ICT Research Institute
(Iran Telecommunication Research Center)

Journal of Information and Communication Technology

Iranian Society of Information and Communication Technology

Volume 17, Issue 63-64, Spring and Summer 2025, pp. 155-165

Analysis of Traffic Data for Congestion Detection Using Machine Learning Algorithms

Nahid Amani[1], Mohammad Hossein Amerimehr[1], Sara Efazati[1*], Ali Javidani[2]

[1] ICT Research Institute (ITRC), Tehran, Iran
[2] School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran

Received: 01 July 2024, Revised: 20 January 2025, Accepted: 15 February 2025
Paper type: Research

Abstract

Congestion detection is one of the basic elements in guaranteeing the quality of service and is one of the most important measures of efficiency in new-generation telecommunication networks. Congestion leads to the loss of information packets in the network because the sending rate is higher than the capacity in some network links. To control congestion in the network, the first step is to distinguish between the loss of packets due to congestion with the packet loss introduced by other cases, including link failure. Because if the source of the loss is mistakenly identified as congestion, reducing the sending rate in the transmitter does not mitigate the congestion and only reduces throughput and quality of service. Therefore, the main problem of this paper is to identify congestion and distinguish it from the error caused by communication links in a traffic data sample. In solving this problem, various supervised machine learning algorithms including decision tree, random forest, support vector machine, logistic regression, K-nearest neighbor, XGBoost, and neural network have been used. The mentioned algorithms have been evaluated based on different criteria and compared with each other.

Keywords: Congestion Detection, Machine Learning, XGBoost, Evaluation Criteria of ML Algorithms.

[*] Corresponding Author's email: s.efazati@itrc.ac.ir

دوفصلنامه
فناوری اطلاعات و ارتباطات ایران

سال هفدهم، شماره‌های ۶۳ و ۶۴ ، بهار و تابستان ۱۴۰۴، صفحه ۱۵۵ الی ۱۶۵

تحلیل داده‌های ترافیکی با هدف تشخیص ازدحام با بهره‌گیری از الگوریتم‌های یادگیری ماشین

ناهید امانی[۱]، محمدحسین عامری‌مهر[۱]، سارا افاضاتی[۱*]، علی جاویدانی[۲]

[۱] گروه مدیریت یکپارچه شبکه، پژوهشکده فناوری ارتباطات، پژوهشگاه ارتباطات و فناوری اطلاعات ، تهران، ایران

[۲] دانشکده مهندسی برق و کامپیوتر، دانشکدگان فنی، دانشگاه تهران، تهران، ایران

تاریخ دریافت: ۱۴۰۳/۰۴/۱۱ تاریخ بازبینی: ۱۴۰۳/۱۱/۰۱ تاریخ پذیرش: ۱۴۰۳/۱۱/۲۷

نوع مقاله: پژوهشی

چکیده

شناسایی ازدحام یکی از ارکان پایه‌ای در تضمین کیفیت خدمت به عنوان مهم‌ترین معیار سنجش کارایی در شبکه‌های مخابراتی نسل جدید است. ازدحام منجر به از دست رفتن بسته‌های اطلاعاتی در شبکه میشود که به دلیل بالاتر بودن نرخ ارسال از ظرفیت در برخی از لینک‌های شبکه رخ میدهد. برای کنترل ازدحام در شبکه گام نخست تمایز میان از دست رفتن اطلاعات در اثر ازدحام و یا در اثر سایر موارد از جمله خرابی لینک است زیرا در صورتی که منشا از دست رفتن بسته به اشتباه ازدحام تشخیص داده شود کاهش نرخ ارسال در فرستنده کمکی به کاهش ازدحام نکرده و تنها موجب کاهش گذردهی و کیفیت سرویس می شود. از این رو مسئله اصلی این مقاله شناسایی ازدحام و تفکیک آن از خطای ناشی از لینک‌های ارتباطی در یک نمونه داده ترافیکی است. در این مقاله برای حل مسئله مذکور از الگوریتم‌های مختلف یادگیری ماشین نظارت‌شده از جمله درخت تصمیم، جنگل تصادفی، ماشین بردار پشتیبان، لجیستیک رگرسیون، K-نزدیکترین همسایه، XGBoost، شبکه عصبی و تقویت درخت تصمیم بهره گرفته شده است. الگوریتم‌های مذکور بر اساس معیارهای مختلف از جمله دقت، درستی، F1-measure، حساسیت و AUC مورد ارزیابی قرار گرفته و با یکدیگر مقایسه شده‌اند. این ارزیابی بر اساس روش K-fold Cross Validation انجام شده است. نتایج شبیه‌سازی نشان می‌دهد که الگوریتم XGBoost به لحاظ تمام معیارهای ارزیابی نسبت به دیگر الگوریتم‌ها عملکرد بهتری در زمینه تشخیص ازدحام دارد.

کلیدواژگان: تشخیص ازدحام، یادگیری ماشین، XGBoost، معیارهای ارزیابی الگوریتم‌های یادگیری ماشین

* رایانامهٔ نویسنده مسؤول: s.efazati@itrc.ac.ir

۱- مقدمه

امروزه با توسعه سریع فناوری‌ها، تقاضا برای سرویس‌های متنوع مخابراتی با کیفیت‌های مورد انتظار متنوع به طور گسترده‌ای افزایش یافته است. از این رو فراهم کردن کیفیت خدمت (QoS) و جلب رضایت‌مندی مشتریان از اهداف اساسی نسل‌های جدید شبکه‌های مخابراتی ازجمله 5G، B5G و 6G می‌باشد[۱-۲]. در راستای رسیدن به هدف تضمین QoS راهکارهای مختلفی توسط محققان ارائه و پیاده‌سازی شده است. یکی از راهکارهای کلیدی در تضمین QoS کنترل ازدحام در ترافیک شبکه است[۳]. منظور از ازدحام شرایطی در شبکه است که بسته‌های ارسالی دقیقا به علت شلوغی و ازدحام شبکه گم می‌شوند. به عبارت دیگر ازدحام به حالتی گفته می‌شود که نرخ ارسال در برخی از لینک‌های شبکه از ظرفیت لینک بیشتر شده که موجب از دست رفتن بسته یا تاخیر زیاد در ارسال بسته می‌شود. در روش‌های سنتی تشخیص و کنترل ازدحام به طور ضمنی از تاخیر و اتلاف بسته برای تشخیص ازدحام و تغییر سایز پنجره ازدحام (CWND) استفاده می‌شود. به عنوان مثال در روش TCP Tahoe در ابتدای الگوریتم CWND تا رسیدن به حد آستانه افزایش می‌یابد و مرتبا دو برابر می‌شود [۴]. سپس در گام بعد در صورتی که از دست رفتن بسته مشاهده نشود (ACK بسته دریافت شود)، CWND به اندازه یک واحد افزایش می‌یابد. اما در صورت اتلاف بسته حدآستانه برابر با نصف مقدارCWND فعلی قرار داده می‌شود و CWND نیز به مقدار اولیه بازنشانی می‌شود. در الگوریتم-های دیگر مثل TCP Reno و TCP New Reno نیز مشابه با TCP Tahoe با توجه به از دست رفتن بسته، CWND و بالطبع نرخ ارسال تغییر می‌کند تا ازدحام در شبکه کنترل شود [۵-۶]. در روش TCP Vegas، تغییر CWND بر اساس میزان تاخیر ارسال بسته (RTT) انجام می‌شود [۵].

مشکل روش‌های سنتی کنترل ازدحام آن است که نمی‌توانند منشا از دست رفتن بسته را تشخیص دهند و هر اتلاف بسته را ناشی از ازدحام تلقی کرده و راهکارهای کنترلی را اعمال می‌کنند. در حالیکه اتلاف بسته به ویژه در شبکه‌های بی‌سیم می‌تواند ناشی از بروز خطا در لینک‌های ارتباطی به واسطه عواملی همچون نویز، محوشدگی کانال بی‌سیم، تداخل، تحرک کاربران و handover، قطع شدن به دلیل پوشش محدود رادیویی، تاخیر و... باشد. عدم تشخیص ازدحام از سایر عوامل از دست رفتن بسته و اعمال مکانیزم‌های کنترل ازدحام مانند کاهش CWND بی‌مورد بوده و تنها باعث افت گذردهی و عملکرد شبکه می‌شود. لذا تشخیص منشا از دست رفتن بسته و تشخیص ازدحام امری بسیار مهم و قابل توجه به شمار می‌رود. در روش‌های سنتی، تشخیص ازدحام به طور ضمنی از روی تاخیر و اتلاف بسته بر اساس یک سری قوانین از پیش‌تعریف شده محاسبه می‌شود [۷]. در حالیکه روش‌های مبتنی بر قانون‌های از پیش تعریف شده بهینه نبوده و در شرایط واقعی ممکن است عملکرد مناسبی از خود ارائه ندهند. از این رو روش‌های هوشمند مبتنی بر یادگیری ماشین در حل این مسئله مطرح می‌شوند.

در روش مبتنی بر یادگیری ماشین از روی وضعیت شبکه تشخیص ازدحام انجام می‌شود. لذا تشخیص ازدحام مبتنی بر یادگیری ماشین از هوشمندی و دقت بیشتری نسبت به روش‌های سنتی تشخیص ازدحام برخوردار است [۸]. در یادگیری ماشین با تجزیه و تحلیل داده‌هایی که به عنوان ورودی دریافت می شود فرایند یادگیری انجام شده و عملکرد سیستم به طور خودکار از طریق تجربه بهبود می‌یابد. از میان انواع الگوریتم‌های یادگیری ماشین، الگوریتم‌های یادگیری نظارت‌شده از دقت بالاتری در حل مسئله تشخیص ازدحام برخوردارند [۹-۱۰]. در این روش آموزش بر اساس داده‌های برچسب‌دار انجام می‌شود و الگوریتم سعی می‌کند پارامترهای خود را بر اساس این داده‌ها بروزرسانی کند به نحوی که خطای پیش‌بینی داده‌ها به حداقل برسد.

در این مقاله به حل مسئله تشخیص ازدحام و تفکیک آن از سایر عوامل از دست رفتن بسته با بهره‌گیری از یادگیری ماشین نظارت شده پرداخته می‌شود. هدف اصلی پیداکردن الگوریتم یادگیری ماشین نظارت‌شده‌ای است که بتواند بهترین عملکرد را بر اساس معیارهای ارزیابی مختلف در حل مسئله تشخیص ازدحام ارائه دهد.

در این زمینه، [۱۱] به بررسی تشخیص ازدحام در یک شبکه حسگری بی‌سیم با استفاده از یادگیری ماشین پرداخته است. در این مقاله برای تشخیص ازدحام از دو الگوریتم شبکه عصبی و ماشین بردار پشتیبان استفاده شده است. البته در این مقاله بحث تفکیک ازدحام از دیگر خطاهای ارتباطی وجود ندارد بلکه با دریافت سه پارامتر مختلف از شبکه، رخ دادن ازدحام در شبکه پیش‌بینی می‌شود.

در [۱۲] از الگوریتم‌های مبتنی بر یادگیری ماشین برای شناسایی ازدحام استفاده شده است. الگوریتم یادگیری ماشین استفاده شده در این مقاله الگوریتمی مبتنی بر الگوریتم ماشین بردار پشتیبان است که به شناسایی و کنترل ازدحام در شبکه‌های مبتنی بر حسگر اینترنت اشیا می‌پردازد.

[۱۳] یک مرور جامع بر روی مقالاتی که با استفاده از یادگیری عمیق به شناسایی ازدحام پرداخته‌اند انجام داده است. [۱۴-۱۵]

نیز از الگوریتم‌های مبتنی بر یادگیری عمیق برای شناسایی ازدحام استفاده کرده‌اند.

در [۱۶] به مسئله تفکیک ازدحام از دیگر خطاهای ارتباطی با استفاده از یادگیری ماشین پرداخته است. در این مقاله از الگوریتم‌های یادگیری ماشین نظارت‌شده از قبیل درخت تصمیم، جنگل تصادفی و شبکه عصبی استفاده شده است.

همین نویسندگان در مقاله‌ای دیگر از الگوریتم یادگیری نظارت شده‌ای با عنوان تقویت درخت تصمیم و همان مجموعه داده قبلی استفاده کرده‌اند و با نتایج مقاله قبل مقایسه کرده‌اند و نتایج بهتری را به دست آورده‌اند [۱۷].

در مقاله حاضر الگوریتم‌های یادگیری نظارت شده متعددی در مسئله تشخیص ازدحام به کار گرفته شده‌اند، از جمله درخت تصمیم، جنگل تصادفی، ماشین بردار پشتیبان، لجستیک رگرسیون، K-نزدیکترین همسایه، XGBoost، شبکه عصبی چند لایه و تقویت درخت تصمیم. نسبت به مقالات [۱۶] و [۱۷] که نزدیکی بیشتری با این مقاله دارند، از الگوریتمی با دقت بالاتر با نام XGBoost در حل مساله تشخیص ازدحام استفاده شده است و نشان داده شده که این الگوریتم به لحاظ تمام معیار-های ارزیابی از جمله درستی، دقت، حساسیت، F1-measure و AUC عملکردی بهتر از سایر الگوریتم‌های مذکور از خود ارائه می‌دهد. در جدول ۱ مقالات مرتبط با مقاله حاضر آورده شده است.

جدول ۱. کارهای مرتبط با مقاله حاضر

مرجع	مسئله	الگوریتم مورد استفاده
[۴]	کنترل ازدحام به صورت سنتی	TCP Tahoe
[۵]، [۶]	کنترل ازدحام به صورت سنتی	TCP Tahoe TCP Reno TCP New Reno TCP Vegas
[۱۱]	تشخیص ازدحام با استفاده از یادگیری ماشین	Neural network SVM
[۱۲]	کنترل ازدحام با استفاده از یادگیری ماشین	SVM
[۱۶]، [۱۷]	تفکیک ازدحام از دیگر خطاهای ارتباطی با استفاده از یادگیری ماشین	Decision Tree, Random forest, Neural Network, Decision Tree Boosting
مقاله حاضر	تفکیک ازدحام از دیگر خطاهای ارتباطی با استفاده از یادگیری ماشین	Logistic Regression, SVM, Random Forest, KNN, XGBoost, Decision Tree, Neural Network

چارچوب اصلی مقاله حاضر بر اساس موارد زیر بنا شده است:

۱. به کارگیری الگوریتم‌های متنوع یادگیری نظارت شده شامل درخت تصمیم، جنگل تصادفی، ماشین بردار پشتیبان، لجستیک رگرسیون ، K-نزدیکترین همسایه، XGBoost، شبکه عصبی چند لایه و تقویت درخت تصمیم
۲. به کارگیری معیارهای ارزیابی مختلف شامل درستی[1] دقت[2]، F1-measure، حساسیت[3] و AUC در ارزیابی الگوریتم‌ها
۳. مقایسه عملکرد الگوریتم‌های مذکور بر اساس معیارهای ارزیابی و انتخاب بهترین آن‌ها برای حل مسئله تشخیص ازدحام با یکدیگر.

ساختار مقاله در ادامه بدین شرح تنظیم شده است. در بخش دوم به معرفی اجمالی الگوریتم‌های یادگیری نظارت‌شده پرداخته می‌شود. سپس شیوه مقایسه الگوریتم‌های یادیگیری ماشین و معیارهای ارزیابی آن‌ها در بخش سوم مورد بررسی قرار می‌گیرد. در بخش چهارم نتایج شبیه‌سازی ارائه شده و در نهایت در بخش پنجم جمع‌بندی نهایی صورت می‌پذیرد.

۲- مرور اجمالی الگوریتم‌های یادگیری ماشین نظارت شده

همانطور که قبلا اشاره شد الگوریتم‌های یادگیری ماشین نظارت شده بر اساس یک مجموعه داده برچسب‌دار عمل کرده و آموزش می‌بینند. در این بخش به معرفی اجمالی الگوریتم‌های مورد استفاده در این مقاله پرداخته می‌شود.

الف) الگوریتم درخت تصمیم

این الگوریتم یکی از قدیمی‌ترین و در عین حال محبوب‌ترین الگوریتم‌های یادگیری ماشین نظارت‌شده در کاربردهای تصمیم-گیری و طبقه‌بندی است [۱۸]. در این الگوریتم فرآیند تصمیم‌گیری از طریق شاخه‌بندی ویژگی‌ها از مهم‌ترین ویژگی به کم‌اهمیت‌ترین ویژگی انجام می‌شود. هر شاخه نشانگر یک تصمیم یا قانون است. ایده اصلی آن است که مجموعه‌ای از تصمیمات متوالی برای رسیدن به یک نتیجه خاص اتخاذ می‌شوند. مزیت اصلی این الگوریتم در مقایسه با سایر الگوریتم‌ها، پیچیدگی محاسباتی کم در کنار دقت بالا می‌باشد.

ب) الگوریتم جنگل تصادفی

[1] Accuracy
[2] Precision
[3] Recall

الگوریتم جنگل تصادفی بر پایه الگوریتم درخت تصمیم ساخته شده است[۱۸-۱۹]. ایده اصلی این روش آن است که خروجی چندین درخت تصمیم را با یکدیگر ادغام کند. بنابراین به جای تکیه بر خروجی یک درخت تصمیم، نتایج چندین درخت تصمیم را جمع‌آوری کرده و بر اساس میانگین یا اکثریت آرا پیش‌بینی نهایی را ارائه می‌دهد.

ج) الگوریتم ماشین بردار پشتیبان

این الگوریتم برای طبقه‌بندی و رگرسیون استفاده می‌شود. مبنای کار طبقه‌بند SVM دسته‌بندی خطی داده‌ها است و در تقسیم خطی داده‌ها سعی می‌شود خطی انتخاب شود که حاشیه اطمینان بیشتری داشته باشد.

د) الگوریتم لجستیک رگرسیون

این الگوریتم روشی ساده و کارآمد برای مسائل طبقه‌بندی باینری و خطی است و به طور گسترده برای طبقه‌بندی در صنعت به کار گرفته می‌شود [۲۰]. در این روش از یک ابر صفحه در یک فضای m بعدی برای جداسازی نقاط داده با تعداد m ویژگی در کلاس‌هایشان استفاده می‌شود.

ه) الگوریتم K-نزدیکترین همسایه

این الگوریتم که به منظور طبقه‌بندی و رگرسیون مورد استفاده قرار می‌گیرد بر این ایده استوار است که نقاط داده مشابه برچسب‌های مشابه دریافت می‌کنند [۲۱-۲۲]. این تشابه بر اساس مقیاس فاصله بین نقاط داده که می‌تواند به عنوان مثال فاصله اقلیدسی در نظر گرفته شود تعیین می‌شود. عملکرد الگوریتم KNN در عین سادگی و محبوبیت در انتخاب، تحت تأثیر انتخاب پارامتر K و متریک فاصله قرار می‌گیرد. بنابراین تنظیم دقیق پارامتر K برای نتایج بهینه ضروری است.

و) الگوریتم XGBoost

الگوریتم XGBoost یا به عبارتی الگوریتم تقویت گرادیان حداکثر با ترکیب الگوریتم درخت تصمیم و الگوریتم تقویت گرادیان ایجاد شده است [۲۳]. الگوریتم تقویت گرادیان یک مدل پایه طبقه‌بندی ایجاد می‌کند و سپس طی مراحلی عملکرد خود را بهبود می‌بخشد تا به یک مدل پیش‌بینی قوی برسد. به این صورت که در هر مرحله مدل جدیدی که ایجاد می‌شود برای رفع خطاهای مدل قبلی است. XGBoost با ترکیب پیش‌بینی‌های چندین مدل جداگانه درخت‌های تصمیم‌گیری، یک مدل پیش‌بینی جدید می‌سازد. عملکرد مناسب، دقت بالا و انعطاف‌پذیری این روش آن را به یکی از روش‌های پرکاربرد در حوزه‌های مختلف تبدیل کرده است.

ز) الگوریتم شبکه عصبی

الگوریتم شبکه عصبی بر اساس رفتار بیولوژیکی نورون‌های عصبی در مغز عمل می‌کند [۲۴-۲۵]. یک شبکه عصبی مصنوعی تشکیل شده است از گره‌هایی معادل نورون‌های مصنوعی که توسط روابطی معادل سیناپس‌های عصبی با یکدیگر در ارتباط هستند. این الگوریتم شامل یک لایه ورودی، چندین لایه تصمیم‌گیری و یک لایه خروجی است. تعداد لایه‌ها، گره‌ها، روابط و حدود آستانه مشخص شده در هر گره مواردی هستند که به صورت تجربی و خاص هر مسئله تنظیم می‌شوند.

ح) الگوریتم تقویت درخت تصمیم

این الگوریتم در راستای تقویت عملکرد الگوریتم درخت تصمیم ساخته شده است. در این روش ابتدا یک مدل اولیه ساخته می‌شود و به تدریج مدل بهبود می‌یابد. به عبارت دقیق‌تر، هر مدل با استفاده از مدل قبلی و با افزایش وزن نمونه‌های یادگیری که در مدل قبل به اشتباه طبقه‌بندی شده‌اند ساخته می‌شود. سپس پیش‌بینی نهایی با استفاده از مجموعه مدل‌های بدست آمده و بر اساس کلاس اکثریت در بین همه کلاس‌ها ساخته می‌شود [۲۶].

۳- ارزیابی و مقایسه الگوریتم‌های یادگیری ماشین نظارت شده

یکی از اهداف اصلی در این مقاله ارزیابی الگوریتم‌های مختلف یادگیری نظارت‌شده و مقایسه آن‌ها بر اساس معیارهای متفاوت ارزیابی است. مساله مطرح شده در این مقاله تشخیص ازدحام و شناسایی و تفکیک آن از خطای ارتباطی در یک نمونه داده برچسب‌دار است. از آنجا که جنس این مساله از نوع طبقه‌بندی است از روش متداول K-fold Cross Validation جهت ارزیابی طبقه‌بندی انجام شده استفاده می‌شود [۲۷]. روش Cross-Validation شامل دو مرحله اصلی است: تقسیم داده‌ها به زیر مجموعه‌هایی به نام folds و چرخش آموزش و اعتبارسنجی میان آن‌ها. مرحله تقسیم داده‌ها معمولاً ویژگی‌های زیر را دربردارد:

- اندازه هر fold تقریباً یکسان است.
- داده‌ها را می‌توان به صورت تصادفی در هر fold انتخاب کرد.
- همه fold ها، به جز یکی که برای اعتبار سنجی استفاده می‌شود، برای آموزش مدل استفاده می‌شوند. آن fold اعتبارسنجی باید چرخانده شود تا زمانی که همه fold ها یک بار و تنها یک بار به یک fold اعتبارسنجی تبدیل شوند. در

این الگوریتم، عموما توصیه می‌شود که هر نمونه در یک و تنها یک قسمت قرار گیرد.

حرف K در نام این الگوریتم مشخص‌کننده آن است که مجموعه داده‌ها به چند قسمت تقسیم می‌شود. بسیاری از کتابخانه‌های پایتون از k=10 به عنوان یک مقدار پیش‌فرض استفاده می‌کنند که نشان‌دهنده این است که ۹۰٪ داده‌ها برای آموزش و ۱۰٪ از داده‌ها برای اعتبارسنجی استفاده خواهد شد.

چارچوب کلی مقایسه میان الگوریتم‌ها در این مقاله بدین شرح است که در ابتدا پیش‌پردازش‌های مورد نظر بر روی نمونه داده‌های برچسب‌دار موجود انجام می‌شود. سپس داده‌های پیش‌پردازش شده بر اساس روش k-fold به صورت تصادفی به k دسته تقسیم می‌شوند تا برای آموزش و اعتبارسنجی الگوریتم یادگیری مورد نظر از آن‌ها استفاده شود. الگوریتم مورد نظر اجرا شده و بر اساس داده‌های آموزشی و داده‌های اعتبارسنجی که در روش k-fold به آن داده شده معیار ارزیابی مشخص شده را محاسبه می‌کند. در ادامه به معرفی معیارهای ارزیابی مد نظر در این مقاله پرداخته شده است.

۳-۱- معیارهای ارزیابی

در شناسایی الگو معیارهای ارزیابی متداول برای مسائل از جنس طبقه‌بندی، عموما درستی، دقت ، حساسیت ، F1-Measure و AUC هستند. معیارهایی همچون دقت، درستی و F1-measure و حساسیت که بر اساس ارزیابی تعداد نمونه‌های درست و نادرست در یک مجموعه داده سنجیده می‌شوند از ابزاری به نام ماتریس درهم‌ریختگی در ارزیابی خود استفاده می‌کنند. از سوی دیگر معیار AUC که یک معیار عددی در سنجش عملکرد یک مدل یادگیری است از منحنی ROC مشتق می‌شود. در ادامه به شرح کامل ماتریس در هم‌ریختگی و منحنی ROC پرداخته شده و سپس معیارهای ارزیابی مستخرج از آن‌ها مورد بررسی قرار گرفته‌اند. ماتریس درهم-ریختگی، عملکرد یک مدل یادگیری ماشین را بر روی مجموعه ای از داده‌های آزمایشی خلاصه می‌کند. در حقیقت این ماتریس، ابزاری برای نمایش تعداد نمونه‌های درست و نادرست بر اساس پیش‌بینی-های مدل است. از این ماتریس برای اندازه‌گیری عملکرد مدل‌های طبقه‌بندی، که هدف آن پیش‌بینی یک برچسب طبقه‌بندی برای هر نمونه ورودی است، استفاده می‌شود. برای یک مساله طبقه‌بندی دو کلاسه، ۴ عنصر زیر در ماتریس درهم‌ریختگی تعریف می‌شوند:

- مثبت واقعی (TP): زمانی رخ می‌دهد که مدل به درستی، یک داده مثبت را پیش‌بینی کند.
- منفی واقعی (TN): زمانی رخ می‌دهد که مدل به درستی، یک داده منفی را پیش‌بینی کند.
- مثبت کاذب (FP): زمانی رخ می‌دهد که مدل به اشتباه، یک داده منفی را مثبت پیش‌بینی کند.
- منفی کاذب (FN): زمانی رخ می‌دهد که مدل به اشتباه، یک داده مثبت را منفی پیش‌بینی کند.

هنگام ارزیابی عملکرد یک مدل طبقه‌بندی، بررسی ماتریس درهم‌ریختگی ضروری است. این ماتریس یک تجزیه و تحلیل کامل از پیش‌بینی‌های مثبت واقعی، منفی واقعی، مثبت کاذب و منفی کاذب ارائه می‌دهد و درک عمیق‌تر از دقت، درستی و اثربخشی کلی مدل در تمایز طبقاتی را تسهیل می‌کند.

منحنی ROC نموداری است که ارزیابی یک سیستم طبقه‌بندی دودویی را به نمایش می‌گذارد. منحنی ROC توسط ترسیم نرخ مثبت واقعی یا TPR بر حسب نرخ مثبت کاذب یا FPR به دست می‌آید که با فرمول‌های زیر قابل محاسبه هستند.

$$TPR = \frac{TP}{TP + FN} \quad (۱)$$

$$FPR = \frac{FP}{TN + FP} \quad (۲)$$

این مقادیر برای آستانه‌های مختلف در طبقه‌بندی محاسبه می‌شوند و در منحنی ROC معادل یک نقطه می‌شوند. با اتصال این نقاط به یکدیگر، منحنی ROC ایجاد می‌شود. منحنی ROC دارای سه بخش است.

- بالای خط نیمساز: در این بخش نقاطی قرار گرفته‌اند که مقدار حساسیت آن‌ها نسبت به FPR بیشتر است. به معنای آن‌که در این بخش، نرخ مثبت صحیح بیشتر از نرخ مثبت کاذب است. قرار گرفتن نقاط در این محیط مطلوب است.
- روی خط نیمساز: در این بخش، مقدار عددی نرخ مثبت صحیح و نرخ مثبت کاذب با یکدیگر برابر است. این نقاط چندان مطلوب ما نیستند.
- پایین خط نیمساز: در این قسمت نقاطی قرار گرفته‌اند که مقدار حساسیت آن‌ها نسبت به FPR کمتر است. یعنی در این بخش، نرخ مثبت صحیح کمتر از نرخ مثبت کاذب است. قرار گرفتن نقاط در این بخش اصلاً مطلوب نیست.

۳-۱-۱- معیار ارزیابی دقت

یکی از معیارهای ارزیابی دقت می‌باشد. در حقیقت این معیار بیان می‌کند که چند درصد از داده‌هایی که کلاس آن‌ها مثبت تشخیص

داده شده، واقعا مثبت هستند. دقت به بیان ریاضی به صورت زیر تعریف می‌شود:

$$Precision = \frac{TP}{TP + FP} \quad (۳)$$

۳-۱-۲- معیار ارزیابی حساسیت

از دیگر معیارهای ارزیابی حساسیت است. این معیار بیان می‌کند که چند درصد از داده‌هایی که کلاس آن‌ها مثبت بوده، به درستی مثبت تشخیص داده شده‌اند. حساسیت به بیان ریاضی به صورت زیر تعریف می‌شود:

$$Recall = \frac{TP}{TP + FN} \quad (۴)$$

۳-۱-۳- معیار ارزیابی F1-Measure

معیار F1-Measure تجمیع دو معیار دقت و حساسیت بوده و هر دو جنبه عملکردی این دو معیار را در بر دارد. این معیار به بیان ریاضی به صورت زیر تعریف می‌شود:

$$F1 - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (۵)$$

۳-۱-۴- معیار ارزیابی درستی

معیار درستی نیز همانند F1-Measure عناصر ماتریس درهم‌ریختگی را در خود تجمیع کرده است. عملکرد طبقه‌بند با در نظر گرفتن تنها همین عدد درستی به طور کامل مورد سنجش قرار گرفته و قابلیت مقایسه الگوریتم‌های مختلف را فراهم می‌ورد. درستی به بیان ریاضی به صورت زیر تعریف می‌شود:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (۶)$$

۳-۱-۵- معیار ارزیابی AUC

به سطح زیر منحنی ROC، AUC گفته می‌شود. مقدار عددی AUC عددی بین صفر تا یک است و نشان می‌دهد قدرت تشخیص یک طبقه‌بند چقدر است. اگر این عدد به یک نزدیک باشد به معنای آن است که نقاط موجود روی منحنی ROC عموماً در بالای خط نیمساز قرار گرفته‌اند و میزان نرخ مثبت صحیح بالا است. اعداد AUC نزدیک به ۰,۵ همان برابری نرخ مثبت صحیح و نرخ مثبت کاذب را نشان می‌دهد و نشان می‌دهد که طبقه‌بند نزدیک به حالت تصادفی داده‌ها را طبقه‌بندی می‌کند. اعداد کمتر از ۰,۵ بیانگر بالاتر بودن نرخ مثبت کاذب در مقایسه با نرخ مثبت صحیح است و نشان می‌دهد که طبقه‌بند حتی بدتر از حالت تصادفی عمل می‌کند.

۴- نتایج شبیه سازی

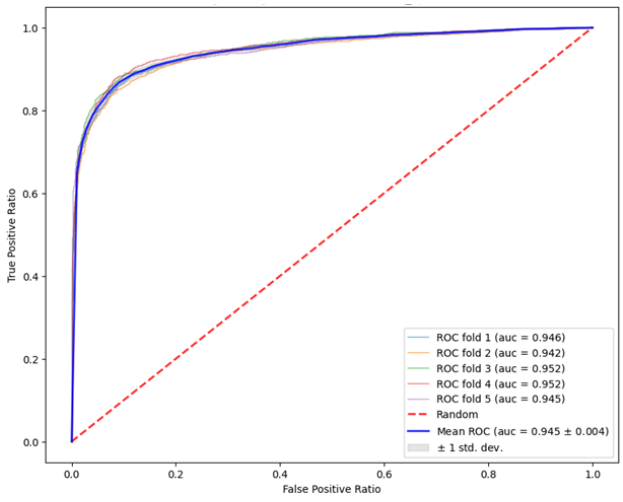
در این قسمت نتایج آموزش الگوریتم‌های مختلف یادگیری ماشین نظارت شده روی مجموعه داده مورد بررسی قرار می‌گیرد. برای این منظور از زبان برنامه‌نویسی پایتون بهره گرفته شده و الگوریتم‌ها در این محیط پیاده‌سازی می‌شوند. مجموعه داده نمونه مورد استفاده در مقاله حاضر توسط [۱۷] تولید شده که از طریق لینک ارائه شده در [۲۸] قابل دریافت است. این مجموعه داده با شبیه ساز شبکه ns-2 تولید شده و برای تولید مشاهدات در آن، از روش زیر استفاده شده است: در هربار توپولوژی شبکه به‌طور تصادفی تولید شده و شبکه در مدت زمان ثابتی شبیه‌سازی شده است. ترافیک داده‌ها به‌صورت تصادفی تولید شده و تمام خطاهای رخ داده در بازه زمانی مشخص شده در پایگاه داده جمع آوری شده است. این روش تا زمانی تکرار می‌شود که تعداد مشاهدات کافی در پایگاه داده ایجاد شود. در این مجموعه داده، تعداد ۳۵۴۴۱ مشاهده جمع‌آوری شده است که این مشاهدات مربوط به بیش از هزار توپولوژی تصادفی متفاوت هستند. برای تولید توپولوژی تصادفی، ابتدا تعداد تصادفی گره (بین ۱۰ تا ۶۰۰) انتخاب شده است و سپس به صورت تصادفی اتصالات بین این گره‌ها برقرار شده است. با توجه به ترافیک، حداقل ۶۰ درصد از جریان‌ها، جریان‌های TCP بودند و بقیه به‌طور تصادفی از بین TCP و سایر انواع ترافیک بر اساس UDP انتخاب شدند. فرستنده‌ها، گیرندگان و مدت زمان هر ترافیک به صورت تصادفی تنظیم شده‌اند. بنابراین، پایگاه داده حاوی خطاهای مربوط به دوره‌های کوتاه و طولانی TCP است.

این مجموعه داده شامل ۱۰۰ ویژگی است که در آن آماره‌های مختلفی از قبیل میانگین، انحراف معیار، کمینه، بیشینه و ... در مورد اطلاعاتی از شبکه مانند RTT، زمان بین رسیدن بسته‌ها در مقصد، اندازه پنجره CWND و ... وجود دارد. همچنین اطلاعاتی از قبیل تعداد بسته‌های از دست رفته و زمان بین رسیدن بسته‌ها تقسیم بر RTT نیز موجود است. از سوی دیگر هر داده دارای برچسب A یا C است که C به معنای ازدحام و A به معنای خطاهای ارتباطی است [۱۱].

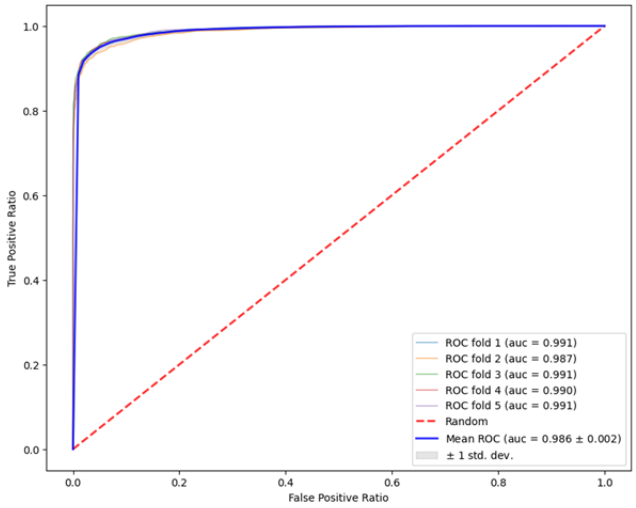
به منظور استانداردسازی ویژگی‌های مستقل موجود در داده‌ها در یک محدوده ثابت از نرمال‌سازی استفاده می‌شود. نرمال‌سازی به بهبود فرایند یادگیری ماشین کمک می‌کند زیرا باعث می‌شود اطمینان حاصل شود که در طول فرآیند یادگیری به هر مشخصه توجه می‌شود. بدون نرمال‌سازی، ویژگی‌های مقیاس بزرگ‌تر می‌توانند بر یادگیری مسلط شوند و نتایج ناهنجاری ایجاد کنند. در اینجا فرض بر آن است که مجموعه داده تنها توسط نرمال‌ساز Z-

Score مورد پیش‌پردازش قرار گرفته است. این تکنیک مقادیر ویژگی‌ها را طوری میزان می‌کند که میانگین آن‌ها برابر ۰ و انحراف معیار آنها برابر ۱ شود. در این مقاله به منظور تشخیص ازدحام از الگوریتم‌های یادگیری نظارت شده درخت تصمیم، جنگل تصادفی، ماشین بردار پشتیبان، لجستیک رگرسیون، K- نزدیکترین همسایه، XGBoost و شبکه عصبی استفاده می‌شود. برای الگوریتم شبکه عصبی از یک شبکه با سه لایه مخفی استفاده شده است. تعداد نورون‌ها در هر لایه مخفی برابر با ۱۰ و تعداد نورون‌ها در لایه خروجی با توجه به وجود دو کلاس برابر ۲ می‌باشد.

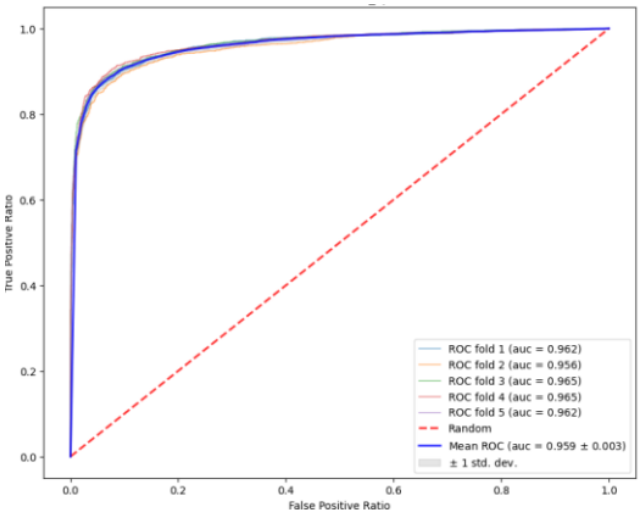
از روش ارزیابی K-fold Cross Validation با K=5 جهت ارزیابی همه الگوریتم‌های یادگیری ماشین استفاده شده است. نمودار منحنی ROC برای fold های مختلف و همچنین متوسط ROC برای الگوریتم‌های مختلف یادگیری ماشین ترسیم شده است. این نمودارها در شکل‌های ۱ تا ۷ رسم شده است. همانطور که در تمام این نمودارها مشخص است نقاط روی این منحنی‌ها با نیمساز ۴۵ درجه فاصله زیادی دارند. همچنین عدد AUC نیز برای همه الگوریتم‌ها بسیار نزدیک به ۱ است. لذا بر اساس توضیحات بخش ۳ میزان نرخ مثبت صحیح برای همه الگوریتم‌های مورد بررسی، بالا است. هم چنین مقایسه بین منحنی ROC الگوریتم‌های مختلف نشان می‌دهد که الگوریتم XGBoost عملکرد بهتری به لحاظ شکل منحنی دارد. به عبارت دیگر معیار AUC (که سطح زیرنمودار منحنی ROC را نشان می‌دهد) برای الگوریتم XGBoost نسبت به الگوریتم‌های دیگر بالاتر است. همچنین برای مقایسه بین الگوریتم XGBoost و الگوریتم تقویت درخت تصمیم ارائه شده در [۱۷] نیز منحنی ROC برای الگوریتم تقویت درخت تصمیم در شکل ۸ رسم شده است. بررسی این شکل نشان می‌دهد که معیار AUC برای الگوریتم XGBoost نسبت به الگوریتم درخت تصمیم بالاتر است.



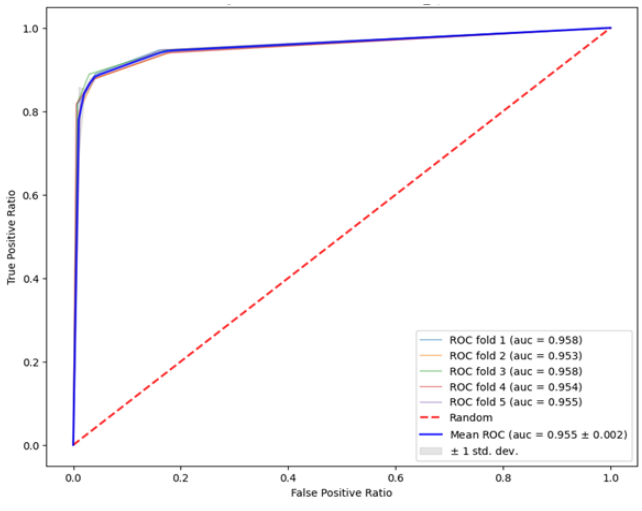
شکل ۱. منحنی ROC طبقه‌بند لاجستیک رگرسیون



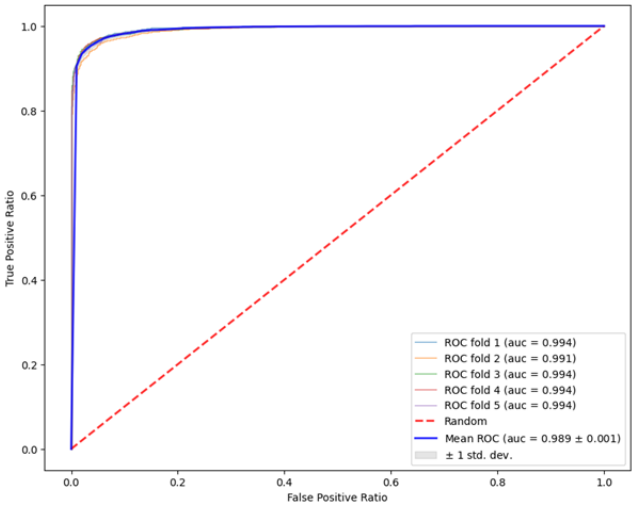
شکل ۲. منحنی ROC طبقه‌بند جنگل تصادفی



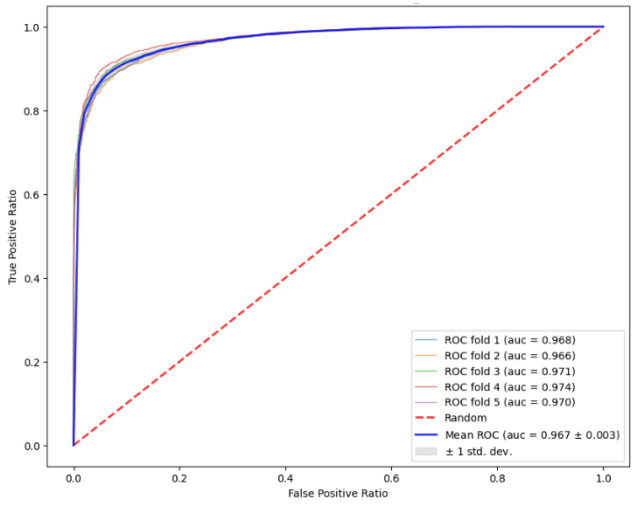
شکل ۳. منحنی ROC طبقه‌بند ماشین بردار پشتیبان



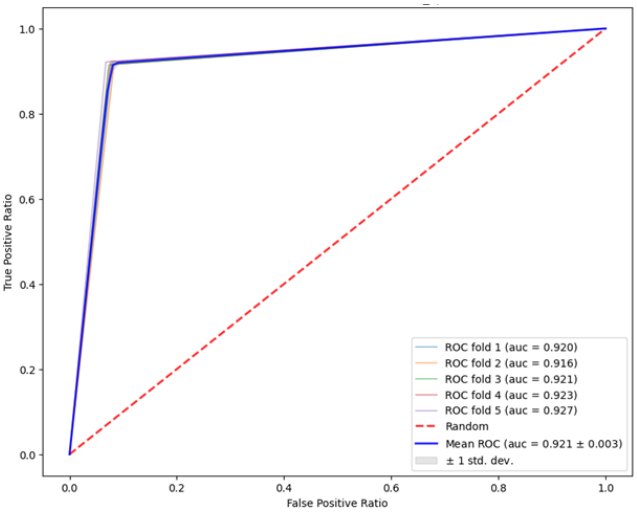
شکل ۴. منحنی ROC طبقه‌بندk نزدیکترین همسایه



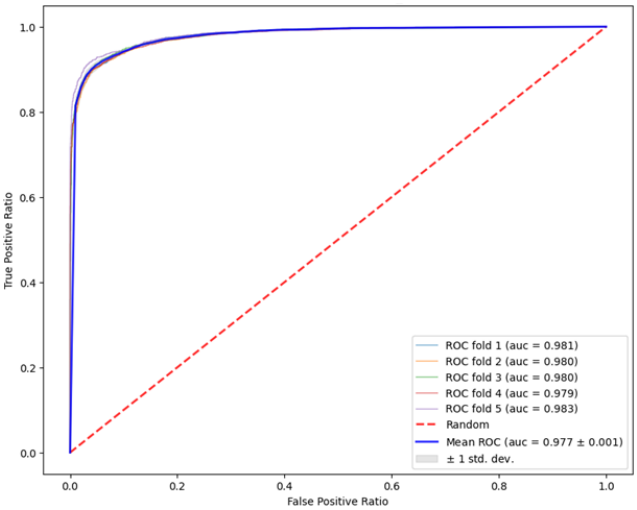
شکل ۵. منحنی ROC طبقه‌بند XGBoost



شکل ۸. منحنی ROCطبقه بند تقویت درخت تصمیم



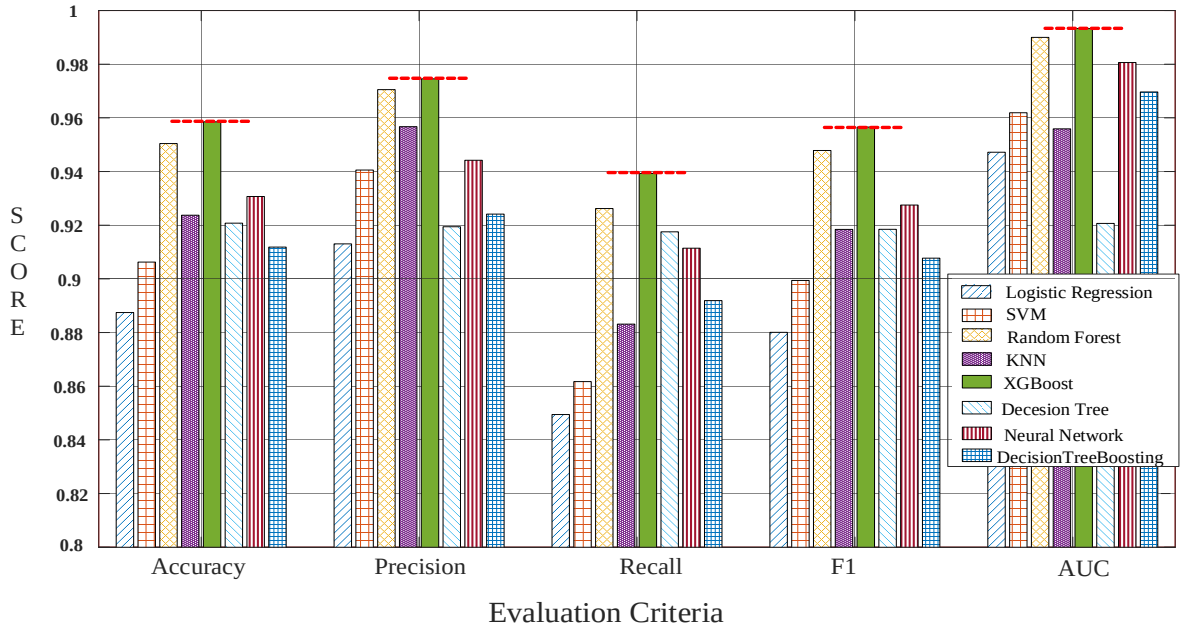
شکل ۶. منحنی ROC طبقه‌بند درخت تصمیم



شکل ۷. منحنی ROC طبقه‌بند شبکه عصبی

در ادامه الگوریتم‌های مختلف به لحاظ معیارهای ارزیابی گوناگون با یکدیگر مقایسه می‌شوند. بدین منظور برای هر کدام از foldها، معیارهای ارزیابی دقت، درستی، حساسیت، F1-Measure و ROC-AUC محاسبه شده است و در نهایت میانگین تمامی foldها به عنوان میانگین کلی آن معیار محاسبه شده است. این نمودارها در شکل ۹ آورده شده است.

در شکل ۹ خلاصه نتایج به دست آمده از آموزش الگوریتم‌های مختلف یادگیری ماشین روی مجموعه داده زمانی که از نرمال‌ساز Z-Score استفاده شده نمایش داده شده است. همانگونه که مشخص است، الگوریتم XGBoost توانسته است از سایر الگوریتم‌ها در همه شاخص‌ها عملکرد بهتری ارائه دهد. بعد از این الگوریتم، الگوریتم‌های جنگل تصادفی و شبکه عصبی در جایگاه بعدی قرار دارند. برای مجموعه داده‌های حوزه شبکه، تمایز بین خطاهای ازدحام و پیوند، مستلزم یافتن الگوهای دقیق در جریان ترافیک شبکه و از دست رفتن بسته‌ها است. توانایی XGBoost در مدل‌سازی روابط غیرخطی پیچیده و تعاملات بین ویژگی‌ها، آن را برای این امر بسیار مناسب می‌سازد. علاوه بر این، شبکه پویا و دارای تغییرات ناگهانی است. سازگاری XGBoost به آن کمک می‌کند حتی با تغییر رفتار شبکه، همچنان موثر باقی بماند. به طور خلاصه، ترکیب XGBoost از پیچیدگی مدل، منظم‌سازی، مدیریت مجموعه داده‌های نامتعادل و مکانیسم موثر هرس درخت، آن را برای کارهای طبقه‌بندی پیچیده، ایده‌آل می‌سازد. عملکرد قوی آن در شاخص‌های دقت، درستی، حساسیت، F1-Measure و ROC-AUC گواهی بر پیش‌بینی موثر الگوهای پیچیده در داده‌های شبکه است که سبب می‌شود اطمینان بالایی برای استقرار عملی در تمایز بین خطاهای ازدحام و پیوند تضمین شود.



شکل ۱. مقایسه عملکرد الگوریتم‌های مختلف بر اساس معیارهای ارزیابی درستی, دقت، حساسیت، F1 و AUC

هر دو جنگل تصادفی و شبکه‌های عصبی جایگزین‌های قوی برای XGBoost با نقاط قوت خود ارائه می‌دهند. جنگل تصادفی، کاهش واریانس و مقاومت مناسبی را از طریق روش تجمیعی ارائه می‌دهد که آن را برای طبقه‌بندی شبکه بسیار قابل اعتماد می‌کند. در واقع الگوریتم جنگل تصادفی، چندین درخت تصمیم می‌سازد و آن‌ها را با هم ادغام می‌کند تا پیش‌بینی دقیق و پایدارتری داشته باشد. شبکه‌های عصبی انعطاف‌پذیری و قابلیت یادگیری بی‌نظیری را به ویژه در یادگیری الگوهای پیچیده لازم برای تمایز مؤثر در رفتار شبکه فراهم می‌کنند. این امر ناشی از قابلیت مدل‌سازی و یادگیری روابط غیرخطی به علت ساختار معماری چندلایه و وجود توابع فعال‌سازی می‌باشد. بررسی نتایج شکل ۹ نشان می‌دهد که به لحاظ معیارهای مختلف ارزیابی، الگوریتم جنگل تصادفی پس از الگوریتم XGBoost در رتبه دوم قرار دارد. همچنین الگوریتم شبکه عصبی نیز در تمام معیارها در رتبه سوم قرار دارد (به جز معیار حساسیت که اختلاف کمی با الگوریتم لجستیک رگرسیون دارد). قرار گرفتن این دو الگوریتم در رتبه‌های دوم و سوم نشان‌دهنده کارایی آن‌ها در مدیریت پیچیدگی‌ها و پویایی داده‌های ترافیک شبکه است. همچنین با توجه به نتایج شبیه‌سازی در شکل ۹ مقایسه بین الگوریتم XGBoost و الگوریتم تقویت درخت تصمیم نشان می‌دهد که به لحاظ تمام معیارهای ارزیابی XGBoost عملکرد بهتری را ارائه می دهد. به عنوان مثال به لحاظ معیار ارزیابی درستی، الگوریتم XGBoost دارای درستی ۹۶ درصد در تفکیک کلاس ازدحام از کلاس خطاهای ارتباطی است در حالی که این عدد برای الگوریتم تقویت درخت تصمیم ۹۱ درصد می‌باشد.

۵- نتیجه‌گیری

هدف اصلی این مقاله تشخیص ازدحام با دقت بالا با بهره‌گیری از الگوریتم‌های یادگیری ماشین نظارت شده بود و مسئله اصلی با عنوان شناسایی ازدحام و تفکیک آن از خطای ناشی از لینک‌های ارتباطی مورد بررسی قرار گرفت. برای حل مسئله مذکور از الگوریتم‌های یادگیری ماشین نظارت شده‌ی متعدد از جمله درخت تصمیم، جنگل تصادفی، ماشین بردار پشتیبان، لجیستیک رگرسیون، K-نزدیکترین همسایه، XGBoost، شبکه عصبی و تقویت درخت تصمیم استفاده گردید. در راستای یافتن بهترین روش حل مساله از میان روش‌های مذکور عملکرد این الگوریتم‌ها با کمک معیارهای مختلف از جمله دقت، درستی، F1-measure، حساسیت و AUC ارزیابی و با یکدیگر مقایسه شد. نتایج شبیه‌سازی نشان داد که الگوریتم XGBoost توانسته است بر اساس همه معیارهای ارزیابی از سایر الگوریتم‌ها عملکرد بهتری ارائه دهد و الگوریتم‌های جنگل تصادفی و شبکه عصبی در جایگاه‌های بعدی قرار گرفتند. همچنین مقایسه بین الگوریتم XGBoost و الگوریتم تقویت درخت تصمیم ارائه شده در [۱۷] بیانگر عملکرد بهتر الگوریتم XGBoost بر اساس تمام معیارهای ارزیابی است. به عنوان مثال به لحاظ معیار ارزیابی درستی، الگوریتم XGBoost دارای درستی ۹۶ درصد در تفکیک کلاس ازدحام از کلاس خطاهای ارتباطی است اما درستی الگوریتم تقویت درخت تصمیم ۹۱ درصد می‌باشد. لذا الگوریتم XGBoost به عنوان یک روش کارامد برای حل مسئله تشخیص ازدحام در شبکه می‌تواند مورد بهره برداری قرار گیرد.

در ادامه فعالیت‌های انجام شده در این مقاله به شبیه‌سازی محیط‌های پیچیده‌تر، از قبیل شبکه با فرضیاتی نظیر تحرک کاربران یا وجود فیدینگ در کانال به منظور واقعی تر شدن مسئله، با استفاده از شبیه‌ساز شبکه NS3 پرداخته و مسئله شناسایی ازدحام را در آن محیط‌ها مورد بررسی قرار خواهیم داد. همچنین استفاده از انواع الگوریتم‌های شبکه عصبی عمیق از جمله LSTM برای پیش‌بینی ازدحام از دیگر مطالعاتی است که در ادامه مطالعه حاضر انجام خواهد شد.

مراجع

[1] Z. Zhang; Y. Zhou; L. Teng; W. Sun; C. Li and X.Min, "Quality-of-Experience Evaluation for Digital Twins in 6G Network Environments," in *IEEE Transactions on Broadcasting*, pp. 1-13, 2024.

[2] R. Prasad and A. Sunny, "QoS-Aware Scheduling in 5G Wireless Base Stations," in *IEEE/ACM Transactions on Networking*, vol. 32, no. 3, pp. 1999-2011, June 2024.

[3] J. P. Astudillo León, L. J. Cruz Llopis and F. J. Rico-Novella, "A machine learning based Distributed Congestion Control Protocol for multi-hop wireless networks," in *Computer Networks*, vol. 231, July 2023.

[4] J. Lorincz, Z. Klarin and J. Ožegovic, "A Comprehensive Overview of TCP Congestion Control in 5G Networks: Research Challenges and Future Perspectives," in *MDPI Sensors*, vol. 21, no. 13, pp. 4510, 2021.

[5] L. Machora. "A survey of transmission control protocol variants," in *World Journal of Advanced Research and Reviews*, vol. 21, no. 3, pp. 1828– 1853, 2024.

[6] I. Stoica, "A Comparative Analysis of TCP Tahoe, Reno, New-Reno, SACK and Vegas," University of California Berkeley, CA, 2005.

[7] Z. Tafa, Zhilbert, and V. Milutinovic. "Machine learning in congestion control: A survey on selected algorithms and a new roadmap to their implementation." *arXiv preprint* arXiv:2112.15522 2021.

[8] T. Zhang and S. Mao, "Machine Learning for End-to-End Congestion Control," in *IEEE Communications Magazine*, vol. 58, no. 6, pp. 52-57, June 2020.

[9] Huiling Jiang, Qing Li, Yong Jiang, GengBiao Shen, Richard Sinnott, Chen Tian, Mingwei Xu, "When machine learning meets congestion control: A survey and comparison", in *Computer Networks*, vol. 192, pp. 1-28, 2021.

[10] M. B. M. Kamel, I. Ahmed Najm and A. Khalaf Hamoud, "Congestion Control Prediction Model for 5G Environment Based on Supervised and Unsupervised Machine Learning Approach," in *IEEE Access*, vol. 12, pp. 91127-91139, 2024

[11] J. Alejandrino, R. Concepcion, S. Lauguico, M. G. Palconit, A. Bandala and E. Dadios, "Congestion Detection in Wireless Sensor Networks Based on Artificial Neural Network and Support Vector Machine," in 12th International Conf. on Humanoid, Nanotechnology, Inf. Tech., Commun. and Control, Environment, and Management (HNICEM), Manila, Philippines, 2020, pp. 1-6.

[12] P.T. Kasthuribai, "Optimized support vector machine based congestion control in wireless sensor network based Internet of Things," in *Int. J. Computer Network and Application*, vol. 8, no. 4, pp. 444-454, 2021.

[13] N. Kumar, and M. Raubal. "Applications of deep learning in congestion detection, prediction and alleviation: A survey." in *Transportation Research Part C: Emerging Technologies* vol.133 pp. 103432, 2021.

[14] G. W. Geremew, , J. Ding. "Elephant Flows Detection Using Deep Neural Network, Convolutional Neural Network, Long Short-Term Memory, and Autoencoder." in *Journal of Computer Networks and Communications*, pp. 1-18, 2023.

[15] M. Labonne, C. Chatzinakis and A. Olivereau, "Predicting Bandwidth Utilization on Network Links Using Machine Learning," in European Conf. on Networks and Commun. (EuCNC), Dubrovnik, Croatia, 2020, pp. 242-247.

[16] P. Geurts, I. El Khayat and G. Leduc, "A machine learning approach to improve congestion control over wireless computer networks," in Fourth IEEE International Conf. on Data Mining (ICDM'04), Brighton, UK, 2004, pp. 383-386.

[17] I. El Khayat, P. Geurts, and G. Leduc, 'Improving TCP in Wireless Networks with an Adaptive Machine-Learnt Classifier of Packet Loss Causes', in NETWORKING, 2005, pp. 549–560.

[18] J. D. Kelleher, B. M. Namee and A. D. Arcy. Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies. MIT press, 2020.

[19] T. K. Ho, "Random Decision Forests," in Proceedings of the 3rd International Conf. on Document Analysis and Recognition, Montreal, QC, 14–16 Aug., 1995, pp. 278–282.

[20] J. Tolles, WJ. Meurer, "Logistic Regression: Relating Patient Characteristics to Outcomes," in *JAMA*, Vol. 316, no. 5, pp. 533-534, Aug. 2016.

[21] E. Fix, J. L. Hodges, "Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties," in International Statistical Review, vol. 57, no.3, pp. 238-247, 1989.

[22] R. O. Duda and P. E. Hart, Pattern classification. John Wiley & Sons, 2006.

[23] T. Chen, "Story and Lessons behind the evolution of XGBoost," 2016.

[24] LeCun, Y. Bengio, and G. Hinton, "Deep learning," in *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[25] D. Shi, W. Ping, A. Khushnood. "A survey on deep learning and its applications," in *Computer Science Review*, vol. 40, pp. 100379, 2021.

[26] Y. Freund, R. E Schapire. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," in *J. of Computer and System Sciences*, vol. 55, no. 1, pp. 119-139, 1997.

[27] C. M. Bishop, and N. M. Nasrabadi. Pattern recognition and machine learning. Vol. 4. No. 4. New York: springer, 2006.

[28] [Online]. Available: https://people.montefiore.uliege.be/geurts/Papers/Boosting-DT/BD-Fifo.dat.gz