای پنجره‌گریدی فیلتر Convolution برابر با [۲، ۳، ۵، ۷، ۱۱]، فیلتر Max-Pooling برابر با [۲، ۳، ۵، ۷] و Dropout برابر با [۰/۱، ۰/۲، ۰/۳، ۰/۴، ۰/۵، ۰/۶، ۰/۷، ۰/۸] جستجو و در بین بازه‌ها، به ترتیب ۲، ۳ و ۰/۷ انتخاب شدند. شکل (۵) معماری نهایی این مدل است. ماتریس پیچیدگی در جدول (۱۰) نمایش و میانگین FScore این مدل برابر با ۸۵/۲۶ درصد محاسبه شد.

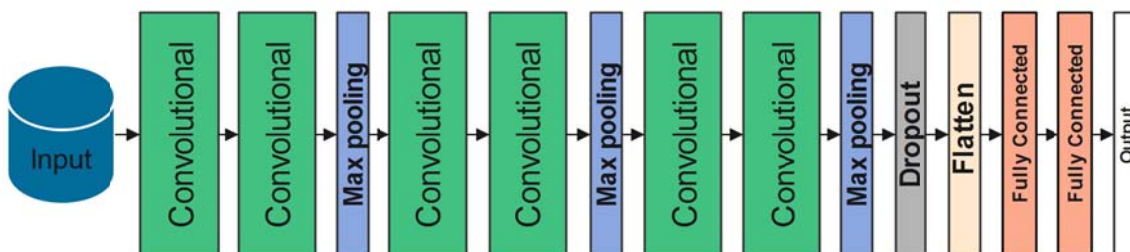
جدول ۹. لایه‌های در نظر گرفته شده مدل CNN

ط	ح	ز	و	ه	د	ج	ب	الف	
مجموعه‌ای از داده با ابعاد (۲۳۰×۴) برای هر تقسیم‌بندی									لایه ورودی
۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	Convolutional
۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	۳۲	Convolutional
✓	✓	✓	✓	✓	×	×	×	×	Max-pooling
۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	×	Convolutional
۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	۶۴	×	Convolutional
✓	✓	✓	✓	✓	×	×	×	×	Max-pooling
۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	×	×	Convolutional
۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	۱۲۸	×	×	Convolutional
✓	✓	✓	✓	✓	×	×	×	×	Max-pooling
✓	✓	✓	✓	×	×	×	×	×	Dropout
✓	✓	✓	×	×	×	×	×	×	FC
✓	✓	×	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	×	×	×	FC
✓	×	×	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	✓	✓	✓	FC(Softmax)
۰/۸۳۷	۰/۸۴۱	۰/۸۳۱	۰/۸۴۳	۰/۸۲۰	۰/۷۹۰	۰/۷۵۵	۰/۷۱۸	۰/۷۰۹	دقت

۴-۵-۲- مدل LSTM

این مدل نیز به مانند مدل CNN، معماری و لایه‌های مناسب را با در نظر گرفتن جدول (۱۱) انتخاب می‌کند. تعداد نورون لایه FC برابر ۱۲۸ در نظر گرفته شده است.

مدل (د) توانسته بهترین دقت را در بین مدل‌های دیگر کسب کند. ماتریس پیچیدگی مدل LSTM همراه با مقادیر Recall، Precision و FScore در جدول (۱۲) نمایش داده شده است که FScore میانگین برابر با ۸۴/۸۰ درصد محاسبه شده است.



شکل ۵. معماری مدل CNN

جدول ۱۰. ماتریس پیچیدگی مدل CNN

کلاس پیش‌بینی شده								کلاس واقعی
Recall	مجموع	قطار	اتومبیل	اتوبوس	دوچرخه	پیاده‌روی		
۰/۹۶۴	۹۱۳	۰	۲	۸	۲۳	۸۸۰	پیاده‌روی	
۰/۸۴۶	۳۳۷	۰	۵	۱۳	۲۸۵	۳۴	دوچرخه	
۰/۷۲۴	۴۰۲	۰	۶۱	۲۹۱	۱۵	۳۵	اتوبوس	
۰/۷۳۰	۲۹۳	۱	۲۱۴	۶۲	۵	۱۱	اتومبیل	
۰/۶۱۲	۴۹	۳۰	۱۴	۱	۰	۴	قطار	
-	-	۰/۹۶۸	۰/۷۲۳	۰/۷۷۶	۰/۸۶۹	۰/۹۱۳	Precision	
-	-	۰/۷۵۰	۰/۷۲۷	۰/۷۴۹	۰/۸۵۷	۰/۹۳۸	F_Score	

جدول ۱۱. لایه‌های در نظر گرفته شده مدل LSTM

الف ب ج د ه و ز ح ط									
مجموعه‌ای از داده با ابعاد (۲۳۰×۴) برای هر تقسیم‌بندی									لایه ورودی
۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	LSTM
۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	×	×	LSTM
✓	✓	✓	✓	✓	×	×	×	×	Dropout
۱۰۰	۱۰۰	۱۰۰	۱۰۰	×	×	×	×	×	LSTM
✓	✓	×	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	×	✓	×	FC
✓	✓	✓	×	×	×	×	×	×	Dropout
✓	✓	✓	×	×	×	×	×	×	FC
✓	×	×	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	✓	✓	✓	FC(Softmax)
۰/۸۰۵	۰/۸۱۴	۰/۸۲۲	۰/۸۴۰	۰/۸۰۶	۰/۸۴۸	۰/۸۳۱	۰/۸۲۶	۰/۸۱۷	دقت

جدول ۱۲. ماتریس پیچیدگی LSTM

کلاس پیش‌بینی شده							
Recall	مجموع	قطار	اتومبیل	اتوبوس	دوچرخه	پیاده‌روی	
۰/۹۷۱	۹۱۳	۰	۳	۲	۲۱	۸۸۷	پیاده‌روی
۰/۸۳۴	۳۳۷	۰	۶	۱۶	۲۸۱	۳۴	دوچرخه
۰/۷۱۶	۴۰۲	۲	۶۰	۲۸۱	۱۶	۳۶	اتوبوس
۰/۷۱۰	۲۹۳	۴	۲۰۸	۶۸	۳	۱۰	اتومبیل
۰/۵۵۱	۴۹	۲۷	۱۶	۳	۰	۳	قطار
-	-	۰/۸۱۸	۰/۷۱۰	۰/۷۶۴	۰/۸۷۵	۰/۹۱۴	Precision
-	-	۰/۶۵۸	۰/۷۱۰	۰/۷۳۹	۰/۸۵۴	۰/۹۴۲	F_Score

کلاس واقعی

۴-۵-۳- مدل ConvLSTM

ابعاد ورودی در این مدل متفاوت از مدل‌های قبلی یادگیری عمیق می‌باشد. ورودی این مدل یک تنسور ۵ بعدی به شکل (کانال، ستون، سطر، فاصله زمانی، تعداد نمونه‌ها) می‌باشد. داده‌ها در دو مدل قبل به شکل سری زمانی به ۲۳۰ نقطه تقسیم شده بودند، در این مدل، این تعداد به n زیرمجموعه m تایی تقسیم می‌شود. سطر داده‌ها برابر با ۱، فاصله زمانی برابر n ، ستون‌ها برابر m و کانال‌ها برابر با تعداد ویژگی‌های نقطه‌ای در نظر گرفته می‌شوند. چون در هر سفر ۲۳۰ نقطه وجود دارد، مقادیر n و m به ترتیب ۵ و ۴۶ محاسبه می‌شوند. جدول (۱۳) لایه‌های در نظر گرفته شده در این مدل را نشان می‌دهد. تعداد نورون‌های لایه FC، ۲۵۶ در نظر گرفته شده‌اند. مدل (۵) توانسته بالاترین دقت را کسب کند. سپس با استفاده از یک پنجره گریدی مقادیر فیلتر Convolution برابر با $[(1,1)]$ ، $(1,7)$ ، $(1,5)$ ، $(1,3)$ ، $(1,2)$ و مقادیر Dropout با $0/8$ ، $0/7$ ، $0/6$ ، $0/5$ ، $0/4$ ، $0/3$ ، $0/2$ ، $0/1$ جستجو شده‌اند که به ترتیب مقادیر $(1,3)$ و $0/5$ به بالاترین دقت میانگین FScore برابر با $85/41$ درصد دست یافته‌اند. جدول (۱۴) ماتریس پیچیدگی این مدل را نشان می‌دهد.

۴-۶- ارزیابی نتایج

با توجه به نامتقارن بودن برچسب داده‌های ورودی، بهترین ارزیابی نهایی دقت، FScore می‌باشد. مقادیر FScore محاسبه

شده حالت‌های حمل‌ونقلی بر اساس مدل‌های کلاسه بندی، در شکل (۶) نمایش داده شده است. مدل LightGBM در حالت استفاده از اتوبوس با $92/91$ درصد، حالت استفاده از اتومبیل با $92/47$ درصد و استفاده از قطار با $90/11$ درصد و مدل CatBoost در حالت استفاده از دوچرخه با $95/48$ درصد و در حالت پیاده‌روی با کسب 98 درصد توانسته‌اند بهترین FScore را در بین مدل‌های دیگر بدست آورند. تمامی محاسبات با استفاده از سرویس ابری Google Colaboratory انجام شده است.

زمان محاسبه بر اساس حجم ورودی داده‌ها در نمودار شکل (۷) نمایش داده شده است. با توجه به این نمودار مدل LightGBM با روند زمان محاسباتی کمتر بهترین مدل در بین مدل‌های تقویت‌کننده و مدل CNN بهترین مدل بر اساس صرف زمان در بین مدل‌های یادگیری عمیق می‌باشد. مطابق جدول (۱۵) مدل‌های LightGBM با انتخاب 23 ویژگی، CatBoost با 30 ویژگی، XGBoost با 21 ویژگی، GB با انتخاب 26 ویژگی به ترتیب با میانگین FScore $95/49$ ، $95/08$ ، $95/18$ ، $94/88$ درصد توانسته‌اند مقدار ارزیابی بالاتری از 3 مدل یادگیری عمیق، LSTM، CNN، ConvLSTM، به ترتیب با F-Score $85/14$ ، $85/26$ ، $84/80$ درصد کسب کنند. بهترین F-Score در مدل‌های تقویت‌کننده، مدل LightGBM و در مدل‌های یادگیری عمیق، مدل ConvLSTM می‌باشد.

جدول ۱۳. لایه‌های در نظر گرفته شده مدل ConvLSTM

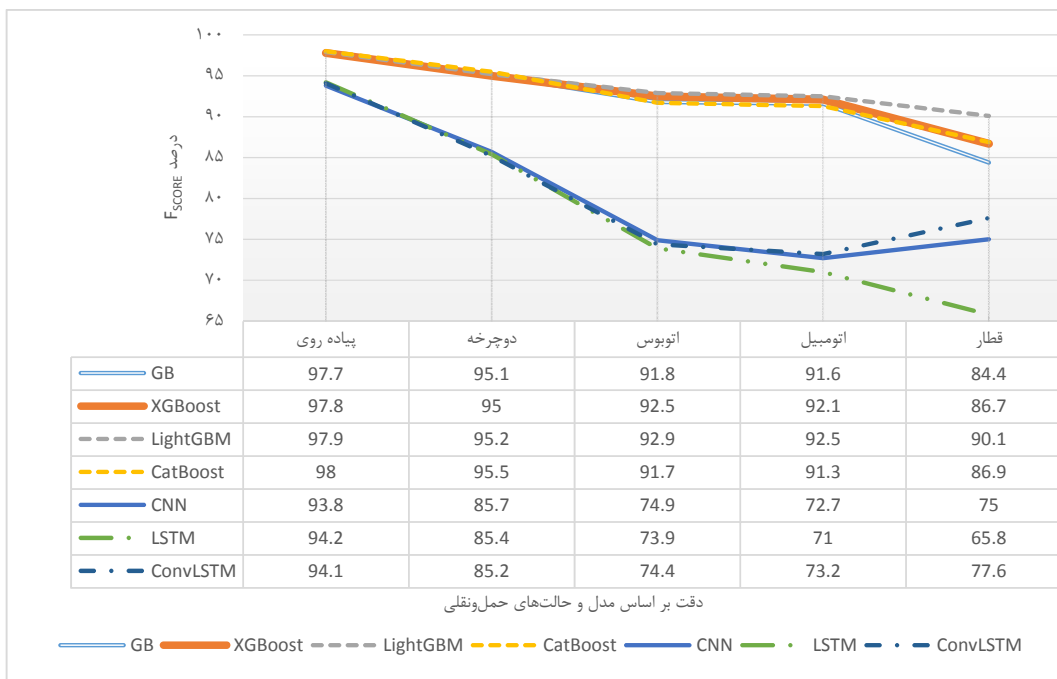
الف ب ج د ه و ز ح ط ی										
مجموعه‌ای از داده با ابعاد (۵×۱×۴۶×۴) برای هر تقسیم‌بندی										
لایه ورودی	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	ConvLSTM
۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	۱۰۰	ConvLSTM
✓	✓	✓	✓	✓	✓	✓	✓	✓	×	Dropout
۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	×	×	ConvLSTM
۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	۲۰۰	×	×	ConvLSTM
✓	✓	✓	✓	✓	✓	✓	×	×	×	Dropout
۳۰۰	۳۰۰	۳۰۰	×	×	×	×	×	×	×	ConvLSTM
۳۰۰	۳۰۰	۳۰۰	×	×	×	×	×	×	×	ConvLSTM
✓	✓	×	×	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	FC
✓	✓	✓	✓	✓	✓	×	×	✓	×	Dropout
✓	×	×	✓	✓	×	×	×	×	×	FC
✓	×	×	✓	×	×	×	×	×	×	Dropout
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	FC(Softmax)
۰/۸۴۲	۰/۸۲۵	۰/۸۰۴	۰/۸۴۰	۰/۸۴۱	۰/۸۴۵	۰/۸۴۴	۰/۸۴۲	۰/۸۲۸	۰/۸۰۴	دقت

جدول ۱۴. ماتریس پیچیدگی ConvLSTM

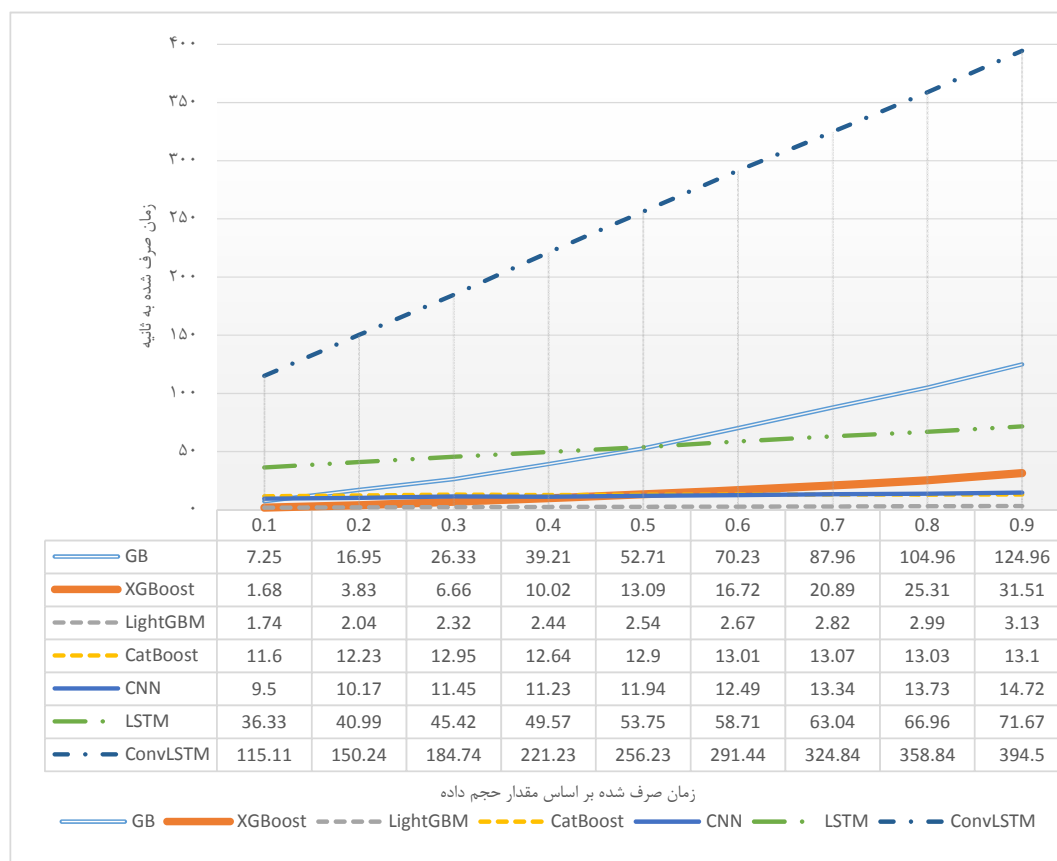
کلاس پیش‌بینی شده								کلاس واقعی
Recall	مجموع	قطار	اتومبیل	اتوبوس	دوچرخه	پیاده‌روی		
۰/۹۷۶	۹۱۳	۰	۰	۴	۱۸	۸۹۱	پیاده‌روی	
۰/۸۱۹	۳۳۷	۰	۲	۱۸	۲۷۶	۴۱	دوچرخه	
۰/۷۲۹	۴۰۲	۰	۶۱	۲۹۳	۱۴	۳۴	اتوبوس	
۰/۷۱۷	۲۹۳	۳	۲۱۰	۶۷	۳	۱۰	اتومبیل	
۰/۶۷۳	۴۹	۳۳	۸	۴	۰	۴	قطار	
-	-	۰/۹۱۷	۰/۷۴۷	۰/۷۵۹	۰/۸۸۷	۰/۹۰۹	Precision	
-	-	۰/۷۷۶	۰/۷۳۲	۰/۷۴۴	۰/۸۵۲	۰/۹۴۱	F_Score	

جدول ۱۵. میانگین دقت بر اساس مدل‌های کلاسه‌بندی

مدل کلاسه‌بندی	تعداد ویژگی انتخاب شده	میانگین دقت (درصد)	سرعت (۸۰ درصد داده‌ها به ثانیه)
GB	۲۶	۹۴/۸۸	۱۰۴/۹۶
XGBoost	۲۱	۹۵/۱۸	۲۵/۳۱
LightGBM	۲۳	۹۵/۴۹	۲/۹۹
CatBoost	۳۰	۹۵/۰۸	۱۳/۰۳
CNN	---	۸۵/۲۶	۱۴/۷۲
LSTM	---	۸۴/۸۰	۷۱/۶۷
ConvLSTM	---	۸۵/۴۱	۳۹۴/۵



شکل ۶. مقدار دقت Fscore محاسبه شده بر اساس مدل‌های کلاسه‌بندی



شکل ۷. سرعت محاسبه بر اساس مدل‌های کلاسه‌بندی

۵- نتیجه گیری

حمل و نقل هوشمند نقش مهمی در زندگی روزمره شهروندان دارد و هر کشور پیشرفته ملزم به توسعه سیستم‌های حمل و نقلی خود در جهت تسهیل زندگی در شهر می‌باشد. پیش‌نیاز شناخت بسیاری از مشکلات حمل و نقلی، پیش‌بینی استفاده از حالت‌های حمل و نقلی شهروندان می‌باشد. گسترش فناوری GNSS و تلفن‌های هوشمند توانایی ثبت حجم عظیمی از داده‌ها در کمترین زمان را فراهم آورده و موجب شده تا بتوان سریع‌تر و با دقت‌تر الگوهای پنهان حرکتی افراد را کشف کرد. در این پژوهش از داده‌های با کیفیت منبع باز GeoLife به همراه سه مدل یادگیری عمیق و چهار مدل تقویت‌کننده استفاده شده تا به وسیله داده‌های GNSS به تنهایی بتوان پیش‌بینی حالت‌های حمل و نقلی پیاده‌روی، استفاده از دوچرخه، استفاده از اتومبیل، استفاده از اتوبوس و استفاده از قطار را انجام داد. پیش‌پردازش داده‌ها بر اساس فیلتر حدآستانه با توجه به ویژگی‌های طبیعی هر حالت حمل و نقلی انجام پذیرفته است. با استخراج ویژگی‌های نقطه‌ای شامل: سرعت، شتاب، جرک و تغییر جهت، مدل‌های CNN، LSTM، ConvLSTM که زیر مجموعه روش یادگیری عمیق می‌باشند، پیاده‌سازی شده‌اند. داده‌های ورودی این مدل‌ها از چهار کانال تشکیل شده که هر کانال یک ویژگی نقطه‌ای را به ترتیب زمان ثبت شده سفر در خود ذخیره می‌کند. سفرها در این مدل به تعداد میانه نقاط موجود در مسیر محدود شده‌اند. مجموعه‌ای از معماری‌های مختلف برای این روش پیاده‌سازی شده و با تنظیم اندازه فیلترها، بهترین آنها شناسایی و انتخاب می‌شوند. از روش توقف زود هنگام و step decay schedule استفاده شده تا به ترتیب از بیش‌برازش جلوگیری و بهترین نقطه تابع هزینه یافت شود. در روش‌های تقویت‌کننده تعداد ۵۶ ویژگی سفر و سه ویژگی پیشرفته استخراج و پس از انتخاب ویژگی بر اساس روش ترکیبی مدل‌های GB، XGBoost، LightGBM و CatBoost پیاده‌سازی و پارامترهای آن تنظیم شده تا بهترین دقت را برآورد کنند. پس از پیاده‌سازی، مدل LightGBM در حالت‌های استفاده از اتوبوس، استفاده از اتومبیل و استفاده از قطار و همچنین مدل CatBoost در حالت‌های پیاده‌روی و استفاده از دوچرخه توانسته‌اند بهترین دقت را کسب کنند. در حالت کلی نیز مدل LightGBM بالاتر از بقیه مدل‌ها با کسب میانگین F_{Score} ۹۵/۴۹ درصد، توانسته بهترین پیش‌بینی را با سرعت بالاتری از بقیه مدل‌ها انجام دهد. انتخاب این مدل نشان‌دهنده ارزش

۶- پی‌نوشت‌ها

1. Location Based Services (Lbs)
2. Global Navigation Satellite System (Gnss)
3. Boosting Method
4. Jerk
5. Gradient Boosting
6. Extreme Gradient Boosting
7. Lightweight Gradient Boosting Machines
8. Category Boosting
9. Convolutional Neural Network
10. Long Short Term Memory
11. Convolutional Long Short Term Memory
12. Filter Based
13. Wrapper Based
14. Forward Search
15. Categorical Cross-Entropy
16. Loss Function
17. Regularization
18. Gradient-Based One-Side Sampling
19. Exclusive Feature Bundling
20. Categorical Feature
21. Greedy Target-Based Statistics
22. Prior Value
23. Feed Forward
24. Rectified Linear Units
25. Recurrent Neural Networks
26. Forget Gate
27. Input Gate
28. Output Gate
29. Cell State
30. Confusion Matrix
31. Imbalanced

of the 22nd acm sigkdd international conference on knowledge discovery and data mining.

-Chon, J. and H. Cha, (2011), "Lifemap: A smartphone-based context provider for location-based services", *IEEE Pervasive Computing* 10(2), pp.58-67.

-Chung, E.-H. and A. Shalaby, (2005), "A trip reconstruction tool for GPS-based personal travel surveys", *Transportation Planning and Technology* 28(5), pp.381-401.

-Cui, G., J. Luo and X. Wang, (2018), "Personalized travel route recommendation using collaborative filtering based on GPS trajectories", *International journal of digital earth* 11(3), pp.284-307.

-Dabiri, S. and K. Heaslip, (2018), "Inferring transportation modes from GPS trajectories using a convolutional neural network." *Transportation research part C: emerging technologies* 86, pp.360-371.

-Das, R. D. and S. Winter, (2016), "Detecting urban transport modes using a hybrid knowledge driven framework from GPS trajectory", *ISPRS International Journal of Geo-Information* 5(11), pp.207.

-Dorogush, A. V., V. Ershov and A. Gulin, (2018), "CatBoost: gradient boosting with categorical features support", *arXiv preprint arXiv:1810.11363*.

-Endo, Y., H. Toda, K. Nishida and A. Kawanobe, (2016), "Deep feature extraction from trajectories for transportation mode estimation", *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer.

-Friedman, J. H., (2001), "Greedy function approximation: a gradient boosting machine", *Annals of statistics*, pp.1189-1232.

-Ge, R., S. M. Kakade, R. Kidambi and P. Netrapalli, (2019), "The step decay schedule: A near optimal, geometrically decaying learning rate procedure for least squares", *arXiv preprint arXiv:1904.12838*.

-Gers, F. A., J. Schmidhuber and F. Cummins (1999), "Learning to forget: Continual prediction with LSTM".

-Guo, M., S. Liang, L. Zhao and P. Wang (2020), "Transportation Mode Recognition With Deep Forest Based on GPS Data", *IEEE Access* 8, pp.150891-150901.

-Han, J., M. Kamber and J. Pei, (2011), "Data mining concepts and techniques third edition", *The Morgan Kaufmann Series in Data Management Systems* 5(4), pp.83-124.

۷-مراجع

-Adler, J. L. and V. J. Blue, (1998), "Toward the design of intelligent traveler information systems", *Transportation Research Part C: Emerging Technologies* 6(3), pp.157-172.

-Bantis, T. and J. Haworth, (2017), "Who you are is how you travel: A framework for transportation mode detection using individual and environmental characteristics", *Transportation Research Part C: Emerging Technologies* 80, pp.286-309.

-Bao, J., Y. Zheng, D. Wilkie and M. Mokbel, (2015), "Recommendations in location-based social networks: a survey", *GeoInformatica* 19(3), pp.525-565.

-Bedogni, L., M. Di Felice and L. Bononi, (2016), "Context-aware Android applications through transportation mode detection techniques", *Wireless communications and mobile computing* 16(16), pp.2523-2541.

-Bengio, Y., P. Frasconi and P. Simard, (1993), "The problem of learning long-term dependencies in recurrent networks", *IEEE international conference on neural networks*, IEEE.

-Biljecki, F., H. Ledoux and P. Van Oosterom, (2013), "Transportation mode-based segmentation and classification of movement trajectories", *International Journal of Geographical Information Science* 27(2), pp.385-407.

-Bolbol, A., T. Cheng, I. Tsapakis and J. Haworth, (2012), "Inferring hybrid transportation modes from sparse GPS data using a moving window SVM classification", *Computers, Environment and Urban Systems* 36(6), pp.526-537.

-Caruana, R., S. Lawrence and L. Giles, (2001), "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping", *Advances in neural information processing systems*, pp.402-408.

-Chamoso, P., A. González-Briones, S. Rodríguez and J. M. Corchado, (2018), "Tendencies of technologies and platforms in smart cities: a state-of-the-art review", *Wireless Communications and Mobile Computing*.

-Chatzimilioudis, G., A. Konstantinidis, C. Laoudias and D. Zeinalipour-Yazti, (2012), "Crowdsourcing with smartphones", *IEEE Internet Computing* 16(5), pp.36-44.

-Chen, T. and C. Guestrin, (2016), "Xgboost: A scalable tree boosting system", *Proceedings*

- Novaković, J. D., A. Veljović, S. S. Ilić, Ž. Papić and T. Milica, (2017), "Evaluation of classification models in machine learning." *Theory and Applications of Mathematics & Computer Science* 7(1), pp.39-46.
- Nwankpa, C., W. Ijomah, A. Gachagan and S. Marshall, (2018), "Activation functions: Comparison of trends in practice and research for deep learning." *arXiv preprint arXiv:1811.03378*.
- Pan, G., G. Qi, Z. Wu, D. Zhang and S. Li, (2012), "Land-use classification using taxi GPS traces", *IEEE Transactions on Intelligent Transportation Systems* 14(1), pp.113-123.
- Payne, S., (2015), "Study on key performance indicators for intelligent transport systems: final report in support of the implementation of the EU Legislative Framework on ITS (Directive 2010/40/EU)".
- Phithakkitnukoon, S., T. Horanont, G. Di Lorenzo, R. Shibasaki and C. Ratti, (2010), "Activity-aware map: Identifying human daily activity pattern using mobile phone data, International workshop on human behavior understanding, Springer.
- Prokhorenkova, L., G. Gusev, A. Vorobev, A. V. Dorogush and A. Gulin, (2017), "CatBoost: unbiased boosting with categorical features", *arXiv preprint arXiv:1706.09516*.
- Quessada, M. S., R. S. Pereira, W. Revejes, B. Sartori, E. N. Gottsfritz, D. D. Lieira, M. A. da Silva, G. P. Rocha Filho and R. I. Meneguette (2020), "ITSMEI: An intelligent transport system for monitoring traffic and event information", *International Journal of Distributed Sensor Networks* 16(10): 1550147720963751.
- Scheiner, J. and C. Holz-Rau, (2007), "Travel mode choice: affected by objective or subjective determinants?", *Transportation* 34(4), pp. 487-511.
- Scherer, D., A. Müller and S. Behnke, (2010), "Evaluation of pooling operations in convolutional architectures for object recognition", *International conference on artificial neural networks*, Springer.
- Shi, X., Z. Chen, H. Wang, D.-Y. Yeung, W. K. Wong and W.-c. Woo, (2015), "Convolutional LSTM network: A machine learning approach for precipitation nowcasting", *arXiv preprint arXiv:1506.04214*.
- Simonyan, K. and A. Zisserman, (2014), "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*.
- Hochreiter, S., (1998), "The vanishing gradient problem during learning recurrent neural nets and problem solutions", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6(02), pp.107-116.
- Jahangiri, A. and H. Rakha, (2014), "Developing a support vector machine (SVM) classifier for transportation mode identification by using mobile phone sensor data", *Transportation Research Board 93rd Annual Meeting*.
- Jarašūniene, A., (2007), "Research into intelligent transport systems (ITS) technologies and efficiency", *Transport* 22(2), pp.61-67.
- Jović, A., K. Brkić and N. Bogunović, (2015), "A review of feature selection methods with applications", 2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO), Ieee.
- Ke, G., Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye and T.-Y. Liu, (2017), "Lightgbm: A highly efficient gradient boosting decision tree", *Advances in neural information processing systems* 30, pp. 3146-3154.
- Kingma, D. P. and J. Ba, (2014), "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*.
- Kitchin, R., (2014), "The real-time city? Big data and smart urbanism." *GeoJournal* 79(1): pp.1-14.
- Krizhevsky, A., I. Sutskever and G. E. Hinton, (2017), "ImageNet classification with deep convolutional neural networks", *Communications of the ACM* 60(6), pp.84-90.
- Langley, R. B., (1997), "Innovation: the GPS error budget", *GPS world* 8(3), pp.51-56.
- Li, J., X. Pei, X. Wang, D. Yao, Y. Zhang and Y. Yue, (2021), "Transportation mode identification with GPS trajectory data and GIS information", *Tsinghua Science and Technology* 26(4), pp.403-416.
- Lipton, Z. C., C. Elkan and B. Naryanaswamy (2014). Optimal thresholding of classifiers to maximize F1 measure. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer.
- Nawaz, A., H. Zhiqiu, W. Senzhang, Y. Hussain, I. Khan and Z. Khan, (2020), "Convolutional LSTM based transportation mode learning from raw GPS trajectories." *IET Intelligent Transport Systems* 14(6), pp.570-577.

- Venkatesh, B. and J. Anuradha, (2019), "A review of feature selection and its methods", *Cybernetics and Information Technologies* 19(1), pp.3-26.
- Vu, T. H., L. Dung and J.-C. Wang, (2016), "Transportation mode detection on mobile devices using recurrent nets", *Proceedings of the 24th ACM international conference on Multimedia*.
- Wang, B., Y. Wang, K. Qin and Q. Xia (2018), "Detecting transportation modes based on LightGBM classifier from GPS trajectory data. 2018 26th International Conference on Geoinformatics, IEEE".
- Xiao, Z., Y. Wang, K. Fu and F. Wu, (2017), "Identifying different transportation modes from trajectory data using tree-based ensemble classifiers", *ISPRS International Journal of Geo-Information* 6(2), pp.57.
- Yang, X., L. Tang, L. Niu, X. Zhang and Q. Li, (2018), "Generating lane-based intersection maps from crowdsourcing big trace data", *Transportation Research Part C: Emerging Technologies* 89, pp.168-187.
- Zheng, Y., H. Fu, X. Xie, W. Ma and Q. Li, (2011), "Geolife GPS Trajectory Dataset-User Guide", Microsoft Research.
- Zheng, Y., Q. Li, Y. Chen, X. Xie and W.-Y. Ma, (2008), "Understanding mobility based on GPS data", *Proceedings of the 10th international conference on Ubiquitous computing*.
- Zheng, Y., L. Liu, L. Wang and X. Xie (2008), "Learning transportation mode from raw gps data for geographic applications on the web", *Proceedings of the 17th international conference on World Wide Web*.
- Song, X., H. Kanasugi and R. Shibasaki, (2016), "Deeptransport: Prediction and simulation of human mobility and transportation mode at a citywide level", *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*.
- Stenneth, L., O. Wolfson, P. S. Yu and B. Xu (2011), "Transportation mode detection using mobile phones and GIS information", *Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems*.
- Stopher, P., C. FitzGerald and J. Zhang, (2008), "Search for a global positioning system device to measure person travel", *Transportation Research Part C: Emerging Technologies* 16(3), pp.350-369.
- Su, X., H. Caceres, H. Tong and Q. He, (2016), "Online travel mode identification using smartphones with battery saving considerations", *IEEE Transactions on Intelligent Transportation Systems* 17(10): pp.2921-2934.
- Tamane, S. C., N. Dey and A. E. Hassanien (2020), "Security and Privacy Applications for Smart City Development, Springer".
- Tang, L., Z. Kan, X. Zhang, X. Yang, F. Huang and Q. Li, (2016), "Travel time estimation at intersections based on low-frequency spatial-temporal GPS trajectory big data", *Cartography and Geographic Information Science* 43(5), pp.417-426.
- Tang, L., X. Yang, Z. Dong and Q. Li, (2016), "CLRIC: Collecting lane-based road information via crowdsourcing", *IEEE Transactions on Intelligent Transportation Systems* 17(9), pp.2552-2562.

Identifying Transportation Modes from Trajectory Dataset using Boosting and Deep Learning Methods in Intelligent Transportation Systems

Sajjad Sowlati, School of Surveying and Geospatial Engineering, University of Tehran, Tehran, Iran.

Rahim Ali Abbaspour, Associate Professor, School of Surveying and Geospatial Engineering, University of Tehran, Tehran, Iran.

Alireza Chehregan, Assistant Professor, Faculty of Mining Engineering, Sahand University of Technology, Tabriz, Iran.

E-mail: abaspour@ut.ac.ir

Received: March 2023- Accepted: August 2023

ABSTRACT

Due to the rapid expansion of urbanization, the significance of intelligent transportation in facilitating citizen mobility has garnered increased attention. The fundamental prerequisite for establishing and utilizing intelligent transportation services lies in the identification and prediction of transportation modes. With the progression of location technologies, agents, and smartphones, an abundance of information is generated by numerous devices utilizing Global Navigation Satellite Systems (GNSS). In this study, GNSS sensors were employed in smartphones (Geolife dataset) to extract 4 Point features, 59 travel features, and 3 advanced features from the GNSS dataset. Subsequently, four Boosting classification models, namely GB, XGBoost, LightGBM, and CatBoost, were utilized alongside hybrid feature selection techniques. Additionally, three deep learning classification models, including CNN, LSTM, and ConvLSTM, were implemented to predict transportation modes encompassing walking, biking, bus usage, car usage, and train usage. In conclusion, the LightGBM classification model demonstrated superior performance in predicting transportation modes, exhibiting a higher F_{Score} (95.49%) and requiring less time complexity compared to other models.

Keywords: Intelligent Transportation System (ITS), Transportation Modes, Trajectory Data, Deep Learning