Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
Volume 13, Number 2, Summer 2024, pp. 1-12
Received: 01/24/2023, Revised: 03/14/2024, Accepted: 10/30/2024

A Machine Learning-Based Routing Algorithm for Mobile Internet of Things

Maryam Rajabi Chapeshloo[1] and Shahram Babaie[2*]
1, 2- Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran.
[1] maryam.rajabi5858@gmail.com,[2] sh.babaie@iaut.ac.ir

Corresponding author's address: Shahram Babaie, Faculty of Computer Engineering, Islamic Azad University, Tabriz Branch, Tabriz, Iran.

Abstract- The Internet of Things and its applications can be seen in all aspects of human life, and adding mobility features to objects has brought about new challenges. Mobility causes dynamics in topology and instability in the network links, and introduces several hurdles in discovering the path with less overhead and low delay. Most of the routing algorithms are based on the on-demand distance vector routing algorithm based on the source node's request. In recent years, several improvements have been presented on this mechanism, and its multi-path version is one of the notable improvements. Due to the high dynamics of the network and the momentary changes in the mobile Internet of Things, it is impossible to predict all the circumstances and adjust the optimal parameters in advance. Thus, a novel routing strategy based on machine learning and multi-path distance vector routing is introduced in this research. Its goal is to assess network conditions and choose the optimal interface node for packet forwarding. Five machine learning algorithms, including decision trees, random forests, and support vector machines, are used in the proposed approach for learning and evaluation. A number of network parameters, including node velocity, the number of neighboring nodes, buffer size, remaining energy, and the average distance between each node and its neighbors, are taken into account when choosing the best interface node. Distance of each node with its neighbors are considered for selection of the appropriate interface node. The simulation results conducted through Python indicate that the decision tree and the gradient boosting have the best results in the collected data set, and by combining them with the proposed approach, the end-to-end delay is reduced by 30%, and the number of lost packets is reduced by 18%.

Keywords- Internet of Things, routing, AOMDV, machine learning.

مجله علمی رایانش نرم و فناوری اطلاعات
دانشگاه صنعتی نوشیروانی بابل
صفحه مجله: jscit.nit.ac.ir
جلد ۱۳، شماره ۲، تابستان ۱۴۰۳ ، صفحه ۱-۱۲
دریافت: ۱۴۰۱/۱۱/۰۴، بازنگری: ۱۴۰۲/۱۲/۲۴، پذیرش: ۱۴۰۳/۰۸/۰۹

یک الگوریتم مسیریابی مبتنی بر یادگیری ماشین برای اینترنت اشیاء متحرک

مریم رجبی چاپشلو[۱] و شهرام بابائی*[۲]
۱و۲- گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد تبریز، تبریز، ایران.
[1] maryam.rajabi5858@gmail.com و [2] sh.babaie@iaut.ac.ir

* نشانی نویسنده مسئول: شهرام بابائی، تبریز، دانشگاه آزاد اسلامی واحد تبریز، گروه مهندسی کامپیوتر.

چکیده- اینترنت اشیاء و کاربردهای آن امروزه در همه‌ی ابعاد زندگی بشری قابل مشاهده است که اضافه شدن قابلیت تحرک به اشیاء، چالش‌های جدیدی به وجود آورده است. تحرک گره‌ها باعث پویایی در توپولوژی و ناپایداری در لینک‌های شبکه شده و کشف مسیر با سربار و تاخیر کمتر را با چالش‌های متعدد مواجه کرده است. بیشتر الگوریتم‌های مسیریابی مبتنی بر تکنیک مسیر بردار فاصله مبتنی بر درخواست می‌باشند که مبتنی بر درخواست گره مبدأ می‌باشد. در سال‌های اخیر بهبودهایی برروی این مکانیزم ارائه شده است که نسخه چند مسیره آن یکی از این بهبودها می‌باشد. با توجه به پویایی بالای شبکه و تغییرات لحظه‌ایی در اینترنت اشیاء متحرک، امکان پیش‌بینی همه شرایط و تنظیم بهینه پارامترها از قبل وجود ندارد. لذا در این مقاله یک رویکرد مسیریابی مبتنی بر الگوریتم مسیریابی بردار فاصله چند مسیره ارائه می‌شود که با یادگیری ماشین سعی دارد شرایط شبکه را ارزیابی کرده و بهترین گره واسط را برای مسیریابی انتخاب کند. در رویکرد پیشنهادی پنج الگویتم یادگیری ماشین مانند ماشین بردار پشتیبان، جنگل تصادفی و درخت تصمیم‌گیری برای یادگیری مورد بررسی قرار می‌گیرند که چندین پارامتر شبکه مانند سرعت حرکت گره‌ها، تعداد گره‌های همسایه، اندازه بافر، انرژی باقیمانده و میانگین فاصله هر گره با همسایگانش را برای انتخاب گره واسط مناسب مورد بررسی قرار می‌دهند. نتایج شبیه‌سازی‌های انجام شده با زبان پایتون نشان می‌دهد که درخت تصمیم و گرادیان تقویت شده بهترین نتایج را در مجموعه داده جمع‌آوری شده دارند و با ترکیب آنها با رویکرد پیشنهادی، میزان تاخیر آنها به انتها به طور متوسط تا ۳۰ درصد و تعداد بسته‌های گمشده نیز تا ۱۸ درصد کاهش داشته است.

واژه‌های کلیدی: اینترنت اشیا، مسیریابی، مسیریابی بردارفاصله چند مسیره مبتنی بر درخواست، یادگیری ماشین.

۱- مقدمه

اینترنت اشیاء[۱]، به میلیاردها دستگاهی اشاره دارد که در سراسر جهان گسترده شده‌اند، به اینترنت متصل می‌شوند، داده‌ها را گردآوری کرده و با یکدیگر به اشتراک می‌گذارند. طبق گزارش سیسکو در سال ۲۰۲۰ بیش از پنجاه میلیارد دستگاه متصل به اینترنت اشیاء وجود داشت و پیش‌بینی می‌شود که این تعداد در سال‌های آتی رو به رشد خواهد بود و روشن است که زیرساخت‌های موجود قادر به پشتیبانی آنها نخواهند بود. از آنجایی که اینترنت اشیاء از حسگرهای[۲] هوشمند برای نظارت در کاربردهای مختلف زندگی روزمره استفاده می‌کند، این تکنولوژی باعث صرفه‌جویی در هزینه و زمان و مدیریت بهتر زندگی روزمره و در نتیجه بهبود کیفیت زندگی می‌شود. بنابراین اینترنت اشیاء به طور گسترده در کاربردهای متنوعی مانند کنترل سلامتی، نظامی، صنعت، مدیریت انرژی و محیط‌های هوشمند به کار گرفته می‌شوند [۱].

به محیط اینترنت اشیاءی که تجهیزات و المان‌های آن متحرک باشند، اینترنت اشیاء متحرک گفته می‌شود. شبکه موردی متحرک

و شبکه بین‌خودرویی دو تکنولوژی نوینی هستند که می‌توانند بستر مناسب را برای ایجاد ارتباط بین المان‌های اینترنت اشیاء متحرک ایجاد کنند و موجب گسترش این تکنولوژی شوند. در اینترنت اشیاء متحرک به دلیل ماهیت متحرک بودن اشیاء، اتصالات بین دستگاه‌های مبدا و مقصد ناپایدار است. در این نوع اتصالات به دلیل عدم وجود زیرساخت شبکه، مدل‌های حرکتی تصادفی و همچنین تعداد زیاد دستگاه‌های متصل به شبکه، به کارگیری رویکردهای ایستا مناسب نخواهند بود. لازم به ذکر است که ویژگی‌های ذکر شده مسئله مسیریابی را برای این شبکه‌ها به یک مسئله پیچیده تبدیل می‌کند[۲].

در حالت کلی، الگوریتم‌های مسیریابی به پنج گروه مسیریابی مبتنی بر توپولوژی، مبتنی بر خوشه‌بندی، مبتنی بر موقعیت مکانی، مبتنی بر داده توزیع شده و ترکیبی تقسیم‌بندی می‌شوند. الگوریتم‌های مسیریابی مبتنی بر توپولوژی برای انتخاب مسیر از اطلاعات وضعیتی خطوط ارتباطی استفاده می‌کنند و این اطلاعات از گره منبع تا گره مقصد، در حافظه گره‌ها ذخیره می‌شوند و به سه گروه، مکانیزم‌های پیشگیرانه، واکنشی و ترکیبی تقسیم می‌شوند. [۳] پروتکل‌های مبتنی بر موقعیت از اطلاعات مکانی گره‌ها برای مسیریابی استفاده می‌کنند. در حالی که در پروتکل‌های مسیریابی مبتنی بر خوشه‌بندی، ناحیه جغرافیایی و یا گره‌ها به خوشه‌های مجزا از هم تقسیم‌بندی می‌شوند. در پروتکل‌های مسیریابی مبتنی بر داده توزیع شده، داده‌های ارسالی به همه گره‌های موجود در یک منطقه جغرافیایی از پیش تعیین شده ارسال می‌شوند. در نهایت در مسیریابی ترکیبی، سعی می‌شود ترکیبی از رویکردهای مذکور مدنظر قرار گیرد. در ضمن تکنیک‌های متعددی مانند الگوریتم‌های فراابتکاری، استنتاج فازی و الگوریتم‌های یادگیری ماشین برای بهبود عملکرد تکنیک‌های مسیریابی و کاهش تاخیر در هدایت آنها ارائه شده است [۴]. لازم به ذکر است که تکنیک‌های مبتنی بر الگوریتم‌های فراابتکاری نیاز به زمان اجرای بالایی دارند که باعث می‌شود برای اکثر کاربردهای اینترنت اشیاء مناسب نباشند. از طرفی اغلب پروتکل‌های مسیریابی مختلف تلاش می‌کنند بر اساس اطلاعات محدود شبکه و گره‌ها به تصمیم‌های جامعی دست پیدا کنند که برای همیشه کارا باشند. در حالی که با توجه به ماهیت پویای اینترنت اشیاء متحرک لازم است از تکنیک‌هایی استفاده شود که به طور پویا اطلاعات لازم از شبکه را استخراج کنند و تصمیم مناسب را برای لحظه اتخاذ کنند. لذا به نظر می‌رسد که رویکردهای مبتنی بر یادگیری و تصمیم‌گیری گزینه مناسبی برای این شرایط باشند [۵].

در علم یادگیری ماشین به طراحی مدل‌هایی پرداخته می‌شود که قادر هستند با استفاده از داده‌های ورودی و نتایج حاصل آموزش ببینند و از آنها در تصمیم‌گیری‌های آینده استفاده کنند. در یادگیری ماشین، به جای برنامه‌نویسی همه چیز، داده‌ها به یک الگوریتم عمومی داده می‌شود و این الگوریتم است که براساس ورودی‌هایی که به آن اعمال می‌شود، منطق خود را می‌سازد. الگوریتم‌های یادگیری به سه گروه یادگیری نظارت شده، نظارت نشده و یادگیری تقویتی تقسیم می‌شوند [۶]. لذا در این مقاله سعی می‌شود مزیت‌های یادگیری ماشین در پویایی تصمیمات مسیریابی لحاظ گردد. بنابراین در رویکرد پیشنهادی سعی می‌شود ابتدا اطلاعات لازم از کارکرد و ویژگی‌های مختلف اینترنت اشیاء جمع‌آوری شود و سپس با اعمال روش‌های یادگیری ماشین به تولید مدل‌هایی پرداخته شود که مسیریابی را انجام می‌دهند. برای شناسایی بهترین الگوریتم یادگیری ماشین برای سناریو مدنظر، تکنیک‌های مختلفی مورد استفاده قرار می‌گیرد تا میزان دقت و کارایی آنها مورد بررسی قرار گیرد. برای این منظور عملکرد رویکردهای با نظارت و بدون نظارت نیز مورد ارزیابی قرار خواهند گرفت. که می‌توانند عملکرد مختلفی داشته باشند با یکدیگر مقایسه خواهند شد. بدین ترتیب یک مدل یادگیری مناسب جهت ترکیب با پروتکل مسیریابی بردار فاصله چند مسیری[۳] (AOMDV) مناسب برای اینترنت اشیاء متحرک انتخاب خواهد شد.

این مقاله از بخش‌های زیر تشکیل شده است. در بخش۲ به مروری از کارهای پیشین و مرتبط با این تحقیق پرداخته می‌شود. در بخش۳ رویکرد پیشنهادی ارائه شده است، در بخش۴ نتایج ارزیابی و شبیه‌سازی‌های انجام شده مورد بررسی قرار می‌گیرد. در نهایت در بخش ۵ نتیجه‌گیری و پیشنهادات برای کارهای آتی ارائه شده است.

۲- کارهای گذشته

اینترنت اشیای متحرک زیرمجموعه‌ایی از اینترنت اشیا، می‌باشد که به علت تحرک المان‌های شبکه، پویایی بالایی در آن وجود دارد و نحوه مدیریت و مسیریابی را با چالش اساسی روبرو کرده است. همچنین اینترنت اشیای متحرک دارای یک سری ویژگی‌های مشترک با کلاس دیگری از شبکه‌ها به نام شبکه‌های متحرک می‌باشند. این اشتراک‌ها باعث می‌شود که الگوریتم‌های مسیریابی یکسانی برای شبکه متحرک و هم برای اینترنت اشیاء متحرک قابل اجرا باشند. مزیت‌های گسترده یادگیری ماشین منجر به کاربرد موفقیت‌آمیز آن در راه حل‌های مسیریابی برای اینترنت اشیاء متحرک از طریق پروتکل‌هایی مانند پروتکل

مسیریابی نزدیک‌ترین K –همسایه[۴](KNNR) و پروتکل مسیریابی مبتنی بر یادگیری[۵](MLPROPH) شده است. واشیشس[۶] و همکارانش استفاده از یادگیری ماشین را برای خودکارسازی تصمیمات مسیریابی در اینترنت اشیاء متحرک دنبال کرده‌اند. برای این منظور آن‌ها از مدل‌های مخلوط گاوسی و مکانیزم خوشه‌بندی نرم[۷] مبتنی بر یادگیری ماشین برای توسعه یک پروتکل مسیریابی جدید به نام مسیریابی مبتنی بر مدل یادگیری ماشین گوسی[۸] (GMRR) استفاده کرده‌اند. طراحی مسیریابی مبتنی بر مدل یادگیری ماشین گوسی به گونه‌ای است که مزایای پروتکل‌های مسیریابی آگاه از وضعیت شبکه و بدون آگاهی از وضعیت شبکه را ترکیب می‌کند. آن‌ها عملکرد مسیریابی مبتنی بر مدل یادگیری ماشین گوسی را با عملکرد پروتکل مسیریابی نزدیک‌ترین K – همسایه ، مسیریابی با پیشبینی مبتنی بر تاریخچه، پروتکل مسیریابی مبتنی بر یادگیری و پروتکل مسیریابی مبتنی بر احتمال[۹] (PROPHET) با استفاده از شبیه‌سازی‌های اجرا شده در شبیه‌ساز محیط شبکه متحرک[۱۰] مقایسه نموده‌اند. پارامترهای ارزیابی این مقایسه شامل احتمال تحویل[۱۱]، نسبت سربار شبکه[۱۲]، میانگین تعداد گام[۱۳] و تعداد پیام‌های گم شده[۱۴] می‌باشد. نتایج حاصل از شبیه‌سازی‌ها نشان می‌دهند که مسیریابی مبتنی بر مدل یادگیری ماشین گوسی عملکرد بهتری نسبت به پروتکل‌های مورد مقایسه دارد [۷].

در اینترنت اشیاء متحرک وجود یک مسیر متصل پایدار مابین فرستنده و گیرنده امکان‌پذیر نیست. بنابراین مسیریابی در این نوع شبکه‌ها با شبکه‌های موردی سیار[۱۵] متفاوت است[۸]. شبکه‌های موردی سیار وجود یک مسیر ثابت بین فرستنده و گیرنده را قبل از شروع ارتباط و تا زمان تکمیل آن فرض می‌کنند. مسیرها به صورت پویا ساخته می‌شوند زیرا هر گره می‌تواند از بین گره‌های همسایگان یک گره را به عنوان واسط انتخاب کند، با این فرض که آن گره پیام را به گره مقصد نزدیکتر می‌کند یا به خود گره مقصد تحویل می‌دهد. دوراندر[۱۶] و همکارانش یک پروتکل جدید مسیریابی پیش‌بینی مبتنی بر تاریخ [۱۷] برای اینترنت اشیای متحرک بدون زیرساختار را پیشنهاد کرده‌اند که از اطلاعات رفتاری گره‌ها برای یافتن بهترین گره واسط برای مسیریابی استفاده می‌کند. این پروتکل از تکنیک مسیریابی جغرافیایی الهام گرفته شده و از تاریخچه حرکت برای مدل‌سازی رفتار گره‌ها استفاده می‌کند. با استفاده از پیش‌بینی مارکوف[۱۸]، گره واسط بعدی انتخاب می‌شود. نتایج شبیه‌سازی‌ها نشان می‌دهند که این پروتکل از نظر تعداد پیام‌های تحویل داده شده و نسبت سربار در مقادیر آستانه مختلف عملکرد خوبی دارد. همچنین این پروتکل از نظر تعداد پیام‌های ارسالی و نسبت سربار عملکرد بهتری نسبت به پروتکل اپیدمی دارد [۹].

شارما و همکارانش یک پروتکل مسیریابی جدید به نام پروتکل مسیریابی مبتنی بر نزدیک‌ترین K-همسایه پیشنهاد کرده‌اند که پیغام‌ها را از طریق گره‌های میانی به سمت مقصد ارسال می‌کند. این پروتکل در ابتدا رفتار گذشته گره‌ها را در یک مجموعه داده ذخیره می‌کند. در هر زمانی که بایستی تصمیمی در مورد یک گره میانی گرفته شود، این مجموعه داده مورد بررسی قرار می‌گیرد و نمونه‌هایی پیدا می‌شوند که براساس پارامترهای شبکه و با استفاده از الگوریتم نزدیک‌ترین K –همسایه شباهت زیادی به گره میانی دارند. پروتکل مسیریابی نزدیک‌ترین K –همسایه شامل دو مرحله آموزش و مرحله کاربردی می‌باشد. در مرحله آموزش سعی می‌شود که گره‌های شبکه براساس عملکرد شبکه در ارسال پیام‌های شبکه آموزش داده شوند. در حالی‌که در مرحله کاربرد، در مورد اینکه آیا یک گره به عنوان گره واسط مناسب است یا خیر تصمیم‌گیری می‌شود که این کار با مقایسه وضعیت آن با رکوردهای گذشته و با استفاده از الگوریتم نزدیک‌ترین K –همسایه صورت می‌گیرد. در این پروتکل، به دلیل ماهیت غیرقابل پیش‌بینی بودن گره‌ها، ادغام رفتار گذشته گره‌ها با نمونه‌های جدید در تصمیم‌گیری‌ها حیاتی است. برای ارزیابی کارایی این پروتکل، عملکرد آن با پروتکل‌های اپیدمی، مسیریابی با پیش بینی ترکیبی[۱۹](HPBR) و پروتکل مسیریابی مبتنی بر احتمال مقایسه شده است و اثبات شده است که پروتکل پروتکل مسیریابی نزدیک‌ترین K –همسایه به طور موثر متوسط تأخیر، نسبت سربار و میانگین تعداد گام را کاهش و در عین حال احتمال تحویل صحیح پیام را افزایش می‌دهد [۱۰].

گارگ[۲۰] و همکارانش رشد ساختار هوشمند را به منظور مسیریابی بسته‌ها بررسی نموده‌اند که قادر است به صورت هوشمند بسته‌ها را در اینترنت اشیاء هدایت کند و به منظور تعیین مسیر ما بین دو گره موجود اطلاعات را توزیع کند. در این روش، هر مسیریاب از اینکه چه شبکه‌هایی به آن متصل هستند آگاه می‌باشد. از سوی دیگر، تحقیقات زیادی در حال حاضر به منظور شناخت تأثیر حمله سیاه‌چاله[۲۱] بر روی الگوریتم‌های مسیریابی شبکه انجام شده است که بر اساس بردار فاصله زمانی مورد تقاضا[۲۲] (AODV) شکل گرفته است. گراگ و همکارانش تأثیر منطق فازی را در پروتکل مسیریابی بررسی نمودند و تأثیر حمله سیاه‌چاله بر روی مسیریابی مبتنی بر بردار فاصله را مورد آزمایش قرار داده‌اند. مطابق نتایج شبیه‌سازی‌ها، استفاده از منطق فازی در الگوریتم‌های مبتنی بر AODV باعث بهبود عملکرد آنها می‌شود. علاوه بر این تأثیر

استفاده از منطق فازی برروی گم شدن بسته‌ها مورد بررسی قرار گرفته است [۱۱].

در بررسی تحقیقات پیشین مشاهده می‌شود که روش‌های استفاده شده برای ایجاد پروتکل مسیریابی به طور کلی اطلاعات شبکه را به صورت جامع دریافت و در یک گره اختصاصی جمع‌آوری کرده و سپس به پیش‌بینی مسیر ارسالی داده از مبدا به مقصد می‌پردازند. در حالی که اطلاعات مسیر در طی حرکت گره‌ها می‌تواند تغییرات زیادی در طول مسیر داشته باشد. از طرفی روش‌های استفاده شده و اطلاعاتی که در این پروتکل‌ها در نظر گرفته شده است بیشتر وابسته به پیام می‌باشد تا اینکه مرتبط به اطلاعات شبکه باشد. روش پیشنهادی این دو نکته را در نظر گرفته است و روشی پیشنهاد می‌دهد که به شکل زیر می‌توان نوآوری‌های آن را ذکر کرد:

۱- پیش بینی مسیر در هر مرحله و در گره فعلی و تصمیم گیری برای ارسال به گره بعدی در هر قدم به جای مسیر کلی از مبدا به مقصد.
۲- استفاده از اطلاعات وابسته به گره‌های شبکه، مکان و فاصله گره‌ها و همچنین اطلاعات بسته ارسالی.
۳- پیش بینی با دقت بالا و چشم پوشانی از روش‌های احتمالاتی و عدم قطعیت

این ویژگی‌ها باعث شده است که روش پیشنهادی از سایر روش‌ها متمایز شود. از طرف دیگر روش پیشنهادی بر روی پروتکل مسیریابی بردار فاصله چند مسیری که یک روش پایا می‌باشد پیاده سازی شده است که اطلاعات جمع آوری شده در طول مسیر به موقع و با تاخیر کم خواهد بود که در سایر روش‌های پیشنهادی روش دقیقی برای جمع آوری داده‌ها در طول مسیر ارائه نشده است.

۳- رویکرد پیشنهادی

پروتکل مسیریابی بردار فاصله چند مسیری یا AOMDV، نسخه بهبود یافته AODV می‌باشد که به صورت چند مسیره بسته‌های شبکه را براساس تعداد گام‌ها هدایت می‌کند. این پروتکل چند مسیره توانایی زیادی در کاهش تعداد تکرار در کشف مسیر در مقایسه با پروتکل‌های تک مسیره دارد[۱۲]. این پروتکل مسیرهای جایگزین را که از قبل کشف کرده و حفظ می‌کند و در صورتی که مسیر اصلی خراب شد از آن‌ها برای بازیابی استفاده می‌کند. از آنجایی که در مسیریابی AODV چندمسیره، انتخاب مسیر برای ارسال یک بسته از چند مسیر ممکن است، یک مسئله تصمیم‌گیری شکل می‌گیرد که می‌تواند شاخص‌های مختلفی برای تعیین مسیر در نظر گرفته شوند[۱۳]. در نسخه استاندارد، انتخاب مسیر براساس تعداد گام انجام می‌شود. اگرچه در نگاه اول یک تصمیم‌گیری عاقلانه به نظر می‌رسد اما ممکن است در حین انتقال اطلاعات مسیر با مشکلات متعددی روبرو شود که مهم‌ترین آن افزایش تعداد بسته‌ها در بافر بعضی گره‌های واسط می‌باشد. همچنین گره‌های موجود بر روی بعضی از مسیرها انرژی بیشتری مصرف می‌کنند که ممکن است برای همیشه خاموش شوند و در ضمن از کارافتادن بعضی از گره‌ها و یا ایجاد گلوگاه در بعضی از مسیرها ممکن است کل فرایند مسیریابی را با اختلال مواجه کند. به همین دلیل لازم است که در شرایط مختلف، سیاست‌های مسیریابی تغییر کنند که به نظر می‌رسد مدل‌های مبتنی بر پیش‌بینی و یادگیری ماشین گزینه‌های مناسبی برای حل مشکل مذکور باشند. لذا در رویکرد پیشنهادی سعی می‌شود با یادگیری از شرایط موجود تصمیم‌گیری‌ها هوشمندانه و پویا باشند. بنابرین برای نائل شدن به هدف مذکور رویکرد پیشنهادی از مراحل زیر تشکیل می‌شود.

- شبیه‌سازی مسیریابی الگوریتم AODV چندمسیره
- جمع‌آوری داده‌ها در شرایط مختلف شبیه‌سازی برای تاخیرهای مختلف
- استفاده الگوریتم‌های یادگیری ماشین برای ساخت مدل پیش‌بینی تاخیر در مسیرهای متفاوت
- استفاده از مدل ایجاد شده در حین تصمیم‌گیری برای انتخاب مسیر جهت انتقال اطلاعات

۳-۱- مراحل رویکرد پیشنهادی

در رویکرد پیشنهادی در جداول مسیریابی علاوه بر ستون‌های AODV چندمسیره استاندارد، یک ستون اضافی برای هر کدام از رکوردهای مسیریابی ایجاد می‌شود. این ستون بیان کننده اطلاعاتی در مورد وضعیت گره‌های شبکه خواهد بود. در رویکرد پیشنهادی تلاش خواهد شد با استفاده از مدل یادگیری ماشین از بین مسیرهای چندگانه یک مسیر با کمترین تاخیر به عنوان مسیر هدایت بسته‌ها انتخاب شود. این اطلاعات توسط بسته RREP از یک گره به گره دیگر در شبکه منتقل می‌شود. هر گره در حین هدایت این بسته، اطلاعات خود را به اطلاعات آن اضافه کرده و سپس آن را هدایت می‌کند. این روند در طی مسیر از سمت گره مقصد به سمت گره منبع انجام خواهد شد تا اینکه تمامی اطلاعات مسیر به گره مبدا رسیده باشد. بنابراین گره مبدا با داشتن اطلاعات چندین مسیر مختلف می‌تواند با دادن ویژگی‌های مسیر به مدل آموزش دیده یادگیری ماشین، به پیش‌بینی تاخیر موجود در مسیرها بپردازد. در نهایت مسیر با کمترین تاخیر در

بین همه مسیرها به عنوان مسیر انتقال بسته‌ها انتخاب می‌شود. در رویکرد پیشنهادی پیش‌فرض‌های زیر در نظر گرفته شده است.

- انرژی اولیه برای همه دستگاه‌های موجود در شبکه یکسان است.
- میزان مصرف انرژی برای انتقال یک بسته در همه گره‌های شبکه یکسان است.
- سرعت حرکت گره‌ها در جهت‌های مختلف متغیر است.
- بافر همه گره‌ها برای نگهداری بسته‌ها یکسان می‌باشد.
- ورود و خروج بسته‌ها از بافر گره‌ها به صورت اولین ورودی، اولین خروجی[۱۸] خواهد بود.
- در صورت کاهش انرژی یک گره از مقدار آستانه تعیین شده، آن گره از شبکه حذف خواهد شد.

۳-۲- جمع‌آوری داده‌ها در رویکرد پیشنهادی

برای جمع‌آوری داده‌ها جهت یادگیری ماشین باید شبکه‌هایی با ساختارهای مختلف در نظر گرفته شود. بنابراین لازم است سناریوهای مختلفی شبیه‌سازی شوند و نحوه حرکت گره‌های شبکه متغیر در نظر گرفته شوند. لازم به ذکر است که چندین پارامتر از جمله سرعت حرکت گره‌ها، تعداد همسایگان هر گره، متوسط تعداد بسته‌های موجود در بافر هر گره و انرژی باقیمانده هر گره در تاخیر بسته‌های عبوری از آن گره تأثیرگذار می‌باشد. بنابراین سعی می‌شود در حین شبیه‌سازی‌های مختلف سناریوهای متفاوتی پیاده‌سازی شوند تا اطلاعات مناسبی برای یادگیری جمع‌آوری شود. به همین دلیل ابتدا توپولوژی شبکه تغییر داده می‌شود. سپس در هر توپولوژی، معیارهای مختلف شبکه تغییر داده می‌شود و مجموعه داده مورد نیاز جمع‌آوری می‌شود. براساس داده‌های جمع‌آوری شده، میزان تاخیرها در ده کلاس مختلف قرار می‌گیرند. با افزایش شماره کلاس، میزان تاخیر نیز افزایش خواهد یافت. در هر فیلد مجموعه داده جمع‌آوری شده، میزان تاخیر به عنوان برچسب آن فیلد مدنظر گرفته می‌شود. داده‌های جمع‌آوری شده به دو گروه مجموعه داده آموزشی و مجموعه داده آزمایش تقسیم می‌شوند. در رویکرد پیشنهادی ویژگی‌های زیر برای جمع‌آوری مجموعه داده مورد استفاده قرار می‌گیرند که در مراحل بعدی برای پیش‌بینی میزان تاخیر در مسیر مبدا تا مقصد مورد استفاده قرار می‌گیرند.

- متوسط تعداد همسایه‌های هر گره موجود در مسیر
- متوسط سرعت حرکت گره‌های موجود در مسیر
- متوسط تعداد بسته‌های موجود در بافر گره‌های روی مسیر
- متوسط میزان انرژی باقیمانده گره‌های موجود در مسیر
- فاصله هر گره تا گره بعدی
- میانگین فاصله تا بقیه گره‌های همسایه
- تعداد کل گره‌های موجود در مسیر
- میزان تاخیر بسته ارسالی به عنوان برچسب این مجموعه داده در نظر گرفته می‌شود.

۳-۲-۱- پیش‌پردازش داده‌ها

در مجموعه داده جمع‌آوری شده به صورت استاندارد معایبی دارد که باید رفع شوند. به عنوان مثال لازم است که رکوردهای تکراری و غیرمفید حذف شوند که به این فرایند پیش‌پردازش داده‌ها گفته می‌شود. در رویکرد پیشنهادی از دو سیاست نرمال‌سازی و حذف داده‌های خارج از بازه برای پیش‌پردازش مجموعه داده استفاده می‌شود تا فرایند یادگیری و آموزش با دقت بیشتری انجام شود.

- **نرمال‌سازی مجموعه داده**

ویژگی‌های نمونه‌برداری شده در مجموعه داده جمع‌آوری شده، در ده کلاس مختلف قرار گرفته‌اند که بایستی به یک مقیاس مشابه تبدیل شوند که باعث بهبود عملکرد ارزیابی و آموزش رویکرد پیشنهادی می‌شود. برای نرمال سازی مجموعه داده در رویکرد پیشنهادی از روش Z-score استفاده می‌شود که یک تکنیک تغییر مقیاس است که نشان دهنده تعداد انحرافات استاندارد دور از میانگین می‌باشد. برای محاسبه Z-score از رابطه (۱) استفاده می‌شود.

$$x' = \frac{(x - \mu)}{\sigma} \qquad (۱)$$

که در این رابطه μ نشان دهنده میانگین و σ بیان کننده انحراف معیار می‌باشد.

- **حذف مقادیر خارج از بازه**

با توجه به شرایط کاملا پویای شبکه و متغیر بودن پارامترهای ورودی، دیده می‌شود که بعضی از مقادیر مجموعه داده جمع‌آوری شده خارج از بازه متعارف برای یک ویژگی خاص می‌باشند که باید اصلاح شوند. در رویکرد پیشنهادی از روش جایگزینی بیشترین و کمترین مقدار برای اصلاح مقادیر خارج از بازه استفاده می‌شود. به طوری که اگر داده‌ایی کمتر از حد پایین یک ویژگی باشد با مقدار کمینه آن ویژگی در مجموعه و اگر داده‌ایی بیشتر از حد بالای یک ویژگی باشد با بیشترین مقدار آن ویژگی در مجموعه داده جایگزین می‌شود.

۳-۲-۲- یادگیری و دسته‌بندی مجموعه داده

در این مرحله لازم است که رویکرد پیشنهادی با مجموعه داده پردازش شده، آموزش ببیند و سپس مورد ارزیابی قرار بگیرد. در رویکرد پیشنهادی از پنج الگوریتم مختلف ادامه روش‌هایی که برای دسته‌بندی تاخیر استفاده شده است که در ادامه تشریح می‌شوند.

- **دسته‌بندی ماشین بردار پشتیبان**

ماشین بردار پشتیبان[۲۳]یک تکنیک قدرتمند می‌باشد که می‌تواند برای هر دو وظایف رگرسیون و طبقه‌بندی استفاده شود. هدف الگوریتم ماشین بردار پشتیبان ایجاد بهترین خط یا مرز تصمیم است که بتواند فضای n بعدی را به کلاس‌های مجزا تفکیک کند تا داده‌های جدید بتوانند در گروه‌بندی صحیح قرار گیرند. این مرز بهترین تصمیم، ابرصفحه[۲۴] نامیده می‌شود. ماشین بردار پشتیبان نقاط و یا بردارهایی را انتخاب می‌کند که به ایجاد ابرصفحه کمک می‌کنند[۱۴].

- **دسته‌بندی بیز ساده**

طبقه‌بندی‌کننده بیز ساده[۲۵] یک مدل یادگیری ماشین احتمالی است که برای طبقه‌بندی مجموعه داده استفاده می‌شود. اساس این طبقه‌بندی کننده براساس قضیه بیز است. در قضیه بیز، با توجه به اینکه B رخ داده است، می‌توان احتمال وقوع A را با استفاده از رابطه (۲) تخمین زد که در واقع B مدرک و A فرضیه است. فرضی که در اینجا مطرح می‌شود این است که در مجموعه داده رویکرد پیشنهادی ویژگی‌ها مستقل هستند و یک ویژگی خاص بر دیگری تأثیر نمی‌گذارد[۱۵]. از این رو به این دسته‌بند نسخه ساده گفته می‌شود.

$$\boldsymbol{P(A|B) = \frac{P(B|A)P(A)}{P(B)}} \qquad (۲)$$

- **دسته‌بندی درخت تصمیم‌گیری**

درخت تصمیم‌گیری[۲۶] یک الگوریتم یادگیری ماشین نظارت شده است که مشابه نحوه تصمیم‌گیری انسان‌ها، از مجموعه‌ای از قوانین برای تصمیم‌گیری استفاده می‌کند. درخت‌های تصمیم می‌توانند هم وظایف طبقه‌بندی و هم رگرسیون را انجام دهند، اغلب به این روش به عنوان الگوریتم CART اشاره می‌کنند. درخت تصمیم‌گیری از ویژگی‌های مجموعه داده برای ایجاد سوالات بله/خیر استفاده می‌کند و به صورت مداوم مجموعه داده را تا زمانی که تمام اطلاعات متعلق به هر کلاس را جدا کند، ادامه می‌دهد. بدین ترتیب، داده‌ها در یک ساختار درختی سازماندهی می‌شوند. هر بار که سوالی پرسیده می‌شود، یک گره به درخت اضافه می‌شود. نتیجه پرسش یک سوال مجموعه داده را براساس مقدار یک ویژگی تقسیم می‌کند و گره‌های جدیدی ایجاد می‌کند. زمانی که فرآیند تقسیم‌بندی برای تصمیم‌گیری به پایان برسید، آخرین گره‌های درخت که برگ نامیده می‌شوند، ایجاد می‌شوند.

- **دسته‌بندی جنگل تصادفی**

جنگل تصادفی[۲۷] یا جنگل تصمیم تصادفی یک روش یادگیری برای طبقه‌بندی، رگرسیون و سایر وظایف است که مبتنی بر ساختن تعداد زیادی درخت تصمیم در زمان آموزش می‌باشد. برای کارهای طبقه‌بندی، خروجی جنگل تصادفی، کلاسی است که توسط اکثر درختان انتخاب شده است. برای وظایف رگرسیون، میانگین یا میانگین پیش‌بینی درختان منفرد برگردانده می‌شود[۱۶].

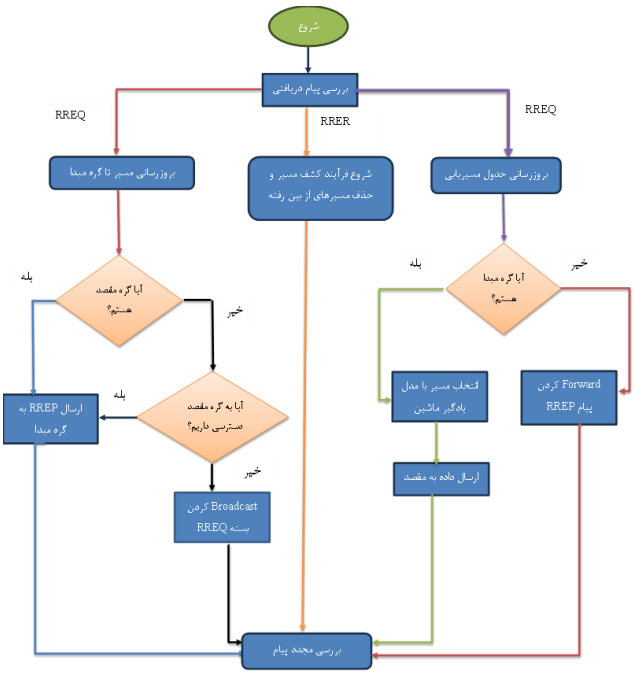
- **دسته‌بندی گرادیان تقویت شده**

گرادیان تقویت شده[۲۸] یک تکنیک یادگیری ماشین است که در کارهای رگرسیون و طبقه‌بندی و سایر موارد استفاده می‌شود. این مدل یک مدل پیش‌بینی را در قالب مجموعه‌ایی از مدل‌های پیش‌بینی ضعیف ارائه می‌دهد که معمولاً درخت‌های تصمیم‌گیری هستند. هنگامی که یک درخت تصمیم یادگیرنده ضعیف باشد، الگوریتم حاصل درختان با گرادیان تقویت شده معمولاً از جنگل تصادفی بهتر است. گرادیان تقویت شده شامل سه جزء اصلی تابع ضرر، یادگیرنده ضعیف و مدل افزایشی می‌باشد. نقش تابع ضرر این است که تخمین بزند که مدل چقدر در پیش‌بینی با داده‌ها بهتر شده است. یادگیرنده ضعیف تکنیکی است که داده‌ها را طبقه‌بندی می‌کند. مدل افزایشی یک رویکرد تکراری و متوالی اضافه کردن یادگیرندگان ضعیف در هر واحد زمان می‌باشد. پس از هر بار تکرار، باید به مدل نهایی نزدیکتر شود. به عبارت دیگر، در هر تکرار باید تابع ضرر کاهش یابد[۱۷].

شکل ۱ روند کلی رویکرد پیشنهادی را نشان می‌دهد. ساختار کلی بسته RREQ در شکل ۲ و بسته RREP در شکل ۳ قابل مشاهده است. در جدول ۱ توضیح فیلدهای این بسته‌ها آورده شده است.

۴- ارزیابی کارایی رویکرد پیشنهادی

در این بخش، عملکرد رویکرد پیشنهادی به واسطه شبیه‌سازی‌های انجام شده مورد ارزیابی قرار می‌گیرد و با رویکردهای مشابه مورد مقایسه قرار می‌گیرد.



شکل ۱: فلوچارت رویکرد پیشنهادی

Source Address	Source Sequence	Hopcount	Broadcast ID
Destination Sequence	Advertised Hopcount	First Hop	Timestamp

شکل ۲: ساختار بسته RREQ

Source Adress	Broadcast ID	Destination address	Destination Sequence
Hopcount	Lifetime	Timestamp	First Hop

شکل ۳: ساختار بسته RREP

جدول ۱: فیلدهای بسته‌های RREP و RREQ

فیلد	توضیحات
Source Address, Destination address	آدرس دستگاه مبدا و دستگاه مقصد
Destination Sequence, Source Sequence	شماره توالی بسته برای مبدا و مقصد
Broadcast ID	شناسه چندپخشی بسته RREQ
Advertised Hop count	بیشترین تعداد گام مربوط به یک مسیر
Lifetime	حداکثر طول عمر یک بسته
First Hop	اولین گام از سمت مبدا به مقصد
Hop count	تعداد گره‌های مابین مبدا و مقصد
Timestamp	لحظه ایجاد بسته

۴-۱- بستر شبیه‌سازی

برای شبیه‌سازی رویکرد پیشنهادی از زبان برنامه‌نویسی پایتون و برای یادگیری ماشین از کتابخانه Jupyter Notebook و برای الگوریتم‌های یادگیری ماشین از Sklearn استفاده شده است و همچنین برای دستکاری و تنظیم و پیش‌پردازش داده‌ها از کتابخانه Pandas استفاده شده است. همچنین برای بصری‌سازی شبیه‌سازی‌ها و نمایش نتایج از کتابخانه Matplotlib استفاده شده است. شبیه‌سازی‌ها بر روی یک کامپیوتر با پردازنده Intel core i7 85500 با فرکانس ۴ گیگاهرتز، دارای حافظه ۱۲ گیگابایت، پردازنده گرافیکی Nvidia GeoForce 930MX و در محیط سیستم عامل ویندوز ۸.۱ انجام شده است. برای شبیه‌سازی رویکرد پیشنهادی لازم است که برخی پیش‌فرض‌ها و پارامترها تعیین شوند که هم در مرحله جمع‌آوری داده‌ها و هم در مقایسه رویکرد پیشنهادی لازم می‌باشند. پارامترهای شبیه‌سازی در جدول ۲ آورده شده است.

جدول ۲: پارامترهای شبیه‌سازی

پارامتر	مقدار
تعداد گره‌های شبکه	۵۰ تا ۳۰۰
تعداد بسته‌های ارسالی در شبکه	۵۰ - ۳۰۰ در هر ثانیه
سرعت حرکت گره‌ها	۳۰ تا ۷۰ متر بر ثانیه
حد آستانه فاصله همسایگی	۷۰ تا ۱۰۰ متر
انرژی اولیه	۰.۵ ژول
اندازه بسته‌های شبکه	۴۰۰۰ بیت
میزان مصرف انرژی در ارسال/ دریافت بسته	$۱۰^{-۹}$* ۰.۵ ژول
ابعاد شبکه	۲۰۰ در ۲۰۰ متر
مدت زمان شبیه‌سازی	۵۰ تا ۳۰۰ ثانیه

پارامترهای شبیه سازی به گونه‌ای تنظیم شده است که این شبکه برای کاربردهایی همچون تبادل اطلاعات بین ماشین‌های خودران که در فاصله کم با یکدیگر در جاده و با سرعت معمولی حرکت می‌کنند مناسب باشد. این شبکه که با اطلاعات گره‌ها به مسیریابی و کارهای مرتبط می‌پردازد در محیط‌های با ابعاد کوچک و با انرژی کم و با سرعت متوسط مانند اطلاعات جدول ۲ انطباق دارند.

۴-۲- پارامترهای ارزیابی

برای ارزیابی رویکرد پیشنهادی پارامترهای متعددی مورد استفاده قرار می‌گیرد. معیارهای دقت، یادآوری و امتیاز اف۱ مهم‌ترین معیارهایی هستند که برای ارزیابی مدلهای دسته‌بندی مورد استفاده قرار می‌گیرند.

- **دقت:** دقت نسبت کلاس‌های مثبت پیش‌بینی‌شده درست به همه موارد مثبت پیش‌بینی‌شده است. به طور عینی، این پارامتر نشان می‌دهد که پیش‌بینی‌های رویکرد پیشنهادی چقدر درست یا دقیق هستند. دقت زمانی مهم خواهد بود که در یک روش دسته‌بندی، مثبت کاذب مهمتر از منفی‌های کاذب باشد.

❖ **یادآوری:** یادآوری نسبت کلاس‌های مثبت پیش‌بینی‌شده درست به همه مواردی است که واقعاً مثبت هستند. به طور عینی، این معیار نشان می‌دهد که یک تکنیک کلاس‌بندی چقدر توانسته است نمونه‌های مثبت واقعی را به درستی پیش‌بینی کند. یادآوری زمانی مهم خواهد بود که در یک روش دسته‌بندی، منفی‌های کاذب مهم‌تر از مثبت‌های کاذب باشد.

❖ **امتیازاف۱**: امتیاز F1 که گاهی اوقات F-Measure نیز نامیده می‌شود، یک معیار ارزیابی می‌باشد که همزمان هم دقت و هم یادآوری را در نظر می‌گیرد و با میانگین هارمونیک دو معیار مذکور محاسبه می‌شود.

لازم به ذکر است که برای محاسبه پارامترهای مذکور لازم است که چهار مقدار مثبت واقعی، مثبت کاذب، منفی واقعی و منفی کاذب محاسبه شوند که به صورت زیر قابل تعریف می‌باشند.

- موارد مثبت واقعی(TP): تعداد موارد مثبتی که به درستی به عنوان مثبت طبقه‌بندی شده‌اند.
- موارد مثبت کاذب(FP): تعداد موارد منفی که به اشتباه به عنوان مثبت طبقه‌بندی شده‌اند.
- منفی واقعی(TN): تعداد موارد منفی که به درستی به عنوان منفی طبقه‌بندی شده‌اند.
- منفی کاذب(FN): تعداد موارد مثبتی که به اشتباه به عنوان منفی طبقه‌بندی شده‌اند.

❖ **ماتریس درهم ریختگی:** یکی از معروف‌ترین پارامترهای ارزیابی تکنیک‌های کلاس‌بندی، ارزیابی ماتریس درهم ریختگی می‌باشد. ماتریس درهم ریختگی به جای ارزیابی دقت مدل خام، پیش‌بینی‌ها را به چند دسته تفکیک می‌کند و به صورت گرافیکی نشان می‌دهد که چگونه ممکن است یک کلاس با کلاس دیگر اشتباه گرفته شود.

۴-۳- بحث و آنالیز کارایی رویکرد پیشنهادی

همان طور که در بخش‌های قبلی گفته شد، برای یادگیری رویکرد پیشنهادی پنج تکنیک مختلف مورد استفاده قرار گرفته است که در اولین ارزیابی، دقت، یادآوری و امتیاز اف ۱ آنها مورد مقایسه قرار می‌گیرد. جدول ۳ خلاصه این مقایسه را نشان می‌دهد. نتایج حاصل نشان می‌دهد که روش‌های درخت تصمیم‌گیری، جنگل تصادفی و گرادیان تقویت شده بهترین عملکرد را در بین همه روش‌ها دارند.

جدول ۳: مقایسه تکنیک‌های مختلف یادگیری ماشین

روش	دقت	یادآوری	امتیاز اف۱
ماشین بردار پشتیبان	۰.۳۸	۰.۴۴	۰.۴۱
بیز ساده	۰.۹۲	۰.۹۳	۰.۹۳
درخت تصمیم‌گیری	۰.۹۹	۰.۹۹	۰.۹۹
جنگل تصادفی	۰.۹۹	۰.۹۹	۰.۹۹
گرادیان تقویت شده	۰.۹۹	۰.۹۹	۰.۹۹

در ارزیابی دیگر، ماتریس درهم ریختگی تکنیک‌های مختلف یادگیری ماشین به کار رفته در رویکرد پیشنهادی مورد ارزیابی قرار گرفته است. شکل ۴ (الف) ماتریس درهم ریختگی روش بیز ساده را نشان می‌دهد این مدل با دقت ۹۲ درصد کلاس‌ها را به درستی تشخیص داده و مقدار یادآوری و امتیاز اف۱ آن برابر ۹۳ می‌باشد. مشکل اساسی این روش زمان آموزش نسبتا بالای آن می‌باشد. همچنین روش بیز ساده یک تکنیک مبتنی بر احتمال می‌باشد و در حالت‌هایی که پیچیدگی داده‌ها بیشتر باشد، از لحاظ دقت و معیارهای ارزیابی عملکرد ضعیفی دارد. لذا در ادامه روند این مقاله، از این تکنیک کلاس‌بندی استفاده نشده است.

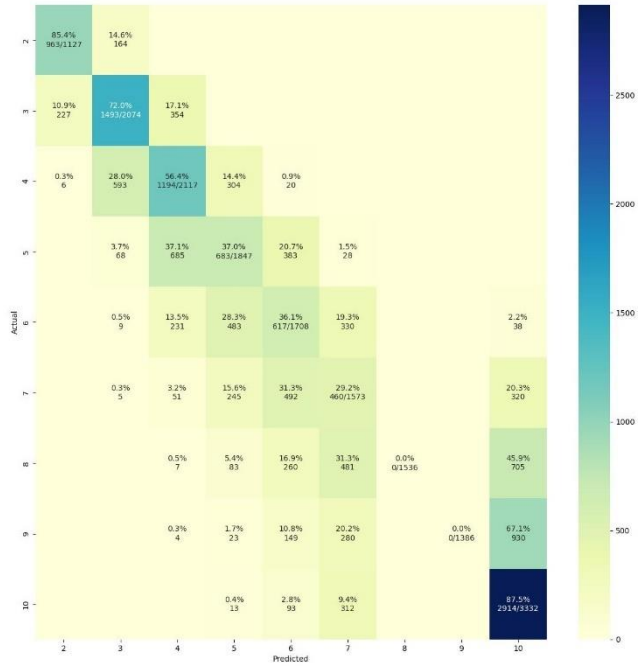
شکل ۴ (ب) ماتریس درهمریختگی مدل ماشین بردار پشتیبان را نشان می‌دهد. روش ماشین بردار پشتیبان به نسبت سایر روش‌ها از لحاظ زمان آموزش، ساخت مدل، دقت و کارایی به شدت ضعیف عمل کرده است. دقت این مدل ۳۸ درصد و یادآوری آن نزدیک به ۴۳ درصد است که مقادیر قابل قبولی حساب نمی‌شود.

شکل ۴ (ب) ماتریس درهمریختگی مدل ماشین بردار پشتیبان را نشان می‌دهد. روش ماشین بردار پشتیبان به نسبت سایر روش‌ها از لحاظ زمان آموزش، ساخت مدل، دقت و کارایی به شدت ضعیف عمل کرده است. دقت این مدل ۳۸ درصد و یادآوری آن نزدیک به ۴۳ درصد است که مقادیر قابل قبولی حساب نمی‌شود.

ماتریس درهم ریختگی مدل جنگل تصادفی در شکل ۵ (الف) نشان داده شده است. این مدل با میزان دقت و میزان یادآوری ۹۹ درصدی یکی از سه مدل کارای استفاده شده در رویکرد پیشنهادی می‌باشد. این مدل در مقایسه با تکنیک‌های درخت تصمیم‌گیری و گرادیان تقویت شده از سرعت کمتری برخوردار است و با افزایش پیچیدگی داده‌ها، این مدل نیز به زمان آموزش بالایی نیاز خواهد داشت. ماتریس دره مریختگی مدل تصمیم‌گیری در شکل ۵ (ب) نشان داده شده است. دقت این مدل تقریبا برابر ۱۰۰ درصد می‌باشد. همچنین این مدل به دلیل ساختار صریح و آسان، قابل درک می‌باشد و میزان یادآوری آن نیز ۱۰۰ درصد می‌باشد. آموزش این مدل بسیار سریع می‌باشد و برای استفاده در رویکرد مسیریابی پیشنهادی بسیار مناسب می‌باشد. بنابراین یکی



(الف)

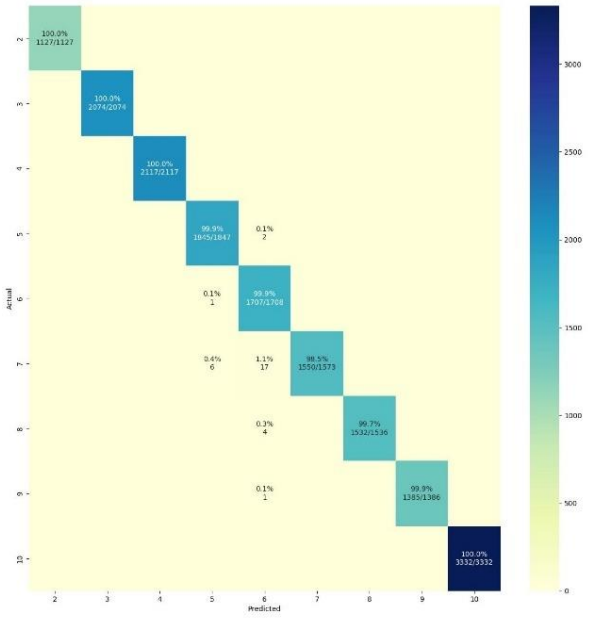


(ب)

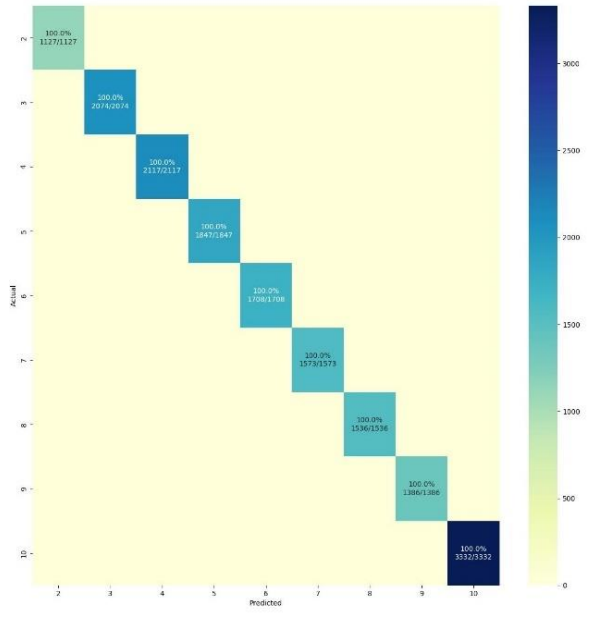
شکل ۴: ماتریس درهم ریختگی الف) بیز ساده. ب) ماشین بردار پشتیبان

از مدل‌هایی که در پروتکل پیشنهادی مورد استفاده قرار گرفته است، مدل درخت تصمیم‌گیری می‌باشد.

روش گرادیان تصادفی براساس پیشرفت درخت تصمیم‌گیری به دست می‌آید و از لحاظ زمان ساخت و آموزش نیز همانند درخت



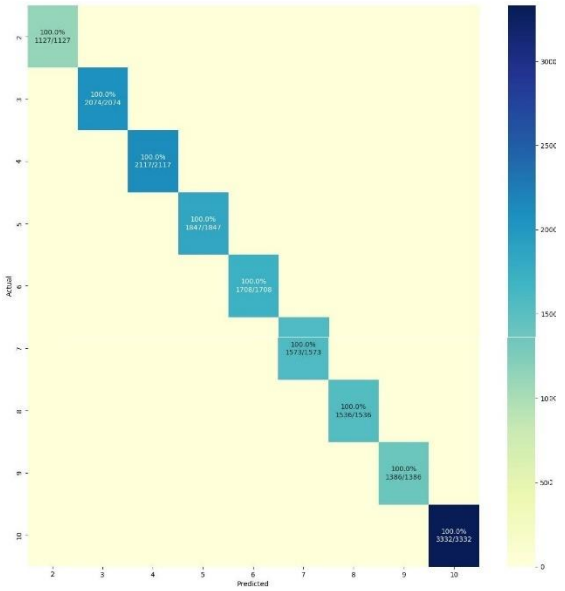
(الف)



(ب)

شکل ۵: ماتریس درهم ریختگی الف) جنگل تصادفی. ب) درخت تصمیم‌گیری

تصمیم عمل می‌کند. دقت و یادآوری این مدل نزدیک به ۱۰۰ درصد می‌باشد. بدین معنی است که تقریبا تمامی تاخیرهای موجود در مسیریابی را به درستی تشخیص می‌دهد. استفاده از این مدل باعث خواهد شد تا به راحتی بتوان پروتکل مسیریابی را با اتکاء به آن پیاده‌سازی کرد. شکل ۶ ماتریس درهمریختگی مدل گرادیان تقویت شده را نشان می‌دهد.



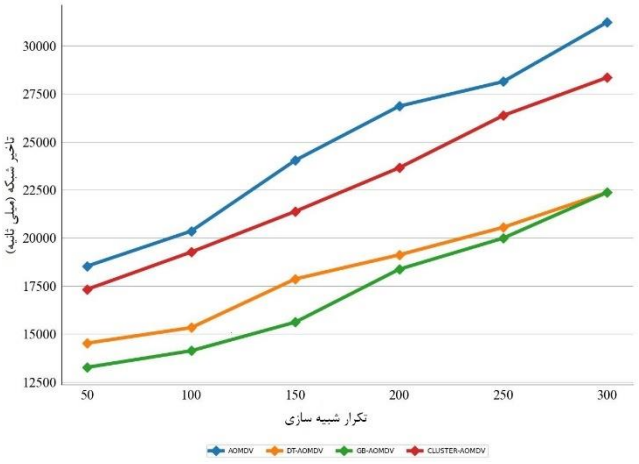
شکل ۶: ماتریس درهم ریختگی روش گرادیان تقویت شده

یکی از مهمترین اهداف رویکرد پیشنهادی کاهش میزان تاخیر شبکه با هدایت بسته‌ها از مسیرهای ایده‌ال می‌باشد. با توجه به اینکه دو روش درخت تصمیم‌گیری و روش گرادیان تقویتی نتایج خوبی داشته‌اند، این دو مدل به عنوان روش‌های دسته‌بندی انتخاب شده‌اند.

تاخیر یک شبکه به عوامل متعددی وابسته است. مهم‌ترین تاخیر شبکه مربوط به هدایت بسته‌ها و مسیریابی می‌باشد که به پارامترهای متعددی وابسته می‌باشد. لازم به ذکر است که افزایش ترافیک شبکه نیز باعث افزایش انتظار بسته‌ها در بافر گره‌های شبکه شده و مستقیما بر تاخیر انتها به انتها تاثیر می‌گذارد. همچنین متحرک بودن گره‌های شبکه، موجب تغییر فاصله مابین گره‌ها شده و در تاخیر انتها به انتها تاثیر گذار باشد. در مرحله جمع‌آوری مجموعه داده، برای برچسب‌گذاری داده‌های جمع‌آوری شده از میزان تاخیر استفاده شده است. مطابق نتایج شبیه‌سازی، تاخیر بین ۱ و ۱۲۷ میلی‌ثانیه متغیر است. در رویکرد پیشنهادی هر ده میلی ثانیه یک کلاس در نظر گرفته شده است و در مجموع ده کلاس مختلف وجود دارد که کلاس دهم به تاخیرهای بیش از ۱۰۰ میلی ثانیه اختصاص داده شده است. تکنیک‌های یادگیری ماشین برای پیش‌بینی این ده کلاس استفاده شده‌اند و مسیری که کمترین میزان تاخیر را داشته باشد به عنوان مسیر ارسال بسته انتخاب می‌کنند.

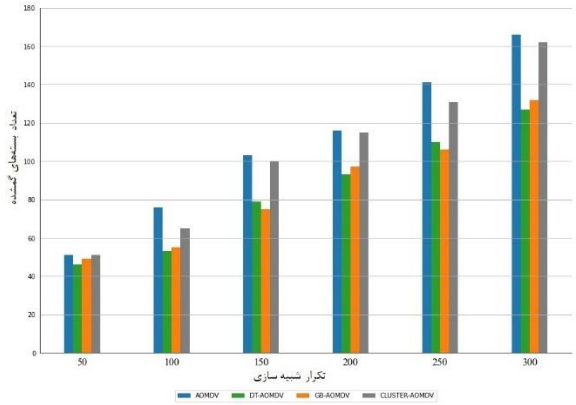
در ادامه، عملکرد رویکرد پیشنهادی با به کارگیری درخت تصمیم(DT-AOMDV) و گرادیان تقویتی (GB-AOMDV) با نسخه AOMDV پایه و زمانی که از k-nearest (CLUSTER-AOMDV) برای خوشه‌بندی استفاده می‌کند، مقایسه می‌شود.

تاخیر انتها به انتهای این روش‌ها در شکل ۷ مقایسه شده است. مطابق شکل، عملکرد رویکرد پیشنهادی در صورت استفاده از گرادیان تقویتی بهترین عملکرد را دارد. علت اصلی این برتری، عملکرد بهتر روش دسته‌بندی استفاده شده و دقت بالای پیش‌بینی این مدل می‌باشد. به دلیل اینکه درخت تصمیم گیری خود زیر ساخت اصلی روش گرادیان تقویتی را تشکیل داده است با برآیند کلی درخت‌های تشکیل دهنده خود اقدام به تصمیم‌گیری می‌کند. با در نظر گرفتن این ویژگی در پیش بینی‌های مختلف در طول زمان بهترین درخت تصمیم گیری موجود در گرادیان تقویتی نتایج نزدیک به درخت تصمیم گیری با بهترین آموزش را پیش بینی می‌کنند که باعث می‌شود به نتایج یکسان دست یابند و این دلیل اصلی همگرایی دو دسته روش به یکدیگر است.



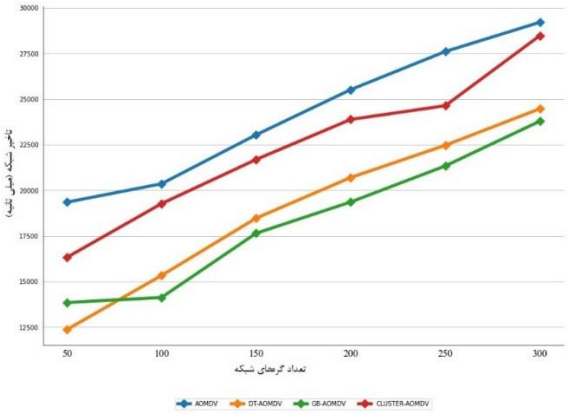
شکل ۷: تاخیر انتها به انتهای روش‌های مختلف با تکرار شبیه‌سازی‌های مختلف

در ارزیابی دیگر سعی شده است که تعداد بسته‌های گم‌شده این رویکردها با تعداد تکرارهای مختلف مقایسه شود. بدیهی است که با افزایش زمان شبیه‌سازی، تعداد بسته‌های ارسالی و دریافتی از منابع و مقاصد مختلف افزایش خواهد داشت که موجب افزایش ترافیک و احتمال گم شدن بسته‌ها می‌شود. در شکل ۸ تعداد بسته‌های گم شده رویکردهای مختلف مقایسه شده است. نتایج حاصل ثابت می‌کنند که DT-AOMDV و GB-AOMDV عملکرد بهتری نسبت رویکردهای مورد مقایسه دارد. طبق پیش‌بینی‌های انجام شده، مدل‌های درخت تصمیم‌گیری و گرادیان تقویت شده با استفاده بهینه از مجموعه داده جمع‌آوری شده قادر هستند با شناسایی بهترین مسیرها، تعداد بسته‌های گم شده را کاهش دهند.



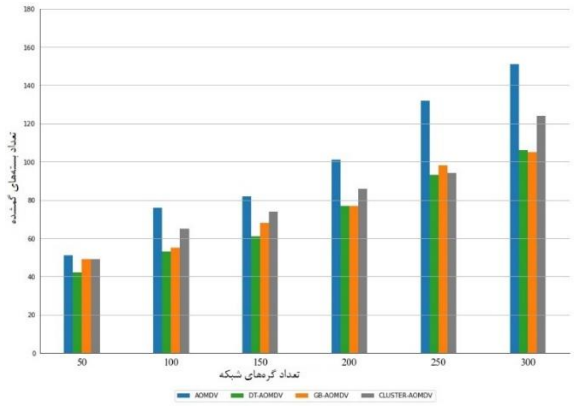
شکل ۸: تعداد بسته‌های از دست رفته روش‌های مختلف با تکرار شبیه‌سازی‌های مختلف

در آزمایش‌های قبلی همیشه تعداد گره‌های شبکه ثابت در نظر گرفته شده بود و مدت زمان شبیه‌سازی متغیر در نظر گرفته شده است. در حالی که تعداد گره‌های شبکه یک پارامتر تاثیرگذار در عملکرد شبکه می‌باشد. در دو ارزیابی دیگر سعی شده است عملکرد رویکردهای مورد مقایسه با تعداد گره‌های مختلف ارزیابی شود. در شکل ۹ تاخیر انتها به انتهای رویکردهای مختلف با تعداد گره‌های مختلف مقایسه شده است. مطابق نتایج حاصل، افزایش تعداد گره موجب افزایش انتها به انتهای همه رویکردهای مورد مقایسه شده است. تفاوت موجود مربوط به روند سیر صعودی و شیب نمودارها می‌باشد. مطابق شکل، رویکرد پیشنهادی مبتنی بر گرادیان تقویت شده بهترین عملکرد را دارد و میزان تاخیر آن نسبت به سایر روش‌ها در شرایط یکسان کمتر می‌باشد.



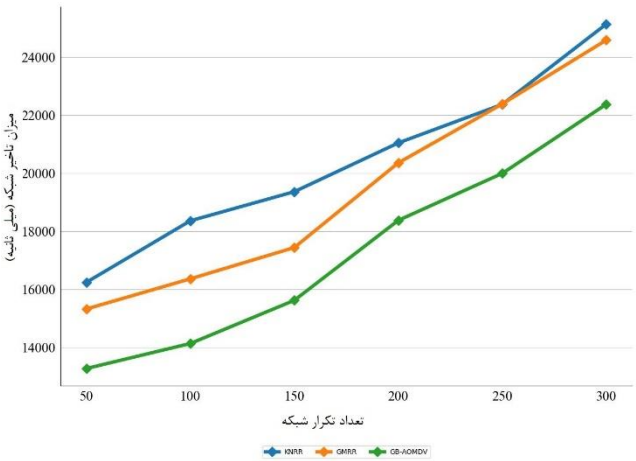
شکل ۹: تاخیر انتها به انتهای روش‌های مختلف با تعداد گره‌های مختلف

هرچه تعداد بسته‌های گمشده کم باشد، تعداد ارسال مجددها و ترافیک شبکه کاهش یافته و در نتیجه میزان تاخیر انتها به انتها نیز کاهش پیدا می‌کند. از این نظر، مدل ایجاد شده توسط روش درخت تصمیم‌گیری و گرادیان تقویت شده عملکرد مشابهی بهم دارند و تعداد بسته کمتری گم شده است. این نتایج در شکل ۱۰ قابل مشاهده است. نتایج این آزمایش با نتایج آزمایش قسمت قبل منطبق بوده و نشان می‌دهد که مدل درخت تصمیم‌گیری و گرادیان تقویتی هر دو با تاخیر کمتری نسبت به دو روش دیگر عمل می‌کنند. در واقع هر چه تعداد گره‌های شبکه افزایش یابد، باعث افزایش پیچیدگی شبکه شده و بر نحوه پیش‌بینی تاخیر مسیرها تاثیر می‌گذارد.



شکل ۱۰: تعداد بسته‌های از دست رفته روش‌های مختلف با تعداد گره‌های مختلف

در آخرین آزمایش روش پیشنهادی با سایر روش‌هایی که در تحقیقات انجام شده است مورد مقایسه قرار می‌گیرد که نتایج در شکل ۱۱ گزارش شده است. همان طور که مشخص است روش پیشنهادی میزان تاخیر کمتری در مقایسه با دو روش KNRR و GMRR داشته است. میزان تاخیر کم همراه با تعداد بسته از دست رفته کمتر نیز می‌باشد. دلیل کاهش میزان تاخیر شبکه به دلیل مسیریابی قدم به قدم روش پیشنهادی نسبت به دو روش دیگر است که کل مسیر را یکباره به دست می‌آورد. از طرفی دقت پیش بینی مدل پیشنهادی که مبتنی بر گرادیان تقویتی است نسبت به دو مدل دیگر بهتر بوده و این دقت منجر به کاهش تاخیر نسبت به دو روش دیگر می‌شود.



شکل ۱۱: میزان تاخیر انتها به انتها در تحقیقات مختلف در مقایسه با روش پیشنهادی

۵- نتیجه‌گیری

در این مقاله یک رویکرد جدید مسیریابی مبتنی بر یادگیری ماشین برای اینترنت اشیاء متحرک ارائه گردید که مبتنی بر پروتکل AOMDV پایه می‌باشد. در رویکرد پیشنهادی سعی گردید برای رفع مشکلات پروتکل AOMDV از یادگیری ماشین استفاده شود که با بهره‌گیری از اطلاعات مختلف شبکه مانند متوسط تعداد همسایه‌های هر گره، متوسط سرعت حرکت گره‌ها، متوسط تعداد بسته‌های موجود در بافر و متوسط انرژی باقیمانده گره‌های شبکه یک مسیر با کمترین تاخیر را انتخاب کند. همچنین پنج رویکرد یادگیری ماشین اعم از ماشین بردار پشتیبان، بیز ساده، درخت تصمیم‌گیری، جنگل تصادفی و گرادیان تقویت شده برای بهبود عملکرد پروتکل AOMDV پایه مورد استفاده قرار گرفتند. مطابق شبیه‌سازی‌های انجام شده در پایتون دو روش درخت تصمیم‌گیری و گرادیان تقویت شده عملکرد بهتری دارند و برای ادامه کار این دو تکنیک انتخاب شده‌اند. برای اثبات بهینگی این تکنیک‌ها، دقت، یادآوری، امتیاز اف ۱ و ماتریس درهم‌ریختگی آنها مورد مقایسه قرار گرفته است. در نهایت تاخیر انتها به انتها و تعداد بسته گم شده رویکرد پیشنهادی با تعداد تکرار مختلف و تعداد گره مختلف با تکنیک‌های مشابه مقایسه شده است. نتایج شبیه‌سازی‌ها نشان می‌دهد که رویکردهای DT-AOMDV و GB-AOMDV بهترین عملکرد را دارند و گزینه‌های مناسبی برای اینترنت اشیاء متحرک می‌باشند. لازم به ذکر است که استفاده از تکنیک‌های یادگیری عمیق و یادگیری تقویت شده می‌تواند به عنوان کارهای آتی این تحقیق مدنظر قرار بگیرند.

مراجع

[1] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, "A Survey on the Internet of Things (IoT) Forensics: Challenges, Approaches, and Open Issues," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 2, pp. 1191–1221, 2020.

[2] A. Razaque, Y. Jararweh, B. Alotaibi, M. Alotaibi, and M. Almiani, "Hybrid energy-efficient algorithm for efficient Internet of Things deployment," *Sustain. Comput. Informatics Syst.*, vol. 35, p. 100715, Sep. 2022.

[3] M. Afrashteh and S. Babaie, "A Route Segmented Broadcast Protocol based on RFID for Emergency Message Dissemination in Vehicular Ad-hoc Networks," *IEEE Trans. Veh. Technol.*, 2020.

[4] P. Fabian, A. Rachedi, C. Gueguen, and S. Lohier, "Fuzzy-Based Objective Function for Routing Protocol in the Internet of Things," *2018 IEEE Glob. Commun. Conf. GLOBECOM 2018 - Proc.*, 2018.

[5] H. Liu and B. Lang, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *Appl. Sci.*, vol. 9, no. 20, p. 4396, Oct. 2019.

[6] M. Bagaa, T. Taleb, J. B. Bernabe, and A. Skarmeta, "A Machine Learning Security Framework for Iot Systems," *IEEE Access*, vol. 8, pp. 114066–114077, 2020.

[7] V. Vashishth, A. Chhabra, and D. K. Sharma, "GMMR: A Gaussian mixture model based unsupervised machine learning approach for optimal routing in opportunistic IoT networks," *Comput. Commun.*, vol. 134, pp. 138–148, Jan. 2019.

[8] S. Boussoufa-Lahlah, F. Semchedine, and L. Bouallouche-Medjkoune, "Geographic routing protocols for Vehicular Ad hoc NETworks (VANETs): A survey," *Veh. Commun.*, vol. 11, pp. 20–31, Jan. 2018.

[9] S. K. Dhurandher, D. K. Sharma, I. Woungang, and S. Bhati, "HBPR: History Based Prediction for Routing in Infrastructure-less Opportunistic Networks," in *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, 2013, pp. 931–936.

[10] D. K. Sharma, Aayush, A. Sharma, and J. Kumar, "KNNR:K-nearest neighbour classification based routing protocol for opportunistic networks," in *2017 Tenth International Conference on Contemporary Computing (IC3)*, 2017, pp. 1–6.

[11] M. Garg, A. Gupta, D. Kaushik, and A. Verma, "Applying machine learning in IoT to build intelligent system for packet routing system," *Mater. Today Proc.*, Nov. 2020.

[12] K. Rama Abirami and M. G. Sumithra, "Evaluation of neighbor credit value based AODV routing algorithms for selfish node behavior detection," *Cluster Comput.*, vol. 22, no. S6, pp. 13307–13316, Nov. 2019.

[13] N. V. Brindha and V. S. Meenakshi, "A secured optimised AOMDV routing protocol in MANET using lightweight continuous multimodal biometric authentication," *J. Ambient Intell. Humaniz. Comput.*, pp. 1–17, Apr. 2022.

[14] F. Kuang, S. Zhang, Z. Jin, and W. Xu, "A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection," *Soft Comput.*, vol. 19, no. 5, pp. 1187–1199, 2015.

[15] B. Karakostas, "Event Prediction in an IoT Environment Using Naïve Bayesian Models," *Procedia Comput. Sci.*, vol. 83, pp. 11–17, 2016.

[16] J. Zhang, M. Zulkernine, and A. Haque, "Random-forests-based network intrusion detection systems," *IEEE Trans. Syst. Man, Cybern. Part C -Applications Rev.*, vol. 38, no. 5, pp. 649–659, 2008.

[17] P. Bedi, N. Gupta, and V. Jindal, "I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems," *Appl. Intell.*, vol. 51, no. 2, pp. 1133–1151, Feb. 2021.

پاورقی‌ها

[1] Internet of Things (IoT)
[2] Sensor
[3] Ad hoc On-Demand Distance Vector (AOMDV)
[4] k-nearest Neighbors Routing
[5] Machine Learning protocol Opp net(MLPROPH)
[6] Vashishth
[7] Soft Clustering Mechanism
[8] Gaussian Model Routing (GMMR)
[9] Probability Routing Protocol Using History of Encounters and Transitivity (PROPHET)
[10] Opportunistic Network Environment (ONE)
[11] Delivery Probability
[12] Network Overhead Ratio
[13] Average Hop Count
[14] Number of Messages Dropped
[15] Mobile Ad hoc Networks (MANET)
[16] Dhurradher
[17] History Based Prediction Routing (HBRP)
[18] Markov Predictor
[19] Hybrid probability Based Protocol (HPBR)
[20] Garg
[21] Black Hole
[22] Ad hoc On-Demand Distance Vector (AODV)
[23] Support Vector Machine (SVM)
[24] Hyperplane
[25] Bayesian Learning
[26] Decision Tree
[27] Random Forest
[28] Gradient Boosting