ترکیب روش‌های تجمیعی داده‌کاوی برای کشف تراکنش‌های تقلب در کارت‌های اعتباری

سعید بختیاری* زهرا نصیری، سید محمد صادق حجازی
[1] گروه فتا، دانشکده اطلاعات، دانشگاه امین، تهران ایران
[2] گروه کامپیوتر، دانشکده فنی و مهندسی، مؤسسه آموزش عالی آل طه، تهران، ایران
[3] گروه کامپیوتر، دانشکده فنی و مهندسی، مؤسسه آموزش عالی پردیسان، مازندران، ایران

چکیده

استفاده از کارت‌های اعتباری، جهت پرداخت آسان پول از طریق تلفن همراه، اینترنت، دستگاه‌های خودپرداز و غیره روزبه‌روز گسترده‌تر می‌شود. در کنار محبوبیت استفاده از کارت‌های اعتباری، مشکلات امنیتی مختلفی مانند تقلب وجود می‌آید. همان‌طور که روش‌های امنیتی به‌روز می‌شوند، متقلبان نیز روش‌های خود را به‌روز می‌کنند که این امر موجب نگرانی بانک‌ها و مشتریان آنها می‌شود؛ به همین دلیل راه حل‌های مختلفی جهت تشخیص، پیش‌بینی و پیش‌گیری از تقلب در کارت‌های اعتباری حائز اهمیت هستند. یکی از راه‌حل‌ها روش داده‌کاوی و یادگیری ماشین است که افزایش دقت و کارایی یکی از با اهمیت‌ترین مسائل در این زمینه است. در این مقاله روش‌های Gradient Boosting را که زیرمجموعه روش‌های تجمیعی و یادگیری ماشین هستند، بررسی کرده، با اعمال مهندسی ویژگی و با ترکیب روش‌ها نرخ خطا را کاهش و دقت تشخیص را بهبود می‌دهیم. در روش پیشنهادی دو الگوریتم LightGBM و XGBoost را با برخی روش‌های متداول دیگر مقایسه، سپس آنها را با استفاده از روش‌های تجمیعی میانگین‌گیری ساده و وزن‌دار ترکیب می‌کنیم و درنهایت مدل‌ها به‌وسیله معیارهای AUC و Recall و F1 - score و Precision و Accuracy ارزیابی‌شده‌اند. مدل پیشنهادی پس از اعمال مهندسی ویژگی با استفاده از روش میانگین‌گیری وزن‌دار به‌ترتیب برای روش‌های ارزیابی یادشده به اعدادی معادل ۹۵/۰۸، ۹۰/۵۷، ۸۹/۳۵، ۸۸/۲۸ و ۹۹/۲۷ رسیده است. بر این اساس مهندسی ویژگی و میانگین‌گیری وزن‌دار تأثیر به‌سزایی در بهبود دقت پیش‌بینی و شناسایی داشتند.

واژگان کلیدی: تشخیص تقلب، کارت اعتباری، یادگیری تجمیعی، داده‌کاوی

Combination of Ensemble Data Mining Methods for Detecting Credit Card Fraud Transactions

Saeid Bakhtiari*, Zahra Nasiri, Mohsen Yazdinejad and Seyed Mohammad Sadegh Hejazi

[1] Department of FATA, faculty of engineering, Amin University, Tehran, Iran
[2] Department of Computer Engineering, faculty of engineering, Ale-Taha Institute of Higher Education, Tehran, Iran
[3] Department of Computer Engineering, faculty of engineering, Pardisan Institute of Higher Education, Mazandaran, Iran

Abstract

As we know, credit cards speed up and make life easier for citizens and bank customers. They can use it anytime and anyplace according to their personal needs, instantly, quickly without worrying about carrying a lot of cash with more security. Together, these factors make credit cards one of the most popular forms of online banking. This reason has led to widespread and increasing use for easy payment for purchases made through mobile phones, the Internet, ATMs, and so on. Despite the popularity and ease of payment with credit cards, various security problems are increasing day by day. One of the most

* نویسندهٔ عهده‌دار مکاتبات
* Corresponding author

• تاریخ ارسال مقاله: ۱۴۰۰/۳/۵ • تاریخ پذیرش: ۱۴۰۱/۲/۲۱ • تاریخ انتشار: ۱۴۰۱/۱۲/۲۹ • نوع مطالعه: پژوهشی

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

important and constant challenges in this field is fraud detection in credit card transactions all around the world. Due to the increasing security issues in credit cards, fraudsters are also updating themselves. In general, as the popularity of using credit cards grows, more fraudsters are attracted to it, and credit card security comes into play. So naturally, this worries banks and their customers around the world. Meanwhile, financial information acts as the main factor in market financial transactions. For this reason, many researchers have tried to prioritize various solutions for detecting, predicting, and preventing credit card fraud in their research work and provide essential suggestions that have been associated with significant success. One of the practical and successful methods is data mining and machine learning. One of the most critical parameters in fraud prediction and detection in these methods is fraud detection accuracy. This research intends to examine the Gradient Boosting methods, such as LightGBm and XGBoost, a subset of Ensemble Learning and machine learning methods. By combining these methods, we can identify credit card fraud transactions, reduce error rates, and improve the detection process, which in turn increases efficiency and accuracy. This study compared some typical methods like Random Forest, Logistic Regression, and Navie base with LightGBM and XGBoost algorithms. In this paper, we proposed to merge LightGBM and XGBoost using simple and weighted averaging techniques and then evaluate the models using AUC, Recall, F1-score, Precision, and Accuracy. The proposed model provided values of 95.08, 90.57, 89.35, 88.28, and 99.27, respectively. In addition, we developed features by feature engineering techniques and then applied the feature engineering phase to the models. The results show that applying the feature engineering phase to the weighted average approach significantly improved prediction and detection accuracy.

Keywords: Fraud Detection, Credit Card, Ensemble Learning, Data Mining

۱- مقدمه

در ایالات متحده، چهار شبکه پردازش کارت اعتباری عمده وجود دارد: Visa Card، Master Card، Discover و American Express. شبکه‌های پردازشی تعیین‌کننده دستورالعمل‌های پردازش کارت اعتباری و تسهیل تراکنش‌ها برای مشتریان هستند. Visa Card و Master Card خود کارت اعتباری‌شان را صادر نمی‌کنند، هر زمان کارت اعتباری Visa یا Mastercard را مشاهده کردید بدانید بانک دیگری وجود دارد که کارت اعتباری را صادر می‌کند. از طرف دیگر شبکه‌های American Express و Discover اغلبِ کارت‌های اعتباری‌شان را خود صادر می‌کنند؛ ولی به بانک‌های دیگر هم اجازه صدور کارتشان را می‌دهند. شما می‌توانید از همه شبکه‌های پردازشی استفاده‌کنید؛ اما این را به‌خاطر داشته‌باشید در خارج از ایالات متحده، تاجران کمتری پیدا می‌شوند که American Express و Discover را قبول کنند؛ بنابراین با کارت‌های متصل به این نوع شبکه‌ها، بیشتر مشکل خواهید داشت. تعدادی از شبکه‌های پردازش اصلی نیز تراکنش‌های کارت بدهی[1] را پردازش می‌کنند. کارت‌های بدهی نیز عملکرد مشابه کارت‌های اعتباری دارند، اما یک تمایز عمده بین این دو این است که معاملات کارت بدهی به‌جای یک خط اعتباری از حساب بانکی مصرف‌کننده تأمین می‌شود. اغلب این نوع تراکنش‌ها در ایالات متحده توسط شبکه‌های ویزا و مسترکارت انجام‌می‌شود. بانک‌ها با ویزا و مسترکارت شریک می‌شوند تا کارت‌های بدهی را در دسترس مشتریان خود قرار دهند [1]. با افزایش حجم معاملات تجارت الکترونیکی، کلاه‌برداری کارت اعتباری روزبه‌روز گسترده‌تر می‌شود. با تبدیل‌شدن تجارت الکترونیکی به جریان اصلی و افزایش چندبرابری معاملات برخط، خطرات امنیتی مرتبط با آنها به نگرانی‌های اساسی تبدیل‌شده‌اند. الگوی کلاه‌برداری مالی نیز با توسعه فناوری مدرن تغییر می‌کند و به‌سرعت افزایش می‌یابد که برعکس باعث افزایش سطح تقلب در معاملات کارت اعتباری و خسارات زیادی می‌شود. کشف تقلب در کارت‌های اعتباری همواره پیچیده‌است؛ زیرا رفتار کاربران ثبات ندارد و این تغییراتِ رفتاری پیچیدگی فرآیند را بیشتر می‌کند.

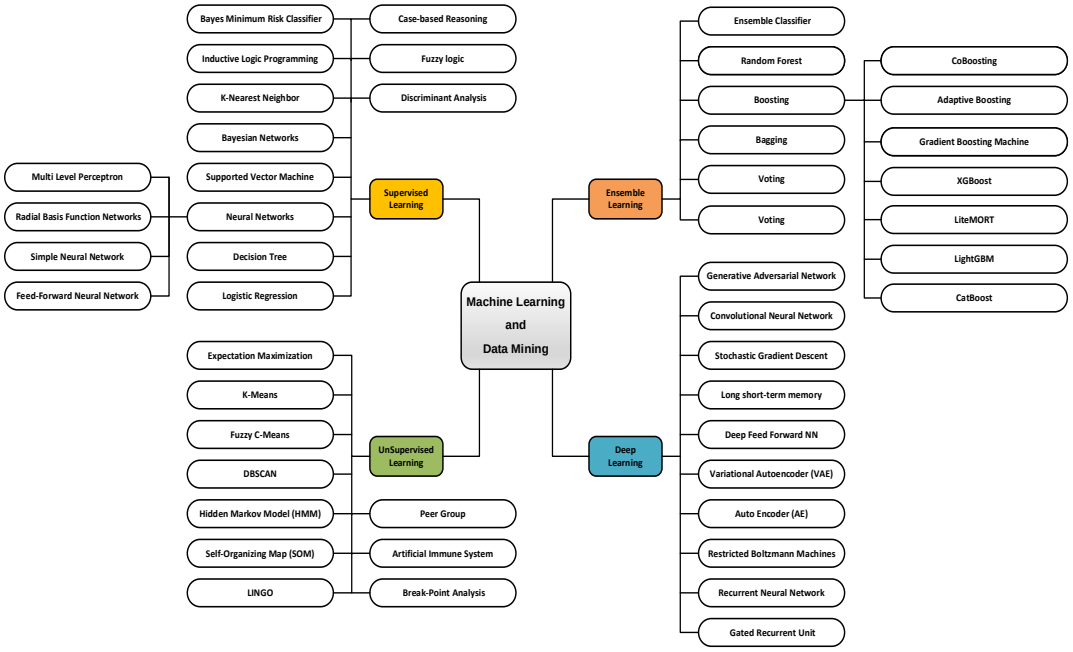
پیش‌گیری از تقلب و کشف تقلب هر دو روش مقابله با تقلب هستند. در پیش‌گیری از تقلب، هدف اصلی جلوگیری از تقلب و تراکنش‌های غیرمجاز است. درحالی‌که در کشف تقلب، هدف تشخیص تراکنش‌های متقلب از تراکنش‌های قانونی است. در سال‌های اخیر چندین مطالعه از تکنیک‌های مختلف داده‌کاوی برای یافتن راه‌حل‌هایی برای این مشکل استفاده کرده‌اند. این تکنیک‌ها مبتنی بر شبکه عصبی[2]، یادگیری عمیق[3]، الگوریتم ژنتیک، مدل مارکوف پنهان[4]، شبکه بیزی[5]، درخت تصمیم[6]، روش خوشه‌بندی[7]، سامانه ایمنی مصنوعی[8]، ماشین بردار پشتیبان[9] و داده‌کاوی که شامل

[1] Debit Card

[2] Neural networks
[3] Deep Learning
[4] Hidden Markov Model
[5] Bayesian Network
[6] Decision Tree
[7] Clustering
[8] Artificial Immune Systems
[9] Support Vector Machine

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]



شکل ۱. روش‌های یادگیری ماشین

Figure 1. Methods of Machine Learning

طبقه‌بندی[1]، خوشه‌بندی، پیش‌بینی، شناسایی نقاط دور افتاده[2] و رگرسیون[3] است است. یادگیری ماشین کاربردی از هوش مصنوعی است که شامل ابزار مختلف یادگیری است. یادگیری را می‌توان: یادگیری تحت نظارت[4]، یادگیری بدون نظارت[5] و یادگیری نیمه‌نظارت[6] قرار داد. الگوریتم‌های یادگیری ماشینی که به‌طورمعمول مورد استفاده قرار می‌گیرند، شامل طبقه‌بندی بیزی[7]، طبقه‌بندی درخت تصمیم[8]، رگرسیون خطی[9]، رگرسیون ترابری[10] و غیره است. (شکل ۱) روش‌های اصلی داده‌کاوی و یادگیری ماشین را به‌صورت اجمالی نشان‌می‌دهد[2].

در سال ۲۰۲۰ هوانگ [3] با استفاده از انتخاب ویژگی[11] در مدل نظارت‌شده، مدل‌های آماری خطی و غیرخطی و مدل‌های یادگیری ماشین از قبیل شبکه عصبی، رگرسیون ترابری، درخت تقویت‌شده[12]، جنگل تصادفی[13] که بر مبنای داده‌های دارایی نیویورک و داده‌های تراکنش کارت اعتباری است پژوهش خود را ارائه داده است. در ژانویه سال ۲۰۱۶ گانین و همکاران [4] یک روش یادگیری نماینده جدید را برای سازگاری دامنه معرفی می‌کند، این روش در چارچوب معماری شبکه عصبی پیاده سازی می‌شود. در مقاله [5] یک روش شبکه عصبی با استفاده از ۱۰ لایه عمیق انکودر خودکار[14] ارائه شده و یک مقایسه دقیق با طبقه‌بندی‌کننده‌های مختلف کلاسیک، یعنی درخت تصمیم‌گیری، ماشین بردار پشتیبان و طبقه‌بندی گروه‌های مختلف انجام شده‌است و از این طبقه‌بندی‌ها در مجموعه‌داده‌های ناهنجاری تراکنش کارت اعتباری استفاده کرده‌اند. در [6] لبیچات و همکاران با استفاده از استراتژی‌های سازگاری دامنه در سامانه‌های کشف تقلب مبتنی بر تراکنش را مورد بررسی قرار دادند. عبدالرضا و همکاران [7]، روش‌های مختلف کشف تقلب را مورد بررسی و طبقه‌بندی قرار دادند و محدودیت‌های عمده و دلایل عدم کارایی روش‌ها را نیز مطرح کردند.. توسعه روش‌های مهندسی ویژگی خودکار[15] به یک مسئله مهم برای بهبود کارایی مدل‌ها تبدیل شده‌است. در [8] لوکاس و همکاران یک مدل مبتنی بر مهندسی ویژگی خودکار برای کشف تقلب در تراکنش کارت‌های اعتباری پیشنهاد دادند. استراتژی مهندسی ویژگی مبتنی بر مدل مارکوف پنهان است. در [9] یک مطالعه مقایسه‌ای از تکنیک‌های مبتنی بر شبکه‌های عصبی، که به مجموعه داده‌ها اعمال می‌شود، انجام شده‌است. در [10] سایت و کارتا با هدف بررسی مزایای

[1] Classification
[2] Outlier
[3] Regression
[4] Supervised
[5] Unsupervised
[6] Semi-Supervised
[7] Bayesian Classifiers
[8] Decision Tree Classifier
[9] Linear Regression
[10] Logistic Regression
[11] Feature Selection
[12] Boostrd Trees
[13] Random Forest
[14] Deep Auto-encoder
[15] Automated Feature Engineering

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]
[DOI: 10.61186/jsdp.19.4.121]

مربوط به اتخاذ استراتژی‌های کشف تقلب پیش‌گیرانه[1]، به‌جای روش‌های برگشت‌پذیر متعارف، به راه حل‌هایی که می‌تواند به‌سمت اجرای مؤثر عملی منجر شود، پرداختند. کیم و همکاران در [11] با در نظر گرفتن عدم توازن نمونه‌ها در تشخیص تقلب یک استراتژی یادگیری جدید با نام قهرمان چالشگر[2] با استفاده از یک روش ترکیبی یادگیری تجمیعی[3] و یادگیری عمیق بر روی کشف تقلب در کارت‌های اعتباری را با استفاده از داده‌های واقعی انجام داده و پیشنهاد کردند. در [12] یک سامانه مبتنی بر یادگیری ماشین تجمیعی به‌منظور کشف خطر اعتبار که در آن تاجران با کدهای MCC[4] نادرست روی سایر قابلیت اطمینان[5] سامانه‌های امتیازدهی[6] تأثیر می‌گذارند و می‌توانند ضررهایی را برای بانک‌ها و سازمان‌های صاحب کارت وارد کنند ارائه دادند. مطالعات مختلفی که در زمینه پیش‌بینی الگوی جرم در گذشته انجام شده‌است نشان می‌دهند جرم یک الگوی جغرافیایی را در فضا و زمان نشان‌می‌دهد. با بیان این نظریه توسط حاجلا و همکاران در ژانویه ۲۰۲۰ در [13] یک رویکرد جدید مبتنی بر خوشه بندی برای شناسایی نقاط مهم[7] برای دسته‌های مختلف جرم با استفاده از وقایع تاریخی به‌عنوان شاخص و یک تکنیک پیش بینی جرم زمانی و مکانی مبتنی بر یادگیری ماشین همراه با تحلیل نقاط مهم دو بعدی ارائه شده‌است. در ژانویه ۲۰۲۰، راتول و همکاران در [14] پژوهشی در مورد جرم و جنایت بر اساس یادگیری ماشین و داده‌کاوی انجام داد. در این پژوهش از بین الگوریتم‌های یادگیری تجمیعی علت استفاده ما از الگوریتم‌های درخت تصمیم با شیب تقویت شده انعطاف پذیری و دقت پیش بینی مناسب و بالا است. در بین این الگوریتم‌ها، روش‌هایی وجود دارد که سازگار با مقادیر گم‌شده[8] نیز هستند. الگوریتم‌های LightGBM و XGBoost دارای کارایی مناسب تری نسبت به‌روش‌هایی چون Lasso، PCA و... در انتخاب ویژگی هستند و نسبت به ماشین بردار پشتیبان نیز سریع‌تر هستند؛ اما در عین حال با وجود عملکرد بسیار مناسب نسبت به سایر روش‌ها نیازمند حافظه بیشتری نیز هستند. در این پژوهش با مقایسه و ترکیب دو الگوریتم LightGBM و XGBoost قصد داریم نرخ خطا را کم و دقت را بالا ببریم و همچنین میزان مصرف حافظه را تا جایی که ممکن است، کاهش دهیم.

۲- کارهای مرتبط

طی دو دهه گذشته سامانه‌های تجمیعی از رشد فزاینده‌ای در جامعه هوش محاسباتی و جامعه یادگیری ماشین برخوردار بوده‌اند و به‌طورکامل سزاوار این توجه بوده‌اند. سامانه‌های تجمیعی ثابت کرده‌اند که در طیف وسیعی از چالش‌ها و کاربردهای دنیای واقعی بسیار کارآمد و متنوع هستند. در اصل برای کاهش واریانس -در نتیجه بهبود دقت- در یک سامانه تصمیم‌گیری خودکار، ایجاد شده‌اند، سامانه‌های تجمیعی با موفقیت برای رفع انواع مشکلات یادگیری ماشین مانند انتخاب ویژگی، تخمین اطمینان[9]، ویژگی‌ها با مقادیر گم‌شده، اصلاح خطا[10]، داده‌های نامتوازن و غیره توسعه‌یافته‌اند. [15]

در این قسمت پژوهش‌های انجام‌شده روی شناسایی و پیش‌گیری تقلب با روش‌های تجمیعی مورد بررسی قرار گرفته است: در ژوئن ۲۰۲۰ گوتیرز-اسپینوزا و همکاران در [16] کشف بازدیدهای جعلی از طریق یادگیری تجمیعی با بررسی مجموعه‌داده رستوران، تشخیص جعلی بودن را ارائه می‌دهد. در فوریه ۲۰۲۰ الطیب الطاهر و همکاران در [17] یک روش هوشمندانه کشف تقلب تراکنش کارت‌های اعتباری را با استفاده از روش LightGBM بهینه شده ارائه داد. در آوریل ۲۰۲۰ آریا و همکاران در [18] یک چارچوب یادگیری تجمیعی عمیق برای کشف تقلب کارت‌های اعتباری جریان داده‌ایی زمان واقعی[11] با استفاده از روش تجمیعی درخت اضافی[12] همراه با یادگیری عمیق برای بهبود دقت پیش‌بینی قطعی و اجتناب از بیش برازش با تراکنش‌های داده‌های واقعی از یک بانک بزرگ را ارائه داده است. در ژانویه ۲۰۲۰، باگا و همکاران در [19] کشف تقلب کارت اعتباری با استفاده از خطوط لوله و یادگیری تجمیعی را ارائه دادند. کوماری و همکاران در [20] به چند رده‌بندی‌کننده گروهی[13] مانند Bagging، جنگل تصادفی، طبقه بندی از طریق رگرسیون و... پرداخته و آنها را با برخی از رده‌بندی‌های منفرد و مؤثر مانند K-نزدیک‌ترین همسایه، شبکه‌های بیزی، ماشین بردار پشتیبان، رده‌بند RBF، پرسپترون چندلایه، درخت تصمیم مقایسه کرده‌اند.

[1] Adopting Preventive Fraud Detection Strategies
[2] Champion-challenger
[3] Ensemble Learning
[4] Merchant Category Code(MCC)
[5] Reliability
[6] Scoring Systems
[7] Hotspot Identification
[8] Missing Values
[9] Reliability Estimation
[10] Error Correction
[11] Real-time Dataflow
[12] Extra Tree Ensemble method
[13] Group Classifier

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]
[DOI: 10.61186/jsdp.19.4.121]

IEEE-CIS در زمینه‌های مختلف هوش مصنوعی و یادگیری ماشین از جمله شبکه‌های عصبی عمیق، سامانه‌های فازی و محاسبات تکاملی کار می‌کند. در این قسمت تعدادی از پژوهش‌های انجام‌شده روی مجموعه‌دادگان تشخیص تقلب IEEE-CIS مورد بررسی قرار گرفته است: در آوریل ۲۰۲۰ ناجادات و همکاران در [21] پژوهش بر روی مجموعه‌داده‌های تشخیص تقلب IEEE-CIS توسط Kaggle با استفاده از مدل‌های یادگیری ماشین و یادگیری عمیق انجام شده‌است و مدل جدیدی که مبتنی بر BiLSTM و BiGRU بود ارائه شده‌است. همچنین مقایسه شش رده‌بندی‌کنندهٔ یادگیری ماشین شامل: شبکه‌های بیزی، Voting، Ada Boosting، جنگل تصادفی، رگرسیون ترابری و نتایج حاصل از رده‌بندی‌کننده یادگیری ماشین نشان‌می‌دهد. در آگوست ۲۰۱۹ گوسویسکا و همکاران در [22] با بیان این مسئله که مهم‌ترین بخش انتخاب مدل و تنظیم هایپرپارامتر ارزیابی عملکرد مدل است و با بیان ضعف‌های مشترک مشهورترین معیارها، مانند AUC، F1-Score، ACC برای رده‌بندی دودویی معیار MAD و از RMSE برای رگرسیون، یا cross-entropy برای رده‌بندی چندلایه به‌منظور حل مسئله روش جدید EPP[1] (قدرت پیش‌بینی‌کننده مبتنی بر Elo) را ارائه می‌دهد. در مارس ۲۰۲۰ ژانگ و همکاران در [23] یک مدل تشخیص تقلب معامله مبتنی بر XGBoost با مهندسی ویژگی و تجسم بر روی مجموعه داده‌های در رقابت تشخیص تقلب IEEE-CIS، Kaggle را ارائه دادند. در آوریل سال ۲۰۲۰ دینگ لینگ و همکاران در [24] با ارائه مدل مبتنی بر LightGBM بر روی مجموعه‌داده تراکنش تشخیص تقلبIEEE-CIS، Kaggle نشان دادند.

۳- پیش‌زمینه

۳-۱- الگوریتم‌های تقویت‌کننده

تقویت‌کننده یک رده‌بندی‌کننده قوی[2] مبتنی بر مجموعه داده‌های آموزش داده‌شده رده‌بندی‌کننده ضعیف[3] است، و یکی از الگوریتم‌های موفق برای یادگیری نظارت شده‌است [25]. روشی برای تبدیل مجموعه یادگیرنده‌های ضعیف به یادگیرنده‌های قوی است. یک یادگیرنده ضعیف دارای خطای کمتر از ۰/۵ و یادگیرنده قوی دارای خطای نزدیک به صفر است. خانواده‌ای از یادگیرندگان ضعیف در کنار هم جمع می‌شوند تا یک یادگیرنده قوی را تشکیل دهند. سه الگوریتم تقویت‌کننده که زیاد استفاده می‌شود: AdaBoost، Gradient Boost و XGBoost هستند[26] .

درخت تصمیم با گرادیان تقویتی[4] یک الگوریتم تقویت‌کننده است که توسط فریدمن ارائه شده‌است، این الگوریتم از چندین درخت تصمیم تشکیل شده‌است و برای تولید هر درخت از روش نزول گرادیان[5] استفاده می‌شود. بر اساس تمام درخت‌های تصمیم منفرد، بهینه‌سازی با به کمینه‌رساندن تابع ضرر[6] به‌عنوان هدف انجام‌می‌شود[27] . در گرادیان تقویتی، بسیاری از مدل‌ها به‌صورت پیوسته آموزش داده می‌شوند. هر مدل جدید با استفاده از روش نزول گرادیان به‌تدریج تابع ضرر را به کمینه می‌رساند. این مدل پیوسته متناسب با مدل‌های جدید، تخمین دقیق‌تری از متغیر پاسخ ارائه می‌دهد. در اصل این الگوریتم از الگوریتم‌های چندگانه ضعیف برای تولید الگوریتمی دقیق‌تر استفاده می‌کند. استفاده از الگوریتم‌های گرادیان تقویتی بیشتر به‌خاطر دقت بالای آنها است[26] .

نحوه کارGBM [۲۸]

گرادیان[7] به خطا یا باقیمانده‌ای گفته‌می‌شود که پس از ساخت یک مدل به‌دست آمده است. تقویت به بهبود اشاره دارد. این روش به‌عنوان ماشین تقویت گرادیان یا GBM شناخته می‌شود. تقویت گرادیان راهی برای بهبود (کاهش) خطای تدریجی است. برای دیدن نحوه کار GBM، فرض کنید یک مدل M (که براساس درخت تصمیم است) داریم و می‌خواهیم آن را بهبود ببخشیم. مدل را به شرح زیر بیان می‌کنیم:

همان‌طور که در جدول (۱) مشاهده می‌کنید، در مرحله ۱ Y متغیر وابسته است و M (x) درخت تصمیم با استفاده از متغیرهای مستقل x است. اکنون می‌خواهیم خطای درخت تصمیم قبلی را پیش بینی کنیم. مرحله ۲ G (x) درخت تصمیم دیگری است که سعی می‌کند خطا را با استفاده از متغیرهای مستقل x پیش‌بینی کند. در مرحله ۳، مشابه مرحله قبل، مدلی ایجاد می‌کنیم که سعی می‌کند error2 را با استفاده از متغیرهای x مستقل پیش بینی کند و درنهایت در مرحله ۴ همه با هم ترکیب می‌شوند.

[1] Elo-based Predictive Power(EPP)
[2] Strong Classifier
[3] Weak Classifier
[4] Gradient Boosting Decision Tree(GBDT)
[5] Gradient Descent method
[6] Loss Function
[7] Gradient

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]
[DOI: 10.61186/jsdp.19.4.121]

الگوریتم ۱. الگوریتم آموزش LightGBM
Algorithm 1. The training of LightGBM

ورودی: مجموعه آموزشی $\{(x_i . y_i)\}_{i=1}^{N}$
خروجی: بردار $\hat{y}_i^{(t)}$ مدل LightGBM

گام ۱. اولین درخت را با مقدار ثابت مقداردهی اولیه می‌کنید:
$$\hat{y}_i^{(0)} = f_0 = 0$$
گام ۲. با کمینه سازی تابع هزینه، درخت بعدی را آموزش دهید:
$$f_t(x_i) = \arg\min_{f_t} L\left(y_i , \hat{y}_i^{(t-1)} + f_t(x_i)\right)$$
گام ۳. مدل بعدی را به‌صورت افزودنی به‌دست آورید:
$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$
گام ۴. مراحل ۲ و ۳ را تکرار کنید تا شرط توقف برقرار شود.
گام ۵. مدل اولیه را به‌دست آورده و به خروجی ارسال کنید:
$$\hat{y}_i^{(t)} = \sum_{t=0}^{M-1} f_t(x_i)$$

Require: input: Training set $\{(x_i . y_i)\}_{i=1}^{N}$
Ensure: output: LightGBM model $\hat{y}_i^{(t)}$
Step 1. Initialize the first tree as a constant:
$\hat{y}_i^{(0)} = f_0 = 0$
Step 2. Train the next tree by minimizing the loss function:
$f_t(x_i) = \arg\min_{f_t} L\left(y_i , \hat{y}_i^{(t-1)} + f_t(x_i)\right)$
Step 3. Get the next model in an additive manner:
$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$
Step 4. Repeat the Step 2 and Step 3 until the model reaches the stop condition.
Step 5. Obtain and return the final model:
$\hat{y}_i^{(t)} = \sum_{t=0}^{M-1} f_t(x_i)$

جدول ۱. مدل GBM
Table1. GBM Model

هاگام	هاعملیات
Step 1:	$Y = M(x) + error$
Step 2:	$error = G(x) + error2$
Step 3:	$error2 = H(x) + error3$
Step 4:	$Y = M(x) + G(x) + H(x) + error3$

LightGBM-۳-۲

LightGBM یکی از الگوریتم‌های تقویت گرادیان است که بر اساس الگوریتم درخت تصمیم‌گیری است که برای رده‌بندی و بسیاری از کارهای دیگر یادگیری ماشین مورد استفاده قرار می‌گیرد. این الگوریتم کپسوله‌ای از انواع داده را فراهم می‌کند که موجب کاهش مصرف حافظه بر روی اشیا داده مانند Numpy، Pandas، Array و... می‌شود. دلیل این امر این است که فقط باید هیستوگرام گسسته را ذخیره کرد. آموزش پیش‌فرض درخت تصمیم در LightGBM استفاده از الگوریتم هیستوگرام است. این گزینه در XGBoost نیز موجود است، اما با مقادیر پیش‌فرض ویژگی‌های از قبل مرتب شده‌است. این الگوریتم تنها از الگوریتم‌های مبتنی بر درخت استفاده می‌کند. LightGBM علاوه‌بر دقت، دارای کارایی بسیار بالا نیز است[27] . این الگوریتم بر اساس الگوریتم‌های درخت تصمیم‌گیری استوار است، در حالی که الگوریتم‌های تقویت‌کننده دیگر عمق یا سطح درخت را تقسیم می‌کنند، این الگوریتم برگ درخت را با بهترین تناسب تقسیم می‌کند؛ بنابراین وقتی همان برگ در LightGBM رشد می‌کند، الگوریتم‌های leaf-wise می‌تواند باعث کاهش تلفات بیشتر از الگوریتم level-wise شود و از این رو دقت بسیار بهتری حاصل می‌شود که به‌ندرت به‌وسیلهٔ هر یک از الگوریتم‌های تقویت‌کننده موجود می‌توان به‌دست‌آورد. در مقایسه با الگوریتم‌های معمول یادگیری ماشین دارای مزایایی چون آموزش سریع‌تر، مصرف حافظه کمتر، یادگیری موازی، پردازش داده در مقیاس بزرگ و... است. همچنین LightGBM از بسط تیلور تابع هزینه و شروط تنظیم برای کنترل پیچیدگی مدل استفاده می‌نماید.

XGBoost -۳-۳

XGBoost یکی از کارآمدترین روش‌های پیاده‌سازی درختان تصمیم‌گیری با گرادیان تقویتی است و به‌عنوان یکی از بهترین الگوریتم‌های یادگیری ماشین شناخته می‌شود. به‌طور خاص، این الگوریتم برای بهینه‌سازی استفاده از حافظه و بهره‌برداری از قدرت محاسبات سخت‌افزاری طراحی شده‌است، XGBoost با افزایش عملکرد نسبت به بسیاری از الگوریتم‌های یادگیری ماشین، زمان اجرا را کاهش می‌دهد. ایده اصلی تقویت، ساختن زیردرختانی از درخت اصلی است، به‌طوری که هر درخت بعدی خطاهای درخت قبلی را کاهش می‌دهد. به این ترتیب، زیرشاخه‌های جدید باقی‌مانده‌های قبلی را به‌منظور کاهش خطای تابع هزینه، به‌روز می‌کنند.

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]
[DOI: 10.61186/jsdp.19.4.121]

الگوریتم ۲. الگوریتم آموزش XGBoost

Algorithm 1. The training of XGBoost

Require: input: Training set $\{(x_i . y_i)\}_{i=1}^{N}$

ورودی: مجموعه آموزشی $\{(x_i . y_i)\}_{i=1}^{N}$

Ensure: output: XGBoost model $\hat{y}_i^{(t)}$

خروجی: بردار $\hat{y}_i^{(t)}$ مدل XGBoost

Step 1. Initialize the first tree as a constant:

گام ۱. اولین درخت را با مقدار ثابت مقداردهی اولیه می کنید:

$$\hat{y}_i^{(0)} = f_0 = 0$$

Step 2. Train the next tree by minimizing the loss function:

گام ۲. با کمینه سازی تابع هزینه، درخت بعدی را آموزش دهید:

$$f_t(x_i) = \arg\min_{f_t} \left(\frac{1}{2} \left[\frac{\left(\sum_{i \in I_L} g_i\right)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i\right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{\left(\sum_{i \in I} g_i\right)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \right)$$

$$g_i = \frac{\partial L\left(y_i , \hat{y}_i^{(t-1)}\right)}{\partial \hat{y}_i^{(t-1)}} \quad , \quad h_i = \frac{\partial^2 L\left(y_i , \hat{y}_i^{(t-1)}\right)}{\partial^2 \hat{y}_i^{(t-1)}}$$

Step 3. Get the next model in an additive manner:

گام ۳. مدل بعدی را به صورت افزودنی بدست آورید:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$

Step 4. Repeat the Step 2 and Step 3 until the model reaches the stop condition.

گام ۴. مراحل ۲ و ۳ را تکرار کنید تا شرط توقف برقرار شود.

Step 5. Obtain and return the final model:

گام ۵. مدل اولیه را بدست آورده و به خروجی ارسال کنید:

$$\hat{y}_i^{(t)} = \sum_{t=0}^{M-1} f_t(x_i)$$

پیش‌بینی را به‌دست آوریم و ارزیابی کنیم.

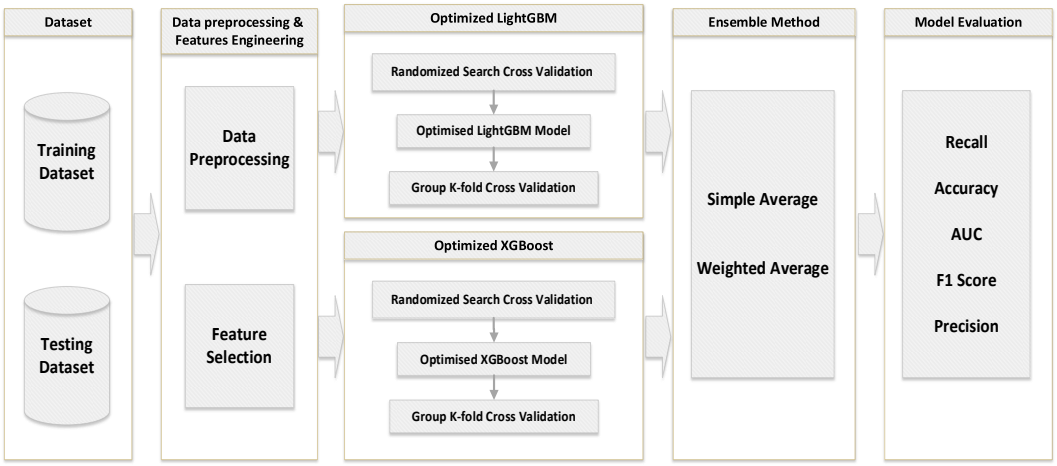
۴- روش پیشنهادی

در این مقاله داده‌های تراکنش و شناسایی را ادغام کرده و پس از مراحل پیش‌پردازش و حل چالش داده‌های گم‌شده، حل عدم توازن داده، کار بر روی داده‌های عددی و غیر عددی و انجام مهندسی ویژگی به آموزش الگوریتم پرداختیم. همان‌طور که در شکل (۲) مشاهده می‌کنید، برپایهٔ الگوریتم LightGBM بهینه‌شده و الگوریتم XGBoost بهینه‌شده به‌وسیله تنظیم هایپرپارامترها ما مدل را آموزش می‌دهیم و سپس با استفاده از روش‌های میانگین گیری ساده و وزن‌دار در یادگیری تجمیعی نتایج دو مدل را با روش با هم ترکیب می‌کنیم تا نتایج نهایی

جدول ۲. بهترین پارامترهای الگوریتم LightGBM

Table 2- The best parameters of the LightGBM algorithm

نام پارامتر	محدوده پارامترها	بهترین مقدار هر پارامتر
num_leaves	200 – 600	256
feature_fraction	0.3 – 0.6	0.5
bagging_fraction	0.3 – 0.7	0.4
min_data_in_leaf	40 – 140	80
max_depth	-1 , 5:11	-1
learning_rate	0.002 – 0.01	0.01
reg_alpha	0.01 : 1	0.01
reg_lambda	0.01 : 1	0.01

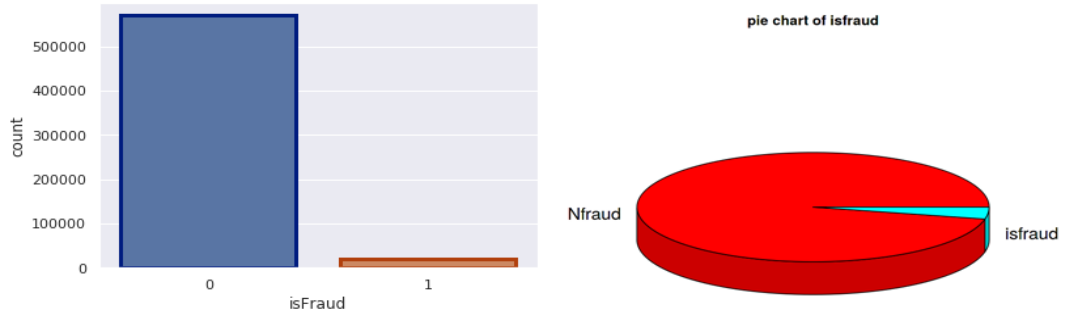


شکل ۲. دیاگرام جریان کار الگوریتم پیشنهادی

Figure 2. Proposed Method Flow Diagram

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]



شکل ۳. نمایش عدم توازن متغییر وابسته "isfraud"

Figure 3. plots of "isfraud" unbalanced target variable

جدول ۳. بهترین پارامترهای الگوریتم XGBoost

Table 3- The best parameters of the XGBoost algorithm

نام پارامتر	محدوده پارامترها	بهترین مقدار هر پارامتر
max_leaves	50 – 400	72
min_child_weight	0 – 10	2
max_depth	-1 , 5	0
learning_rate	0.002 – 0.05	0.04
reg_alpha	0.01 : 1	0.01
n_estimators	800-1000	800
Subsample	0.7-0.9	0.74
colsample_bytree	0.5-1	0.89

۴-۱- فرآیند مدل‌سازی

پس از پیش‌پردازش داده‌ها، به‌ترتیب از الگوریتم LightGBM و الگوریتم XGBoost برای آموزش مدل استفاده می‌کنیم. مقادیر پارامترها از طریق روش Randomize Search cross validation و سعی و خطا با تغییر بازه مقادیر برای به‌دست‌آوردن مقادیر بهینه و بهینه‌سازی مدل به‌دست می‌آیند که در جدول ۲ و ۳ مشاهده می‌کند؛ سپس نتایج پیش‌بینی‌شده به‌وسیلهٔ مدل LightGBM و نتایج پیش‌بینی‌شده به‌وسیلهٔ مدل XGBoost هر دو وارد مدل تجمیعی می‌شوند. سرانجام، ترکیب نتایج پیش‌بینی‌ها به‌روش میانگین گیری انجام‌می‌شود.

۵- پیاده‌سازی

۵-۱- مجموعه‌داده

مجموعه‌داده کشف تقلب کارت‌های اعتباری IEEE-CIS مربوط به سری مسابقات کگل به نشانی https://www.kaggle.com/c/ieee-fraud-detection است.

در زمینه‌های مختلف هوش مصنوعی و یادگیری ماشین‌کار می‌کند، از جمله شبکه‌های عصبی عمیق، سامانه‌های فازی، محاسبات تکاملی. امروز آنها با شرکت خدمات پرداخت برتر جهان، Vesta Corporation همکاری می‌کنند و به‌دنبال بهترین راه‌حل‌ها برای صنعت پیش‌گیری از کلاه‌برداری هستند. شرکت وستا مجموعه داده این مسابقه را ارائه داده است. شرکت وستا پیشگام راه حل‌های پرداخت تضمینی تجارت الکترونیکی است. وستا در سال ۱۹۹۵ تأسیس شد و در روند معاملات پرداخت به‌طورکامل تضمینی کارت غیر موجود (CNP) برای صنعت ارتباطات راه دور پیشگام بود. از آن زمان، وستا به‌صورتی پایدار و محکم توانمندی‌های علم داده و یادگیری ماشین را در سرتاسر جهان گسترش داده و جایگاه خود را در رأس پرداخت‌های تجارت الکترونیکی تقویت کرده است. امروز وستا معاملات سالانه بیش از ۱۸ میلیارد دلار را تضمین می‌کند.

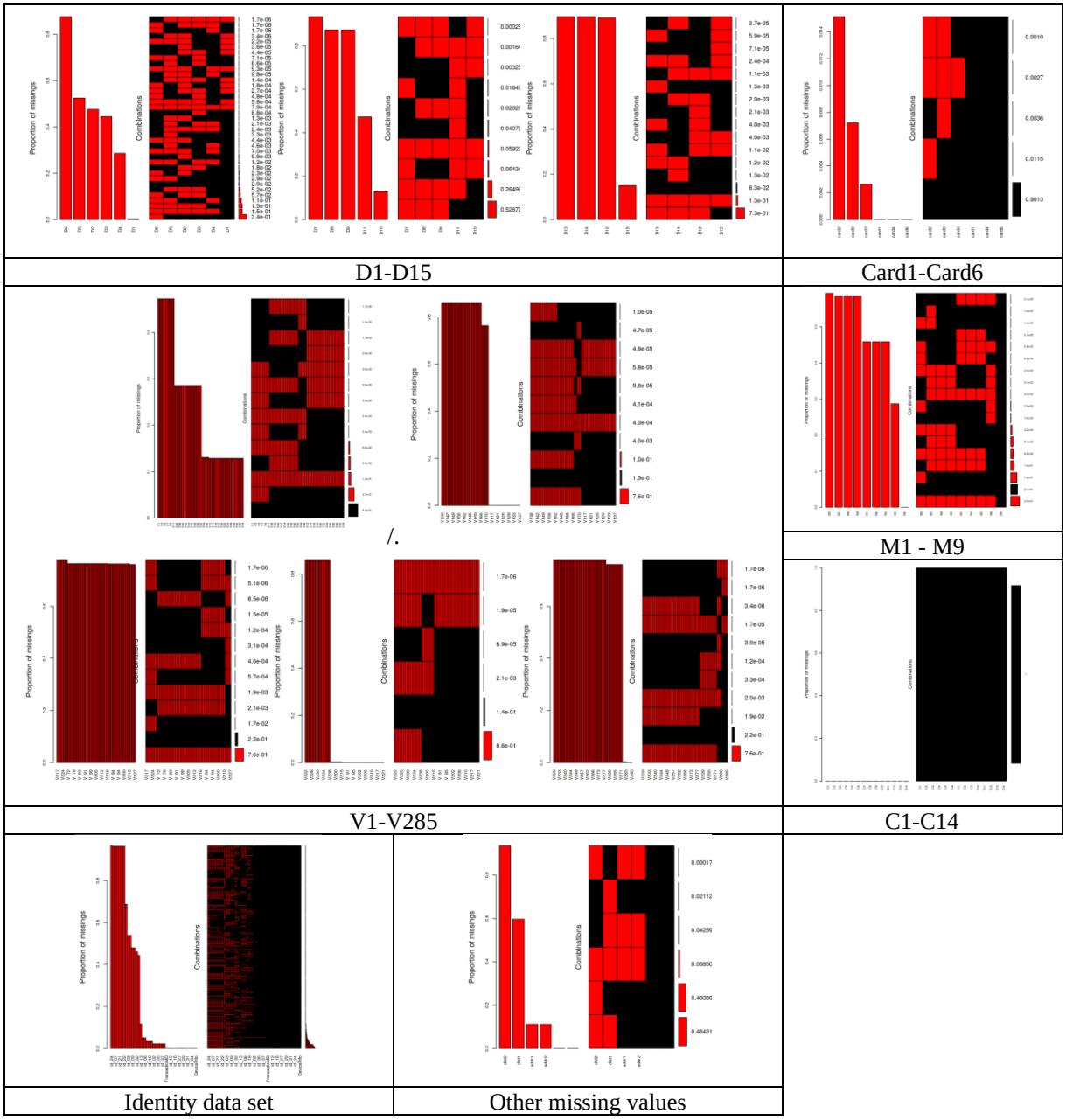
مجموعه‌داده شامل چهار مجموعه که دو مجموعه‌داده آموزش و آزمون برای Transaction data و دو مجموعه‌داده آموزش و آزمون مربوط به identity data است است. حجم بالای مجموعه‌داده و وجود ویژگی‌های بسیار زیاد و متنوع اعم از عددی و غیرعددی باعث می‌شد به فضای بالایی برای حافظه نیاز داشته باشیم و از چالش‌های مهم کار محسوب می‌شد.

ویژگی "isFruad" برچسب ردهٔ حاصل است که در صورت تقلب معادل "fruad" و درصورت سالم‌بودن معادل "Notfruad" نشان داده‌می‌شود. همانطور که در شکل (۳) دیده‌می‌شود، توزیع متغیر پاسخ بسیار نامتوازن است. فقط ۳/۵٪ از تراکنش‌ها در مجموعه‌داده به‌عنوان تقلب تعیین شدند و بقیه به‌عنوان تراکنش‌های سالم شناخته می‌شوند که در نمودار barplot اعداد در محور عمودی نشان‌دهنده تعداد تراکنش‌ها و محور افقی (صفر و یک) به‌ترتیب نشان‌دهنده تراکنش سالم و تقلب است.

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

مجموعه‌داده دارای مقدار زیادی مقادیر گم‌شده است، زمانی که دو مجموعه‌داده ترکیب می‌شود، در ۴۱۱ ویژگی از کل ۴۳۳ ویژگی مقادیر ازدست‌رفته داریم. بیش از ۴۷٪ ویژگی‌ها بالای ۷۰٪ مقدار ازدست‌رفته دارند. ویژگی‌هایی که بیش از ۹۹٪ مقدار از دست رفته داشته‌باشند را می‌توانیم به‌طور کامل حذف کنیم. در شکل (۴) مقادیر گمشده ویژگی‌ها را مشاهده می‌کنید. به‌علت حجم بالا داده‌ها را تفکیک کرده و بررسی انجام شد. همانطور که مشاهده می‌کنید نمودارaggregate plot اطلاعات زیادی در مورد میزان داده‌های گم‌شده می‌دهد. رنگ سیاه شامل داده‌های مشاهده‌شده و قرمز رنگ شامل داده‌های گم‌شده است. اعدادی که سمت راست ملاحظه می‌کنید از تقسیم تعداد مشاهده هر حالت بر کل نمونه‌ها است. برای درک بهتر مثالی را بیان می‌کنیم. در نمودار other missing value نشان‌می‌دهد در ۴۶/۴۳۱٪ کل نمونه‌ها dist1 و dist2 دارای مقادیر گم‌شده هستند و باقی یعنی ویژگی‌های addr1 و addr2 دارای مقادیر مشاهده‌شده هستند.

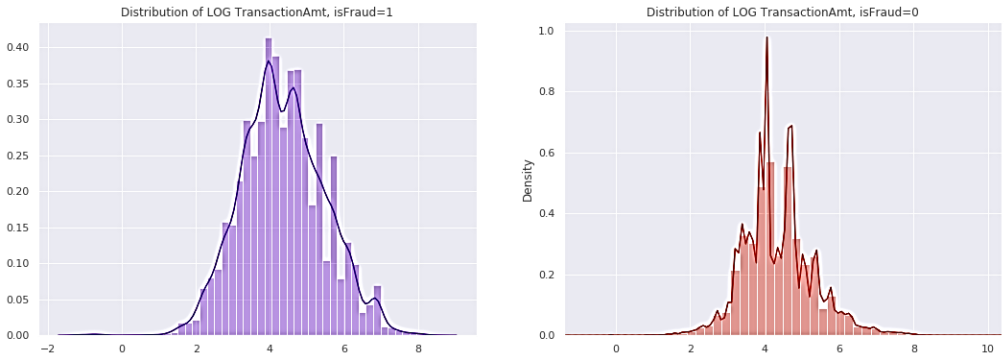


شکل ۴. نمودار aggrplot از ویژگی‌های دارای مقادیر گم‌شده

Figure 4. aggrplot of missing values features

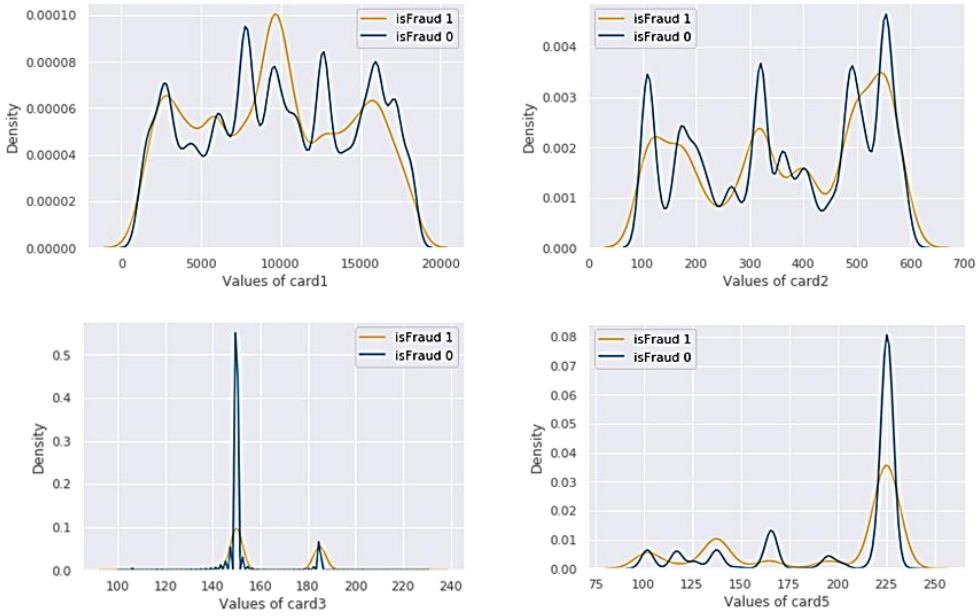
[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]



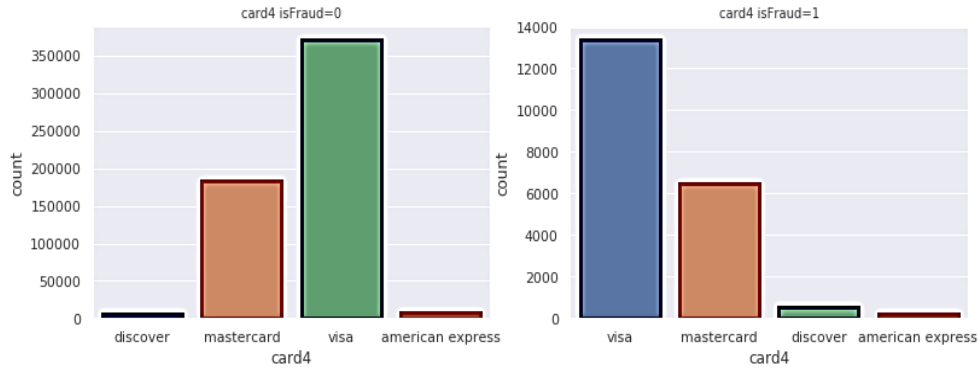
شکل ۵. نمودار توزیع لگاریتم ویژگی TransactionAMT

Figure 5. Distribution of log TransactionAMT



شکل ۶. برخی از متغیرهای categorical، به‌علت داشتن مقادیر منحصربه‌فرد زیاد رفتاری مشابه متغیرهای عددی دارند.

Figure 6. numerical-like behavior of some categorical features (Card1, Card2, Card3, and Card5)



شکل ۷. مقدار تراکنش‌های نرمال و تقلب در شبکه پردازشی بر اساس ویژگی Card4

Figure 7. The amount of fraud and normal transactions in processing networks based on Card4

۵-۲- داده‌های آموزش و آزمون

مجموعه‌داده را روی یک لپ‌تاپ با پردازنده intel core2Duo و حافظه ۴ گیگابایت اجرا شد، اما به‌علت بالابودن حجم مجموعه‌داده‌ها اجرای الگوریتم ممکن نبود؛ بنابراین مجموعه‌داده‌ها را در فضاهای Python و R در بستر ابری کگل که ۱۶ گیگ رم و ۱۰۰ گیگ هارد در اختیار می‌گذاشت اجرا شده‌است. همچنین در این سرویس می‌توان در حالت‌های مختلف TPU، GPU و CPU الگوریتم را اجرا کرد که آزمایش‌ها در حالت GPU انجام شده‌است

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

داده‌های آموزش در identity dataset شامل ۱۴۴۲۳۳ نمونه و داده‌های آزمون آن شامل ۱۴۱۹۰۷ نمونه و ۴۰ ویژگی است. داده‌های آموزش dataset Transaction شامل ۵۹۰۵۴۰ نمونه و ۳۹۳ ویژگی و داده‌های آزمون آن شامل ۵۰۶۶۹۱ نمونه است. در ادامه به بررسی برخی از ویژگی‌ها می‌پردازیم.

در شکل (۵) توزیع مبلغ تراکنش به‌صورت لگاریتم ویژگی (TransactionAMT) در نمودارها نشان داده شده‌است که با توجه به نمودارها می‌توان نتیجه گرفت که تراکم میانگین مبلغ در تراکنش‌های تقلب بیشتر از تراکنش‌های عادی است، یعنی این میزان در مبالغ رو به بالا بیشتر دیده‌می‌شود و این نوعی هشدار برای جابه‌جایی با مبالغ بالا است. در نتیجه کلاه‌برداران تمرکز بیشتری روی مبالغ بالا دارند که بدین صورت امنیت آن نیز به همان میزان باید توسط شبکه‌های پردازشی و بانک‌ها تأمین شود.

ویژگی‌های card1 تا card6 نشان‌دهنده اطلاعات کارت اعم از نوع کارت، گروه کارت، بانک صادرکننده کارت، کشور و... هستند که از نوع categorical هستند. همانطور که در شکل (۶) مشاهده می‌کنید، برخی از این ویژگی‌ها به‌علت داشتن مقادیر منحصربه‌فرد بسیار، رفتاری مشابه مقادیر پیوسته دارند، مانند card1 و card2 که به‌ترتیب دارای ۱۳۵۵۳ و ۵۰۰ مقدار منحصربه‌فرد هستند. محور عمودی میزان تراکم و محور افقی مقادیر منحصربه‌فرد را نشان‌می‌دهد.

ویژگی card4 شبکه پردازشی تراکنش‌ها را نشان‌می‌دهد. با توجه به شکل (۷) visa card و master card دارای بیشترین تراکنش تقلب و discoverو american express دارای کمترین میزان تراکنش تقلب هستند.

ویژگی card6 انواع کارت‌های اعتباری را نشان‌می‌دهد. همان طور که در شکل (۸) مشاهده می‌کنید، میزان استفاده debit و بعد از آن credit به‌ترتیب بسیار بالا است. و به همان میزان تراکنش تقلب فقط در کارت‌های debit و credit صورت پذیرفته است که debit card دارای تقلب بیشتری نسبت به credit card است؛ ولی با این وجود می‌توان گفت credit card با توجه به حجم کمتر تراکنش‌ها نسبت به debit card بیشتر در معرض خطر است و شاید این بدین معنی باشد که debit card پروتکل‌های امنیتی بیشتری را رعایت می‌کند.

۵-۳- بحث

همان‌طور که گفته شد، از معیارهای مختلفی استفاده شده‌است، اما در مورد AUC توضیحاتی بیان شده‌است و دلیل آن خواست مسابقه بر این معیار بوده است. بر اساس خواست مسابقه در تعیین معیار، یکی از معیارهای ارزیابی را AUC (Area Under the ROC Curve) قرار دادیم. AUC نشان‌دهنده سطح زیر نمودار ROC (Receiver Operating Characteristic) است که هر چه مقدار این عدد مربوط به یک رده‌بند بزرگ‌تر باشد، کارایی نهایی رده‌بند مطلوب‌تر ارزیابی می‌شود. به‌عبارت دیگر، به‌بیشینه‌رساندن AUC ROC به بیشینه‌رساندن همبستگی رتبه هدف و پیش‌بینی است. می‌توان ROC curve را نیز به‌صورتی که در فرمول‌های ۱ و ۲ نشان دادیم، بیان کرد.

$$ROC\ AUC = \frac{Cov\left(y, rank\left(\mu\right)\right)}{Cov\left(y, rank\left(y\right)\right)} * 0.5 + 0.5 \qquad (۱)$$

$$ROC\ AUC = \frac{Cov\left(rank\left(y\right), rank\left(\mu\right)\right)}{Cov\left(rank\left(y\right), rank\left(y\right)\right)} * 0.5 + 0.5 \qquad (۲)$$

استراتژی ارزیابی Group Kfold cross validation است. برای یافتن بهترین پارامترها از روش Randomize Search cross validation و سعی و خطا با تغییر بازه مقادیر تعیین کردیم.

یکی از چالش‌های مجموعه‌داده نامتوازن‌بودن آن است که به‌شدت مشاهده می‌شود. استفاده از پارامتری به نام scale_pos_weight که وزن رده مثبت را تعیین می‌کند تا چالش داده‌های نامتوازن را رفع کند. مقدار پیش‌فرض این پارامتر برابر ۱ است. مقدار ۱ یعنی داده‌ها متوازن هستند. مقدار این پارامتر با روش‌های مختلفی محاسبه می‌شود. مقدار مناسب این پارامتر را همان‌طور که در فرمول ۳ مشاهده می‌کنید، براساس تعداد کل نمونه‌ها تقسیم‌بر تعداد نمونه‌های مثبت من‌های یک محاسبه کردیم. این پارامتر در الگوریتم‌های پیشنهادی موجود است.

$$\frac{Total\ Samples}{Positive\ Samples} - 1 \qquad (۳)$$

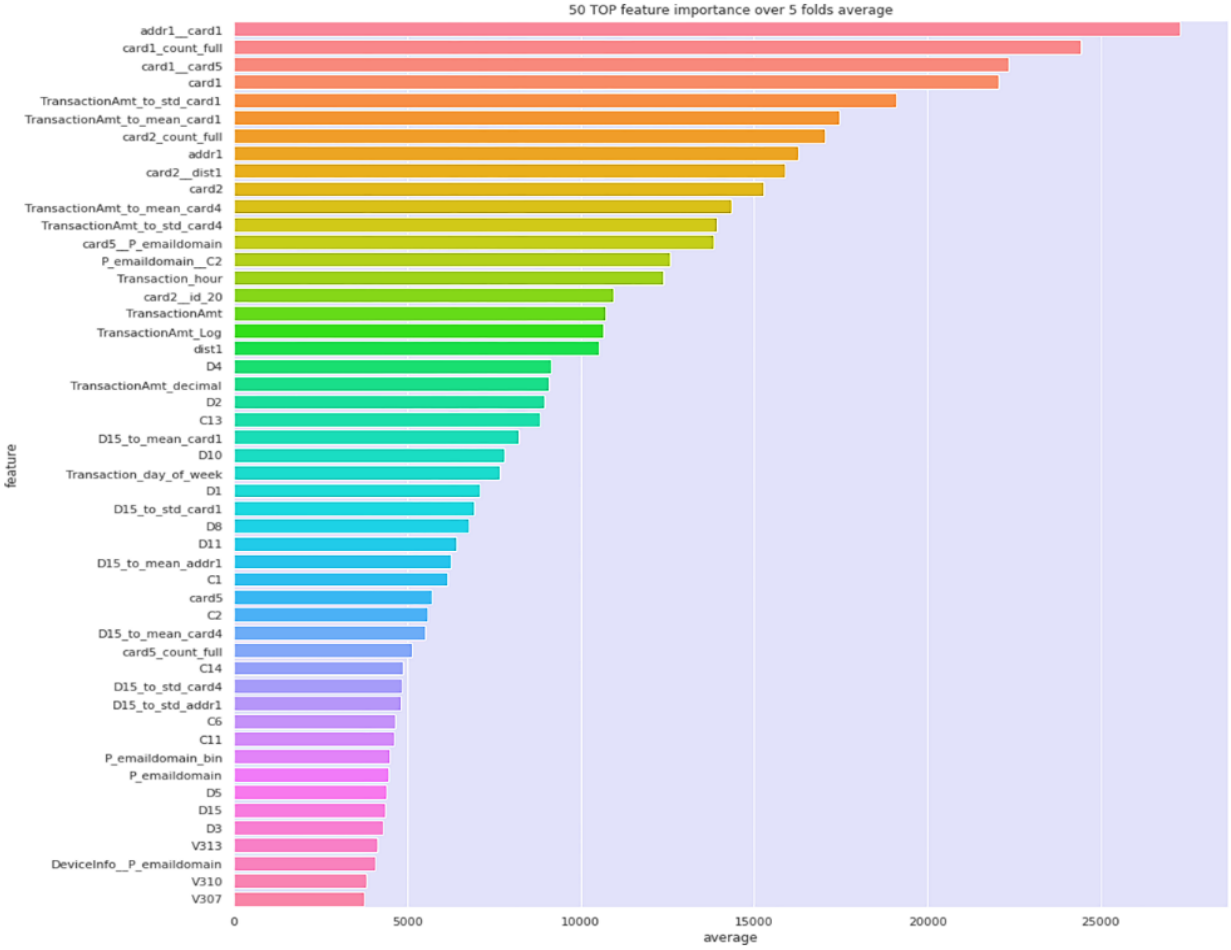
در مجموعه‌داده انواع مختلفی از ویژگی‌ها از جمله غیرعددی وجود دارد که این نوع داده‌ها را مورد بررسی قرار داده و با استفاده از Label Encoding نیاز خود برای رفع این چالش را حل کردیم. البته می‌توان گفت Label Encoding تنها راه انکودکردن نیست. می‌توان از روش‌های دیگر نظیر one-hot encoding نیز استفاده‌کرد. تفاوت روش‌ها در موقعیت استفاده از آن است. اغلب در شبکه‌های عصبی از one-hot encoding بیشتر استفاده می‌شود و برای درخت به‌طورمعمول از Label Encoding استفاده می‌شود که موقعیتی در جایگاه درختی است.

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

مـدل را بـا اسـتفاده از طبقـه‌بنـدی‌کننـده‌هـای LightGBM و XGBoost ، یک‌بار با مـدل LightGBM، یک‌بار با مدل XGBoost و یک‌بـار بـه‌صـورت ترکیبـی بـا استفاده از Averaging methode اجرا کـردیم. در اجـرای نخسـت، بـا اسـتفاده از ویژگـی feature_importance الگوریتم LightGBM، میزان اهمیت ویژگی‌ها بررسی شـد که در شکل (۹) می‌توانید مشاهده کنیـد. پنجـاه مـورد از بااهمیت‌ترین و مؤثرترین ویژگی‌هـای مجموعـه‌دادگـان را براسـاس ۵ بـار اجـرا آورده‌ایـم. ویژگـی‌هـایی کـه وزن پایین‌تری داشتند (ویژگی‌هـایی کـه میـزان اهمیـت آنهـا نزدیک به صفر بـود) را حـذف و دوبـاره الگـوریتم را اجـرا کردیم.

نتایج اجرای الگـوریتم‌هـا بـا اسـتفاده از اسـتراتژی ارزیـابی Group Kfold بـا ۵ فولـد بـا معیارهـای AUC، Accuracy، Percision، Recall و F1 score قبل و بعـد از عملیات مهندسی ویژگی‌ها را در جـدول ۴ تـا ۷ مشـاهده می‌کنیـد. در الگـوریتم تجمیعـی بـه‌روش میـانگین‌گیری وزن‌دار که روش توسعه یافته میانگین گیری ساده است به مـدلی کـه دارای ارزیـابی بهتـری اسـت، وزن بیشـتری اختصاص داده‌می‌شـود، بـه‌دلیـل ارزیـابی بهتـر الگـوریتم XGBoost وزن ۰/۶ و به الگوریتم LightGBM وزن ۰/۴ را اختصاص داده‌شده‌است. نکته حائز اهمیت در وزن‌دهـی که باید به آن توجه داشت این است که مجمـوع وزن‌هـای مدل باید یک باشد. همان‌طورکه در جدول ۴ تا ۷ مشاهده می‌شود، نتایج قبل از انجام مهندسی ویژگی دارای مقـادیر کمتری نسبت به مقادیر پس از مهندسـی ویژگـی اسـت و این بدین معناست که انتخاب ویژگی‌های تأثیرگذار تا چـه اندازه مـی‌توانـد مهـم باشـد. همچنـین ترکیـب تجمیعـی LightGBM و XGBoost به‌روش میانگین گیـری سـاده و وزن‌دار پس از مهندسی ویژگی‌ها با معیـار مسـابقه یعنـی AUC ۹۴/۶۹ و ۹۵/۰۸ نتـایج بهتـری را در مقایسـه بـا حالت‌های دیگر نشان‌مـی‌دهـد. در جـدول (۸) مقایسـه‌ای بین روش‌های یادشده و برخـی روش‌هـای متـداول دیگـر مانند شبکه‌های بیزی، جنگل تصادفی و رگرسیون ترابـری انجام شده‌است، همان طـور کـه مشـاهده مـی‌شـود، روش تجمیعـی میـانگیری وزن‌دار پـس از اعمـال مهندسـی ویژگی‌ها بهترین مقادیر ارزیابی را دارد..



شکل ۹. مهم‌ترین ویژگی‌ها

Figure 9. The most important features

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

جدول ۴. نتایج پیاده‌سازی مدل LightGBM

Table 4. The Result of LightGBM

	بدون استفاده از مهندسی ویژگی					با استفاده از مهندسی ویژگی				
Fold	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***
1	91.14	98.10	83.82	57.05	67.89	92.24	98.23	85.95	59.07	70.02
2	92.19	98.90	84.91	87.34	86.11	93.09	99.07	86.60	89.86	88.20
3	93.32	99.10	87.06	88.87	87.95	94.24	99.21	88.87	90.09	89.47
4	93.60	98.89	87.87	84.39	86.10	95.12	99.21	90.69	89.33	90.01
5	93.11	98.97	86.81	83.98	85.37	94.00	99.10	88.53	85.85	87.17
Avg	92.67	98.79	86.09	80.33	82.68	93.74	98.96	88.13	82.84	84.97

جدول ۵. نتایج پیاده‌سازی مدل XGBoost

Table 5. The Result of XGBoost

	بدون استفاده از مهندسی ویژگی					با استفاده از مهندسی ویژگی				
Fold	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***
1	93.13	98.28	87.72	59.70	71.05	94.90	98.38	91.25	60.79	72.97
2	93.31	99.09	87.02	90.10	88.54	94.37	99.19	89.13	90.51	89.82
3	94.24	99.21	88.87	90.09	89.47	94.98	99.28	90.33	90.54	90.44
4	92.96	99.05	86.35	89.06	87.68	94.81	99.21	90.04	89.75	89.89
5	94.13	99.11	88.77	85.98	87.36	95.16	99.26	90.74	88.31	89.51
Avg	93.55	98.95	87.75	82.99	84.82	94.84	99.06	90.30	83.98	86.53

جدول ۶. نتایج پیاده‌سازی مدل میانگین‌گیری ساده

Table 6. The Result of Simple Average Method

	بدون استفاده از مهندسی ویژگی					با استفاده از مهندسی ویژگی				
Fold	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***
1	93.56	99.12	87.72	78.39	82.79	94.37	99.18	89.31	79.27	83.99
2	93.84	99.14	88.08	90.31	89.18	94.32	99.20	89.03	90.85	89.93
3	94.69	99.25	89.77	90.28	90.02	95.21	99.31	90.79	90.85	90.82
4	93.72	99.12	87.87	89.42	88.64	94.22	99.17	88.85	89.98	89.41
5	94.78	99.23	90.00	88.12	89.05	95.35	99.29	91.11	88.73	89.91
Avg	94.12	99.17	88.69	87.30	87.94	94.69	99.23	89.82	87.94	88.81

جدول ۷. نتایج پیاده‌سازی مدل میانگین‌گیری ویزن دار

Table 7. The Result of Weighted Average Method

	بدون استفاده از مهندسی ویژگی					با استفاده از مهندسی ویژگی				
Fold	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***
1	93.92	99.15	88.42	78.78	83.32	94.90	99.22	90.37	79.76	84.73
2	93.95	99.15	88.29	90.43	89.34	94.65	99.23	89.66	91.14	90.40
3	94.81	99.26	90.00	90.40	90.20	95.56	99.34	91.46	91.15	91.31
4	93.83	99.13	88.09	89.55	88.81	94.55	99.21	89.50	90.28	89.89
5	94.91	99.25	90.25	88.25	89.24	95.72	99.33	91.85	89.07	90.44
Avg	94.28	99.19	89.01	87.48	88.18	95.08	99.27	90.57	88.28	89.35

جدول ۸. مقایسه نتایج نهایی چهار مدل پیشنهادی و سه روش متداول

Table 8. the comparison of total results of four proposed methods and other three common methods

	بدون استفاده از مهندسی ویژگی					با استفاده از مهندسی ویژگی				
Method	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***	***AUC***	***Acc.***	***Recall***	***Prec.***	***F-Meas.***
Naïve base	86.87	95.53	74.62	77.44	70.85	87.60	98.43	75.98	78.86	76.91
Random Forest	88.83	98.51	78.44	78.40	78.42	90.37	98.65	81.42	82.54	81.97
Logistic Regression	89.92	97.21	79.85	89.84	81.79	91.36	98.81	83.36	82.31	82.83
LightGBM	92.67	98.79	86.09	80.33	82.68	93.74	98.96	88.13	82.84	84.97
XGBoost	93.55	98.65	87.75	82.99	84.82	94.84	99.06	90.30	83.98	86.53
Simple Average	94.12	**99.21**	88.69	87.30	87.94	94.69	99.23	89.82	87.94	88.81
Weighted Average	**94.28**	99.19	**89.01**	**87.48**	**88.18**	**95.08**	**99.27**	**90.57**	**88.28**	**89.35**

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

۶- نتیجه‌گیری

شناسایی تقلب در کارت‌های اعتباری به‌عنوان یک مسئله جدی برای سازمان‌های مالی مانند بانک‌ها و شرکت‌های کارت اعتباری شناخته شده‌است. با کشف سریع تراکنش متقلبانه می‌توان از خسارات هنگفت جلوگیری کرد. در صنعت کارت اعتباری، استانداردهای ثابتی برای توسعه مدل کشف تقلب به‌عنوان مجموعه ایی از مدل‌های متنوع وجود داشت. در این پژوهش مطالعه ایی بین مدل‌های XGBoost و LightGBM با معیارهای ارزیابی AUC، Accuracy، Percision، Recall و F1- score انجام شده‌است تا مشخص شود (کدام مدل‌ها در داده‌های تراکنش‌های حجیم دنیای واقعی عملکرد بهتری نسبت به مدل‌های دیگر دارند.) مدل یادگیری تجمیعی به‌روش میانگین‌گیری ساده و میانگین‌گیری وزن‌دار برای توسعه و مقایسه دو مدل معرفی شده‌است.

مجموعه‌داده کشف تقلب کارت‌های اعتباری مربوط به مسابقه IEEE-CIS fraud detection از سایت کگل گرفته شده‌است و روی آن پژوهش و آزمایش لازم انجام شد. در این پژوهش برای کاهش مصرف حافظه مصرفی از تابع reduce_mem_usage استفاده شده‌است که عملکرد مناسبی در کاهش مصرف حافظه از خود نشان داد. در راه‌کار پیشنهادی، به‌وسیله سعی و خطا نتیجه به‌دست آمده نشان داد بهترین نتیجه برای داده‌های گم‌شده زمانی به‌دست می‌آید که به‌جای پرداختن به داده‌های گم‌شده و برآورد آن، از خود الگوریتم‌های سازگار با داده‌های گم‌شده یعنی LightGBM و XGBoost استفاده شود. همچنین تعداد ویژگی‌ها بسیار زیاد بودند که با استفاده از مهندسی ویژگی‌ها و استفاده از ویژگی feature_importance الگوریتم LightGBM در راستای انتخاب ویژگی‌ها مدل کارآمدتری به‌دست آمد.

درمجموع در الگوریتم پیشنهادی جهت بررسی و افزایش عملکرد مدل، دو الگوریتم XGBoost و LightGBM با استفاده از روش‌های میانگین‌گیری ساده و وزن‌دار یادگیری تجمیعی ترکیب شده‌است و سپس با مدل‌های Naïve base، Random Forest، Logistic Regression مقایسه انجام شد که در این میان مدل پیشنهادی نتیجه بهتری در این مجموعه داده داشت.

۷- کارهای آینده

مهندسی ویژگی‌ها اهمیت بالایی در رسیدن به عملکرد مناسب ایفا می‌کند. با بررسی دقیق‌تر ویژگی‌ها و مبحث اتخاب ویژگی‌های جدید می‌توان به مجموعه‌ای دقیق‌تر و با اهمیت‌تر رسید که با هزینه کمتر دقت بیشتری را به ارمغان آورد. همچنین استفاده و ترکیب مدل‌های دیگر هم از نظر تعداد مدل‌ها و هم از نظر نوع الگوریتم آموزشی مانند شبکه‌های عصبی عمیق ممکن است، دقت و عملکرد مدل را افزایش دهد. از طرفی یادگیری افزایشی و خودکار نیز یک کار مهم است که در آینده می‌توان به آن پرداخت. برای کنارآمدن با رانش مفهوم، مدل باید الگوهای تقلبی را از جریان معاملات به‌طور مداوم و بدون فراموش‌کردن دانش موجود بیاموزد. در حالی که یادگیری در مورد داده‌های بسیار نامتوازن در محیط یادگیری استاتیک بررسی شده‌است، یادگیری از جریان داده‌های غیر ثابت قابل بررسی است. برای یک فرایند یادگیری به‌طورکامل خودکار، علاوه‌بر یادگیری مدل، عواملی مانند استخراج داده، تغییر شکل (transformation)، پیش پردازش و ارزیابی مدل نیز باید خودکار باشد. همچنین روش‌های دیگر یادگیری تجمیعی نیز ممکن است در بهبود عملکرد مدل مؤثر باشند که می‌توان در آینده به آنها پرداخت.

۸- مراجع 8-References

[1] t. o. c. cart. [Online]. Available: [https://www.thebalance.com/key-differences-between-visa-mastercard-discover-anmerican-express-4588450#citation-4.

[2] "Performance Evaluation of Credit Card Fraud Transactions using Boosting Algorithms, " International Journal of Electronics Communication and Computer Engineering, vol. 10, no. 6, pp. 262-270, 2019.

[3] J. Huang, "Credit Card Transaction Fraud Using Machine Learning Algorithms, " in 2019 International Conference on Education Science and Economic Development (ICESED 2019), 2020.

[4] Y. Ganin, E. Ustinova, H. Ajakan and P. Germain, "Domain-Adversarial Training of Neural Networks, " The Journal of Machine Learning Research, vol. 17, no. 1, pp. 2030-2096, 2016.

[5] M. Raza and U. Qayyum, "Classical and deep learning classifiers for anomaly detection," 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), pp. 614-618, 2019.

[6] B. Lebichot, Y.-A. L. Borgne, L. He-Guelton, F. Oblé and G. Bontempi, "Deep-learning domain adaptation techniques for credit cards fraud detection," in NNS Big Data and Deep Learning conference, Cham, 2019.

[7] A. A. Abdulrazaq, M. B. Abdulrazaq, I. J. Umoh and E. A. Adedokun, "Fraud Detection

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]

classifiers, " Computational Intelligence in Data Mining, pp. 111-122, 2019.
[21] H. Najadat, O. Altiti, A. A. Aqouleh and M. Younes, "Credit Card Fraud Detection Based on Machine and Deep Learning, " in 11th International Conference on Information and Communication Systems (ICICS), IEEE, 2020.
[22] G. Alicja, M. Bakala, K. Woznica, M. Zwolinski and P. Biecek, "EPP: interpretable score of model predictive power., " arXiv, p. preprint arXiv:1908.09213, 2019 Aug 24.
[23] Z. Yixuan, J. Tong, Z. Wang and F. Gao, "Customer Transaction Fraud Detection Using Xgboost Model, " in International Conference on Computer Engineering and Application (ICCEA), IEEE, 2020 Mar 18.
[24] D. J. G. S. C. a. J. C. Ge, "Credit Card Fraud Detection Using Lightgbm Model., " in International Conference on E-Commerce and Internet Technology (ECIT), IEEE, 2020 Apr 22.
[25] J. Choi, B. Jeong, Y. Park, J. Seo and C. Min, "AN OPTIMAL BOOSTING ALGORITHM BASED ON NONLINEAR CONJUGATE GRADIENT METHOD, " Journal of the Korean Society for Industrial and Applied Mathematics, vol. 22, no. 1, pp. 1-13, 2018.
[26] D. Kavya and K. Chitharanjan, "Performance Evaluation of Credit Card Fraud Transactions using Boosting Algorithms, " International Journal of Electronics Communication and Computer Engineering, vol. 10, no. 6, pp. 262-270, 2019.
[27] Y. Liang, W. Jiyu, W. Wei, C. Yujun, Z. Biliang, C. Zhenkun and L. Zhenzhang, "Product marketing prediction based on XGboost and LightGBM algorithm, " the 2nd International Conference on Artificial Intelligence and Pattern Recognition, pp. 150-153, 2019.
[28] V. K. Ayyadevara, "Gradient Boosting Machine, " Pro Machine Learning Algorithms, pp. 117-134, 01 July 2018.
[29] P. KHANDELWAL, "Which algorithm takes the crown: Light GBM vs XGBOOST?," 12 June 2017. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/06/which-algorithm-takes-the-crown-light-gbm-vs-xgboost/.
[30] S. Mittal and S. Tyagi, "Computational Techniques for Real-Time Credit Card Fraud Detection., " Handbook of Computer Networks and Cyber Security, pp. 653-681, 2020.

in Credit Card and Application of VAT Clustering Algorithm: A Review, " in 2019 2nd International Conference of the IEEE Nigeria Computer Chapter (NigeriaComputConf), 2019, October.
[8] Y. Lucas, P.-E. Portier, L. Laporte, L. He-Guelton, O. Caelen, M. Granitzer and S. Calabretto, "Towards automated feature engineering for credit card fraud detection using multi-perspective HMMs," Future Generation Computer Systems, vol. 102, pp. 393-402, 2020.
[9] I. Sadgali, N. Sael and F. Benabbou, "Comparative Study Using Neural Networks Techniques for Credit Card Fraud Detection, " The Proceedings of the Third International Conference on Smart City Applications, 2019.
[10] R. Saia and S. Carta, "Evaluating the benefits of using proactive transformed-domain-based techniques in fraud detection tasks, " uture Generation Computer Systems, vol. 93, 2019.
[11] E. Kim, J. Lee, H. Shin, H. Yang, S. Cho, S.-k. Nam and e. al, "Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning," Expert Systems with Applications, vol. 128, pp. 214-224, 2019.
[12] C.-H. Su, F. Tu, X. Zhang, B.-C. Shia and T.-S. Lee, "A ENSEMBLE MACHINE LEARNING BASED SYSTEM FOR MERCHANT CREDIT RISK DETECTION IN MERCHANT MCC MISUSE," Journal of Data Science, vol. 17, no. 1, pp. 81-106, 2019.
[13] G. M. C. A. R. Hajela, "A Clustering Based Hotspot Identification Approach For Crime Prediction, " Procedia Computer Science, vol. 167, pp. 1462-1470, 2020.
[14] R. Md and A. Rab, "A Comparative Study on Crime in Denver City Based on Machine Learning and Data Mining., " arXiv preprint arXiv:2001.02802, 2020 Jan 9.
[15] R. Polikar, Ensemble Learning, M. Y. Zhang C., Ed., Boston, Massachusetts: Springer, 19 January 2012.
[16] L. F. A. A. S. N. K. S. J. D. R. S. Gutierrez-Espinoza, "Fake Reviews Detection through Ensemble Learning., " arXiv preprint arXiv:2006.07912, 2020 Jun 14.
[17] A. A. a. S. J. M. Taha, "An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine., " IEEE Access 8, vol. 8, pp. 25579-25587, 2020 Feb 3.
[18] M. H. S. G. Arya, "DEAL–'Deep Ensemble ALgorithm'Framework for Credit Card Fraud Detection in Real-Time Data Stream with Google TensorFlow., " Smart Science, vol. 8, no. 2, pp. 71-83, 2020 Apr 2.
[19] S. A. G. N. G. A. G. Bagga, "Credit Card Fraud Detection using Pipeling and Ensemble Learning, " Procedia Computer Science, vol. 173, pp. 104-112, 2020 Jan 1.
[20] P. Kumari and S. P. Mishra, "Analysis of credit card fraud detection using fusion

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]
[DOI: 10.61186/jsdp.19.4.121]



دکتر سعید بختیاری عضو هیئت علمی و استادیار دانشگاه امین و همچنین مشاور امنیتی سازمان‌های دولتی و بانکی است. در سال ۲۰۰۹، وی کارشناسی مهندسی نرم‌افزار را از دانشگاه آمل و در سال‌های ۲۰۱۱ و ۲۰۱۶ ، به‌ترتیب مدرک کارشناسی‌ارشد و دکترای خود را در زمینه امنیت شبکه و اطلاعات از **UTM** مالزی دریافت کرد. فعالیت‌های مورد علاقه وی رمزنگاری و داده‌کاوی است.

Saeid_bakhtiarii@yahoo.com



زهرا نصیری داوطلب دکترای هوش مصنوعی و فارغ‌التحصیل مهندسی - نرم‌افزار رایانه در دانشگاه آل طه در تهران، ایران است. وی چندین سال تجربه برنامه‌نویسی هوش مصنوعی و یادگیری ماشین دارد. یادگیری ماشین، داده‌کاوی، بهینه‌سازی و رایانش ابری از جمله علایق پژوهشی وی است. وی دارای اعتبار حرفه‌ای در زمینه علوم داده از وزارت علوم، تحقیقات و فناوری است.

Lnasiri007@gmail.com



سید محمد صادق حجازی کارشناس ارشد مهندسی نرم‌افزار رایانه در دانشگاه پردیسان در مازندران، ایران است. وی چندین سال تخصص برنامه‌نویسی و تدریس هوش مصنوعی و یادگیری ماشین دارد. یادگیری ماشین، یادگیری عمیق، تحلیل داده، داده‌کاوی، بهینه‌سازی، رایانش ابری و اینترنت اشیا از جمله علایق پژوهشی وی است.

Sadegh.hejazi@hotmail.com

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-13]

[DOI: 10.61186/jsdp.19.4.121]