**تشخیص ناهنجاری برای تشخیص خطا در شبکه‌های اجتماعی بی‌سیم با استفاده از یادگیری ماشین** 2

یورنس سردا-آلابرن 3، گابریل یوهاس 4، گابریله جمی 5

*دانشگاه پلی‌تکنیک کاتالونیا، بارسلون، اسپانیا* 6*دانشگاه غربی، تیمیشوارا، رومانی* 7*دانشگاه ونیز کا فوسکاری، ایتالیا* 8

چکیده 9

یادگیری ماشین در سال‌های اخیر توجه روزافزونی را در علوم کامپیوتر به خود جلب کرده و انواع بسیاری از روش‌ها پیشنهاد شده‌اند10. در شبکه‌های کامپیوتری، توجه کمی به استفاده از یادگیری ماشین (ML) برای تشخیص خطا شده است 11، که دلیل اصلی آن کمبود مجموعه‌داده (dataset) است12. این امر ناشی از عدم تمایل اپراتورهای شبکه برای اشتراک‌گذاری داده‌های مربوط به زیرساخت‌ها و خرابی‌های شبکه خود است13. در این مقاله، ما تلاش می‌کنیم با استفاده از تکنیک‌های تشخیص ناهنجاری، این شکاف را پر کنیم تا رویدادهای خرابی سخت‌افزاری را در شبکه‌های اجتماعی بی‌سیم تشخیص دهیم14. برای این منظور، ما از ۴ رویکرد یادگیری ماشین بدون نظارت (unsupervised) استفاده می‌کنیم که بر اساس اصول متفاوتی بنا شده‌اند15. ما یک مجموعه‌داده از یک شبکه اجتماعی بی‌سیم در حال کار (production) ساخته‌ایم که شامل ویژگی‌های ترافیکی و غیر ترافیکی، مانند بار پردازنده (CPU) و حافظه است16. برای تحلیل عددی، ما توانایی رویکردهای مختلف ML را در تشخیص یک خرابی ناخواسته درگاه (gateway) که در طول جمع‌آوری داده رخ داد، بررسی کردیم17. نتایج عددی ما نشان می‌دهد که وقتی ویژگی‌های غیر ترافیکی نیز در نظر گرفته می‌شوند، عملکرد همه رویکردهای آزمایش‌شده در تشخیص خرابی درگاه بهبود می‌یابد18. ما می‌بینیم که وقتی به درستی تنظیم شوند، همه روش‌های ML در تشخیص خرابی مؤثر هستند19. با این وجود، با استفاده از مرزهای تصمیم‌گیری و سایر تکنیک‌های تحلیلی، ما تفاوت‌های رفتاری قابل توجهی را در میان روش‌های ML مشاهده می‌کنیم20.

**۱.**

مقدمه 21

تشخیص ناهنجاری (AD) با هدف شناسایی انحرافات از رفتار مورد انتظار انجام می‌شود22. ثابت شده است که شناسایی این رفتارهای غیرمعمول، ابزاری قدرتمند در طیف گسترده‌ای از علوم کاربردی است23. برخی نمونه‌ها شامل تشخیص کلاهبرداری کارت اعتباری، تشخیص پزشکی، فرآیندهای صنعتی و شبکه‌های کامپیوتری است24. اگرچه روش‌های مختلفی را می‌توان برای AD به کار برد 25، پیشرفت‌های اخیر در روش‌های ML و توانایی آن‌ها در یادگیری از داده‌ها، تعداد پیشنهادات AD با استفاده از تکنیک‌های ML را افزایش داده است26.

یادگیری ماشین از یک مجموعه‌داده برای پیش‌بینی‌های احتمالی استفاده می‌کند27. گروهی از نمونه‌های مجموعه‌داده که به عنوان «مجموعه آموزش» (training set) شناخته می‌شوند، برای آموزش الگوریتم استفاده می‌شوند28. سپس، پیش‌بینی‌ها بر روی گروه متفاوتی از نمونه‌ها که «مجموعه آزمایشی» (testing set) نامیده می‌شوند، انجام می‌پذیرد29. در برخی کاربردها، مجموعه آموزش برچسب‌گذاری شده است30. به عنوان مثال، در تشخیص نوری کاراکترها، مجموعه آموزش شامل تصاویر و برچسب‌های متناظر با کاراکترهای آن‌هاست31. این مسائل، «یادگیری با نظارت» (supervised learning) نامیده می‌شوند32. در مقابل، در AD معمولاً از یک مجموعه‌داده بدون برچسب استفاده می‌شود و الگوریتم‌های ML، «بدون نظارت» (unsupervised) نامیده می‌شوند33. در AD، مجموعه آموزش صرفاً به عنوان نمایشی از عملیات عادی مورد انتظار استفاده می‌شود و ناهنجاری‌ها با انحراف از رفتار مورد انتظار شناسایی می‌شوند34.

در حوزه شبکه‌های کامپیوتری، AD عمدتاً به امنیت، به‌ویژه تشخیص نفوذ به شبکه، معطوف بوده است35. در مقابل، ما در این مقاله بر روی «تشخیص خطا» (fault detection) تمرکز می‌کنیم 3636، یکی دیگر از نیازهای اساسی در شبکه‌های کامپیوتری که AD می‌تواند در آن بسیار مورد توجه باشد، اما توجه کمی به آن شده است37373737.

برخی مشکلات، کارهای اندکی را که می‌توان در ادبیات موضوع در مورد AD برای تشخیص خطا یافت، توضیح می‌دهند38. مشکل اصلی، کمبود مجموعه‌داده است39. دلیل آن این است که یک مجموعه‌داده واقع‌بینانه برای تشخیص خطا باید شامل ویژگی‌های مرتبط با ترافیک شبکه و معیارهای سخت‌افزاری، مانند بار پردازنده و استفاده از حافظه باشد40. تولید این نوع مجموعه‌داده از طریق شبیه‌سازی دشوار خواهد بود، بلکه باید از یک بستر آزمایشی واقعی یا در حالت ایده‌آل، از یک شبکه در حال کار به دست آید41. با این حال، به دلایل محرمانگی، اپراتورهای شبکه تجاری این نوع مجموعه‌داده‌ها را از شبکه‌های خود عمومی نمی‌کنند42.

بنابراین، هدف دیگر کار ما ایجاد یک مجموعه‌داده با داده‌های واقعی است43. برای انجام این کار، ما بر روی AD در شبکه‌های اجتماعی بی‌سیم (WCN) تمرکز خواهیم کرد44. شبکه‌سازی اجتماعی، که به عنوان شبکه‌سازی از پایین به بالا نیز شناخته می‌شود، مدلی است که از نیاز به دسترسی گسترده به اینترنت در مناطق محروم، معمولاً مناطق روستایی و کشورهای در حال توسعه، پدید آمده است45. امروزه صدها شبکه اجتماعی وجود دارند که به روش‌های بسیار متنوعی فعالیت می‌کنند46. WCNها شبکه‌های غیرانتفاعی هستند که توسط کاربران خودشان ساخته می‌شوند، معمولاً با نصب آنتن‌های بی‌سیم بر روی سقف خانه‌هایشان47. گاهی اوقات کاربران WCN نه تنها زیرساخت شبکه را می‌سازند، بلکه برای تشکیل ISPهای خرد (micro ISP) با هم متحد می‌شوند و دسترسی به اینترنت و خدمات داخلی مدیریت‌شده توسط جامعه را فراهم می‌کنند48. WCNها شبیه به ارائه‌دهندگان خدمات اینترنت بی‌سیم (WISP) هستند 49که معمولاً کسب‌وکارهای کوچکی با حاشیه سود کاهش‌یافته هستند و هدفشان ارائه اینترنت گسترده به جامعه خود است50.

ماهیت غیرانتفاعی و منابع مشترک شبکه‌های اجتماعی معمولاً کاربران آن‌ها را برای مشارکت در پروژه‌های تحقیقاتی و فراهم کردن دسترسی به داده‌های زیرساخت شبکه خود، پذیرا می‌سازد 51. بنابراین ما با هدف ساخت یک مجموعه‌داده از یک شبکه در حال کار، بر روی WCN تمرکز کرده‌ایم52. به طور خاص، ما Guifi.net را در نظر گرفته‌ایم53. Guifi.net یکی از نمونه‌های برجسته WCN است54. این شبکه در سال ۲۰۰۴ آغاز به کار کرد و در زمان نگارش این مقاله، ۳۶٬۸۸۶ گره فعال را گزارش می‌دهد55. Guifi.net به یک شبکه پیچیده تبدیل شده است که در آن انجمن‌های کاربران خود-تأمین‌کننده با اپراتورهای شبکه تجاری همزیستی دارند56.

در این مقاله، ما یک مجموعه‌داده را که از یک WCN در حال کار که بخشی از Guifi.net است جمع‌آوری کرده‌ایم، تحلیل می‌کنیم57. این WCN در محله‌ای از بارسلون، اسپانیا، مستقر شده است و GuifiSants نامیده می‌شود 58. هنگامی که مجموعه‌داده جمع‌آوری شد، حدود ۶۰ گره وجود داشت59. GuifiSants از نظر توپولوژی، کیفیت پیوندها و رفتار گره‌ها بسیار ناهمگن است60. گره‌هایی در ساختمان‌های بلندتر وجود دارند که دارای چندین آنتن هستند، برخی سکتور و برخی دیگر پارابولیک که پیوندهای نقطه به نقطه با ظرفیت بالا ایجاد می‌کنند61. این گره‌ها نوعی ستون فقرات (backbone) برنامه‌ریزی‌نشده را تشکیل می‌دهند62. سایر کاربران غیر فنی دارای گره‌های نهایی هستند که از یک آنتن واحد ساخته شده‌اند63. شکل ۱ مکان‌های جغرافیایی گره‌های GuifiSants و تصویری از روترهای فضای باز مستقر در یک گره را نشان می‌دهد64646464. بسته به نوع آنتن‌ها، انسداد پیوندها و فناوری‌های WiFi 802.11n/ac مورد استفاده، ظرفیت پیوند از چند مگابیت بر ثانیه تا چند صد مگابیت بر ثانیه متغیر است65. پایداری گره‌ها نیز بسیار متغیر است66. به عنوان مثال، کاربرانی هستند که مرتباً نرم‌افزار را به‌روزرسانی می‌کنند یا اگر اتصال رضایت‌بخش نباشد، گره خود را راه‌اندازی مجدد (reboot) می‌کنند67. گره‌های ستون فقرات پایدارتر هستند و خرابی یا راه‌اندازی مجدد کمتری دارند68. از آنجایی که این یک شبکه مِش (mesh) است، اتصال نسبتاً انعطاف‌پذیر (resilient) است: حتی اگر یک گره ستون فقرات از کار بیفتد، پروتکل مسیریابی مسیرهای جایگزین را مجدداً پیکربندی می‌کند و اکثر گره‌ها همچنان می‌توانند به اینترنت، که سرویس اصلی مصرفی کاربران است، دسترسی داشته باشند69.

AD در چنین WCN به دلیل ماهیت ناهمگن و متنوع آن چالش‌برانگیز است70. با این وجود، ما معتقدیم که این نه تنها یک مطالعه جالب به خودی خود است، بلکه سناریوهای دیگری مانند شبکه‌های بزرگ اینترنت اشیاء (IoT) نیز ممکن است ویژگی‌های تنوع مشابهی داشته باشند، که به ما امکان می‌دهد برخی از درس‌های آموخته‌شده از تحلیل WCN را تعمیم دهیم71.

اکثر مطالعات AD مرتبط با شبکه‌های کامپیوتری که در ادبیات موضوع یافت می‌شوند، از ویژگی‌های مرتبط با ترافیک استفاده می‌کنند72. در مقابل، مجموعه‌داده‌ای که ما تولید کرده‌ایم نه تنها شامل ترافیک، بلکه شامل ویژگی‌هایی از هسته لینوکس است که به عنوان مثال، وضعیت پردازنده و حافظه را منعکس می‌کند 73. چنین ویژگی‌هایی معمولاً توسط ابزارهای نظارت بر شبکه مانند Munin و Nagios جمع‌آوری می‌شوند 74. با این حال، این ابزارها صرفاً گرافیک‌های سری زمانی از ویژگی‌های جمع‌آوری‌شده تولید می‌کنند و تحلیل و تفسیر آن‌ها به مدیر شبکه واگذار می‌شود75. بنابراین ما علاقه‌مندیم بدانیم که آیا گنجاندن این ویژگی‌های اضافی به ویژگی‌های اساسی مبتنی بر ترافیک می‌تواند قابلیت AD را افزایش دهد یا خیر76.

به طور خلاصه، مشارکت‌های اصلی ما در این مقاله به شرح زیر است:

* ما یک مجموعه‌داده تولید کرده‌ایم که مجموعه بزرگی از ویژگی‌های مرتبط نه تنها با ترافیک، بلکه با پارامترهای دیگری مانند پردازنده و حافظه را جمع‌آوری می‌کند77. این مجموعه‌داده از یک WCN در حال کار جمع‌آوری شده و ما آن را در مخزن عمومی Zenodo در دسترس قرار داده‌ایم 78. تا جایی که ما اطلاع داریم، این اولین مجموعه‌داده‌ای است که شامل چنین تنوع غنی از ویژگی‌های به‌دست‌آمده از یک شبکه بی‌سیم در حال کار است که در یک مخزن عمومی موجود است79.
* ما از روش‌های AD با استفاده از این مجموعه‌داده برای انجام تحلیل تشخیص خطا با استفاده از ۴ رویکرد یادگیری بدون نظارت ML مبتنی بر اصول مختلف استفاده کرده‌ایم80. ما عملکرد این روش‌های شناخته‌شده ML را که برای AD در شبکه‌های بی‌سیم به کار می‌روند و رفتار آن‌ها را با تغییر برخی پارامترهای سیستم، مانند اندازه مجموعه‌داده، بررسی می‌کنیم81.

ادامه مقاله به این صورت سازمان‌دهی شده است: بخش ۲ برخی از کارهای مرتبط را ارائه می‌دهد82. بخش ۳، ۴ رویکرد ML مورد استفاده در این مقاله را خلاصه می‌کند83. بخش ۴ (خطا در متن اصلی، باید ۳ باشد) متدولوژی و ویژگی‌های جمع‌آوری‌شده در مجموعه‌داده مورد مطالعه را توصیف می‌کند84. بخش ۵ نتایج عددی را ارائه می‌دهد 85و بخش ۶ مقاله را به پایان می‌رساند86.

**۲.**

کارهای مرتبط 87

تشخیص ناهنجاری به مسئله یافتن الگوهایی در داده‌ها اشاره دارد که با رفتار مورد انتظار مطابقت ندارند88. در آمار، اغلب از اصطلاح «داده پرت» (outlier) برای این مفهوم استفاده می‌شود89. اهمیت AD به این دلیل است که ناهنجاری‌ها در داده‌ها ممکن است نشان‌دهنده رفتارهای نادرست حیاتی در سناریوهای متعدد باشند90. به عنوان مثال، تشخیص کلاهبرداری برای کارت‌های اعتباری، بیمه یا مراقبت‌های بهداشتی، تشخیص آسیب و غیره91. اهمیت این موضوع منجر به تعداد زیادی مقاله، بررسی و حتی کتاب شده است92. به عنوان مثال، بررسی‌های [1, 20] مروری گسترده بر تکنیک‌های تشخیصی ارائه می‌دهند که در حوزه‌های متعددی توسعه یافته‌اند93.

در شبکه‌های کامپیوتری، بیشتر کارهای مربوط به AD به طور سنتی بر روی مسئله امنیت متمرکز بوده‌اند94. در این زمینه، اغلب از اصطلاح «سیستم تشخیص نفوذ» (IDS) استفاده می‌شود95. IDSها معمولاً از ناهنجاری‌ها در ترافیک شبکه برای تشخیص طیف گسترده‌ای از حملات امنیتی، مانند حملات محروم‌سازی از سرویس (DoS)، شناسایی (reconnaissance) و غیره بهره می‌برند96. مروری بر IDSهای پیشنهادی در ادبیات موضوع را می‌توان در [7] یافت97. در [22]، PCA (تحلیل مؤلفه‌های اصلی) برای انجام تحلیل ساختاری ترافیک شبکه معرفی شد98. پیشنهاد آن‌ها توجه قابل توجهی را برای IDS به خود جلب کرد99. به عنوان مثال، در [23, 24] اصلاحاتی در تحلیل PCA برای تشخیص جهش‌ها (spikes) در جریان‌های ترافیکی انجام شده است، که به طور بالقوه حملات شبکه را شناسایی می‌کند 100. PCA برای تشخیص ناهنجاری‌های ترافیک شبکه توسط تعدادی از مقالات نیز مورد انتقاد قرار گرفته است101. در [27] ضعف‌های PCA تحلیل شده و از Commute Distance برای تشخیص حملات DoS با استفاده از اندازه‌گیری‌های ترافیک استفاده شده است102. در [14] استدلال می‌شود که مشکلات PCA ناشی از نقص در پذیرش آن است و روش‌های مناسبی برای اعمال PCA پیشنهاد می‌شود103.

روش‌های ML سابقه طولانی در استفاده برای AD دارند104. تا همین اواخر، بیشتر تحقیقات بر روی استفاده از روش‌های ML با نظارت همراه با مجموعه‌داده‌های مصنوعی یا به شدت حاشیه‌نویسی‌شده انجام می‌شد105. اخیراً این روند به سمت استفاده از مجموعه‌داده‌های واقعی همراه با روش‌های بدون نظارت تغییر کرده است106. با این حال، مقالات تحقیقاتی زیادی وجود ندارند که هم مقایسه‌ای از روش‌های AD بدون نظارت و هم یک مجموعه‌داده واقعی در دسترس عموم را داشته باشند107107107107.

مرور بسیار خوبی از روش‌های بدون نظارت بر روی چندین مجموعه‌داده در [29] آورده شده است108. در اینجا ما عملکرد چندین روش شناخته‌شده AD مانند k-NN، CBLOF و غیره را می‌بینیم109. تحقیقات مرتبط‌تری شامل برخی از همین روش‌ها در [30] انجام شده است که در آن نویسندگان چندین تکنیک AD مبتنی بر خوشه‌بندی را بر روی داده‌های ترافیک شبکه مقایسه می‌کنند110. انواع دیگر روش‌های AD نیز در سیستم‌های مدیریت شبکه با داده‌های شبکه با ابعاد بالا استفاده شده‌اند111.

روش‌های مبتنی بر یادگیری عمیق نیز برای AD استفاده می‌شوند112. در [32] نویسندگان مروری جامع بر روش‌های مختلف یادگیری عمیق برای AD شامل روش‌های بدون نظارت ارائه می‌دهند113. آن‌ها همچنین به درستی اشاره می‌کنند که «توضیح‌پذیری» (explainability) یک مسئله کلیدی است، به‌ویژه هنگام استفاده از روش‌های AD که به طور سنتی به عنوان مدل‌های جعبه-سیاه (black-box) در نظر گرفته می‌شوند114. مسئله توضیح‌پذیری همچنین در [33, 34] در مورد مدل‌های مختلف یادگیری عمیق، که شامل خودرمزگذارها (Autoencoders) [35] و خودرمزگذارهای متغیر (Variational-Autoencoders) [36] هستند، مورد بحث قرار گرفته است115.

بیشتر تحقیقات انجام‌شده در مورد روش‌های بدون نظارت برای AD، عمدتاً بر روی تشخیص نفوذ به شبکه متمرکز است116. [37] یکی از معدود کارهای یافت‌شده در ادبیات موضوع است که در آن AD شبکه با استفاده از ML با یک مجموعه‌داده واقعی بررسی شده است117. با این حال، در [37] نویسندگان از AD برای مطالعه رفتار ردیابی‌های TCP (TCP traces) جهت بررسی عملکرد یک شبکه سلولی 4G استفاده می‌کنند که سناریوی کاملاً متفاوتی نسبت به ماست118.

روش‌های مورد استفاده در ادبیات موضوع را می‌توان به چند نوع تقسیم کرد119. اینها شامل روش‌های مبتنی بر خوشه‌بندی [38-41]، تکنیک‌های مبتنی بر داده پرت [37, 42-45] و تکنیک‌های مبتنی بر محاسبات نرم [46-48] است120. همه روش‌های توصیف‌شده در اینجا علاوه بر مشکل توضیح‌پذیری که قبلاً ذکر شد، دارای مزایا و معایبی هستند121. روش‌های مبتنی بر خوشه‌بندی معمولاً فقط می‌توانند ویژگی‌های پیوسته را مدیریت کنند؛ معیارهای نزدیکی نامناسب بر نرخ تشخیص تأثیر منفی می‌گذارد122. بیشتر روش‌های مبتنی بر داده پرت پیچیده هستند (معمولاً در ترکیب با خوشه‌بندی استفاده می‌شوند) و به شدت به پارامترها وابسته‌اند، در حالی که روش‌های مبتنی بر محاسبات نرم معمولاً به مقادیر زیادی داده تاریخی نیاز دارند و مشکل «بیش‌برازش» (overfitting) مدل‌ها مسلماً مسئله بزرگ‌تری نسبت به روش‌های با نظارت است123. در اصل، انتخاب اینکه از چه نوع روشی و چه پارامترهای آموزشی استفاده شود، بسیار به حوزه مسئله و داده‌های موجود بستگی دارد124.

**۳.**

رویکردهای مبتنی بر یادگیری ماشین 125

در بخش بعدی، ما بر روی آزمایش چندین تکنیک ML بدون نظارت برای AD تمرکز خواهیم کرد126. برای تکنیک‌های ML، AD بر اساس نوع روش‌ها و ویژگی‌های داده‌های موجود به چند دسته تقسیم می‌شود127. ساده‌ترین شکل ناهنجاری‌ها، ناهنجاری‌های نقطه‌ای (point anomalies) هستند که می‌توانند تنها با یک ویژگی مشخص شوند و تشخیص آن‌ها آسان‌تر است128. انواع دیگر ناهنجاری‌ها پیچیده‌تر هستند اما در نهایت درک بسیار عمیق‌تری از عملکردهای درونی یک سیستم و/یا برنامه تحت نظارت ارائه می‌دهند129. این نوع ناهنجاری‌ها در سیستم‌های پیچیده توزیع‌شده جغرافیایی بسیار رایج هستند130.

ناهنجاری‌های زمینه‌ای (Contextual anomalies) در مورد سیستم‌های پیچیده بسیار جالب هستند131. این نوع ناهنجاری‌ها زمانی اتفاق می‌افتند که الگوی خاصی از مقادیر ویژگی‌ها مشاهده می‌شود132. به صورت مجزا، این مقادیر ناهنجار نیستند، اما وقتی در زمینه مشاهده می‌شوند، یک رویداد ناهنجار را نشان می‌دهند133. این نوع ناهنجاری‌ها می‌توانند نشان‌دهنده تنگناهای (bottlenecks) برنامه، خرابی قریب‌الوقوع سخت‌افزار، پیکربندی نادرست نرم‌افزار یا حتی فعالیت مخرب باشند134. آخرین انواع عمده ناهنجاری‌ها که مرتبط هستند، ناهنجاری‌های زمانی (temporal) و ترتیبی (sequential) هستند که در آن‌ها یک رویداد خاص خارج از ترتیب یا در زمان‌های نادرست رخ می‌دهد135. این نوع ناهنجاری‌ها در سیستم‌هایی که رابطه فضایی-زمانی قوی بین ویژگی‌ها دارند، بسیار مهم هستند، که این مورد در سیستم‌های توزیع‌شده پویا مانند شبکه‌های مِش بسیار صادق است136.

یک ملاحظه مهم برای اکثر وظایف AD این واقعیت است که داده‌های باکیفیت و برچسب‌گذاری‌شده به ندرت در دسترس هستند137. علاوه بر این، روش‌های با نظارت که بر روی داده‌های برچسب‌گذاری‌شده آموزش دیده‌اند، قادر به تشخیص رویدادهای ناهنجار جدید یا پیش‌بینی‌نشده نخواهند بود138. روش‌های بدون نظارت این محدودیت را ندارند، با این حال، این روش‌ها دارای چندین نقطه ضعف هستند139. اولاً، آنها در بسیاری از موارد مستعد نرخ بالای «مثبت کاذب» (false-positive) هستند140. این می‌تواند به دلایل زیادی مانند بیش‌برازش، فضاهای ویژگی بزرگ و غیره باشد141. ثانیاً، حتی اگر ناهنجاری‌ها تشخیص داده شوند، خود الگوریتم‌ها معمولاً قادر به ارائه هیچ بینش معناداری در مورد اینکه چه نوع ناهنجاری تشخیص داده شده یا حتی چه چیزی باعث رویداد ناهنجار شده است، نیستند142. در این بخش از مقاله، ما بر روی کاهش برخی از مسائل ارائه‌شده در اینجا در مورد روش‌های تشخیص بدون نظارت تمرکز خواهیم کرد143. توجه داشته باشید که ما هر اندازه‌گیری از مجموعه‌داده را به عنوان «نمونه» (sample) ارجاع خواهیم داد144. بنابراین، ما از اصطلاحات رویداد ناهنجار، نمونه ناهنجار یا به سادگی ناهنجاری، به جای یکدیگر استفاده خواهیم کرد145.

**۳.۱.**

تحلیل مؤلفه‌های اصلی (PCA) 146

تحلیل مؤلفه‌های اصلی (PCA) یک متدولوژی استاندارد است که در کنترل فرآیند آماری به کار می‌رود147. ایده PCA تقریب زدن نمونه‌های n ویژگی با یک تصویر (projection) بر روی یک فضای با ابعاد کمتر $I<n$ است148. PCA به خوبی در ادبیات موضوع مطالعه شده است، بنابراین ما از توصیف این روش صرف‌نظر می‌کنیم و فقط در مورد آماره‌هایی که برای تحلیل خود استفاده کردیم بحث می‌کنیم149. خواننده علاقه‌مند می‌تواند برای جزئیات بیشتر به [14] مراجعه کند150.

به منظور تشخیص ناهنجاری‌های یک امتیاز جدید، چندین شاخص وجود دارد151. ما دریافتیم که در مطالعه ما، بهترین عملکرد مربوط به خطای پیش‌بینی مربع (SPE) است که به عنوان آماره Q نیز شناخته می‌شود152:

$Q\_{j}=\sum\_{i=1}^{n}e\_{ji}^{2}$ 153که در آن $e\_{ji}$ باقیمانده i-ام ویژگی‌های نمونه j است154. (۱) 155

هنگامی که یک ناهنجاری تشخیص داده می‌شود، به یک سیستم تشخیص (diagnosis) برای تعیین علل ریشه‌ای آن نیاز است156. رویکرد کلی استفاده از «نمودارهای مشارکت» (contribution plots) است157. ایده چنین نمودارهایی، تخمین مشارکت هر ویژگی مشاهده‌شده از یک نمونه در یک مقدار آماری خاص است که ناهنجار در نظر گرفته شده است158. روش‌های مختلفی برای ساخت نمودارهای مشارکت پیشنهاد شده است159. در این مقاله، ما از «مشارکت‌های تجزیه کامل» (CDC) استفاده خواهیم کرد160. محاسبه CDC آسان است و عملکرد تشخیص خوبی را در یک سیستم نظارت بر شبکه نشان می‌دهد161. CDC برای آماره Q که در بالا تعریف شد، توسط [49] داده شده است162:

(فرمول ۲ حذف شده است) 163163163163

**۳.۲.**

جنگل ایزوله‌سازی (Isolation Forest) 164

در سال‌های اخیر، آشکارسازهای مبتنی بر گروه (ensemble) در مورد AD محبوبیت پیدا کرده‌اند165. این روش‌ها خروجی‌های چندین الگوریتم به نام «آشکارسازهای پایه» (base detectors) را ترکیب می‌کنند که سپس برای ایجاد یک خروجی یکپارچه استفاده می‌شوند166. این الگوریتم‌ها بر این فرض عمل می‌کنند که برخی الگوریتم‌ها بر روی زیرمجموعه‌ای از داده‌های موجود به خوبی عمل می‌کنند در حالی که برخی دیگر می‌توانند بر روی زیرمجموعه متفاوتی بهتر عمل کنند167. از سوی دیگر، ترکیب گروهی اغلب بهتر از برآوردگرهای (estimators) منفرد عمل می‌کند، به دلیل توانایی آن‌ها در ترکیب خروجی‌های چندین الگوریتم168. بیشتر روش‌های گروهی به داده‌های برچسب‌گذاری‌شده نیاز دارند، با این حال، هم از دیدگاه نظری [51] و هم عملی [52] نشان داده شده است که روش‌های گروهی بدون نظارت ویژگی‌هایی با همتایان با نظارت خود دارند169. ما می‌توانیم یک موازنه بایاس-واریانس (bias-variance) اصلاح‌شده را برای تنظیم تحلیل ناهنجاری فرموله کنیم [51]، این امر بسیاری از الگوریتم‌های با نظارت را قادر می‌سازد تا به وظایف بدون نظارت «تعمیم» یابند170. یکی از مسائل کلیدی که نتیجه مستقیم این تعمیم است، افزایش دشواری در انتخاب «فراپارامترهای» (hyper-parameters) مناسب است171.

جنگل ایزوله‌سازی (Isolation Forest) یک الگوریتم مبتنی بر گروه داده پرت است که از چندین «درخت ایزوله‌سازی» (isolation trees) ساخته شده است172. این الگوریتم زیرفضاهای تصادفی از داده‌ها را کاوش می‌کند173. در اصل، این الگوریتم زیرفضاهای محلی تصادفی را کاوش می‌کند زیرا هر درخت از تقسیم‌های (splits) متفاوتی استفاده می‌کند174. امتیازدهی با تعیین اینکه یافتن یک زیرفضای محلی با ابعاد کم که در آن یک رویداد خاص ایزوله شده است چقدر آسان است، انجام می‌شود175. به عبارت دیگر، فاصله از برگ تا ریشه به عنوان امتیاز داده پرت استفاده می‌شود176. مشابه جنگل تصادفی (Random Forest) که یک روش با نظارت است 177، امتیاز نهایی با میانگین‌گیری طول مسیر هر نقطه داده خاص در درختان ایزوله‌سازی مختلف به دست می‌آید178. در بیشتر سناریوها، جنگل ایزوله‌سازی تحت این فرض کار می‌کند که احتمال تشخیص یا ایزوله کردن یک داده پرت در زیرفضایی با ابعاد کمتر که توسط تقسیم‌های تصادفی ایجاد شده، بیشتر است179.

در طول فاز آموزش، جنگل ایزوله‌سازی معادل بدون نظارت درختان تصمیم‌گیری را می‌سازد180. این درختان باینری هستند و حداکثر N گره برگ دارند که برابر با تعداد نقاط داده در مجموعه آموزش است181. ... (توضیح فرآیند ساخت درخت) ... 182. می‌توان دید که نتیجه آموزش یک درخت باینری نامتعادل خواهد بود و گره‌های پرت معمولاً در ابعاد پایین‌تری نسبت به نقاط عادی ایزوله می‌شوند183. رویکرد ذاتاً تصادفی چندین بار تکرار می‌شود و نتایج میانگین‌گیری می‌شوند (گروهی)184. ... این امر منجر به برخی مسائل هنگام استفاده از این روش بر روی مجموعه‌داده‌های واقعی می‌شود185. با استفاده از زیرنمونه‌گیری (subsampling) می‌توان عملکرد محاسباتی را بهبود بخشید و به ایجاد تنوع که برای روش‌های گروهی حیاتی است، کمک کرد186. ... بنابراین، انتخاب صحیح ویژگی‌ها یا خصیصه i به شدت تحت تأثیر تصادفی بودن است187. کاملاً ممکن است که به دلیل انتخاب نامناسب ویژگی در طول آموزش، مدل‌هایی با عملکرد ضعیف داشته باشیم188.

**۳.۳.**

فاکتور ناهنجاری محلی مبتنی بر خوشه‌بندی (CBLOF) 189

چندین فرض اساسی وجود دارد که پایه و اساس روش‌های بدون نظارت را تشکیل می‌دهند190. تکنیک‌های مبتنی بر نزدیکی (Proximity-based)، یک رویداد (یا نقطه داده) ناهنجار را زمانی تعریف می‌کنند که محل آن کم‌جمعیت باشد191. ... در روش‌های مبتنی بر خوشه (cluster-based)، عدم عضویت یک رویداد نسبت به یک خوشه، فاصله آن از خوشه‌های دیگر، اندازه نزدیک‌ترین خوشه، یا ترکیبی از این عوامل، برای محاسبه امتیاز ناهنجاری استفاده می‌شود192. ... روش‌های مبتنی بر فاصله (Distance-based) امتیاز ناهنجاری خود را بر اساس فاصله یک رویداد تا k-نزدیک‌ترین همسایگانش قرار می‌دهند193. ... روش‌های مبتنی بر چگالی (Density-based) بر اساس تعداد رویدادهایی که در یک منطقه محلی هستند کار می‌کنند 194. ...

CBLOF (Clustering-Based Local Outlier Factor) [55] یک الگوریتم مبتنی بر نزدیکی است که ترکیبی از فاکتور ناهنجاری محلی (LOF) و یک تکنیک خوشه‌بندی است [56]195. LOF امتیاز ناهنجاری (یا پرت) را بر اساس چگالی محلی تنظیم می‌کند196. ... در واقع، CBLOF امتیازی است که در آن ناهنجاری‌ها به عنوان ترکیبی از فاصله محلی تا خوشه‌های مجاور و اندازه خوشه‌هایی که هر رویداد به آن تعلق دارد، تعریف می‌شوند197. بنابراین، رویدادها در خوشه‌های کوچک که در فواصل زیادی تا خوشه‌های مجاور قرار دارند، به عنوان ناهنجاری برچسب‌گذاری می‌شوند198.

**۳.۴.**

خودرمزگذارهای متغیر (VAE) 199

خودرمزگذارهای متغیر (VAE) مدل‌های شبکه عصبی عمیق هستند که برای آموزش بدون نظارت طراحی شده‌اند و می‌توانند برای وظایف AD استفاده شوند [57, 58]200. آنها اغلب همراه با خودرمزگذارها (AE) ذکر می‌شوند که آنها نیز مدل‌های یادگیری عمیق با اجزای توپولوژیکی ظاهراً مشابه هستند: رمزگذار (encoder) و رمزگشا (decoder)201. رمزگذار سعی می‌کند یک نمایش با ابعاد کمتر از داده‌های ورودی را یاد بگیرد (مشابه PCA) و رمزگشا تلاش می‌کند داده‌های ورودی را در ابعاد اصلی بازتولید کند202. ... هنگامی که AEs برای AD استفاده می‌شوند، خطای بازسازی (reconstruction error) آنها می‌تواند به عنوان نوعی امتیاز ناهنجاری استفاده شود [59]203.

از سوی دیگر، VAEها سعی می‌کنند داده‌ها را در یک توزیع نهان (latent distribution) چندمتغیره رمزگذاری کنند که سپس توسط رمزگشا استفاده می‌شود204. تفاوت بین AE و VAE این است که به جای تولید یک بردار نهان، که رمزگشا می‌تواند آن را بازتولید کند، VAEها دو بردار را یاد می‌گیرند که پارامترهای میانگین و واریانس یک توزیع را نشان می‌دهند که بردار نهان از آن نمونه‌برداری شده و توسط تابع رمزگشا برای بازسازی ورودی اصلی استفاده می‌شود205. ...

VAEها مسائل مشترکی با سایر مدل‌های یادگیری عمیق دارند206. اولاً، آموزش آنها نسبتاً کند است و اغلب به سخت‌افزار تخصصی مانند GPGPU نیاز دارد207. مجموعه‌داده‌های بزرگ همچنین به توپولوژی‌های شبکه پیچیده نیاز دارند208. با این حال، اگر توپولوژی‌های پیچیده بر روی مجموعه‌داده‌های کوچک اعمال شوند، این می‌تواند منجر به بیش‌برازش شود209. یک روش تنظیم (regularization) که در کاهش بیش‌برازش مؤثر بوده، افزودن لایه «حذف» (dropout) است [60]210.

نوعی از VAE به نام $\beta-VAE$ برای تشخیص ناهنجاری پیشنهاد شده است [61, 62]211. این نوع، یک فراپارامتر جدید به نام $\beta$ اضافه می‌کند که ظرفیت رمزگذاری گلوگاه نهان را محدود می‌کند و نمایش‌های نهان را تشویق می‌کند تا فاکتورسازی شوند212. لازم به ذکر است که VAE استاندارد را می‌توان حالت خاصی از این نوع جدید در نظر گرفت، و این دو زمانی معادل هستند که $\beta=1$213.

**۳.۵.**

انتخاب ویژگی 214

مجموعه‌داده‌های نامتعادل با ابعاد بالا یکی از چالش‌های اصلی حوزه تحقیقاتی ML هستند [63]215. با استفاده از روش‌هایی که امکان انتخاب دستی یا خودکار یک زیرمجموعه مرتبط را از میان مجموعه اولیه بزرگ و بالقوه افزونه از ویژگی‌های اولیه فراهم می‌کنند، می‌توانیم به طور بالقوه هم دقت و هم سربار محاسباتی روش‌های ML را بهبود بخشیم216. یک مجموعه‌داده که ابعاد کمتری دارد و به خوبی متعادل است، همچنین می‌تواند به طور قابل توجهی بیش‌برازش را محدود کند و تفسیر آسان‌تری از نتایج پیش‌بینی ارائه دهد [64]217.

«کشیدگی» (Kurtosis) را می‌توان بر روی هر ویژگی به عنوان نوعی روش انتخاب ویژگی بدون نظارت محاسبه کرد218. ... (توضیح فرمول‌های ۳ و ۴) ... 219. کشیدگی، معیار «عدم یکنواختی» (non-uniformity) داده‌ها است220. این معیار، شکل توزیع احتمال را توصیف می‌کند221. مقدار کشیدگی بالا معمولاً با درجه بالایی از انحراف/داده‌های پرت مطابقت دارد222. ... یک نقطه ضعف شناخته‌شده معیار کشیدگی این است که تعاملات بین ویژگی‌های مختلف را در نظر نمی‌گیرد223.

«میانگین تفاضل مطلق» (Mean Absolute difference - MAD) تفاضل مطلق از مقدار میانگین را محاسبه می‌کند224. امتیاز بالاتر نشان‌دهنده پتانسیل تمایز بیشتر است [65]225. (فرمول ۵) 226226226226.

**۳.۵.۱.**

مقادیر شِپلی (Shapely values) 227

روش‌های فوق‌الذکر راهی مفید برای شناسایی ویژگی‌هایی هستند که ممکن است منجر به یک مدل پیش‌بینی‌کننده با عملکرد بهتر شوند228. یک رویکرد جدید جالب، نه تنها برای انتخاب ویژگی‌هایی با درجه تأثیر بالا بر پیش‌بینی، بلکه برای توضیح اینکه چرا یک رویداد خاص به عنوان ناهنجار تلقی شده است (در مورد روش‌های بدون نظارت)، «مقادیر شپلی» است [66]229. اینها اولین بار در مطالعه نظریه بازی‌های ائتلافی معرفی شدند230. ... (توضیح فرمول‌های ۶ و ۷) ... 231.

در زمینه مدل‌های ML، $X$ بردار مقادیر ویژگی‌های نمونه‌ای است که باید توضیح داده شود232. مقادیر شپلی متقارن هستند به این معنا که مشارکت برابر منجر به مقادیر شپلی برابر می‌شود و ویژگی‌های بدون مشارکت، مقدار شپلی ۰ دارند233. ...

مسئله اصلی برای محاسبه مقادیر شپلی برای انتخاب ویژگی این است که آنها به یک مجموعه‌داده برچسب‌گذاری‌شده یا پیش‌بینی برای توضیح نیاز دارند234. در مورد ما، ما روش‌های بدون نظارتی داریم که می‌خواهیم ابتدا توضیحی داشته باشیم که چرا یک رویداد خاص به عنوان ناهنجار شناسایی شده است235. این مورد استفاده به خوبی با مقادیر شپلی مطابقت دارد236. با این حال، ما همچنین می‌توانیم اهمیت ویژگی را بر اساس رویدادهای ناهنجار شناسایی‌شده و مقادیر شپلی محاسبه‌شده، محاسبه کنیم237. این مرحله می‌تواند برای کاهش قابل توجه فضای ویژگی مجموعه‌داده استفاده شود، در حالی که همزمان پتانسیل افزایش نرخ تشخیص رویداد ناهنجار را دارد238.

**۴.**

مجموعه‌داده 239

در این مقاله، ما بر روی یک WCN به نام GuifiSants [13] تمرکز خواهیم کرد240. GuifiSants در سال ۲۰۰۹ آغاز به کار کرد و بخشی از Guifi.net است241. گره‌های GuifiSants شامل آنتن‌هایی هستند که با توزیع لینوکس Openwrt [68] فلش شده‌اند و پروتکل مسیریابی مِش BMX6 [69, 70] را اجرا می‌کنند242. GuifiSants یک WCN است که در محله‌ای از شهر بارسلون (اسپانیا) به نام Sants مستقر شده است243. در سال ۲۰۱۲، دانشگاه پلی‌تکنیک کاتالونیا (UPC) برای اهداف تحقیقاتی به GuifiSants پیوست244. در زمان نگارش این مقاله، حدود ۶۰ گره در GuifiSants وجود دارد245245. GuifiSants یک شبکه در حال کار است که ده‌ها کاربر این شبکه را به عنوان تنها دسترسی خود به اینترنت دارند246246246246. ...

مجموعه‌داده از نمونه‌های داده‌ای ساخته شده است که هر ۵ دقیقه از هر گره جمع‌آوری شده‌اند247. مجموعه‌داده به صورت عمومی در Zenodo [19] در دسترس است248. این کار با استفاده از یک اتصال ssh دائمی از یک سرور نظارت مرکزی به هر گره در مِش انجام می‌شود که برای اجرای دستورات استاندارد سیستم استفاده می‌شود249. سپس خروجی این دستورات برای به دست آوردن داده‌ها تجزیه (parse) می‌شود250. این روش این مزیت را دارد که هیچ تغییر یا نرم‌افزار اضافی نیاز به نصب در گره‌ها ندارد251. این یک شرط مهم است زیرا کاربران، مالک گره‌های خود هستند252.

داده‌ها با خواندن متغیرهای هسته لینوکس موجود از طریق فایل سیستم /proc به دست می‌آیند253. به عنوان مثال، /proc/net/dev برای خواندن شمارنده‌هایی با تعداد بایت‌ها و بسته‌های ارسالی و دریافتی بر روی هر رابط 254؛ /proc/stat که در آن اطلاعاتی در مورد فعالیت هسته وجود دارد 255؛ /proc/meminfo برای استفاده از حافظه و غیره256. متغیرهای هسته دو نوع هستند: (i) مقادیر مطلق، به عنوان مثال، میانگین بار ۱ دقیقه‌ای پردازنده، و (ii) شمارنده‌هایی که به طور یکنواخت افزایش می‌یابند، به عنوان مثال، تعداد بسته‌های ارسالی257. ما متغیرهای هسته از نوع شمارنده را با تقسیم تفاضل بین دو نمونه متوالی بر تفاضل مُهرهای زمانی (timestamps) مربوطه بر حسب ثانیه، به «نرخ» (rates) تبدیل کرده‌ایم258. ما نمونه‌های نرخ منفی را که هنگام راه‌اندازی مجدد یک گره یا زمانی که یک شمارنده به مقدار حداکثر خود می‌رسد و دوباره شروع می‌شود، رخ می‌دهند، حذف کرده‌ایم259.

مجموعه‌داده شامل ویژگی‌های ترافیکی و غیر ترافیکی است260. ویژگی‌های ترافیکی از شمارنده‌های موجود در فایل سیستم /proc/net/dev لینوکس به دست می‌آیند261. برای هر گره، ما شمارنده‌های بایت‌ها و بسته‌های دریافتی و ارسالی را بر روی رابط‌های اترنت و وای‌فای در نظر گرفته‌ایم262. ... (ویژگی‌های نرخ مانند eth.txb.rate-24) ... 263.

ویژگی‌های غیر ترافیکی شامل تعداد فرآیندها، میانگین بار پردازنده، softirq، iowait، زمان اجرای فرآیند در حالت هسته و کاربر، تعویض زمینه (context switches) و غیره است264. ... لیست کامل ویژگی‌ها با توضیحات مختصر را می‌توان در مخزن عمومی مجموعه‌داده [19] یافت265.

در مجموع ۶۳ گره در مجموعه‌داده جمع‌آوری شد، با مجموع ۲۳۸۷ ویژگی266. برای ارزیابی اثربخشی روش‌های ML در تشخیص ناهنجاری‌ها، به این صورت عمل کردیم267. در ۱۴ آوریل ۲۰۲۱، یکی از دو درگاه مِش (گره ۲۴) از کار افتاد و جایگزین شد268. به دلیل خرابی گره، نمونه‌های این گره بین ساعت ۰۱:۵۵ تا ۱۷:۴۰ روز ۱۴ آوریل موجود نبودند269. برای مجموعه آزمایشی، ما از نمونه‌های جمع‌آوری‌شده در یک بازه ۳ روزه که خرابی رخ داد استفاده کردیم: ۱۳، ۱۴ و ۱۵ آوریل (۶۹۴ نمونه)270. برای مجموعه آموزش، ما از نمونه‌های جمع‌آوری‌شده در طول ۴ هفته قبل از مجموعه آزمایشی استفاده کردیم (۷۲۳۷ نمونه)271. شکل ۲ ویژگی ترافیکی sum.xb.rate-24 گره ۲۴ را در طول مجموعه آموزش و آزمایشی نشان می‌دهد272272272272.

**۵.**

آزمایش‌ها و نتایج 273

در این بخش، ما آزمایش‌های خود را با استفاده از روش‌های AD بدون نظارت تشریح خواهیم کرد274. تمام آزمایش‌ها بر روی یک سرور IBM Power SC922 با ۱۶۰ پردازنده Power9 با فرکانس ۳.۷ گیگاهرتز، ۶۴۴ گیگابایت رم و ۴ پردازنده گرافیکی Nvidia V100 32 GB GDDR5 با NVLlink اجرا شدند275.

ما از ۴ روش ML ارائه‌شده در بخش ۳ استفاده کرده‌ایم276. این روش‌ها به این دلیل انتخاب شدند که رویکردهای کاملاً متفاوتی برای AD هستند277. آزمایش‌ها در پایتون با استفاده از کتابخانه‌های scikit-learn [73] برای PCA و pyod [74] برای Isolation Forest، CBLOF و VAE اجرا شدند278.

آزمایش‌ها از مجموعه‌ای از مراحل تثبیت‌شده پیروی می‌کنند279. ابتدا، داده‌های خام قالب‌بندی، پاک‌سازی و نرمال‌سازی می‌شوند280. سپس، ما چندین روش تشخیص مبتنی بر ML را اجرا می‌کنیم تا بهترین فراپارامترهای عملکردی را پیدا کنیم281. سپس روش‌های انتخاب ویژگی برای بهبود این نتایج اولیه اعمال می‌شوند282. در نهایت، ما مجموعه دوم آزمایش‌ها را با زیرمجموعه جدید داده‌های اصلی اجرا می‌کنیم و نتیجه نهایی را تحلیل می‌کنیم283. ...

در آزمایش‌های ما، ما آن ویژگی‌هایی را که واریانس پایینی در بازه زمانی انتخاب‌شده داشتند، هرس (prune) کردیم284. ... داده‌های آموزش شامل ۷۲۳۷ نمونه است که هر کدام پس از هرس کردن ۱۵۸۵ ویژگی دارند، در حالی که مجموعه آزمایشی ۶۹۴ نمونه با همان فضای ویژگی مجموعه آموزش است285.

نرمال‌سازی داده‌ها می‌تواند در صورت اعمال نادرست بر روی مجموعه‌داده‌های نامتعادل، عملکرد روش‌های ML را به طور جدی کاهش دهد286. ... ما تکنیک min-max را انتخاب کرده‌ایم که داده‌ها را در محدوده [0, 1] نرمال می‌کند287. ... برای PCA، ما دریافتیم که راحت‌تر است که نرمال‌سازی min-max را فقط بر روی ویژگی‌های غیر ترافیکی اعمال کنیم288. ویژگی‌های ترافیکی با تقسیم هر بردار ویژگی بر میانگین بیشینه هر مجموعه ویژگی ترافیکی نرمال‌سازی شدند289. PCA به محدوده‌های ویژگی نرمال‌شده بسیار حساس است290.

اکثر کتابخانه‌ها برای روش‌های تشخیص ناهنجاری و داده پرت بدون نظارت، نیاز به تعریف یک «فاکتور آلودگی» (contamination factor) مورد انتظار برای آموزش دارند291. از آنجایی که رویدادهای ناهنجار به ندرت رخ می‌دهند و ما می‌خواهیم تا حد امکان از مثبت کاذب جلوگیری کنیم، ما آلودگی را بر روی مقدار پایین $\alpha=$ 0.005 تنظیم کردیم292. ... برای مجموعه آموزش ما با ۷۲۳۷ نمونه، این فاکتور آلودگی ۳۷ ناهنجاری به دست می‌دهد293. توجه داشته باشید که در ۶۹۴ نمونه مجموعه آزمایشی، در شرایط عادی ما ۴ ناهنجاری را پیش‌بینی می‌کردیم294. با این حال، به دلیل خرابی درگاه که در طول بازه آزمایشی رخ می‌دهد، ما انتظار داریم که روش‌های مختلف ML تعداد بیشتری ناهنجاری را تشخیص دهند295.

برای هر یک از این الگوریتم‌ها، ما مجبور بودیم فراپارامترها را به صورت دستی تنظیم کنیم تا رویدادهای ناهنجار در داخل چارچوب زمانی که می‌دانیم رویدادهای ناهنجار رخ داده‌اند، گروه‌بندی شوند296. جدول ۱ فراپارامترهای مورد استفاده برای هر روش تشخیص ML را نشان می‌دهد297297297297. این مقادیر با استفاده از «جستجوی تصادفی» (Random Search) با یک فضای جستجوی فراپارامتر تعریف‌شده به صورت دستی، تنظیم دقیق شدند298. در مورد PCA، تنها پارامتر، تعداد مؤلفه‌های اصلی (PC) است299. برای ارزیابی تعداد PC، ما از روش رایج انتخاب تعداد ابعادی که ۹۵٪ واریانس باقیمانده را حفظ می‌کنند، استفاده کردیم300. ... پارامترهای VAE شامل نرخ حذف (dropout rate) هستند که می‌تواند به تعمیم مدل کمک کند و از بیش‌برازش جلوگیری کند301. ... ما اندازه دسته (batch size) را ۳۲ تنظیم کردیم302. ... هدف نهایی ما این است که ببینیم روش‌های ML فوق‌الذکر در دستگاه‌هایی که ممکن است در یک سناریوی لبه (Edge)/مه (Fog) قرار داشته باشند، چگونه رفتار خواهند کرد303.

... مسلماً تنظیم جنگل ایزوله‌سازی (IF) دشوارترین بود304. همانطور که در بخش ۳.۲ توضیح داده شد، IF مراحل زیادی دارد که به میزان قابل توجهی تحت تأثیر تصادفی بودن هستند305. این در بسیاری از موارد مفید است، اما می‌تواند نتایج متناقضی به همراه داشته باشد306. ...

نتایج همه آزمایش‌ها در جدول ۲ آمده است307307307. برای هر الگوریتم، ما زمان‌های آموزش و استنتاج (inference) و همچنین تعداد ناهنجاری‌های شناسایی‌شده در مجموعه آزمایشی، و اینکه چه تعداد از آنها در داخل بازه خرابی درگاه قرار می‌گیرند (۱۲۲ نمونه در طول بازه خرابی جمع‌آوری شد) را فهرست می‌کنیم308. از آنجایی که خرابی درگاه به وضوح یک وضعیت ناهنجار است، تعداد نمونه‌های شناسایی‌شده به عنوان ناهنجار توسط الگوریتم‌ها که متعلق به بازه خرابی درگاه هستند، می‌تواند به عنوان یک معیار عملکرد استفاده شود309. برای هر الگوریتم، جدول ۲ ناهنجاری‌های شناسایی‌شده با استفاده از همه ویژگی‌ها و فقط ویژگی‌های ترافیکی را نشان می‌دهد310. با مقایسه ناهنجاری‌های شناسایی‌شده در هر دو حالت، ما می‌بینیم که در نظر گرفتن همه ویژگی‌ها به طور قابل توجهی عملکرد الگوریتم‌ها را افزایش می‌دهد311. به عنوان مثال، با VAE با همه ویژگی‌ها، ۱۳۴ ناهنجاری تشخیص داده شد که از این تعداد، تمام ۱۲۲ نمونه در طول بازه خرابی درگاه جمع‌آوری شده بودند312. با استفاده از ویژگی‌های ترافیکی به تنهایی، تنها ۲۳ ناهنجاری تشخیص داده شد که از این تعداد، تنها ۱۲ مورد در بازه خرابی درگاه بودند313.

گام بعدی در آزمایش‌های ما، آزمودن روش‌های ML با تغییر اندازه مجموعه‌داده بود314. برای این منظور، ما ابتدا تعداد نمونه‌های مورد استفاده برای مجموعه آموزش را تغییر دادیم315. نتایج در شکل ۳ آورده شده است316. ... شکل ۳ عملکرد کاملاً متفاوتی را در روش‌های مختلف ML نشان می‌دهد317. در حالی که PCA، CBLOF و VAE هنگام در نظر گرفتن نمونه‌های تعداد کمی از روزها، تعداد زیادی ناهنجاری و در نتیجه مثبت کاذب تولید می‌کنند، جنگل ایزوله‌سازی تنها تعداد کمی ناهنجاری را تشخیص می‌دهد و در نتیجه تعداد زیادی منفی کاذب (false negatives) دارد318. از سوی دیگر، هنگام استفاده از نمونه‌های بیش از ۱۳ روز، ناهنجاری‌های شناسایی‌شده توسط VAE تقریباً ثابت باقی می‌مانند و به درستی تمام نمونه‌های داخل بازه خرابی درگاه را به عنوان ناهنجار تشخیص می‌دهند319. ... ما می‌توانیم نتیجه بگیریم که در آزمایش‌های ما، VAE قادر است به طور قابل اعتمادی ناهنجاری‌ها را با تعداد کمتری نمونه تشخیص دهد320.

در مرحله بعد، ما روش‌های ML را با تغییر تعداد ویژگی‌ها بررسی کردیم321. ما از یک تا همه ویژگی‌ها را، که بر اساس اهمیت مرتب شده بودند، انتخاب کردیم و تعداد ناهنجاری‌ها را محاسبه کردیم322. برای محاسبه اهمیت ویژگی، ما از نمودار مشارکت آماره Q PCA (بخش ۳.۱) و روش‌های انتخاب دقیق‌شده در بخش ۳.۵ استفاده کردیم323. ... جنگل ایزوله‌سازی برای محاسبه مقادیر شپلی پیش‌بینی‌های مجموعه آزمایشی انتخاب شد324. ...

شکل ۴ تعداد ناهنجاری‌های شناسایی‌شده در مجموعه آزمایشی ... را با تغییر تعداد ویژگی‌های مورد استفاده برای آموزش نشان می‌دهد325. ... این شکل نشان می‌دهد که روش‌های ML مورد مطالعه بسته به تعداد ویژگی‌های مورد استفاده، رفتار بسیار متفاوتی دارند326. جنگل ایزوله‌سازی حساس‌ترین است327. ... از سوی دیگر، VAE پایدارترین نشان داده شده است328. ... در مورد روش‌های انتخاب ویژگی، شکل ۴ نشان می‌دهد که «کشیدگی» (kurtosis) بدترین عملکرد را ارائه می‌دهد329. برای روش‌های دیگر، شکل ۴ نشان می‌دهد که هیچ برنده مشخصی وجود ندارد330.

... شکل ۵ نشان می‌دهد که ناهنجاری‌های همپوشان شناسایی‌شده در میان ۴ روش ML در مجموعه آموزش چگونه است331. ... با این حال، اگر مجموعه آزمایشی را در نظر بگیریم، توافق بالایی در میان همه روش‌های ML وجود دارد، همانطور که در شکل 5.(b) نشان داده شده است 332. البته، این به دلیل بازه خرابی درگاه است، جایی که بیشتر ناهنجاری‌ها یافت می‌شوند333. به عنوان مثال، شکل 5.(c) نشان می‌دهد که PCA و VAE تمام ۱۲۲ نقطه در بازه خرابی درگاه را به عنوان ناهنجار تشخیص می‌دهند334.

برای به دست آوردن بینشی در مورد روش‌های ML مورد مقایسه، شکل ۶ یک تصویر ۲ بعدی PCA از مجموعه آموزش (مؤلفه‌های اصلی PC1، PC2) را برای جنگل ایزوله‌سازی، CBLOF و VAE نشان می‌دهد335. ... ما به راحتی می‌توانیم تفاوت‌های بین نحوه «تصمیم‌گیری» هر روش ML در مورد اینکه چه چیزی یک رویداد ناهنجار است و چه چیزی نیست را ببینیم336. مرز تصمیم‌گیری محاسبه‌شده برای جنگل ایزوله‌سازی از اهمیت ویژه‌ای برخوردار است337. ... شکل ۶ به وضوح نشان می‌دهد که بیش‌برازش می‌تواند در مورد جنگل ایزوله‌سازی یک مشکل باشد338.

برای تفسیر ناهنجاری‌ها، شکل ۷ یک نقشه حرارتی (heatmap) SHAP از ناهنجاری‌های به‌دست‌آمده با استفاده از مجموعه‌های آموزش و آزمایشی را به ترتیب در زیرشکل‌های (a) و (b) نشان می‌دهد339. ... در شکل ۷ می‌توان دید که شدت رنگ‌های ناهنجاری‌های به‌دست‌آمده برای مجموعه آموزش بسیار روشن‌تر و پراکنده‌تر هستند، در حالی که برای مجموعه آزمایشی شدیدتر و مکررتر هستند340. این به وضوح نشان می‌دهد که گروهی از ویژگی‌ها وجود دارند که به طور قابل توجهی تحت تأثیر خرابی درگاه در طول مجموعه آزمایشی قرار می‌گیرند و بنابراین، مقدار شپلی بالایی برای بیشتر ناهنجاری‌های شناسایی‌شده در مجموعه آزمایشی به آنها اختصاص داده می‌شود341. همچنین جالب است که در شکل ۷ توجه داشته باشید که هیچ یک از ویژگی‌هایی که بیشترین مشارکت را در ناهنجاری‌های یافت‌شده با استفاده از مجموعه آموزش (به‌دست‌آمده با تنظیم فاکتور آلودگی) دارند، با آن ویژگی‌هایی که بیشترین مشارکت را با استفاده از مجموعه آزمایشی (ناشی از خرابی درگاه) دارند، مطابقت ندارند 342. این واقعیت نشان می‌دهد، همانطور که انتظار می‌رود، ناهنجاری‌های شناسایی‌شده در هر دو مجموعه از ماهیت متفاوتی برخوردارند343.

شکل ۸ یک سری زمانی از دو ویژگی را نشان می‌دهد که بیشترین مشارکت را در AD در مجموعه آزمایشی دارند (همانطور که در شکل 7.(b) نشان داده شده است)344. اینها نرخ تعویض زمینه پردازنده گره ۴۲ (ctxt.rate-42) و نرخ بایت‌های ارسالی گره ۷ (txb.rate-7) هستند345. توجه داشته باشید که ctxt.rate-42 در طول عملیات عادی مقدار میانگین حدود ۲۰۰ و در طول خرابی درگاه ۰ دارد346. این به این دلیل است که در طول خرابی درگاه، گره ۴۲ از شبکه قطع شده بود و هیچ نمونه‌ای جمع‌آوری نشد347. این نمونه‌های گمشده روی ۰ تنظیم شدند348. از سوی دیگر، شکل ۸ نشان می‌دهد که txb.rate-7 در طول خرابی درگاه مقدار بالاتر غیرمعمولی دارد349. این به این دلیل است که گره ۷ نزدیک به درگاهی است که از کار افتاده بود و در طول خرابی، بیشتر ترافیکی را که به درگاه دیگر در مِش هدایت می‌شد، جذب کرد350.

**۵.۱.**

بحث 351

نتایج آزمایش ما جالب بوده است، ما دریافتیم که هر ۴ روش تشخیص ML انتخاب‌شده بر روی مجموعه‌داده ارائه‌شده به خوبی عمل می‌کنند352. متأسفانه، در مورد روش‌های انتخاب ویژگی، اینطور نیست353. روشی که در بیشتر موارد بهترین عملکرد را دارد، مبتنی بر محاسبه مقادیر شپلی است که از نظر محاسباتی گران است354. ... VAE با فضای ویژگی کامل بهترین عملکرد را داشت در حالی که جنگل ایزوله‌سازی پس از انتخاب ویژگی مبتنی بر مقدار شپلی، بیشترین بهبود را داشت355. به طور کلی، عملکرد CBLOF با زمان‌های آموزش و استنتاج نسبتاً سریع، سازگارترین بود356. روش انتخاب MAD نیز کاهش قابل توجهی در فضای ویژگی، به‌ویژه برای روش‌های جنگل ایزوله‌سازی و VAE، فراهم کرد357.

یک مسئله که در طول آزمایش‌های ما مطرح شد این است که ما نمی‌دانیم چند نوع رویداد ناهنجار در مجموعه‌داده وجود دارد358. هر ۴ روش، نمونه‌ها را به عنوان ناهنجار یا عادی برچسب‌گذاری می‌کنند اما تمایزی در مورد انواع ناهنجاری قائل نمی‌شوند359. ... این اطلاعات اگرچه مفید هستند، اما هنوز به این سؤال پاسخ نمی‌دهند: چند نوع ناهنجاری وجود دارد و کاهش فضای ویژگی چگونه بر پیش‌بینی تأثیر می‌گذارد؟360. بسیار محتمل است که با کاهش فضای ویژگی، ما به طور مؤثری ظرفیت روش‌های تشخیص خود را از تشخیص هر نوع ناهنجاری دیگری حذف کنیم361.

برای پاسخ به این سؤال، ما تصمیم گرفتیم ناهنجاری‌های شناسایی‌شده را با استفاده از روشی خوشه‌بندی کنیم که هر نقطه داده (رویداد) را مجبور به تعلق به یک خوشه نکند و همچنین مفهوم «نویز» (noise) را داشته باشد362. برای این کار، ما الگوریتم «خوشه‌بندی فضایی مبتنی بر چگالی سلسله‌مراتبی کاربردها با نویز» (HDDBSCAN) [77] را انتخاب کردیم که خود نیز روشی است که برای AD استفاده می‌شود363. ... ما حداقل اندازه خوشه را ۳۰ تنظیم کردیم364. برای سازگاری، ما از ناهنجاری‌های شناسایی‌شده توسط جنگل ایزوله‌سازی استفاده کردیم365. شکل ۱۱ نشان می‌دهد که HDDBSCAN یک خوشه حاوی ۹۷ رویداد ناهنجار و ۲۰ رویداد ناهنجار شناسایی‌شده به عنوان نویز را شناسایی کرد366. از این ناهنجاری‌ها، ۱۰۷ مورد مربوط به بازه خرابی درگاه بودند که ۱۴ مورد از آنها به عنوان نویز و ۹۳ مورد در خوشه طبقه‌بندی شدند367. تفسیر ما از این نتایج این است که بیشتر رویدادهای ناهنجاری که ما شناسایی کرده‌ایم از یک نوع هستند که مربوط به خرابی درگاه است368.

وقتی نوبت به پیچیدگی روش‌های ML انتخاب‌شده می‌رسد، ... IF الگوریتمی است که به‌ویژه برای پردازش حجم بالایی از داده‌ها با پیچیدگی زمانی خطی ... مناسب است369. به طور مشابه، CBLOF نیز برای حجم‌های زیاد داده $O(N \log N)$ در بهترین حالت مناسب است [55]370. در نهایت، VAE به دلیل پیچیدگی محاسباتی بالای خود بدنام است که عمدتاً بر اساس توپولوژی شبکه مورد استفاده در طول آموزش است371. با این حال، در آزمایش‌های ما این پیچیدگی اضافی منجر به نتایج جالبی شد372. VAE توانست در حالی که توپولوژی حداقلی داشت، از سایر روش‌ها بهتر عمل کند373. ...

ما همچنین باید اشاره کنیم که آزمایش‌های ما شامل چندین روش ML اضافی بود374. ... به عنوان مثال، ما آزمایش‌هایی را با استفاده از روش‌های مبتنی بر GAN مانند ALAD [79] و AnoGAN [80] انجام دادیم375. ... متأسفانه نتایج تجربی در هر دو مورد ضعیف بود376. ... به طور مشابه، ما چندین آزمایش با استفاده از $\beta-VAE$ ... اجرا کردیم377. با این حال، بهترین نتایج با استفاده از مقدار پیش‌فرض $\beta$ برابر با 1.0 ... به دست آمد378. ... ما همچنین از پیاده‌سازی Deep-SVDD [82] در آزمایش‌های خود استفاده کردیم که نتایج آن به طور قابل توجهی بدتر از الگوریتم‌های ارائه‌شده در بخش ۴ بود (باید بخش ۳ باشد)379. ...

ما نتیجه می‌گیریم که عدم عملکرد پیش‌بینی‌کننده برخی از روش‌های مورد استفاده در آزمایش‌های ما به دلیل فضای ویژگی بزرگ مجموعه‌داده ما است380. ... کار آینده بر روی پرداختن به این مسئله و شامل فرمول‌بندی مجدد مجموعه‌داده‌های آموزش و آزمایشی شامل گراف‌های جداول مسیریابی مورد استفاده در شبکه مِش و همچنین گنجاندن روش‌های AD اضافی تمرکز خواهد کرد381.

**۶.**

نتیجه‌گیری 382

در دهه گذشته، روش‌های یادگیری ماشین (ML) منفجر شده‌اند و در تعداد فزاینده‌ای از زمینه‌ها به کار گرفته شده‌اند383. شبکه‌های کامپیوتری نیز از این قاعده مستثنی نیستند و کارهای متعددی از ML برای بررسی تشخیص نفوذ به شبکه استفاده کرده‌اند384. تشخیص خطا در شبکه‌های کامپیوتری یک کاربرد جذاب برای ML است که توجه کمی به آن شده است، عمدتاً به دلیل کمبود مجموعه‌داده‌های موجود385. در این مقاله، ما تلاش می‌کنیم با انجام تشخیص ناهنجاری برای تشخیص خطا با استفاده از ML، این شکاف را پر کنیم386. MLهای انتخاب‌شده برای مطالعه ما بر اساس ۴ اصل بسیار متفاوت هستند387. اولین مورد، تحلیل مؤلفه‌های اصلی (PCA) است388. PCA یک روش شناخته‌شده است که به طور گسترده در صنعت تولید استفاده شده است389. ... دومین مورد، جنگل ایزوله‌سازی (IF) است390. IF بر اساس ایده ساخت درختان تصمیم‌گیری است که نمونه‌ها را به کلاس‌های مختلف جدا می‌کنند391. ... سومین مورد، فاکتور ناهنجاری محلی مبتنی بر خوشه‌بندی (CBLOF) است که از اصل خوشه‌بندی استفاده می‌کند392. ... در نهایت، ما از خودرمزگذارهای متغیر (VAE) استفاده کرده‌ایم که مبتنی بر یک مدل شبکه عصبی عمیق است393.

برخلاف سایر کارهای یافت‌شده در ادبیات موضوع، ما یک مجموعه‌داده از یک شبکه در حال کار ساخته‌ایم394. برای مجموعه‌داده خود، ما از بازه‌ای استفاده کردیم که در آن یک خرابی ناخواسته درگاه رخ داده بود395. این رویداد نادر به ما این امکان را داد که عملکرد روش‌های ML مورد مطالعه را با بررسی اینکه چه تعداد از نمونه‌های جمع‌آوری‌شده در طول بازه خرابی درگاه به عنوان ناهنجار پرچم‌گذاری می‌شوند، تحلیل کنیم396. مجموعه‌داده با جمع‌آوری نمونه‌ها هر ۵ دقیقه ساخته شد397. نمونه‌های داده نه تنها دارای ویژگی‌های ترافیکی، بلکه ویژگی‌های غیر ترافیکی مانند استفاده از پردازنده، استفاده از حافظه و غیره نیز هستند398. ما از یک دوره ۴ هفته‌ای برای مجموعه آموزش و یک دوره ۳ روزه برای مجموعه آزمایشی استفاده کردیم399. خرابی درگاه به مدت ۱۶ ساعت در اواسط مجموعه آزمایشی رخ می‌دهد400.

**نتایج اصلی را می‌توان به شرح زیر خلاصه کرد:** 401

* هنگامی که به درستی تنظیم شوند، ناهنجاری‌های ناشی از خرابی درگاه به خوبی توسط همه روش‌های ML ثبت می‌شوند402.
* خارج از بازه خرابی، چندین جهش ترافیکی نیز به عنوان ناهنجاری علامت‌گذاری می‌شوند403. این جهش‌ها، با این حال، می‌توانند به عنوان نویز و نتیجه الگوهای ترافیکی نامنظم کاربران در نظر گرفته شوند404.
* روش یادگیری عمیق مبتنی بر VAE بهتر از سایر روش‌های ML عمل می‌کند، اما به قیمت هزینه محاسباتی قابل توجهی بالاتر405.
* **در نظر گرفتن سایر ویژگی‌های مرتبط با پردازنده و حافظه علاوه بر ویژگی‌های ترافیکی، به طور قابل توجهی تعداد نقاط شناسایی‌شده به عنوان ناهنجار را در داخل بازه خرابی درگاه افزایش می‌دهد**406.
* از میان روش‌های انتخاب ویژگی که ما آزمایش کردیم، «میانگین تفاضل مطلق» (MAD) و «مقادیر شپلی» (SHAP) بهترین عملکرد را داشتند407. محاسبه MAD بسیار آسان‌تر است، در حالی که SHAP درک بهتری از اینکه چرا رویدادها به عنوان ناهنجار پرچم‌گذاری می‌شوند (توضیح‌پذیری) ارائه می‌دهد408.

منابع 409

[1] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM Comput. Surv. 41 (3) (2009) 1–58. 410

[2] C.C. Aggarwal, Outlier Analysis, Springer, 2017. 411

[3] M. Ahmed, A.N. Mahmood, J. Hu, A survey of network anomaly detection techniques, J. Netw. Comput. Appl. (60) cite\_start 19–31. 412

[4] D.P. Kumar, T. Amgoth, C.S.R. Annavarapu, Machine learning algorithms for wireless sensor networks: A survey, Inf. Fusion 49 (2019) 1–25. 413[5] M. Mohri, A. Rostamizadeh, A. Talwalkar, Foundations of Machine Learning, MIT Press, 2018. 414

[6] K.P. Murphy, Probabilistic Machine Learning: An Introduction, MIT Press, 2022. 415

[7] L.N. Tidjon, M. Frappier, A. Mammar, Intrusion detection systems: A cross-domain overview, IEEE Commun. Surv. Tutor. 21 (4) (2019) 3639–3681. 416

[8] G. Fernandes, J.J. Rodrigues, L.F. Carvalho, J.F. Al-Muhtadi, M.L. Proença, A comprehensive survey on network anomaly detection, Telecommun. Syst. 70 (3) (2019) 447–489. 417

[9] D. Vega, R. Baig, L. Cerdà-Alabern, E. Medina, R. Meseguer, L. Navarro, A technological overview of the guifi.net community network, Comput. Netw. 9 (2) (2015) 260–278, http://dx.doi.org/10.1016/j.comnet.2015.09.023. 418[10] Y. Ben David, Connecting the Last Billion (Ph.D. thesis), UC Berkeley, 2015. 419

[11] Guifi.net, Open, free and neutral network internet for everybody, 2021, http: //guifi.net/en. (Accessed 13 January 2021). 420

[12] L. Cerdà-Alabern, R. Baig, L. Navarro, On the guifi.net community network economics, Comput. Netw. 168 (2020) 107067. 421

[13] GuifiSants, Xarxa oberta, lliure i neutral del barri de sants, 2021, http://sants. guifi.net/. (Accessed January 2021). 422

[14] J. Camacho, A. Pérez-Villegas, P. Garcí a Teodoro, G. Maciá-Fernández, PCA based multivariate statistical network monitoring for anomaly detection, Comput. Secur. 59 (https://www.google.com/search?q=2016) 118–137. 423

[15] D.H. Hoang, H.D. Nguyen, A PCA-based method for IoT network traffic anomaly detection, in: 2018 20th International Conference on Advanced Communication Technology, ICACT, IEEE, 2018, pp. 381–386. 424

[16] I.K. Savvas, A.V. Chernov, M.A. Butakova, C. Chaikalis, Increasing the quality and performance of N-dimensional point anomaly detection in traffic using PCA and DBSCAN, in: 2018 26th Telecommunications Forum, TELFOR, IEEE, 2018, pp. 1–4. 425[17] Munin networked resource monitoring tool, http://munin-monitoring.org. 426[18] Nagios, The Industry Standard In IT Infrastructure Monitoring, https://www. nagios.org. 427

[19] L. Cerdà-Alabern, Dataset for Anomaly Detection in a Production Wireless Mesh Community Network, Zenodo, 2022, http://dx.doi.org/10.5281/zenodo. 6169917. 428

[20] V. Hodge, J. Austin, A survey of outlier detection methodologies, Artif. Intell. Rev. 22 (2) (2004) 85–126. 429[21] S. Northcutt, J. Novak, Network Intrusion Detection, Sams Publishing, 2002. 430

[22] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E.D. Kolaczyk, N. Taft, Structural analysis of network traffic flows, in: Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems, 2004 pp. 61–72. 431

[23] Z.R. Zaidi, S. Hakami, T. Moors, B. Landfeldt, Detection and identification of anomalies in wireless mesh networks using principal component analysis (PCA), J. Interconnect. Netw. 10 (04) (2009) 517–534. 432

[24] Z.R. Zaidi, S. Hakami, B. Landfeldt, T. Moors, Real-time detection of traffic anomalies in wireless mesh networks, Wirel. Netw. 16 (6) (2010) 1675–1689. 433[25] C. Pascoal, M.R. De Oliveira, R. Valadas, P. Filzmoser, P. Salvador, A. Pacheco, Robust feature selection and robust PCA for internet traffic anomaly detection, in: 2012 Proceedings IEEE Infocom, IEEE, 2012, pp. 1755–1763. 434[26] H. Ringberg, A. Soule, J. Rexford, C. Diot, Sensitivity of PCA for traffic anomaly detection, in: Proceedings of the 2007 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, 2007 pp. 109–120. 435

[27] N.L.D. Khoa, T. Babaie, S. Chawla, Z. Zaidi, Network anomaly detection using a commute distance based approach, in: 2010 IEEE International Conference on Data Mining Workshops, IEEE, 2010, pp. 943–950. 436

[28] A.B. Nassif, M.A. Talib, Q. Nasir, F.M. Dakalbab, Machine learning for anomaly detection: A systematic review, IEEE Access 9 (2021) 78658–78700, http://dx. doi.org/10.1109/ACCESS.2021.3083060. 437[29] S. Goldstein, A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data, PLoS One 11 (4) (https://www.google.com/search?q=2016) 1–31, http://dx.doi. org/10.1371/journal.pone.0152173. 438

[30] M. Ahmed, A.N. Mahmood, Novel approach for network traffic pattern analysis using clustering-based collective anomaly detection, Ann. Data Sci. 2 (2015) 111–130. 439[31] X. Chun-Hui, S. Chen, B. Cong-Xiao, L. Xing, Anomaly detection in network management system based on isolation forest, in: 2018 4th Annual International Conference on Network and Information Systems for Computers, ICNISC, 2018, pp. 56–60, http://dx.doi.org/10.1109/ICNISC.2018.00019. 440

[32] G. Pang, C. Shen, L. Cao, A.V.D. Hengel, Deep learning for anomaly detection: A review, ACM Comput. Surv. 54 (2) (2021) http://dx.doi.org/10.1145/3439950. 441[33] N. Takeishi, Y. Kawahara, On anomaly interpretation via Shapley values, 2020, arXiv:2004.04464. 442

[34] L. Antwarg, R.M. Miller, B. Shapira, L. Rokach, Explaining anomalies detected by autoencoders using Shapley additive explanations, Expert Syst. Appl. 186 (2021) 115736, http://dx.doi.org/10.1016/j.eswa.2021.115736. 443

[35] C. Zhou, R.C. Paffenroth, Anomaly detection with robust deep autoencoders, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 665–674, http://dx.doi.org/10.1145/3097983. 3098052. 444

[36] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, 2014, arXiv:1312. 6114. 445

[37] M. Moulay, R.G. Leiva, V. Mancuso, P.J.R. Maroni, A.F. Anta, Ttrees: Automated classification of causes of network anomalies with little data, in: 2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM, IEEE, 2021, pp. 199–208. 446[38] K. Sequeira, M. Zaki, ADMIT: Anomaly-based data mining for intrusions, in: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’02, Association for Computing Machinery, New York, NY, USA, 2002, pp. 386–395, http://dx.doi.org/10.1145/775047.775103. 447

[39] Y.F. Zhang, Z.Y. Xiong, X.Q. Wang, Distributed intrusion detection based on clustering, in: 2005 International Conference on Machine Learning and Cybernetics, vol. 4, 2005, pp. 2379–2383 Vol. 4, http://dx.doi.org/10.1109/ICMLC. 2005.1527342. 448

[40] M.H. Bhuyan, D.K. Bhattacharyya, J.K. Kalita, An effective unsupervised network anomaly detection method, in: Proceedings of the International Conference on Advances in Computing, Communications and Informatics, ICACCI ’12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 533–539, http://dx.doi.org/10.1145/2345396.2345484. 449

[41] N. Hu, Z. Tian, H. Lu, X. Du, M. Guizani, A multiple-kernel clustering based intrusion detection scheme for 5G and IoT networks, Int. J. Mach. Learn. Cybern. 12 (2021) http://dx.doi.org/10.1007/s13042-020-01253-w. 450

[42] M.E. Otey, A. Ghoting, S. Parthasarathy, Fast distributed outlier detection in mixed-attribute data sets, Data Min. Knowl. Discov. 12 (2–3) (2006) 203–228, http://dx.doi.org/10.1007/s10618-005-0014-6. 451

[43] M. Bhuyan, D.K. Bhattacharyya, J. Kalita, A multi-step outlier-based anomaly detection approach to network-wide traffic, Inform. Sci. 348 (https://www.google.com/search?q=2016) http://dx. doi.org/10.1016/j.ins.https://www.google.com/search?q=2016.02.023. 452

[44] P. Casas, J. Mazel, P. Owezarski, Unsupervised network intrusion detection systems: Detecting the unknown without knowledge, Comput. Commun. 35 (7) (2012) 772–783, http://dx.doi.org/10.1016/j.comcom.2012.01.016. 453

[45] O. Iraqi, H. El Bakkali, Application-level unsupervised outlier-based intrusion detection and prevention, Secur. Commun. Netw. 2019 (2019) 1–13, http://dx. doi.org/10.1155/2019/8368473. 454

[46] M. Khan, Rule based network intrusion detection using genetic algorithm, Int. J. Comput. Appl. 18 (2011) 26–29, http://dx.doi.org/10.5120/2303-2914. 455

[47] H. Alsaadi, R. Almuttairi, O. Ucan, O. Bayat, An adapting soft computing model for intrusion detection system, Comput. Intell. 01 (2021) http://dx.doi.org/10. 1111/coin.12433. 456[48] A. Shenfield, D. Day, A. Ayesh, Intelligent intrusion detection systems using artificial neural networks, ICT Express 4 (2) (2018) 95–99, http://dx.doi.org/ 10.1016/j.icte.2018.04.003, SI on Artificial Intelligence and Machine Learning. 457

[49] C.F. Alcala, S.J. Qin, Analysis and generalization of fault diagnosis methods for process monitoring, J. Process Control 21 (3) (2011) 322–330. 458

[50] P. Miller, R.E. Swanson, C.E. Heckler, Contribution plots: A missing link in multivariate quality control, Appl. Math. Comput. Sci. 8 (4) (1998) 775–792. 459

[51] C.C. Aggarwal, S. Sathe, Theoretical foundations and algorithms for outlier ensembles, SIGKDD Explor. Newsl. 17 (1) (2015) 24–47, http://dx.doi.org/10. 1145/2830544.2830549. 460

[52] C.C. Aggarwal, Outlier ensembles: Position paper, SIGKDD Explor. Newsl. 14 (2) (2013) 49–58, http://dx.doi.org/10.1145/2481244.2481252. 461

[53] F.T. Liu, K.M. Ting, Z.H. Zhou, Isolation-based anomaly detection, ACM Trans. Knowl. Discov. Data 6 (1) (2012) http://dx.doi.org/10.1145/2133360.2133363. 462

[54] F.T. Liu, K.M. Ting, Z.H. Zhou, Isolation forest, in: 2008 Eighth IEEE International Conference on Data Mining, 2008, pp. 413–422, http://dx.doi.org/10.1109/ ICDM.2008.17. 463

[55] Z. He, X. Xu, S. Deng, Discovering cluster-based local outliers, Pattern Recognit. Lett. 24 (9) (2003) 1641–1650, http://dx.doi.org/10.1016/S0167-8655(03 00003-5. 464

[56] M.M. Breunig, H.P. Kriegel, R.T. Ng, J. Sander, LOF: Identifying density-based local outliers, SIGMOD Rec. 29 (2) (2000) 93–104, http://dx.doi.org/10.1145/ 335191.335388. 465[57] R. Zheng, J. Gu, Anomaly detection for power system forecasting under data corruption based on variational auto-encoder, in: 8th Renewable Power Generation Conference, RPG 2019, 2019, pp. 1–6, http://dx.doi.org/10.1049/cp.2019.0461. 466[58] R. Yao, C. Liu, L. Zhang, P. Peng, Unsupervised anomaly detection using variational auto-encoder based feature extraction, in: 2019 IEEE International Conference on Prognostics and Health Management, ICPHM, 2019, pp. 1–7, http://dx.doi.org/10.1109/ICPHM.2019.8819434. 467[59] Y. Aizenbud, O. Lindenbaum, Y. Kluger, Probabilistic robust autoencoders for anomaly detection, 2021, arXiv:2110.00494. 468

[60] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (56) (2014) 1929–1958, http://jmlr.org/papers/v15/srivastava14a.html. 469

[61] C.P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, A. Lerchner, Understanding disentangling in β-VAE, CoRR (2018) arXiv:1804.03599. 470470470[62] L. Zhou, W. Deng, X. Wu, Unsupervised anomaly localization using VAE and beta-VAE, 2020, http://dx.doi.org/10.48550/ARXIV.2005.10686. 471[63] J. Cai, J. Luo, S. Wang, S. Yang, Feature selection in machine learning: A new perspective, Neurocomputing 300 (2018) 70–79, http://dx.doi.org/10.1016/ j.neucom.2017.11.077. 472[64] C. Suman, S. Tripathy, S. Saha, Building an effective intrusion detection system using unsupervised feature selection in multi-objective optimization framework, 2019, arXiv preprint arXiv:1905.06562. 473

[65] A.J. Ferreira, M.A.T. Figueiredo, Efficient feature selection filters for high dimensional data, Pattern Recognit. Lett. 33 (13) (2012) 1794–1804, http: //dx.doi.org/10.1016/j.patrec.2012.05.019. 474

[66] L.S. Shapley, 17. A value for n-person games, in: H.W. Kuhn, A.W. Tucker (Eds.), Contributions to the Theory of Games (AM-28), vol. II, Princeton University Press, https://www.google.com/search?q=2016, pp. 307–318, http://dx.doi.org/10.1515/9781400881970-018. 475[67] C. Molnar, Interpretable Machine Learning, Independently published, 2022. 476

[68] OpenWrt Project, OpenWrt project: Welcome to the OpenWrt project, 2021, https://openwrt.org/. (Accessed January 2021). 477

[69] BMX6 mesh networking protocol, http://bmx6.net. (Accessed January 2021). 478[70] L. Cerdà-Alabern, A. Neumann, L. Maccari, Experimental evaluation of BMX6 routing metrics in a 802.11an wireless-community mesh network, in: 2015 3rd International Conference on Future Internet of Things and Cloud, 2015 pp. 770–775. 479

[71] GuifiSants, qMp Sants-UPC, 2021, http://dsg.ac.upc.edu/qmpsu. (Accessed January 2021). 480[72] L. Cerdà-Alabern, A. Neumann, P. Escrich, Experimental evaluation of a wireless community mesh network, in: The 16th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM’13, ACM, Barcelona, Spain, 2013. 481

[73] F. Pedregosa, et al., Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830. 482

[74] Y. Zhao, Z. Nasrullah, Z. Li, Pyod: A Python toolbox for scalable outlier detection, J. Mach. Learn. Res. 20 (96) (2019) 1–7, http://jmlr.org/papers/v20/19-011. html. 483[75] A. Mahfouz, A. Abuhussein, D. Venugopal, S. Shiva, Ensemble classifiers for network intrusion detection using a novel network attack dataset, Future Internet 12 (11) (2020) http://dx.doi.org/10.3390/fi12110180. 484

[76] S.M. Lundberg, S.I. Lee, A unified approach to interpreting model predictions, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 4765–4774. 485

[77] L. McInnes, J. Healy, S. Astels, Hdbscan: Hierarchical density based clustering, J. Open Source Softw. 2 (11) (2017) 205, http://dx.doi.org/10.21105/joss.00205. 486

[78] M. Ester, H.P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD ’96, AAAI Press, 1996, pp. 226–231. 487

[79] H. Zenati, M. Romain, C.S. Foo, B. Lecouat, V.R. Chandrasekhar, Adversarially learned anomaly detection, 2018, http://dx.doi.org/10.48550/ARXIV.1812. 02288. 488

[80] T. Schlegl, P. Seeböck, S.M. Waldstein, U. Schmidt-Erfurth, G. Langs, Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, 2017, http://dx.doi.org/10.48550/ARXIV.1703.05921. 489[81] M. Fil, M. Mesinovic, M. Morris, J. Wildberger, Beta-VAE reproducibility: Challenges and extensions, 2021, http://dx.doi.org/10.48550/ARXIV.2112.14278, https://arxiv.org/abs/2112.14278. 490

[82] P. Liznerski, L. Ruff, R.A. Vandermeulen, B.J. Franks, M. Kloft, K.R. Müller, Explainable deep one-class classification, 2020, http://dx.doi.org/10.48550/ARXIV. 2007.01760. 491[83] H. Alshammari, O. Ghorbel, M. Aseeri, M. Abid, Non-negative matrix factorization (NMF) for outlier detection in wireless sensor networks, in: 2018 14th International Wireless Communications & Mobile Computing Conference, IWCMC, 2018, pp. 506–511, http://dx.doi.org/10.1109/IWCMC.2018.8450421. 492