مقاله پژوهشی

روش ترکیبی جدیدی مبتنی بر الگوریتم‌های هوشمند جهت تشخیص نفوذ در SDN-IoT

ذکریا رئیسی، فضل‌الله ادیب‌نیا و مهدی یزدیان دهکردی

چکیده: در سال‌های اخیر، کاربرد اینترنت اشیا در جوامع به‌طور گسترده‌ای رشد یافته و از طرفی، فناوری جدیدی به نام شبکه‌های نرم‌افزارمحور جهت حل چالش‌های اینترنت اشیا پیشنهاد شده است. چالش‌های موجود در این شبکه‌های نرم‌افزارمحور و اینترنت اشیا موجب گردیده که امنیت SDN-IoT به یکی از نگرانی‌های مهم این شبکه‌ها تبدیل شود. از طرف دیگر، الگوریتم‌های هوشمند فرصتی بوده که به‌کارگیری آنها در موارد متعددی از جمله امنیت و تشخیص نفوذ، موجب پیشرفت چشم‌گیری شده است. البته سیستم‌های تشخیص نفوذ جهت محیط SDN-IoT، همچنان با چالش نرخ هشدار غلط بالا مواجه هستند. در این مقاله یک روش ترکیبی جدید مبتنی بر الگوریتم‌های هوشمند پیشنهاد شده که جهت دسترسی به نتایج خوبی در زمینه تشخیص نفوذ، الگوریتم‌های نظارتی دروازه بازگشتی مکرر و طبقه‌بند غیرنظارتی k- میانگین را ادغام می‌کند. نتایج شبیه‌سازی نشان می‌دهند که روش پیشنهادی با بهره‌گیری مزایای هر کدام از الگوریتم‌های ادغام‌شده و پوشش معایب یکدیگر، نسبت به روش‌های دیگر مانند روش Hamza دارای دقت بیشتری و بالاخص نرخ هشدار غلط کمتری است. همچنین روش پیشنهادی توانسته نرخ هشدار غلط را به ۱٫۱٪ کاهش داده و دقت را در حدود ۹۹٪ حفظ کند.

کلیدواژه: **شبکه‌های نرم‌افزارمحور، الگوریتم‌های هوشمند، اینترنت اشیا، تشخیص نفوذ، یادگیری ماشین.**

۱- مقدمه

زیرساخت اینترنت اشیا (IoT)[1] از تلفن‌های هوشمند گرفته تا شهرهای هوشمند به‌آرامی در حال نفوذ به جنبه‌های متفاوت زندگی بشر است؛ به‌گونه‌ای که طی سال‌های آتی به جزئی جدانشدنی از زندگی بشر تبدیل می‌شود [۱]. همچنین با برنامه‌های گسترده در راستای بهبود اینترنت اشیا انتظار می‌رود در سال‌های آتی، استفاده از این فناوری چندین برابر شود. با این حال ماهیت ناهمگن دستگاه‌های اینترنت اشیا و محدودیت‌های ارتباطی، مدیریت شبکه و امنیت را به یک عمل دشوار برای مجریان آن تبدیل کرده است. در واقع رمزگذاری داده‌ها یا نرم‌افزارهای ضدویروس نمی‌توانند چالش‌های امنیتی پیچیده این حوزه را مدیریت کنند.

این مقاله در تاریخ ۶ بهمن ماه ۱۴۰۱ دریافت و در تاریخ ۵ خرداد ماه ۱۴۰۲ بازنگری شد.

ذکریا رئیسی، دانشکده مهندسی کامپیوتر ، دانشگاه یزد، یزد، ایران، (email: Zakaria.raisi@stu.yazd.ac.ir).

فضل‌الله ادیب‌نیا، دانشکده مهندسی کامپیوتر ، دانشگاه یزد، یزد، ایران، (email: fadib@yazd.ac.ir).

مهدی یزدیان دهکردی، دانشکده مهندسی کامپیوتر ، دانشگاه یزد، یزد، ایران، (email: yazdian@yazd.ac.ir).

1. Internet of Things

اینترنت اشیا فناوری است که هر شیء یا دستگاهی را در هر نقطه از مسیر یا شبکه به دستگاه‌های دیگر متصل می‌کند. تمام اشیای متصل باید در زیست‌بوم[2] اینترنت اشیا با استفاده از شناسه منحصربه‌فردی آدرس‌دهی شوند [۲]. مطالعات گسترده‌ای در زمینه اینترنت اشیا انجام شده که مرور آنها سه چالش اصلی برای اینترنت اشیا را نشان می‌دهد. اولین چالش، ناهمگونی شبکه است. چالش دوم آن است که اینترنت اشیا نیازمند اتخاذ معماری بسیار گسترده به‌خصوص در برنامه‌های کاربردی مانند شهرهای هوشمند و شبکه‌های هوشمند می‌باشد. سوم آنکه اینترنت اشیا نیازمند معرفی پروتکل‌های جدیدی برای رسیدگی به مسائل خاص مربوط به محدودیت قدرت و محاسبات حسگرهای شبکه می‌باشد [۳] و [۴].

بررسی ادبیات موضوع نشان می‌دهد که زیرساخت‌های اینترنت اشیای مبتنی بر شبکه‌های نرم‌افزارمحور (SDN-IoT)[3] می‌تواند به میزان قابل ملاحظه‌ای در مدیریت چالش امنیت، مؤثر واقع شود [۵]. علت این تأثیرات مثبت، عدم وجود آسیب‌پذیری‌ها و چالش‌های شبکه‌های سنتی در محیط SDN-IoT می‌باشد. شبکه نرم‌افزارمحور، یک نمونه شبکه‌ای جدید است که انعطاف‌پذیری، قابلیت مقیاس‌پذیری و امنیت را بهبود می‌بخشد [۶] و [۷] و بنابراین می‌تواند تا حدی تأثیرات منفی ناشی از چالش‌های موجود در اینترنت اشیا را مدیریت کند. مطالعات بسیاری در مورد استقرار شبکه‌های نرم‌افزارمحور به‌عنوان یک مدل شبکه‌ای برای اینترنت اشیا انجام شده است [۸] تا [۱۰]. ایده اصلی در شبکه‌های نرم‌افزارمحور، تفکیک صفحه کنترل و داده می‌باشد [۱۱]. صفحه داده برای سوئیچ جریان داده‌هاست؛ در حالی که صفحه کنترل، بخش جدیدی در شبکه را به نام کنترل‌کننده، معرفی می‌کند. صفحه داده شامل دستگاه‌های سخت‌افزاری مانند سوئیچ‌ها، روترها، فایروال‌ها و سیستم‌های تشخیص نفوذ (IDS)[4] است. این دستگاه‌ها هیچ پردازشی جهت پرکردن جدول صفحه داده نیاز ندارند و منطق شبکه به‌طور مستقل به کنترلر منتقل گردیده است. کنترلر، خدماتی مانند وضعیت شبکه و اطلاعات هم‌بندی[5] را ارائه می‌دهد و جهت جابه‌جایی اطلاعات از کنترلر به صفحه انتقال و برعکس، از پروتکلی به نام OpenFlow استفاده می‌شود [۱۲]. با وجود این همچنان امنیت در شبکه‌های نرم‌افزارمحور، یکی از اصلی‌ترین نگرانی‌ها در محیط SDN-IoT می‌باشد.

از یک سو شبکه‌های نرم‌افزارمحور برای حل چالش‌های اینترنت اشیا از قبیل انعطاف‌پذیری و مقیاس‌پذیری، مطرح شده است اما از طرف دیگر، نگرانی‌های ناشی از تأمین امنیت شبکه‌های نرم‌افزارمحور به نگرانی‌های

2. Ecosystem
3. Software Defined Network-Internet of Things
4. Intrusion Detection Systems
5. Topology

امنیتی در اینترنت اشیا افزوده شده است؛ لذا نگرانی در مورد برقراری امنیت SDN-IoT به‌عنوان یکی از نگرانی‌های اصلی در زمینه چالش‌های اینترنت اشیا می‌تواند توجه بسیاری را به خود جلب نماید [۴] و [۱۳].

شبکه‌های نرم‌افزارمحور به‌خاطر کنترل‌کننده مرکزی، دید کلی شبکه در راستای سهولت فرایند مدیریت و اجرای سیاست‌ها را فراهم می‌کنند؛ همچنین خطاها را در زمان پیکربندی و تغییر سیاست‌های شبکه، کاهش و قابلیت همکاری را افزایش می‌دهند. تهدیدات امنیتی، چالشی حیاتی در سیستم‌های شبکه‌های سنتی است و این تهدیدها در شبکه‌های نرم‌افزارمحور تشدید می‌شود. در واقع مزایای این مدل از شبکه با تهدیدهای اضافی همراه است که در شبکه‌های سنتی وجود نداشت. بررسی‌های امنیتی انجام‌شده بر روی پروتکل OpenFlow نشان می‌دهند که حملات مختلفی در این پروتکل صورت گرفته است. برای مثال، جداول جریان و کانال‌های ارتباطی بین دستگاه‌ها و کنترل‌کننده می‌توانند تحت تأثیر حملات منع سرویس (DoS)[1] قرار گیرند. کانال ارتباطی بین کنترلر و سوئیچ با یک اتصال TCP با پروتکل رمزنگاری TLS برای امن‌کردن کانال ارتباطی شروع می‌شود و بدون یک روش رمزنگاری، ارتباط بین کنترلر و دستگاه‌های انتقال در معرض حملات مرد میانی است [۱۱].

در نمونه مطالعاتی دیگری، Kloti و همکاران [۱۲] یک تحلیل امنیتی بر روی پروتکل OpenFlow انجام داده‌اند. نتایج مطالعات آنها نشان می‌دهد که حمله DoS، کانال‌های ارتباطی و جدول جریان‌ها را تهدید می‌کند. علاوه بر این، حملات دست‌کاری به‌طور قابل ملاحظه‌ای جداول قوانین را با افزودن قوانین از منابع غیرقابل اعتماد مورد هدف قرار داده است.

مروری بر ادبیات نشان می‌دهد که وجود سیستم‌های تشخیص نفوذ کارآمد در یک محیط SDN-IoT به‌دلیل رشد قابل توجه کاربرد اینترنت اشیا در جنبه‌های متفاوت زندگی و برجسته‌شدن مسئله رعایت امنیت در این گونه محیط‌ها ضروری به نظر می‌رسد. سیستم‌های تشخیص نفوذ، سیستم‌های نرم‌افزاری یا سخت‌افزاری اختصاص‌داده‌شده جهت نظارت بر جریان ترافیک در مقابل تهدیدهای امنیتی هستند. سیستم‌های تشخیص نفوذ استاندارد شامل سه مرحله جمع‌آوری اطلاعات از شبکه، تجزیه و تحلیل و سپس پاسخ مناسب در صورت وجود تهدید می‌باشند. همچنین برای تجزیه و تحلیل ترافیک جمع‌آوری‌شده، سه روش مبتنی بر امضا یا سوءاستفاده، تشخیص غیرعادی[2] و مبتنی بر پروتکل‌های Stateful وجود دارد.

امنیت داده‌های سایبری همچنان یک چالش برای جامعه یادگیری ماشین به حساب می‌آید؛ زیرا حجم بالای ترافیک، تفکیک رفتار غیرعادی از عادی را دشوار می‌کند. در واقع تصمیم‌گیری جهت جداسازی رفتار عادی از غیرعادی، هسته اصلی یک سیستم هوشمند تشخیص نفوذ است و به‌عنوان مؤلفه‌ای که مسئول اعلام هشدارها در زمان شناسایی تهدیدات بالقوه است در نظر گرفته می‌شود [۱۴]. از طرفی روش‌های سنتی برای انجام این وظیفه، دقت کافی را ندارند. با توجه به این موضوع در سال‌های اخیر، الگوریتم‌های یادگیری ماشین به‌ویژه روش‌های مبتنی بر شبکه‌های عصبی بازگشتی پیشنهاد شده‌اند [۱۵] تا [۱۷].

هدف اصلی مقاله با توجه به ضرورت سیستم‌های تشخیص نفوذ در SDN-IoT، ارائه یک سیستم تشخیص نفوذ کارآمد با به‌کارگیری یک رویکرد ترکیبی جدید مبتنی بر الگوریتم‌های هوشمند LSTM[3] و GRU می‌باشد. این مقاله بر چهار محور اصلی بعدی متمرکز شده است:

۱) کاهش نرخ هشدار غلط و بهبود دقت نتایج خروجی

۲) سازگاری روش پیشنهادی با معماری SDN-IoT به‌طوری که تمام جوانب ممکن در این فناوری در نظر گرفته شود.

۳) استفاده از مجموعه داده‌ای که متناسب با ویژگی‌های موجود در این فناوری باشد. در واقع در ارزیابی روش پیشنهادی از مجموعه داده‌های قدیمی که هیچ شباهتی با ترافیک این فناوری ندارد، استفاده نشده است.

۴) قابلیت پیاده‌سازی راحت. بسیاری از روش‌های قبلی از شیوه‌های پیچیده‌ای استفاده می‌کنند که یا به سخت‌افزار اضافه نیاز دارند یا پیاده‌سازی آنها بسیار سخت است.

در ادامه، پیشینه پژوهش به‌اختصار مرور می‌گردد و سپس در بخش دو روش پیشنهادی و توضیحات مربوط ارائه شده است. در بخش سوم، آزمایش‌هایی جهت ارزیابی کارایی روش پیشنهادی انجام گردیده و نتایج آزمایش‌ها و ارزیابی کارایی بیان شده است. سپس در بخش چهارم، مقایسه‌ای بین روش پیشنهادی و برخی روش‌های پیشین مشابه انجام شده و نهایتاً در بخش پایانی، نتایج حاصل از پژوهش جاری و نمونه‌ای از کارهای آتی بیان گردیده است.

در اینجا به آشنایی با ادبیات موضوع، مروری بر پیشینه پژوهش و شناسایی رویکردهای هوشمند پیشین در زمینه مدیریت چالش امنیت در یک محیط SDN-IoT پرداخته می‌شود.

اینترنت اشیا یک فناوری نوظهور است که زیست‌بوم هوشمند را قادر می‌سازد تا فناوری‌های ناهمگن را کنار هم قرار دهد [۱۸] و [۱۹]. طیف گسترده‌ای از دستگاه‌های اینترنت اشیا، دلالت بر وجود احتمال آسیب‌پذیری‌هایی است که از طریق آن می‌توان دستگاه‌ها را به خطر انداخت. با افزایش سریع تعداد دستگاه‌های اینترنت اشیا، عملکرد شبکه مانند مقیاس‌پذیری، امنیت و حفظ QoS یک مسئله اساسی است و به بخشی از نیازمندی‌های اصلی اینترنت اشیا تبدیل شده است [۲۰] تا [۲۳].

شبکه‌های نرم‌افزارمحور، رویکردی است که جهت مقابله با چالش‌های ناشی از ظهور دستگاه‌های اینترنت اشیا در شبکه در نظر گرفته شده است. شبکه‌های نرم‌افزارمحور با جداسازی صفحه کنترل و صفحه داده، برنامه‌نویسی و انعطاف‌پذیری در مدیریت شبکه را فراهم می‌کنند. هدف از این جداسازی، امکان تنظیم پیکربندی شبکه با انعطاف‌پذیری و برنامه‌نویسی بیشتر است [۲۴].

در حالت کلی می‌توان دلایل ادغام شبکه‌های نرم‌افزارمحور و اینترنت اشیا را به‌صورت خلاصه در موارد زیر بیان کرد [۷]، [۲۵] و [۲۶]:

- موجب سهولت در جمع‌آوری داده‌ها و تحلیل، تصمیم‌گیری و فرایند کنترل می‌شود.
- منابع شبکه با استفاده از فناوری شبکه‌های نرم‌افزارمحور بهینه شده است. برنامه‌های پرمصرف اینترنت اشیا در یکپارچه‌سازی با شبکه‌های نرم‌افزارمحور، اپراتورهای شبکه را قادر می‌سازد تا کاربران، دستگاه‌ها و گروه‌ها را مدیریت کنند.

در ادامه این بخش، برخی از کارهای پژوهشی پیشین در زمینه اینترنت اشیا، شبکه‌های نرم‌افزارمحور و SDN-IoT در راستای شناسایی ایده‌های اصلی نهفته در آنها و ویژگی‌های روش‌های تشخیص نفوذ مرور گردیده است.

1. Denial of Service
2. Anomaly
3. Long Short-Term Memory

جدول ۱: مقایسه روش‌های مطرح قبلی و ویژگی‌های آنها.

روش	ACC	TPR	FPR	مزایا و معایب
Hamza و همکاران [۲۷]	بالا	–	–	الگوریتم بیز و استفاده از MUD برای به‌دست‌آوردن امضای داده‌های طبیعی و غیرطبیعی. کارایی بهتر از Snort دارد. ذکرنکردن مقدار ACC و FP.
Hamza و همکاران [۲۹]	۹۷٫۵	۷۲٫۳	۲٫۴	استفاده از MUD که نیاز به سخت‌افزار اضافی دارد و همچنین استفاده از ۴ الگوریتم به‌صورت ترکیبی که بسیار وقت‌گیر می‌باشد. این در حالیست که TPR بسیار پایین است.
Silveira و همکاران [۳۲]	۹۳	–	۶	روش پیشنهادی با رگرسیون منطقی دقت خوبی دارد؛ اما FPR نیز بالاست.
Wani و همکاران [۳۳]	۹۵٫۹	۹۶٫۴	۷	الگوریتم MLP دقت بالا دارد؛ ولی همچنان FPR بالاست و فقط جهت شناسایی حمله DDOS است.
روش پیشنهادی	۹۹	۹۹	۱٫۱	دقت و TPR بالا، FPR بسیار پایین و مدت زمان آموزش کم از مزایای روش پیشنهادی است.

Hamza و همکاران [۲۷] در روش پیشنهادی خود سعی می‌کنند تا سیاست‌های MUD را به قوانین جریان ترجمه کنند که می‌تواند با استفاده از SDN اجرا شود و حملاتی را بدون نیاز به IDS شناسایی کند. در این مقاله نویسندگان سیستمی را ایجاد می‌کنند که سیاست‌های MUD را به قوانین جریان‌ها ترجمه می‌کند که به‌صورت فعال در سوئیچ‌های شبکه پیکربندی می‌شوند و همچنین بر اساس اتصالات زمان اجرای DNS اضافه می‌شوند.

Shravanya و همکاران [۲۸] روش پیشنهادی خود را در ایجاد یک معماری امن برای کنترل‌کننده شبکه‌های نرم‌افزارمحور در نظر گرفته‌اند که در برابر حمله DoS قوی باشد. در واقع در این نوع حملات، سوئیچ شبکه‌های نرم‌افزارمحور مانند یک میزبان حمله‌کننده و کنترل‌کننده شبکه‌های نرم‌افزارمحور به‌عنوان سرور قربانی عمل می‌کنند. نویسندگان بعد از پیاده‌سازی معماری مورد نظر و الگوریتم‌های درخت تصمیم و KNN به‌دقتی در حدود ۹۹٪ دست پیدا کرده‌اند.

Hamza و همکاران [۲۹] جهت مقایسه از الگوریتم Snort استفاده کرده‌اند. این مقایسه نشان می‌دهد که Snort از میان ۴۰ نوع حمله موجود، فقط قادر است که ۲ تا از آنها را تشخیص دهد؛ زیرا امضا برای حملات دیگر در Snort وجود ندارد و نمی‌تواند بقیه حملات را تشخیص دهد. روش پیشنهادی در مقایسه با Snort در مقابل حملات جدید بسیار خوب عمل کرده و توانسته دقتی در حدود ۹۹٪ داشته باشد.

Meidan و همکاران [۳۰] سه الگوریتم DNN، SVM و LGBM را در نظر گرفته‌اند و با استفاده از داده‌های حاصل از دستگاه‌های اینترنت اشیا، این الگوریتم‌ها را آموزش داده‌اند. همچنین معیارهای ارزیابی در این مقاله عبارتند از اندازه طبقه‌بند، مدت زمان آموزش طبقه‌بند، مدت زمان آموزش هر جریان از مجموعه داده و *FPR*. نویسندگان اظهار داشته‌اند که الگوریتم LGBM بهترین عملکرد را ارائه کرده است.

Salman و همکاران [۳۱] یک چارچوب امنیتی را برای تشخیص ترافیک مخرب ارائه داده‌اند که جهت شناسایی نوع ترافیک ایجادشده و شناسایی حملات شبکه استفاده گردیده است. نویسندگان مقایسه‌ای بین الگوریتم‌های مختلف یادگیری ماشین و جنگل تصادفی انجام داده و اظهار داشته‌اند که بهترین نتیجه برای تشخیص غیرعادی ترافیک با دقت ۹۷٪ به‌دست آمده است.

Silveira و همکاران [۳۲] یک سیستم تشخیص نفوذ مبتنی بر امضا را با جایگذاری الگوریتم‌های جنگل تصادفی، رگرسیون منطقی و گرادیان تقویتی در کنترلر پیاده‌سازی نمودند. سپس مجموعه داده‌های موجود را با استفاده از سوئیچ‌ها انتقال داده تا عمل تشخیص انجام شود. نویسندگان اظهار داشته‌اند که نتایج، دقت بالای ۹۳٪ نرخ هشدار کاذب (FAR) را نشان می‌دهد.

Wani و همکاران [۳۳] یک مکانیزم امنیتی مبتنی بر شبکه نرم‌افزارمحور را برای شناسایی و کاهش DDoS در شبکه‌های اینترنت اشیا ارائه دادند. روش پیشنهادی آنها جهت شناسایی حملات با استفاده از MLP آموزش دیده و سپس عمل دسته‌بندی حملات را انجام می‌دهد. نویسندگان اظهار داشته‌اند که روش پیشنهادی آنها دقتی در حدود ۹۹٪ و *FPR* و *FNR* به‌ترتیب ۷٪ و ۳٪ ارائه نموده است.

Novaes و همکاران [۳۴] برای بررسی و شناسایی حملات منع سرویس توزیع‌شده و حملات Portscan در محیطهای شبکه‌های نرم‌افزار، روش LSTM-FUZZY را ارائه دادند. این سیستم دارای سه مرحله مجزای تعیین خصوصیات، تشخیص ناهنجاری و کاهش آنهاست. نتایج ارزیابی نشان‌دهنده کارایی قابل قبول رویکرد نویسندگان در مدیریت شبکه، شناسایی و کاهش وقوع حملات است.

Xu و همکاران [۳۵] یک استراتژی دفاع حمله DDoS را بر اساس طبقه‌بندی ترافیک پیشنهاد کردند. برای بهبود انعطاف‌پذیری و کاهش بار شبکه‌های نرم‌افزارمحور در برابر حملات، DDoS از معماری مجازی‌سازی عملکرد شبکه (SDNFV) و استراتژی طبقه‌بندی ترافیک استفاده کرده است. نویسندگان در ماژول تشخیص حمله با کمک یادگیری تقویتی، الگوریتم طبقه‌بندی جنگل تصادفی را بهبود داده‌اند؛ به‌طوری که دقت ماژول طبقه‌بندی به ۹۹٫۵۴٪ رسیده است.

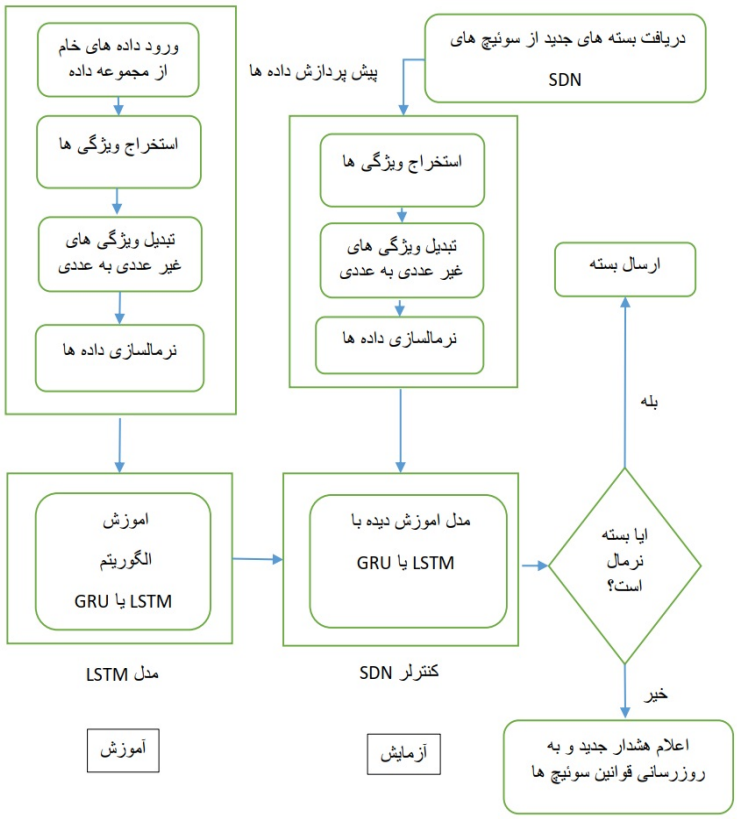
Khan و همکاران [۳۶] مقایسه‌ای بین الگوریتم‌های مختلف یادگیری ماشینی انجام داده‌اند که می‌تواند برای شناسایی و پیش‌بینی هر گونه داده مخرب یا غیرعادی برای مجموعه داده‌ای از ویژگی‌های محیطی ایجادشده استفاده گردد. نویسندگان اظهار داشته‌اند که در آزمایش‌های خود به دقتی حدود ۹۶٫۵٪ و زمان اجرای کمتر از ۰٫۲ ثانیه دست یافته‌اند.

Abdullahi و همکاران [۳۷] یک بررسی سیستماتیک از حملات تشخیص امنیت سایبری در اینترنت اشیا را با استفاده از روش‌های هوش مصنوعی ارائه داده‌اند. نویسندگان معتقدند که با توجه به توسعه سریع اینترنت اشیا در حوزه‌های مختلف، مقادیر زیادی داده به‌طور مداوم تولید می‌شود که نیاز به تمرکز بیشتر بر حریم خصوصی و امنیت دارند. آنها در بررسی خود از الگوریتم‌های SVM، RNN و RF استفاده کرده‌اند و به نتایج قابل قبولی دست یافته‌اند.

خلاصه‌ای از روش‌های مطرح قبلی و مقایسه آنها در جدول ۱ نشان داده شده است.

۲- روش پیشنهادی

شبکه نرم‌افزارمحور، نمونه‌ای از شبکه‌های جدید است و انعطاف‌پذیری، قابلیت مقیاس‌پذیری و امنیت را بهبود می‌بخشد؛ لذا می‌تواند چالش‌های موجود در IoT را مانند امنیت حل کند. سیستم‌های تشخیص نفوذ، سیستم‌های نرم‌افزاری یا سخت‌افزاری اختصاص‌داده‌شده برای نظارت بر جریان ترافیک در مقابل تهدیدهای امنیتی هستند. با توجه به ضرورت



شکل ۱: معماری سیستم تشخیص نفوذ پیشنهادی.

سیستم‌های تشخیص نفوذ در SDN-IoT، یک سیستم تشخیص نفوذ با استفاده از الگوریتم هوشمند برای این محیط پیشنهاد شده است. بنابراین پژوهش جاری از یک رویکرد ترکیبی جدید مبتنی بر الگوریتم‌های هوشمند برای سیستم تشخیص نفوذ کارآمد در SDN-IoT استفاده کرده است. رویکرد پیشنهادی مقاله، الگوریتم‌های LSTM و GRU را جهت دستیابی به نتایج دقیق‌تر و کاهش نرخ هشدار خطا ادغام نموده و در ادامه رویکرد پیشنهادی در دو بخش مجزا تشریح شده است.

۲-۱ معماری رویکرد پیشنهادی

یک شبکه عصبی بازگشتی (RNN)[1] سنتی اگر به اندازه کافی بزرگ باشد از نظر تئوری باید قادر به تولید دنباله‌های پیچیده باشد؛ اما در عمل مشاهده می‌شود که این شبکه برای یادگیری دنباله‌های طولانی‌مدت به مشکل برمی‌خورد. در واقع این ویژگی باعث می‌شود یک شبکه عصبی بازگشتی سنتی در مدل‌سازی ساختارهای بلندمدت، ضعیف عمل کند. این فراموشی باعث می‌گردد تا این نوع از شبکه‌ها در زمان تولید دنباله‌ها در معرض ناپایداری قرار گیرند. در واقع اگر پیش‌بینی‌های شبکه، تنها وابسته به چند ورودی اخیر باشد، شانس بسیار کمی برای تصحیح و جبران اشتباهات گذشته توسط شبکه وجود دارد. مشکل ناپایداری به‌طور ویژه در زمان مواجهه با داده اعشاری، وخیم می‌شود؛ زیرا پیش‌بینی‌ها می‌توانند از دامنه‌ای که داده‌های آموزشی بر روی آن قرار گرفته‌اند، فاصله بگیرند.

استفاده از حافظه می‌تواند به‌عنوان راه‌حل به‌مراتب بهتر و تأثیرگذارتری نسبت به بقیه راه‌حل‌ها مطرح گردد. LSTM یک معماری شبکه عصبی بازگشتی است که برای ذخیره‌سازی و دسترسی بهتر به اطلاعات نسبت به نسخه سنتی آن، طراحی شده و به‌گونه‌ای عمل می‌کند که مقادیر نورون‌های لایه مخفی توسط فعال‌سازهای محلی که نزدیک به آنهاست، تحت تأثیر قرار می‌گیرند. همچنین وزن‌های شبکه توسط محاسباتی که بر روی تمام دنباله رخ می‌دهد، تأثیر می‌پذیرند که این به‌منزله حافظه بلندمدت است [۱۵]. در شبکه RNN، شبکه در هر گام زمانی از ابتدا بازنویسی می‌شود؛ اما یک شبکه LSTM دارای توانایی تصمیم‌گیری نسبت به حفظ حافظه فعلی از طریق دروازه‌های موجود می‌باشد. یعنی اگر واحد LSTM، ویژگی مهمی در دنباله ورودی گام‌های ابتدایی تشخیص دهد، آن گاه به‌سادگی می‌تواند این اطلاعات را طی مسیر طولانی منتقل کند. بدین ترتیب LSTM وابستگی‌های بلندمدت احتمالی را دریافت و حفظ می‌نماید و از طرف دیگر قادر است مشکل محوشدن گرادیان را که در شبکه‌های عصبی وجود دارد مدیریت کند [۱۶]. محوشدگی گرادیان، مشکلی است که در هنگام آموزش شبکه‌های عصبی مصنوعی با استفاده از روش یادگیری پس از انتشار خطا یا همان مبتنی بر گرادیان، اتفاق می‌افتد. در این نوع روش‌ها به‌منظور به‌روزرسانی پارامترهای شبکه عصبی از گرادیان استفاده می‌شود و هر پارامتر با توجه به میزان اثری که در نتیجه نهایی شبکه دارد مورد تغییر قرار می‌گیرد. این کار با استفاده از مشتق جزئی تابع خطا نسبت به هر پارامتر در هر تکرار فرایند آموزش صورت می‌پذیرد. مشکل محوشدگی در واقع اشاره به این مسئله دارد که مقادیر گرادیان‌ها با حرکت به سمت ابتدای شبکه تدریجاً به حدی کوچک می‌شوند که تغییرات خاصی روی وزن‌ها صورت نمی‌گیرد و به این علت، فرایند آموزش شدیداً کند می‌شود؛ بنابراین این مسئله در بدترین حالت باعث توقف فرایند آموزش می‌گردد و به‌واسطه عمق زیاد شبکه رخ می‌دهد. برای حل این مشکل، LSTM با استفاده از مقادیر حالت مخفی، توابع فعال‌ساز را به شکلی تحت تأثیر قرار می‌دهد که وزن‌ها به‌واسطه تمام دنباله و نه یک ورودی، تحت تأثیر قرار گیرند.

معماری سیستم تشخیص نفوذ پیشنهادی در شکل ۱ آمده است. همان طور که مشاهده می‌شود در ابتدا ویژگی‌های موجود در مجموعه داده استخراج می‌گردد و سپس داده‌های غیرعددی به داده‌های عددی تبدیل

1. Recurrent Neural Network

می‌گردد. در گام بعدی داده‌ها با اعمال روش نمره استاندارد[1] نرمال‌سازی می‌شود و الگوریتم LSTM جهت آموزش داده‌های نرمال‌سازی‌شده مورد استفاده قرار می‌گیرد. با ورود بسته‌های ناشناخته به سوئیچ‌های شبکه، این بسته‌ها برای کنترل‌کننده ارسال می‌شوند و بعد از استخراج ویژگی‌ها و انجام پیش‌پردازش روی مدل آموزش‌دیده بررسی می‌شود. در بررسی انجام‌شده بر روی مدل آموزش‌دیده، مشخص می‌گردد که آیا بسته دریافت‌شده نرمال است یا مخرب. اگر بسته نرمال باشد به مقصد مورد نظر خود ارسال شده و قوانین بر روی سوئیچ‌ها تغییر پیدا می‌کند؛ ولی اگر بسته مخرب باشد، هشداری برای مدیر شبکه ارسال شده و آدرس IP مورد نظر نیز به‌عنوان حمله‌کننده، شناسایی می‌شود. بنابراین دریافت بسته جدید از این میزبان نیز با تغییر قوانین در سوئیچ‌ها متوقف می‌گردد.

۲-۲ الگوریتم‌های پایه LSTM

LSTM توسعه یا نوعی از شبکه عصبی RNN است. در واقع RNN سنتی از حافظه کوتاه‌مدت در صورتی که توالی طولانی باشد رنج می‌برد و طبقه‌بند برای حمل اطلاعات از مراحل قبلی تا موارد بعدی دچار مشکل می‌شود. بنابراین شبکه عصبی مکرر از مشکل گرادیان رنج می‌برد و LSTM به‌عنوان یک راه حل طراحی شده است. این الگوریتم لایه‌های پنهان با بلوک‌های حافظ جایگزین شده‌اند که مشکل فراموشی در یادگیری الگوی دنباله‌های طولانی حل شده است. LSTM در معماری پایه خود، مفهوم استفاده از مکانیزم دروازه را برای واحدهای تنظیم‌کننده جریان اطلاعات در نظر می‌گیرد. مکانیزم دروازه شامل سه دروازه یعنی ورودی، خروجی و دروازه فراموشی است و این دروازه‌ها وظیفه دارند که تصمیم بگیرند کدام توالی داده برای ذخیره یا دورانداختن مناسب است. دروازه‌های ورودی و خروجی، جریان اطلاعات ورودی و خروجی سلول‌ها را کنترل می‌کنند. علاوه بر این اگر اطلاعات نامناسب باشد، دروازه فراموشی برای تنظیم مجدد اطلاعات حالت قبلی خود وارد عمل می‌شود [۷].

۲-۳ الگوریتم‌های مورد استفاده (LSTM و GRU)

یک مشکل در آموزش معماری اولیه RNN، وابستگی بسیار طولانی مدت به توالی است که یادگیری مؤثر را دشوار می‌کند. این مسئله می‌تواند زمینه‌ساز انگیزه اصلی برای به‌کارگیری شبکه‌های نوع LSTM باشد. LSTM سعی می‌کند با معرفی دروازه‌ها، مکانیزم‌هایی در شبکه‌های عصبی عمیق ایجاد کند که مجاز هستند تا مشخص کنند که چه اطلاعاتی را پردازش کنند. در یک سلول LSTM سه دروازه ورودی، فراموش و خروجی وجود دارد. دروازه ورودی مشخص می‌کند کدام قسمت از ورودی، اجازه ورود به سلول را دارد. دروازه فراموشی با کنترل وضعیت داخلی سلول تعیین می‌کند کدام بخش باید به‌روزرسانی شود. سرانجام، دروازه خروجی آنچه را که به خارج از سلول بر روی خط لوله خروجی رانده می‌شود، کنترل می‌کند. معمول‌ترین قوانین به‌روزرسانی برای یک سلول LSTM به‌صورت (۱) تا (۶) تعریف می‌شود

$$i_t = \sigma(w_i x_t + v_i h_{t-1} + b_i) \quad (۱)$$

$$f_t = \sigma(w_f x_t + v_f h_{t-1} + b_f) \quad (۲)$$

$$o_t = \sigma(w_o x_t + v_o h_{t-1} + b_o) \quad (۳)$$

$$\bar{c}_t = \tanh(w_c x_t + v_c h_{t-1} + b_c) \quad (۴)$$

$$c_t = f_t c_{t-1} + i_t \bar{c}_t \quad (۵)$$

$$h_t = o_t \tanh c_t \quad (۶)$$

در (۱) تا (۶) i_t، f_t و o_t به‌ترتیب دروازه ورودی، فراموشی و خروجی و b_i، b_f و b_o نیز مقادیر ثابتی هستند که به آنها بایاس گفته می‌شود. همچنین σ و $\bar{c}_t$ به‌ترتیب تابع سیگموئید و حالت سلول کاندیدا را نشان می‌دهد و h_t خروجی سلول در مرحله t است. پارامترهای قابل تنظیم در واقع ماتریس‌های w و v هستند.

الگوریتم LSTM دارای پیچیدگی زمانی بالایی است؛ لذا برای حل این مشکل، ادغام الگوریتم‌های LSTM و GRU پیشنهاد شده است. شبکه عصبی GRU بهبودیافته، شبکه عصبی LSTM است. شبکه عصبی GRU پس از تنظیم مجدد دروازه، اطلاعات را پردازش نموده و زمان آموزش را کاهش می‌دهد. بدین منظور، این شبکه عصبی تعداد دروازه‌های LSTM را از ۳ دروازه به ۲ دروازه کاهش می‌دهد. در واقع روش GRU با LSTM فرق زیادی ندارد و فقط تعداد دروازه‌ها را کاهش داده و وظایف هر کدام از دروازه‌ها را در حد امکان در یکدیگر ادغام کرده است [۳۸] و [۳۹]. دروازه‌های به‌روزرسانی و تنظیم مجدد در GRU به‌صورت (۷) تا (۱۱) محاسبه شده است

$$l_u = \sigma(w_u[x_t + c_{t-1}] + b_u) \quad (۷)$$

$$l_r = \sigma(w_r[x_t + c_{t-1}] + b_r) \quad (۸)$$

$$c_t = l_u \bar{c}_t + (1 - l_u) c_{t-1} \quad (۹)$$

$$o_t = \text{softmax}(w_r c_t + b_o) \quad (۱۰)$$

$$\bar{c}_t = \tanh(w_c[x_t] + l_r c_{t-1} + b_c) \quad (۱۱)$$

در (۷) تا (۱۱)، l_u و l_r به‌ترتیب دروازه‌های به‌روزرسانی و تنظیم مجدد و همچنین w_u، w_r و w_c وزن‌های شبکه عصبی هستند.

محوشدگی گرادیان، مشکلی است که هنگام آموزش شبکه‌های عصبی مصنوعی با استفاده از روش یادگیری پس‌انتشار خطا یا همان مبتنی بر گرادیان اتفاق می‌افتد. جهت رفع مشکل محوشدگی گرادیان می‌توان از راه‌حل‌های بعدی استفاده کرد:

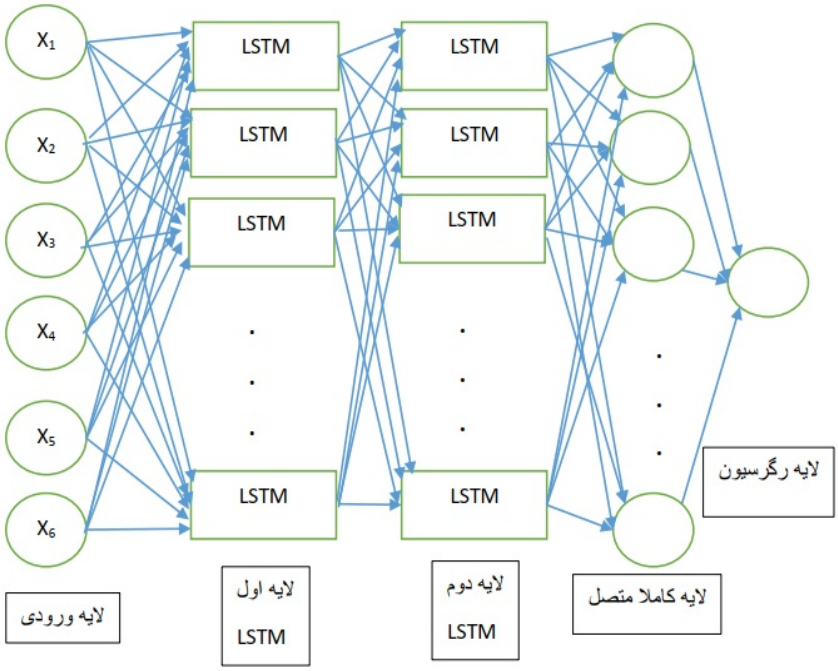
- مقداردهی اولیه وزن‌ها به‌طوری که احتمال رخدادن محوشدگی گرادیان کمینه شود.
- استفاده از شبکه عصبی IRNN یا Echo State
- استفاده از شبکه‌های عصبی بازگشتی LSTM و GRU که دقیقاً برای این موضوع طراحی شده‌اند و در روش پیشنهادی از این مورد استفاده شده است.

در روش پیشنهادی در این مقاله، بعد از آنکه داده‌ها با استفاده از ترکیب الگوریتم‌های LSTM و GRU آموزش دیدند، داده‌های جدید را برای انجام عمل پیش‌بینی دریافت می‌کند. لذا بعد از عمل رگرسیون، به‌کارگیری یک طبقه‌بند کارآمد برای طبقه‌بندی پیش‌بینی‌های انجام‌شده، ضروری به‌نظر می‌رسد.

۳- پیاده‌سازی و نتایج تجربی

در این بخش مقاله، مطالب مربوط به پیاده‌سازی روش پیشنهادی و نتایج تجربی آزمایش‌های انجام‌شده برای ارزیابی کارایی روش پیشنهادی ارائه گردیده و برای درک بهتر مطالب، این بخش در دو زیربخش مجزا سازمان‌دهی شده است.

1. Z-Score



شکل ۲: معماری LSTM پیاده‌سازی‌شده در روش پیشنهادی.

جدول ۲: ابرپارامترهای اصلی برای بهینه‌سازی LSTM.

ابرپارامترها	دامنه	مقدار انتخاب‌شده
تعداد لایه‌های مخفی	۱، ۲، ۳، ۴، ۵	۲
Batch size	۱۰، ۲۰، ۳۰، ۴۰، ۵۰، ۸۰	۸۰
نرخ یادگیری	۰/۰۱، ۰/۰۰۵، ۰/۰۰۱	۰/۰۱
LSTM unit number	۳۲، ۶۴، ۱۲۸، ۲۵۶	۶۴ تا ۱۲۸
Dropout	۰/۵، ۰/۲، ۰/۱	۰/۲
Gradient threshold	۰/۰۱، ۰/۰۲، ۰/۰۳، ۰/۰۴	۰/۰۱

جدول ۳: ابرپارامترهای اصلی جهت بهینه‌سازی GRU.

ابرپارامترها	دامنه	مقدار انتخاب‌شده
تعداد لایه‌های مخفی	۱، ۲، ۳، ۴، ۵	۲
Batch size	۱۰، ۲۰، ۳۰، ۴۰، ۵۰، ۸۰	۸۰
نرخ یادگیری	۰/۰۱، ۰/۰۰۵، ۰/۰۰۱	۰/۰۱
LSTM unit number	۳۲، ۶۴، ۱۲۸، ۲۵۶	۶۴ تا ۱۲۸
Dropout	۰/۵، ۰/۲، ۰/۱	۰/۲
Gradient threshold	۰/۰۱، ۰/۰۲، ۰/۰۳، ۰/۰۴	۰/۰۱

۳-۱ پیاده‌سازی روش پیشنهادی

جهت پیاده‌سازی روش پیشنهادی از محیط برنامه‌نویسی متلب نسخه ۲۰۲۰ بر روی سیستمی با مشخصات Intel Core i۷ – ۲۶۲۰M CPU و RAM ۶ GB استفاده شده است. در واقع ابزار متلب با توجه به پشتیبانی بسیار قوی از الگوریتم‌های هوشمند مورد استفاده قرار گرفته و همچنین این نرم‌افزار، کتابخانه‌های جدید یادگیری ماشین را به‌خوبی پشتیبانی می‌کند.

LSTM ۳-۱-۱

در روش پیشنهادی، استفاده از الگوریتم‌های LSTM و GRU برای آموزش شبکه و استفاده از k-میانگین برای طبقه‌بندی داده‌هاست. معماری LSTM پیاده‌سازی‌شده در روش پیشنهادی در شکل ۲ نشان داده شده و همچنین ابرپارامترهای اصلی برای بهینه‌سازی LSTM در جدول ۲ آمده است.

GRU ۳-۱-۲

در مقایسه با LSTM، GRU می‌تواند ضمن بهبود ساختار کنترل گیت، مشکل گرادیان در RNN را حل کند. از آنجا که تعداد دروازه‌ها از ۳ به ۲ تغییر یافته است، پارامترهای آموزش کاهش یافته و در نتیجه سرعت آموزش افزایش پیدا می‌کند. GRU از دو GRU متشکل از یک تابع سیگموئید و یک عمل ضرب برای کنترل انتخاب اطلاعات استفاده می‌نماید.

معماری GRU مشابه معماری LSTM است و لایه‌های موجود در آن هیچ فرقی ندارند. تنها تفاوت آن، وجود گره‌های GRU به‌جای گره‌های LSTM است. معماری این گره‌ها باعث کاهش مدت زمان آموزش می‌شود و سربارای که در آموزش LSTM وجود دارد در این معماری وجود ندارد. در جدول ۳ ابرپارامترهای موجود در بهینه‌سازی الگوریتم GRU نشان داده شده است.

در بهینه‌سازی این پارامترها، استفاده از برخی الگوریتم‌های تکاملی می‌تواند مفید باشد؛ اما این روش‌ها وقت‌گیر هستند و لذا به‌روزرسانی این ابرپارامترها و انتخاب مقدار بهینه به‌صورت آزمون و خطا می‌تواند از نظر زمان به‌صرفه باشد؛ زیرا انتخاب مقدار نزدیک به مقدار بهینه نیز کافی است و دقت الگوریتم زیاد تغییر نمی‌کند. جهت محاسبه خطا در زمان‌های مختلف و به‌روزرسانی مقادیر وزن‌ها از تابع خطا استفاده شده است. محاسبه خطا به‌صورت (۱۲) است

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_o)^2} \qquad (۱۲)$$

در واقع برای مقایسه نتایج پیش‌بینی‌شده با مقادیر اصلی و کاهش خطا از (۱۲) استفاده شده است. هدف اصلی در تعریف تابع RMSE، حداقل‌سازی خطاست. در (۱۲) y_i و y_o به‌ترتیب مقادیر واقعی و پیش‌بینی‌شده هستند.

k-میانگین ۳-۱-۳

رگرسیون، یک روش برای پیش‌بینی و خوشه‌بندی، یک روش برای طبقه‌بندی است که افراد (نقاط داده) را به گروه‌های مختلف (خوشه‌ها)

جدول ۴: حملات موجود در مجموعه داده UNSW-NB۱۵.

برچسب	اندازه	توزیع (٪)
Normal	۵۶۰۰۰	۳۱٫۹۴
Backdoor	۱۷۴۶	۱
Analysis	۲۰۰۰	۱٫۱۴
Fuzzers	۱۸۱۸۴	۱۰٫۳۰
Shellcode	۱۱۳۳	۰٫۶۵
Reconnaissance	۱۰۴۹۱	۵٫۹۸
Exploits	۳۳۳۹۳	۱۹٫۰۴
DoS	۱۲۲۶۴	۶٫۹۹
Worms	۱۳۰	۰٫۰۷
Generic	۴۰۰۰۰	۲۲٫۸۱
Total	۱۷۵۳۴۱	۱۰۰

طبقه‌بندی می‌کند و الگوریتمی مؤثر برای داده‌های با حجم زیاد است. خوشه‌بندی یک روش داده‌کاوی بدون نظارت است و این بدان معناست که این نوع الگوریتم‌ها به مجموعه داده‌های آموزشی نیاز ندارند. از محبوب‌ترین روش‌های خوشه‌بندی، خوشه‌بندی مبتنی بر بخش‌بندی و مبتنی بر سلسله‌مراتب است. خوشه‌بندی $k-$میانگین که در آن k تعداد خوشه‌ها را نشان می‌دهد، نوعی خوشه‌بندی مبتنی بر بخش‌بندی است. در خوشه‌بندی $k-$میانگین، هر خوشه توسط مرکز (یا میانگین) نقاط داده در خوشه تعریف می‌شود.

جهت شروع فرایند خوشه‌بندی، $k-$میانگین ابتدا به انتخاب مقدار k یا همان تعداد خوشه‌ها نیاز دارد و سپس k نقطه داده به‌عنوان مرکز اصلی انتخاب می‌شود. می‌توان k مرکز را به‌صورت تصادفی یا بر اساس توزیع داده انتخاب کرد. بعد از آن فاصله هر نقطه داده تا هر مرکز محاسبه می‌شود و یک نقطه داده، عضو خوشه‌ای است که از همه خوشه‌های دیگر به آن نزدیک‌تر است. هنگامی که همه نقاط داده درون یک خوشه قرار می‌گیرند، مرکز نقاط یک خوشه مجدداً محاسبه می‌شود. فواصل هر نقطه داده تا هر مرکز جدید نیز محاسبه می‌شود و به‌دنبال آن دسته‌بندی مجدد نقاط بر اساس خوشه‌ها بر اساس قانون کمترین فاصله، انجام می‌شود [۴۰].

روش‌ها و الگوریتم‌های متعددی برای تبدیل داده‌ها به گروه‌های هم‌شکل یا مشابه وجود دارد. الگوریتم $k-$میانگین یکی از ساده‌ترین الگوریتم‌ها در داده‌کاوی به‌خصوص در حوزه یادگیری نظارت‌نشده است. درجه پیچیدگی محاسباتی این الگوریتم برابر با $O(n^{dk+1})$ است که n تعداد اشیا، d بعد ویژگی‌ها و k تعداد خوشه‌ها می‌باشد. همچنین پیچیدگی زمانی برای این الگوریتم برابر با $O(nkdi)$ است که البته منظور از i تعداد تکرارهای الگوریتم برای رسیدن به جواب بهینه است. با توجه به اینکه بعد از پیش‌بینی داده‌ها توسط LSTM و GRU داده‌های جدید دارای ۱ ویژگی و ۲ خوشه هستند، پیچیدگی زمانی و محاسباتی بسیار کم است. از طرف دیگر با توجه به آنکه داده‌های مورد استفاده در این مرحله به‌صورت منظم هستند، انتخاب مرکز اولیه که یکی از مشکلات این الگوریتم است دیگر تأثیر زیادی نخواهد داشت.

۳-۲ نتایج تجربی

جهت ارزیابی روش پیشنهادی، برخی آزمایش‌ها، انجام و سپس نتایج آنها گزارش شده است. این بخش مقاله در سه زیربخش سازمان‌دهی گردیده است.

۳-۲-۱ مجموعه داده

علی‌رغم بعضی اقدامات، هنوز دو مجموعه داده NSL-KDD و KDDCUP۹۹ مشکلاتی دارند. در سال‌های اخیر، برخی از محققان مشاهده کرده‌اند که نقص این دو مجموعه داده تأثیر منفی بر عملکرد طبقه‌بندی سیستم‌های تشخیص نفوذ خواهد داشت. اول اینکه کمبود نمونه‌های حملات سطح پایین که در شبکه‌های امروزی رایج است، وجود دارد. در واقع حملات سطح پایین به‌تدریج با گذشت زمان، خصوصیت خود را از دست داده و به ترافیک عادی تبدیل می‌شوند که نوعی حمله پنهانی هستند. دوم اینکه مجموعه داده‌های آزمایشی این دو مجموعه داده، شامل بخشی از داده‌های جدید هستند و منجر به افزایش نرخ هشدار غلط می‌شوند. سوم اینکه با تغییر محیط شبکه با گذشت زمان، داده‌های زاید زیادی در این دو مجموعه داده وجود دارد که بسیار نامعقول به نظر می‌رسند. به‌طور خلاصه، اگرچه این دو مجموعه داده، سهم قابل توجهی در توسعه تشخیص نفوذ داشته‌اند، اکنون برای بازتاب فضای واقعی شبکه کافی نیستند [۴۱] و بنابراین در این پژوهش از مجموعه داده عمومی UNSW-NB۱۵ استفاده می‌شود که توسط آزمایشگاه مرکز امنیت سایبری استرالیا (ACCS) ایجاد گردیده است. رفتارهای غیرطبیعی در UNSW-NB۱۵ به ۹ دسته اصلی تقسیم می‌شوند و هر کلاس رفتار غیرطبیعی نیز به رفتارهای خاص حمله تقسیم می‌گردد. حملات موجود در مجموعه داده UNSW-NB۱۵ در جدول ۴ نشان داده شده و همچنین تعداد داده‌های استفاده‌شده جهت آموزش و آزمایش شبکه در جدول ۵ آمده است.

مجموعه داده UNSW-NB۱۵ دارای تعداد ویژگی‌های زیادی است؛ اما چون قرار است که ترافیک حمله این مجموعه داده با ترافیک نرمال مجموعه داده SDN-IoT ادغام شود در این مجموعه داده، ویژگی‌های مشترک یعنی زمان ارسال، آدرس مبدأ، آدرس مقصد، نوع پروتکل و طول بسته انتخاب می‌شوند. در مجموعه داده‌ها- با توجه به اینکه ویژگی نوع پروتکل از نوع رشته است- برای تبدیل این ویژگی به‌صورت صحیح به شکل بعدی عمل می‌شود. در ابتدا به‌جای اسامی هر کدام از پروتکل‌ها از اعداد صحیح در جدول ۶ که نگاشت نوع پروتکل در مجموعه داده‌ها به عدد صحیح را نشان می‌دهد، استفاده شده است. همچنین نقطه‌ای را که در بین آدرس‌ها و زمان‌ها وجود دارد حذف کرده و سپس با استفاده از نرمال‌سازی، همه ستون‌ها به‌روزرسانی شده است.

۳-۲-۲ معیارهای ارزیابی

در این مقاله برای ارزیابی، پارامترهای TNR، TPR، FNR، FPR و ACC، محاسبه و هر کدام از این معیارها در ادامه توصیف شده است.

- معیار TNR: معیاری است جهت نشان‌دادن نمونه‌هایی که سیستم آنها را به‌اشتباه، عضو دسته منفی تشخیص داده است؛ در حالی که این نمونه‌ها عضو دسته مثبت هستند. این معیار با کمک (۱۳) محاسبه می‌شود. در (۱۳)، TN[1] تعداد مواردی است که در آن هر دو کلاس واقعی و پیش‌بینی‌شده، منفی هستند. همچنین FP[2] تعداد مواردی است که در آن کلاس واقعی داده‌ها منفی است؛ اما کلاس پیش‌بینی‌شده، مثبت است

$$TNR = \frac{TN}{TN + FP} \qquad (۱۳)$$

1. True Negative
2. False Positive

جدول ۵: مجموعه داده‌های استفاده‌شده برای آموزش و آزمایش شبکه.

نام مجموعه داده	داده‌های طبیعی	داده‌های غیرطبیعی	تعداد داده‌های آموزشی	تعداد داده‌های آزمایشی	توزیع داده‌ها
SI-dataset	۳۰۰۰۰	۱۵۰۰۰	۳۱۵۰۰	۱۳۵۰۰	۷۰٪ آموزشی ۳۰٪ آزمایشی
SI-UNSW-dataset	۳۰۰۰۰	۱۵۰۰۰	۳۱۵۰۰	۱۳۵۰۰	۷۰٪ آموزشی ۳۰٪ آزمایشی

جدول ۶: نگاشت نوع پروتکل در مجموعه داده‌ها به عدد صحیح.

شماره تخصیص‌یافته	پروتکل	شماره تخصیص‌یافته	پروتکل	شماره تخصیص‌یافته	پروتکل
۴۳	PKIX-CRL	۲۲	DIAMETER	۱	BJNP
۴۴	RSL	۲۳	DISTCC	۲	DHCP
۴۵	RSVP	۲۴	ESP	۳	DNS
۴۶	SAB	۲۵	Gearman	۴	HTTP
۴۷	SIP	۲۶	GQUIC	۵	ICMP
۴۸	SSDP	۲۷	GSMTAP	۶	IGMPv۲
۴۹	SSL	۲۸	GTP	۷	MDNS
۵۰	SSLv۲	۲۹	HTTP/XML	۸	NTP
۵۱	STUN	۳۰	ICP	۹	SSH
۵۲	TCPCL	۳۱	IGMPv۳	۱۰	SSHv۲
۵۳	TFP over TCP	۳۲	IPv۴	۱۱	TCP
۵۴	TLSv۱	۳۳	ISAKMP	۱۲	TLSv۱.۲
۵۵	TLSv۱.۲	۳۴	LISP	۱۳	UDP
۵۶	TLSv۱.۳	۳۵	MIH	۱۴	KNXnet/IP
۵۷	UDPENCAP	۳۶	MPTCP	۱۵	ADwin Config
۵۸	WOW	۳۷	NBNS	۱۶	AJP۱۳
۵۹	XMPP/XML	۳۸	NDPS	۱۷	ASAP
۶۰	Ospf	۳۹	NVMe/TCP	۱۸	AX۴۰۰۰
۶۱	Rdp	۴۰	OCSP	۱۹	BFD Control
		۴۱	OpcUa	۲۰	BROWSER
		۴۲	PCEP	۲۱	DCERPC

- معیار پوشش $TPR(Recall)$: این معیار میزان حساسیت یک روش در تشخیص موارد مثبت است. در (۱۴) TP[1] مواردی است که کلاس واقعی و پیش‌بینی‌شده، هر دو مثبت هستند و نیز FN[2] مواردی است که در آن موارد کلاس واقعی داده‌ها مثبت است؛ اما کلاس پیش‌بینی‌شده، منفی است

$$TPR = \frac{TP}{TN + FP} \quad (۱۴)$$

- معیار FNR: این معیار نشان‌دهنده مواردی است که دسته‌بند، آنها را عضو دسته منفی تشخیص داده است؛ در حالی که آنها عضو دسته مثبت هستند. این معیار با (۱۵) محاسبه می‌شود

$$FNR = \frac{FN}{FN + TP} \quad (۱۵)$$

- معیار FPR: این معیار نشان‌دهنده مواردی است که دسته‌بند، آنها را مثبت تشخیص داده است؛ در حالی که آنها عضو دسته منفی هستند و به‌صورت (۱۶) محاسبه می‌شود

$$FPR = \frac{FP}{FP + TN} \quad (۱۶)$$

- معیار دقت یا ACC: دقت معیاری برای بررسی میزان خوب‌بودن یک الگوریتم است و به‌صورت (۱۷) محاسبه می‌شود

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (۱۷)$$

۳-۲-۳ ارزیابی روش پیشنهادی

در این بخش، کارایی روش پیشنهادی با انجام آزمایش‌های متعدد ارزیابی شده و جهت ارزیابی هر کدام از الگوریتم‌های LSTM و GRU دو آزمایش متفاوت انجام گردیده است. در آزمایش اول برای هر کدام از این الگوریتم‌ها، مجموعه داده SI در مرحله آموزش و مجموعه داده SI-UNSW در مرحله آزمون استفاده شده است. در آزمایش دو، مجموعه داده SI-UNSW در مرحله آموزش و مجموعه داده SI در مرحله آزمون مورد استفاده قرار گرفته و نتایج آزمایش اول و دوم برای الگوریتم‌های LSTM و GRU در جداول ۷ و ۸ ارائه شده است. در این جداول در آزمایش ۱ از مجموعه داده‌های مشابه، جهت آموزش و آزمون استفاده شده است؛ یعنی داده‌های استفاده‌شده جهت آموزش و آزمون از یک مجموعه داده هستند.

1. True Positive
2. False Negative

جدول ۷: نتایج آزمایش اول و دوم برای الگوریتم‌های LSTM.

نتایج آزمایش با مجموعه داده‌های آموزش	معیارهای ارزیابی LSTM						
	ACC	FPR	FNR	TPR	TNR	Precision	F۱
آزمایش با مجموعه داده SI	۹۸٫۸	۱٫۱	۱٫۱	۹۸٫۸	۹۸٫۹	۹۹٫۴	۹۹٫۱
آموزش با SI آزمایش با مجموعه داده SI-UNSW	۹۹٫۷	۰٫۵	۰٫۱	۹۹٫۹	۹۹٫۴	۹۹٫۷	۹۹٫۸
آموزش با SI-UNSW آزمایش با مجموعه داده SI	۹۷٫۱	۰٫۰۷	۴٫۱	۹۵٫۸	۹۹٫۹	۹۹٫۹	۹۷٫۸

جدول ۸: نتایج آزمایش اول و دوم برای الگوریتم‌های GRU.

نتایج آزمایش با مجموعه داده‌های آموزش	معیارهای ارزیابی GRU						
	ACC	FPR	FNR	TPR	TNR	Precision	F۱
آزمایش با مجموعه داده SI	۹۹	۱٫۱	۰٫۹	۹۹	۹۸٫۸	۹۹٫۴	۹۹٫۲
آموزش با SI آزمایش با مجموعه داده SI-UNSW	۹۹٫۹	۰	۰٫۰۷	۹۹٫۹	۱	۱	۹۹٫۹
آموزش با SI-UNSW آزمایش با مجموعه داده SI	۹۵٫۶	۱٫۶	۵٫۵	۹۴٫۵	۹۸٫۳	۹۹٫۲	۹۶٫۸

جدول ۹: نتایج مقایسه کارایی روش پیشنهادی و سایر روش‌های مشابه.

روش	ACC	TPR	FPR	مزایا و معایب
Silveira و همکاران [۳۲]	۹۳	–	۶	روش پیشنهادی با رگرسیون منطقی دقت خوبی دارد؛ اما FPR نیز بالاست.
Wani و همکاران [۳۳]	۹۵٫۹	۹۶٫۴	۷	الگوریتم MLP دقت بالایی دارد؛ اما همچنان FPR بالاست و فقط جهت شناسایی حمله DDOS است.
Hamza و همکاران [۴۰]	بالا	–	–	الگوریتم بیز و استفاده از MUD برای به‌دست‌آوردن امضای داده‌های طبیعی و غیرطبیعی و نیز کارایی بهتر از Snort ذکرنکردن مقدار ACC و FP
Hamza و همکاران [۴۱]	۹۷٫۵	۷۲٫۳	۲٫۴	استفاده از MUD که نیاز به سخت‌افزار اضافی دارد و همچنین استفاده از ۴ الگوریتم به‌صورت ترکیبی که بسیار وقت‌گیر می‌باشد؛ این در حالیست که TPR بسیار پایین است.
روش پیشنهادی	۹۹	۹۹	۱٫۱	دقت و TPR بالا، FPR بسیار پایین و مدت زمان آموزش کمتر از مزایای روش پیشنهادی است.

همان طور که نتایج آزمایش‌ها در جداول ۷ و ۸ نشان داده است، الگوریتم GRU و LSTM تقریباً دارای نتایج یکسانی بوده‌اند به‌جز زمان آموزش که GRU حدود ۳۰٪ از الگوریتم LSTM بهتر بوده است.

در آزمایش ۲، ابتدا داده‌ها با مجموعه داده SI آموزش دیده و سپس با مجموعه داده UNSW مورد آزمایش قرار گرفته‌اند. در واقع در مجموعه داده آزمون، حملات جدیدی وجود دارند که در داده‌های آموزشی، وجود نداشته‌اند. هدف اصلی در این آزمایش، بررسی شبکه برای حملات جدید بوده که نتایج، دقتی در حدود ۹۹٫۹٪ را نشان می‌دهند.

در آزمایش ۳، ابتدا داده‌ها با مجموعه داده SI-UNSW آموزش دیده و سپس با مجموعه داده SI مورد آزمون قرار گرفته‌اند. همچنین نتایج ارزیابی الگوریتم‌ها نشان می‌دهند که برعکس آزمایش ۲، در اینجا دقت کاهش یافته و *FNR* نیز افزایش داشته است. در واقع دلیل این امر آن است که ترافیک غیرنرمالی که در مرحله آزمایش مورد استفاده قرار گرفته است از همان شبکه‌ای که داده‌های نرمال به‌دست آمده‌اند به وجود آمده؛ اما در مرحله آموزش مورد استفاده قرار نگرفته و این موضوع باعث شده که شبکه، این داده‌ها را تا حدی شبیه داده‌های نرمال شناسایی کند؛ هرچند دقت و بقیه معیارهای ارزیابی هنوز مناسب بوده و جهت شناسایی حملاتی که در مرحله آموزش استفاده نشده‌اند بسیار عالی عمل کرده‌اند.

٤- مقایسه روش پیشنهادی با سایر روش‌ها

در این بخش، مقایسه کارایی روش پیشنهادی و سایر روش‌های قبلی مشابه انجام شده و سپس نتایج مقایسه در جدول ۹ ارائه گردیده است. در این جدول، مقایسه‌ای بین روش پیشنهادی و روش‌های دیگر انجام شده و همان طور که مشاهده می‌شود، روش Zhang و همکاران [۴۱] با استفاده از مشخصات موجود در MUD هر دستگاه، سیستم تشخیص نفوذ آموزش می‌بینند. سپس در مرحله دیگری با استفاده از ویژگی‌های موجود در سرآیند جریان‌ها، سیستم تشخیص نفوذ، آموزش مجدد می‌بینند. در واقع ابتدا در مرحله اول با PCA تعداد ویژگی‌ها کاهش می‌یابد و سپس با استفاده از X-means فرایند کلاس‌بندی انجام شده و سپس خروجی هر دو مرحله در اختیار الگوریتم‌های boundary detection و زنجیره مارکوف برای خوشه‌بندی نهایی قرار می‌گیرد. همان طور که مشاهده می‌شود این روش از دو مرحله استفاده می‌کند که برای محیط SDN-IoT مناسب نیست؛ زیرا دستگاه‌های اینترنت اشیا در مدت زمان بسیار کوتاه، ترافیک بسیار زیادی تولید می‌کنند و سوئیچ‌های شبکه، درخواست‌های زیادی را برای کنترلر می‌فرستند که با دو مرحله‌کردن فرایند شناسایی با تأخیر مواجه می‌شود. از طرفی دیگر هنوز استاندارد اضافه‌کردن MUD به دستگاه‌های اینترنت اشیا رواج پیدا نکرده است.

٥- نتیجه‌گیری و پژوهش‌های آتی

در این مقاله، یک روش جدید ترکیبی با بهره‌مندی از مزایای استراتژی به‌کاررفته در الگوریتم‌های یادگیری ماشین LSTM و GRU در راستای بهبود عملکرد یک سیستم تشخیص نفوذ در محیط SDN-IoT، پیشنهاد و سه ویژگی مهم در روش پیشنهادی بیان شده است:

- روش پیشنهادی با درنظرگرفتن ترافیک بسیار زیاد دستگاه‌های اینترنت اشیا، زمان پاسخ سیستم تشخیص نفوذ به سوئیچ‌ها و ... پیشنهاد شده است.
- استفاده از مجموعه داده مرتبط با ترافیک SDN-IoT: مرور ادبیات نشان می‌دهد که در اغلب پژوهش‌های موجود SDN-IoT، توجه خاصی نسبت به استفاده از مجموعه داده‌های به‌روزنشده است.
- قابلیت پیاده‌سازی راحت و بدون نیاز به سخت‌افزارهای خاص: متاسفانه در بسیاری از پژوهش‌ها نسبت به سهولت پیاده‌سازی روش‌ها به‌صورت عملی، توجه نشده است.

نتایج ارزیابی نشان می‌دهند که روش پیشنهادی، نتایج خوبی در زمینه تشخیص نفوذ در محیط SDN-IoT دارد. عدم وجود یک مجموعه داده استاندارد جهت آموزش روش‌های پیشنهادی در این زمینه، چالشی است که می‌تواند پژوهشگران را سردرگم کند؛ بنابراین ارائه یک مجموعه داده استاندارد متناسب با SDN-IoT می‌تواند انجام پژوهش در این زمینه را تسهیل نماید. همچنین برای کارهای آتی می‌توان از الگوریتم‌های دیگر یادگیری ماشین مانند RBM، DNN، CNN و ... استفاده کرد. در واقع الگوریتم‌های بازگشتی در آموزش وقت‌گیر هستند؛ لذا ارائه یک روش که بتواند مدت زمان آموزش را کاهش دهد می‌تواند مفید واقع شود.

مراجع

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645-1660, Sep. 2013.

[2] R. Kushwah, P. K. Batra, and A. Jain, "Internet of things architectural elements, challenges and future directions," in *Proc. 6th Int. Conf. on Signal Processing and Communication, ICSC'20,* 5 pp., Noida, India, 5-7 Mar. 2020.

[3] A. Koohang, C. S. Sargent, J. H. Nord, and J. Paliszkiewicz, "Internet of Things (IoT): from awareness to continued use," *International J. of Information Management*, vol. 62, Article ID: 102442-, Feb. 2020.

[4] U. Farooq, N. Tariq, M. Asim, T. Baker, and A. Al-Shamma'a, "Machine learning and the internet of things security: solutions and open challenges," *J. of Parallel and Distributed Computing*, vol. 162, pp. 89-104, Apr. 2022.

[5] P. Mishra, A. Biswal, S. Garg, R. Lu, M. Tiwary, and D. Puthal, "Software defined internet of things security: properties, state of the art, and future research," *IEEE Wireless Communications*, vol. 27, no. 3, pp. 10-16, Jun. 2020.

[6] A. E. Omolara, *et al.*, "The internet of things security: a survey encompassing unexplored areas and new insights," *Computers & Security*, vol. 112, Article ID: 102494, Jan. 2022.

[7] X. Guo, H. Lin, Z. Li, and M. Peng, "Deep-reinforcement-learning-based QoS-aware secure routing for SDN-IoT," *IEEE Internet of Things J.*, vol. 7, no. 7, pp. 6242-6251, Dec. 2019.

[8] P. K. Sharma, J. H. Park, Y. S. Jeong, and J. H. Park, "SHSec: SDN based secure smart home network architecture for internet of things," *Mobile Networks and Applications*, vol. 24, pp. 913-924 2018.

[9] S. Rathore, B. W. Kwon, and J. H. Park, "BlockSecIoTNet: blockchain-based decentralized security architecture for IoT network," *J. of Network and Computer Applications*, vol. 143, pp. 167-177, Oct. 2019.

[10] H. Honar Pajooh, M. Rashid, F. Alam, and S. Demidenko, "Multi-layer blockchain-based security architecture for internet of things," *Sensors*, vol. 21, no. 3, Article ID: 772, 2021.

[11] A. Dawoud, S. Shahristani, and C. Raun, "Deep learning and software-defined networks: towards secure IoT architecture," *Internet of Things*, vol. 3-4, pp. 82-89, Oct. 2018.

[12] N. McKeown, *et al.*, "OpenFlow: enabling innovation in campus networks," *ACM Sigcomm Computer Communication Review*, vol. 38, no. 2, pp. 69-74, Apr. 2008.

[13] M. Babiker Mohamed, *et al*., "A comprehensive survey on secure software-defined network for the Internet of Things," *Trans. on Emerging Telecommunications Technologies*, vol. 33, no. 1, Article ID: e4391, Jan. 2022.

[14] D. Sovilj, P. Budnarain, S. Sanner, G. Salmon, and M. Rao, "A comparative evaluation of unsupervised deep architectures for intrusion detection in sequential data streams," *Expert Systems with Applications*, vol. 159, Article ID: 113577, Nov. 2020.

[15] C. W. Chang, C. Y. Chang, and Y. Y. Lin, "A hybrid CNN and LSTM-based deep learning model for abnormal behavior detection," *Multimedia Tools and Applications*, vol. 81, no. 2, pp. 1-19, Apr. 2022.

[16] K. Smagulova and A. P. James, "A survey on LSTM memristive neural network architectures and applications," *the European Physical J. Special Topics*, vol. 228, no. 10, pp. 2313-2324, Oct. 2019.

[17] N. Alqudah, M. Y. Qussai, "Machine learning for traffic analysis: a review," Procedia Computer Science, vol. 170, pp. 911-916, 2020.

[18] S. K. Tayyaba, M. A. Shah, O. A. Khan, and A. W. Ahmed, "Software defined network (SDN) based internet of things (IoT) a road ahead," in *Proc. of the International Conf. on Future Networks and Distributed Systems*, Article ID: 15, 8 pp., Cambridge, UK, 19-20, Jul. 2017.

[19] R. Bhatia, S. Benno, J. Esteban, T. V. Lakshman, and J. Grogan, "Unsupervised machine learning for network-centric anomaly detection in IoT," in *Proc. of the 3rd ACM Context Workshop on Big Data, Machine Learning and Artificial Intelligence for Data Communication Networks*, pp. 42-48, Orlando, FL, USA, 9-9 Dec .2019.

[20] D. Arellanes and K. K. Lau, "Evaluating IoT service composition mechanisms for the scalability of IoT systems," *Future Generation Computer Systems*, vol. 108, pp. 827-848, Mar. 2020.

[21] S. Bera, S. Misra, and A. V. Vasilakos, "Software-defined networking for internet of things: a survey," *IEEE Internet of Things J.*, vol. 4, no. 6, pp. 1994-2008, Aug. 2017.

[22] M. Singh and G. Baranwal, "Quality of Service (QoS) in internet of things," in *Proc. 3rd Int. Conf. on Internet of Things: Smart Innovation and Usages, IoT-SIU'18,* 6 pp., Bhimtal, India, 23-24 Feb. 2018.

[23] T. A. Nguyen, D. Min, and E. Choi, "A hierarchical modeling and analysis framework for availability and security quantification of IoT infrastructures," *Electronics*, vol. 9, no. 1, Article ID: 155, Jan. 2020.

[24] R. Swami, M. Dave, and V. Ranga, "Voting-based intrusion detection framework for securing software-defined networks," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 24, Article ID: e5927, 25 Dec. 2020.

[25] I. Rabet, *et al.*, "SDMob: SDN-based mobility management for IoT networks," *J. of Sensor and Actuator Networks*, vol. 11, no. 1, Article ID: 8, 2022.

[26] B. Alzahrani and N. Fotiou, "Enhancing internet of things security using software-defined networking," *J. of Systems Architecture*, vol. 110, Article ID: pp. 101779, Nov. 2020.

[27] A. Hamza, H. H. Gharakheili, and V. Sivaraman, "Combining MUD policies with SDN for IoT intrusion detection," in *Proc. of the Workshop on IoT Security and Privacy*, 7 pp., Budapest, Hungary, 20-20, Aug. 2018.

[28] G. Shravanya, N. H. Swati, R. P. Rustagi, and O. Sharma, "Securing distributed SDN controller network from induced DoS attacks," in *Proc., IEEE International Conf. on Cloud Computing in Emerging Markets, CCEM'19*, pp. 9-16, Bengaluru, India, 19-20 Sept. 2019.

[29] A. Hamza, H. H. Gharakheili, T. A. Benson, and V. Sivaraman, "Detecting volumetric attacks on lot devices via SDN-based monitoring of mud activity," in *Proc. of the ACM Symp. on SDN Research*, pp. 36-48, San Jose, CA, USA, 3-4 Apr. 2019.

[30] C. Xu, H. Lin, Y. Wu, X. Guo, and W. Lin, "An SDN FV-based DDoS defense technology for smart cities," *IEEE Access*, vol. 7, pp. 137856-137874, 2019.

[31] O. Salman, I. H. Elhajj, A. Chehab, and A. Kayssi, "A machine learning based framework for IoT device identification and abnormal traffic detection," *Trans. on Emerging Telecommunications Technologies*, vol. 33, no. 3, Article ID: e3743, Mar. 2022.

[32] F. A. F. Silveira, F. Lima-Filho, F. S. D. Silva, A. D. M. B. Junior, and L. F. Silveira, "Smart detection-IoT: a DDoS sensor system for Internet of Things," in *Proc. Int. Conf. on Systems, Signals, and Image Processing, IWSSIP'20,* pp. 343-348, Niteroi, Brazil, 1-3 Jul. 2020.

[33] A. Wani and S. Revathi, "DDoS detection and alleviation in IoT using SDN (SDN IoT-DDoS-DA)," *J. of the Institution of Engineers (India): Series B*, vol. 101, no. 3, pp. 117-128, Apr. 2020.

[34] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proenca, "Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment," *IEEE Access*, vol. 8, pp. 83765-83781, 2020.

[35] Y. Meidan, *et al.*, "A novel approach for detecting vulnerable IoT devices connected behind a home NAT," *Computers & Security*, vol. 97, Article ID: 101968, Oct. 2020.

[36] S. H. Khan, A. R. Arko, and A. Chakrabarty, "Anomaly detection in IoT using machine learning," In: S. Misra, A. K. Tyagi, V. Piuri, and L. Garg (Eds.), *Artificial Intelligence for Cloud and Edge Computing Springer, Chap*, pp. 237-254, 2022.

[37] M. Abdullahi, *et al*., "Detecting cybersecurity attacks in internet of things using artificial intelligence methods: a systematic literature review," *Electronics*, vol. 11, no. 2, Article ID: e3743198, 2022.

[38] M. V. Assis, L. F. Carvalho, J. Lloret, and M. L. Proença Jr, "A GRU deep learning system against attacks in software defined networks," *J. of Network and Computer Applications*, vol. 177, Article ID: 102942, 2021.

[39] A. S. Alshra'a, A. Farhat, and J. Seitz, "Deep learning algorithms for detecting denial of service attacks in software-defined networks," *Procedia Computer Science*, vol. 191pp. 254-263, 2021.
[40] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451-461, Feb. 2003.
[41] J. Zhang, Y. Ling, X. Fu, X. Yang, G. Xiong, and R. Zhang, "Model of the intrusion detection system based on the integration of spatial-temporal features," *Computers & Security*, vol. 89, Article ID: 101681, Feb. 2020.

ذکریا رئیسی در سال ۱۳۹۴ مدرک کارشناسی مهندسی کامپیوتر خود را از دانشگاه سیستان و بلوچستان و در سال ۱۳۹۹ مدرک کارشناسی ارشد مهندسی فناوری اطلاعات خود را از دانشگاه یزد دریافت نمود. زمینه‌های علمی مورد علاقه نام‌برده شامل موضوعاتی مانند شبکه‌های کامپیوتری، امنیت شبکه، سیستم‌های توزیعی و مدیریت شبکه‌های کامپیوتری می‌باشد.

فضل‌الله ادیب‌نیا در سال ۱۳۶۵ مدرک کارشناسی مهندسی کامپیوتر خود را از دانشگاه صنعتی اصفهان و در سال ۱۳۶۸ مدرک کارشناسی‌ارشد مهندسی برق خود را از دانشگاه صنعتی شریف و در سال ۱۳۷۸ مدرک دکترای مهندسی کامپیوتر خود را از دانشگاه برمن آلمان دریافت نمود. و هم‌اکنون دانشیار دانشکده مهندسی کامپیوتر دانشگاه یزد می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شبکه‌های کامپیوتری، مکان‌یابی در شبکه‌ها، امنیت رایانه و شبکه، زنجیره بلوکی و سیستم‌های توزیعی است.

مهدی یزدیان دهکردی مدرک کارشناسی مهندسی کامپیوتر گرایش نرم‌افزار را در سال ۱۳۸۵ از دانشگاه یزد و مدرک کارشناسی ارشد و دکترای خود را به‌ترتیب در سال‌های ۱۳۸۸ و ۱۳۹۴ در رشته مهندسی کامپیوتر گرایش هوش مصنوعی از دانشگاه شیراز اخذ کرد. وی از سال ۱۳۹۴ در دانشکده مهندسی کامپیوتر دانشگاه یزد مشغول به فعالیت گردید و در حال حاضر عضو هیأت علمی این دانشکده می‌باشد. زمینه‌های پژوهشی مورد علاقه ایشان بینایی ماشین، یادگیری ماشین، یادگیری عمیق و تحلیل داده است.