

A Hybrid Intrusion Detection System for RPL-Based IoT Networks Using Decision Tree and CNN-BiLSTM Models

Mohammad Fazeli^{1*}, Mohsen Raji², Mohammad Mojtaba Fazeli³

¹ Department of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran

² Department of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran

³ Department of Computer Engineering, Islamic Azad University, Sepidan Branch, Sepidan, Iran

Received: ..., Revised: ..., Accepted: ...

Paper type: ...

Abstract

The Routing Protocol for Low-Power and Lossy Networks (RPL) is a widely adopted standard in Internet of Things (IoT) networks, enabling efficient and stable communication. However, its vulnerability to cyberattacks poses a significant threat. Traditional intrusion detection methods, although effective, are computationally complex and resource-intensive, rendering them unsuitable for resource-constrained IoT devices. This paper presents a lightweight, two-layer intrusion detection system (IDS) for RPL-based IoT networks. The first layer employs a decision tree model to detect attacks, offering accurate and rapid detection with minimal computational resources. The second layer utilizes a hybrid Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) model to classify attack types. This combination enables automatic extraction of deep features from network traffic data and learning of temporal dependencies and complex attack patterns. Experimental results on the ROUT-4-2023 dataset, comprising four common RPL attacks (Blackhole, Flooding, Version Number, and Rank Attacks), demonstrate the proposed method's superior performance, achieving an overall accuracy of 97% and precision, recall, and F1-scores of approximately 96%. The attack detection time is significantly reduced to 0.062 seconds. These findings confirm the efficacy and high speed of the proposed system in RPL-based IoT environments, making it a suitable solution for securing resource-constrained IoT networks.

Keywords: Intrusion Detection, Internet of Things (IoT), IoT Security, RPL Protocol, RPL attacks, Lightweight Detection Model, Layered Security Architecture, Attack Classification, Decision Tree (DT), CNN-BiLSTM.

** Corresponding Author's email:*
mohammad.fazeli@hafez.shirazu.ac.ir

سال ... ، شماره ... ، فصل ... ، صفحه ... الی ...

یک سامانه تشخیص نفوذ ترکیبی برای شبکه‌های اینترنت اشیاء مبتنی بر RPL با بهره‌گیری از مدل‌های درخت تصمیم و CNN-BiLSTM

محمد فاضلی^{۱*}، محسن راجی^۲، محمد مجتبی فاضلی^۳

^۱ دانشکده مهندسی برق و کامپیوتر، دانشگاه شیراز، شیراز، ایران

^۲ دانشکده مهندسی برق و کامپیوتر، دانشگاه شیراز، شیراز، ایران

^۳ دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد سپیدان، سپیدان، ایران

تاریخ دریافت: ... تاریخ بازبینی: ... تاریخ پذیرش: ...

نوع مقاله: ...

چکیده

پروتکل مسیریابی برای شبکه‌های کم‌مصرف و پراتلاف (RPL) ^۱ به عنوان یک پروتکل استاندارد شده در مسیریابی شبکه‌های اینترنت اشیاء شناخته شده و به طور گسترده ای در این شبکه‌ها استفاده می شود. اما این پروتکل در برابر حملات سایبری آسیب پذیر است. روش‌های سنتی تشخیص نفوذ، به دلیل پیچیدگی محاسباتی و مصرف بالای منابع، برای شبکه‌های اینترنت اشیاء مناسب نیستند. در این پژوهش، یک سامانه تشخیص نفوذ دو لایه‌ای سبک‌وزن برای شناسایی و طبقه‌بندی حملات در شبکه‌های اینترنت اشیاء مبتنی بر RPL ارائه شده است. در لایه اول، از یک مدل درخت تصمیم ^۲ برای تشخیص وقوع حمله استفاده می‌شود. این مدل، عملکردی دقیق و سریع دارد و نیاز به منابع محاسباتی زیادی ندارد. در لایه دوم، برای طبقه‌بندی دقیق نوع حمله، از ترکیبی از شبکه عصبی کانولوشنی و حافظه بلند-کوتاه مدت دوطرفه ^۳ (CNN-BiLSTM) استفاده می‌شود. این ترکیب، توانایی استخراج خودکار ویژگی‌های عمیق از داده‌های ترافیک شبکه و یادگیری وابستگی‌های زمانی و الگوهای پیچیده حملات را دارد. نتایج آزمایش‌ها بر روی مجموعه داده ROUT-4-2023 با چهار نوع چهار نوع حمله متداول RPL از جمله سیاه‌چاله ^۴، سیل ^۵، شماره نسخه ^۶ و کاهش رتبه ^۷ نشان می‌دهد که روش پیشنهادی با صحت کلی ۹۷٪ و مقادیر دقت، بازخوانی و امتیاز F1 حدود ۹۶٪، عملکرد بهتری نسبت به روش‌های موجود دارد. زمان شناسایی حمله نیز به حدود ۰.۰۶۲ ثانیه کاهش یافته است. نتایج به دست آمده، کارایی مناسب و سرعت بالای سامانه پیشنهادی را در محیط‌های اینترنت اشیاء مبتنی بر RPL نشان می‌دهد.

کلیدواژه‌گان: تشخیص نفوذ، اینترنت اشیاء (IoT)، امنیت اینترنت اشیاء، مدل سبک وزن تشخیص نفوذ، معماری امنیتی لایه‌ای.

* رایانامه نویسنده مسؤول:

mohammad.fazeli@hafez.shirazu.ac.ir

^۳ Convolutional Neural Network-Bidirectional Long Short-Term Memory

^۴ Blackhole

^۵ Flooding

^۶ Version Number

^۷ Decreased Rank

^۱ Routing Protocol for Low-Power and Lossy Networks

^۲ Decision Tree

۱- مقدمه

پشتیبان⁷ (SVM)، درخت تصمیم (DT) و بیز ساده (NB⁸) و همچنین از مدل‌های یادگیری عمیق نظیر شبکه عصبی کانولوشنی (CNN⁹)، شبکه حافظه بلند-کوتاه مدت (LSTM¹⁰) در سیستم‌های تشخیص نفوذ در شبکه‌های IoT استفاده شده است. از سوی دیگر، با توجه به تفاوت بنیادی که پروتکل مسیریابی RPL دارد، برای شبکه‌های مبتنی بر RPL سامانه‌های تشخیص نفوذ خاصی با بهره‌گیری از مدل‌های مختلف یادگیری ماشین و عمیق مانند جنگل تصادفی (RF¹¹)، شبکه عصبی چندلایه (MLP)، ماشین بردار پشتیبان (SVM)، شبکه LSTM، و مدل ترکیبی CNN-LSTM ارائه شده است.

با وجود دستیابی به دقت نسبتاً مطلوب در تشخیص انواع حملات، بسیاری از پژوهش‌های پیشین از مدل‌های پیچیده و سنگین مانند شبکه‌های عمیق چندلایه یا مدل‌های ترکیبی با پارامترهای زیاد استفاده کرده‌اند که به منابع محاسباتی و انرژی بالایی نیاز دارند و در نتیجه، اجرای آن‌ها در محیط‌های واقعی IoT با منابع محدود عملاً دشوار یا غیر ممکن است.

در این پژوهش، یک سامانه‌ی تشخیص نفوذ دولایه‌ای سبک‌وزن و کارآمد برای شناسایی و طبقه‌بندی حملات در شبکه‌های اینترنت اشیا مبتنی بر RPL ارائه شده است. در لایه‌ی نخست از سیستم تشخیص نفوذ پیشنهادی، به منظور عملکرد دقیق، سریع و بدون نیاز به منابع محاسباتی زیاد از یک مدل درخت تصمیم برای تشخیص وقوع حمله استفاده می‌شود. سپس در لایه دوم، ترکیبی از شبکه عصبی کانولوشنی و حافظه بلند-کوتاه مدت دوطرفه (CNN-BiLSTM) به کار گرفته می‌شود تا نوع حمله مشخص شود. علت انتخاب این مدل به دلیل توانایی بالای آن در استخراج خودکار ویژگی‌های عمیق از داده‌های ترافیک شبکه توسط لایه‌های CNN و یادگیری وابستگی‌های زمانی و الگوهای پیچیده حملات از طریق

اینترنت اشیا (IoT)¹ به عنوان یکی از فناوری‌های تحول‌آفرین قرن حاضر، با اتصال میلیاردها دستگاه به شبکه‌های ارتباطی، امکان ایجاد سیستم‌های هوشمند در حوزه‌هایی نظیر خانه‌های هوشمند، حمل‌ونقل، کشاورزی دقیق و مراقبت‌های بهداشتی را فراهم کرده است. این فناوری با بهره‌گیری از دستگاه‌های متصل، داده‌های عظیمی تولید می‌کند که تحلیل آن‌ها می‌تواند به بهبود کیفیت زندگی و بهینه‌سازی فرآیندها منجر شود [۱]. با این حال، محدودیت‌های منابع در دستگاه‌های IoT، از جمله توان پردازشی پایین، حافظه محدود و مصرف انرژی بالا، چالش‌های جدی را در طراحی و پیاده‌سازی این سیستم‌ها ایجاد کرده است.

یکی از پروتکل‌های کلیدی در شبکه‌های IoT، پروتکل مسیریابی RPL است که توسط گروه ویژه مهندسی اینترنت (IETF)² برای شبکه‌های کم‌مصرف و با تلفات بالا (LLN)³ استاندارد شده است. RPL با استفاده از ساختار گراف جهت‌دار بدون چرخه (DODAG⁴) و معیارهای بهینه‌سازی مانند مصرف انرژی یا تعداد گام‌ها، مسیریابی کارآمدی را در این شبکه‌ها فراهم می‌کند [۲]. با این وجود، این پروتکل در برابر حملات سایبری نظیر حمله سیاه‌چاله، سیل‌آسا و دستکاری رتبه⁵ آسیب‌پذیر است. این حملات می‌توانند عملکرد شبکه را مختل کرده و امنیت آن را به خطر اندازند. روش‌های سنتی مانند رمزنگاری سنگین یا سیستم‌های تشخیص نفوذ (IDS)⁶ به دلیل نیاز به منابع محاسباتی بالا، برای محیط‌های محدود منابع IoT ناکارآمد هستند [۳].

برای غلبه بر این چالش‌ها، رویکردهای نوینی مانند یادگیری ماشین و یادگیری عمیق برای تشخیص نفوذ در شبکه‌های IoT پیشنهاد شده‌اند. به عنوان نمونه، از مدل‌های سنتی مانند ماشین بردار

¹ The Internet of Things

² Internet Engineering Task Force

³ Low-Power and Lossy Networks

⁴ Destination Oriented Directed Acyclic Graph

⁵ Rank Manipulation

⁶ Intrusion Detection System

⁷ Support Vector Machine

⁸ Naive Bayes

⁹ Convolutional Neural Network

¹⁰ Long Short-Term Memory

¹¹ Random Forest

۱-۲- مدل‌های یادگیری عمیق برای تشخیص نفوذ

در سیستم‌های تشخیص نفوذ برای شبکه‌های IoT، مدل‌های یادگیری عمیق نتایج چشمگیری داشته‌اند.

در [۵]، از شبکه‌های حافظه بلند-کوتاه‌مدت (LSTM) برای شناسایی حملات DoS^۱ در محیط‌های IoT استفاده شده است. این مدل توانسته وابستگی‌های زمانی داده‌ها را به خوبی یاد بگیرد و دقت تشخیص بالاتری نسبت به روش‌های سنتی ارائه دهد، اما همچنان به منابع محاسباتی زیادی برای آموزش و استنتاج نیاز دارد.

در ادامه، [۶]، یک مدل شبکه عصبی پیچشی زمانی (TCNN^۲) همراه با مهندسی ویژگی پیشرفته را معرفی کرد که توانست ویژگی‌های حملات را با دقت بالاتری استخراج کند و عملکرد مناسبی ارائه دهد، اما همچنان نیاز به توان محاسباتی بالایی دارد.

همچنین در [۷]، یک مدل یادگیری عمیق فضایی-زمانی نیمه‌نظارتی^۳ معرفی شده که نیاز به داده‌های برچسب‌گذاری شده را کاهش می‌دهد و تعمیم‌پذیری بهتری نسبت به مدل‌های نظارتی کامل ارائه می‌کند. با این حال، پیچیدگی بالای مدل، پیاده‌سازی آن در گره‌های کم‌منبع LLN را محدود می‌سازد.

به‌طور مشابه، در [۸]، ترکیبی از شبکه‌های کانولوشنی (CNN) و حافظه بلند-کوتاه‌مدت دوطرفه (BiLSTM) برای طبقه‌بندی حملات RPL در شبکه‌های هوشمند IoT پیشنهاد شد. این مدل با بهره‌گیری همزمان از ویژگی‌های مکانی و زمانی، دقتی بیش از ۹۰٪ به دست آورد اما دقت و زمان محاسبه همپنان چالش است.

۲-۲- مدل‌های یادگیری گروهی و ترکیبی برای

تشخیص تشخیص نفوذ

چندین پژوهش اخیر بر استفاده از مدل‌های یادگیری گروهی^۴ تمرکز کرده‌اند تا دقت و کارایی تشخیص حملات RPL را افزایش دهند.

در [۹]، یک مدل گروهی شامل درخت تصمیم، بیز ساده، شبکه عصبی مصنوعی و نزول گرادینت تصادفی معرفی شد که دقت حدود ۸۶٪ به دست آورد، اما دقت و زمان محاسبه همپنان چالش است.

در [۱۰]، یک چارچوب یادگیری گروهی مبتنی بر انتخاب ویژگی ژنتیکی پیشنهاد شد که با ترکیب طبقه‌بندی‌کننده‌هایی مانند جنگل تصادفی و AdaBoost^۵ توانست نرخ تشخیص بالایی در شناسایی حملاتی مانند سیل^۶ و دستکاری نسخه RPL به دست

BiLSTM بوده، که همین امر موجب افزایش دقت طبقه‌بندی، کاهش نرخ هشدار کاذب و بهبود شناسایی حملات ناشناخته و پیچیده می‌شود. علاوه بر این، به دلیل ساختار بهینه و وزن‌های فشرده، این مدل نسبت به بسیاری از مدل‌های عمیق پیشرفته‌تر به مصرف محاسباتی و حافظه کمتری نیاز دارد که باعث می‌شود برای استفاده در سیستم‌های IoT با منابع محدود مناسب‌تر از سایر مدل‌ها باشد. عملکرد مدل پیشنهادی با استفاده از مجموعه داده ROUT-4-2023 ارزیابی می‌شود که شامل چهار نوع حمله متداول RPL از جمله سیاه‌چاله، سیل آسا، شماره‌نسخه و کاهش‌رتبه ارزیابی شده است. نتایج آزمایش‌ها نشان می‌دهد که روش پیشنهادی با دستیابی به صحت کلی ۹۷٪ و مقادیر دقت، بازخوانی و امتیاز F1 حدود ۹۶٪، عملکرد بهتری نسبت به روش‌های موجود دارد. همچنین، زمان شناسایی حمله به حدود ۰.۰۶۲ ثانیه کاهش یافته است. نتایج به دست آمده بیانگر کارایی مناسب و سرعت بالای سامانه پیشنهادی در محیط‌های اینترنت اشیا مبتنی بر RPL با دستگاه‌های با محدودیت منابع است.

ادامه این مقاله به شرح زیر سازماندهی شده است: بخش ۲ کارهای مرتبط با امنیت RPL و تشخیص حمله مبتنی بر یادگیری ماشین و یادگیری عمیق را مرور می‌کند. بخش ۳ مدل پیشنهادی و تنظیمات مربوطه و مجموعه داده استفاده شده را معرفی می‌کند، بخش ۴ نتایج تجربی را ارائه می‌دهد و عملکرد چارچوب تشخیص را بر اساس مجموعه داده ROUT-4-2023 ارزیابی می‌کند. در نهایت، بخش ۵ این مقاله را به پایان می‌رساند و خلاصه نتیجه حاصل از این پژوهش را ارائه می‌دهد.

۲- کارهای مرتبط

در سال‌های اخیر، به دلیل گسترش دستگاه‌ها در در شبکه‌های اینترنت اشیا (IoT) و وابستگی روزافزون آن‌ها به پروتکل‌های مسیریابی، تحقیقات گسترده‌ای در زمینه تشخیص نفوذ و شناسایی حملات در شبکه‌های مبتنی بر RPL انجام شده است. این مطالعات عمدتاً بر دو محور اصلی متمرکز بوده‌اند:

- (۱) مدل‌های یادگیری عمیق برای تشخیص نفوذ
 - (۲) روش‌های یادگیری گروهی و بهینه‌سازی ترکیبی برای بهبود دقت و کارایی.
- در ادامه به کارهای انجام گرفته در هر محور پرداخته می‌شود.

^۴ Ensemble Learning

^۵ Adaptive Boosting

^۶ DODAG Information Solicitation

^۱ Denial of Service

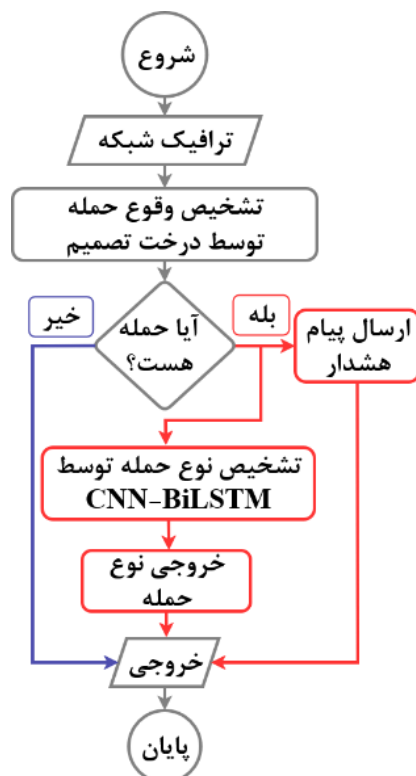
^۲ Temporal Convolutional neural network

^۳ Semi-supervised Spatiotemporal DL

پیش‌پردازش داده‌ها، آموزش مدل و فرآیند ارزیابی به تفصیل شرح داده می‌شوند.

۱-۳- نمای کلی از روش پیشنهادی

شکل ۱ نمای کلی روش پیشنهادی تشخیص نفوذ را نمایش می‌دهد. در این مدل، ابتدا ترافیک شبکه وارد یک مدل سبک‌وزن درخت تصمیم (DT) می‌شود که با سرعت بالا ترافیک ورودی را تجزیه و تحلیل کرده و نمونه‌های مشکوک را شناسایی و علامت‌گذاری می‌کند. سپس در صورت شناسایی رفتار غیرعادی، یک پیام هشدار به خروجی ارسال می‌شود. این لایه به عنوان یک فیلتر اولیه، نقش مؤثری در حذف بخش عمده‌ای از ترافیک‌های بی‌خطر با حداقل سر بار محاسباتی ایفا می‌کند. در مرحله بعد، ترافیک علامت‌گذاری شده به لایه دوم ارسال می‌شود. این لایه از ترکیب یک شبکه عصبی کانولوشنی و حافظه بلند-کوتاه مدت دوطرفه (CNN-BiLSTM) تشکیل شده است که تجزیه و تحلیل عمیق‌تری انجام داده و نوع دقیق حمله RPL را شناسایی می‌نماید. در نهایت، خروجی سیستم شامل تشخیص نهایی و نوع حمله شناسایی شده خواهد بود.



شکل ۱. روند کار مدل پیشنهادی

آورد، اما پیچیدگی محاسباتی بالای آن، کاربرد عملی‌اش را در دستگاه‌های IoT با منابع محدود دشوار می‌کند.

در [۱۱]، یک مدل ترکیبی بهینه‌سازی شده مبتنی بر الگوریتم آموزش-یادگیری (TLBO^۱) و ماشین یادگیری شدید (ELM^۲) معرفی شد که با انتخاب بهینه ویژگی‌ها، سرعت آموزش و دقت بالایی را به دست آورد، اما اجرای آن روی گره‌های کم‌منبع مورد بررسی کامل قرار نگرفته است.

در نهایت، [۱۲]، یک سیستم تشخیص نفوذ مبتنی بر یادگیری گروهی برای شبکه‌های RPL ارائه می‌کند که با ترکیب چندین الگوریتم، نسبت به مدل‌های تک‌لایه‌ای، دقت بالاتری در شناسایی انواع حملات ارائه می‌دهد.

به طور کلی، هرچند مدل‌های ترکیبی و مبتنی بر یادگیری عمیق دقت و انعطاف‌پذیری بالایی در تشخیص نفوذ ارائه داده‌اند، اما بیشتر این روش‌ها محدودیت‌های منابع دستگاه‌های IoT را نادیده گرفته‌اند. این غفلت موجب شده است که بسیاری از مدل‌های موجود، علی‌رغم عملکرد مناسب در محیط‌های شبیه‌سازی، در گره‌های واقعی و کم‌منبع IoT قابل پیاده‌سازی نباشند. در این پژوهش، با هدف رفع این چالش و پر کردن این خلأ، یک چارچوب تشخیص نفوذ دولایه و سبک‌وزن ارائه می‌شود که با ترکیب هوشمند الگوریتم‌های کلاسیک و عمیق، توازن مؤثر میان دقت تشخیص و کارایی محاسباتی برقرار می‌سازد. در لایه‌ی نخست، از یک درخت تصمیم سبک‌وزن برای شناسایی سریع ناهنجاری‌ها استفاده می‌شود و در لایه‌ی دوم، ترکیب شبکه عصبی کانولوشنی و حافظه بلند-کوتاه مدت دوطرفه (CNN-BiLSTM) به منظور طبقه‌بندی دقیق انواع حملات RPL به کار گرفته می‌شود. چارچوب پیشنهادی با حفظ دقت بالا و کاهش چشمگیر پیچیدگی محاسباتی، امکان پیاده‌سازی مقیاس‌پذیر و کارآمد در شبکه‌های IoT مبتنی بر RPL را فراهم می‌سازد.

۳- روش پیشنهادی دولایه‌ای برای تشخیص نفوذ در

شبکه‌های اینترنت اشیا مبتنی بر RPL

در این بخش، مدل تشخیص نفوذ دولایه‌ای و ترکیبی پیشنهادی برای شبکه‌های اینترنت اشیا مبتنی بر پروتکل مسیریابی RPL با بهره‌گیری از یادگیری ماشین و یادگیری عمیق معرفی می‌شود. به این منظور، ابتدا دید کلی از این سیستم تشخیص ارائه می‌گردد و سپس، توضیح مفصلی درباره‌ی تنظیمات آزمایشی و مجموعه داده مورد استفاده در این پژوهش ارائه خواهد شد. در ادامه نیز مراحل

² Extreme Learning Machine

¹ Teaching learning-based optimization

۳-۲- تنظیمات آزمایشی

روش پیشنهادی در پایتون با استفاده از TensorFlow 2.10.1 پیاده‌سازی و بر روی یک رایانه شخصی مجهز به پردازنده گرافیکی NVIDIA GeForce GTX 1660 Ti با ۶ گیگابایت VRAM اجرا شد. این سیستم از CUDA 11.8 با درایور NVIDIA نسخه ۲۲.۰۶ برای فعال کردن شتابدهی GPU استفاده کرد. محیط بر روی سیستم عامل ویندوز اجرا شد.

۳-۳- مجموعه داده (ROUT-4-2023)

ما برای آزمایش مدل پیشنهادی از مجموعه داده‌ی ارائه شده در [۴] استفاده کردیم که یک مجموعه ترافیک شبکه IoT با تمرکز بر حملات RPL است. این مجموعه داده‌ها از طریق شبیه‌سازی‌های Cooja از یک شبکه 6LoWPAN تحت عملکرد عادی و چهار سناریوی حمله RPL ایجاد شده است. جدول ۱ تعداد رکوردهای جریان مربوط به هر نوع حمله RPL را در مجموعه داده‌ی [۴] خلاصه می‌کند.

جدول ۱. اندازه برچسب مجموعه داده ROUT-4-2023 [۴]

انواع حمله ها	تعداد ردیف‌ها
حمله سیاه‌چاله (Blackhole)	۴۰۴۱۳۴
حمله شماره نسخه (DODAG)	۴۶۸۰۶۰
حمله سیل آسا (Flooding)	۳۹۸۷۸۲
حمله کاهش رتبه (Rank)	۳۶۸۹۹۹

این مجموعه شامل حدود ۱.۶۴ میلیون رکورد جریان است که حدود ۱.۰۶ میلیون از آنها با برچسب "عادی" و ۰.۵۸ میلیون با برچسب "حمله" مشخص شده‌اند. چهار کلاس حمله عبارتند از:

- سیاه‌چاله (ارسال انتخابی) - گره‌های مخرب تمام بسته‌ها را رها می‌کنند
 - سیل آسا - گره‌ها شبکه را با پیام‌های کنترلی بیش از حد مثلاً (DIS) غرق می‌کنند تا آن را تحت الشعاع قرار دهند
 - حمله شماره نسخه - DODAG یک مهاجم شماره نسخه RPL را جعل می‌کند تا به روزرسانی‌های مسیریابی را مختل کند
 - رتبه کاهش یافته - یک گره رتبه‌ای غیرقانونی پایین را برای جذب و مسیریابی نادرست ترافیک تبلیغ می‌کند.
- با توجه به تعداد مثال‌های موجود در این مجموعه داده، هر حمله به خوبی نمایش داده شده است، اگرچه ترافیک عادی کلی تقریباً

دو برابر حجم حملات است (~۶۵٪ عادی در مقابل ۳۵٪ حمله)، که می‌تواند به عنوان نزدیکی این مجموعه داده به فعالیت عادی شبکه نیز در نظر گرفته شود. هر نمونه داده (یک بازه زمانی از ترافیک شبکه) توسط ۱۶ ویژگی توصیف می‌شود، همانطور که در جدول ۲ نشان داده شده است.

جدول ۲. ویژگی‌های مجموعه داده ROUT-4-2023 [۴]

نام ویژگی	توضیحات
TIME	زمان شبیه‌سازی
SOURCE	گره مبدأ IP آدرس
DESTINATION	گره مقصد IP آدرس
LENGTH	طول بسته
INFO	اطلاعات بسته
TR	نرخ ارسال (در هر ۱۰۰۰ میلی‌ثانیه)
RR	نرخ دریافت (در هر ۱۰۰۰ میلی‌ثانیه)
TAT	میانگین زمان ارسال
RAT	میانگین زمان دریافت
TPC	تعداد بسته‌های ارسالی (در هر ثانیه)
RPC	تعداد بسته‌های دریافتی (در هر ثانیه)
TTT	مجموع زمان ارسال
TRT	مجموع زمان دریافت
DAO	تعداد بسته‌های DAO
DIS	تعداد بسته‌های DIS
DIO	تعداد بسته‌های DIO
CATEGORY	نوع حمله یا وضعیت عادی
LABEL	برچسب عادی/مخرب

۳-۴- پیش پردازش داده

قبل از آموزش، مجموعه داده‌ها چندین مرحله پیش‌پردازش را طی می‌کنند تا از سازگاری و آمادگی مدل اطمینان حاصل شود. شکل ۲ این مراحل را نشان می‌دهد که شامل موارد زیر است:

منصفانه‌ای از قابلیت تعمیم مدل را تضمین می‌کند. پس از اتمام مراحل پیش‌پردازش، برای کاهش هزینه‌ی محاسباتی و افزایش تعمیم‌پذیری، فرآیند مهندسی ویژگی دومرحله‌ای طراحی شده است که مرحله‌ی نخست آن در **الگوریتم ۲** خلاصه می‌شود. در این مرحله، یک مدل درخت تصمیم اولیه روی مجموعه‌ی

الگوریتم ۱: اصلاح عدم تعادل کلاس SMOTE

- 1: Input $\leftarrow$ minority instances (M_T); oversampling percentage ($N\%$); Value for nearest neighbors (k); granularity parameter (α)
- 2: Output $\leftarrow$ synthetic data (S_D)
- 3: $S^* \leftarrow$ the minority class instances (M_T)
- 4: $y \leftarrow$ mechanism to perform feature ranking
- 5: $c \leftarrow$ granularity parameter (y, α)
- 6: $P \leftarrow$ maximum value of cas 'r'
- 7: for $i \in T$ do
- 8: for $i' \in i' \neq i$ do
- 9: IMOWAD ($x_i, x_{j'}, w$); distance measure operator
- 10: end for
- 11: for $k \leftarrow 1$ do N
- 12: $x_k \leftarrow$ selects random samples from M_T
- 13: $x'_k \leftarrow$ perform interpolation to generate synthetic data
- 14: $T^* \leftarrow$ synthetic data added to S^*
- 15: $S_D \leftarrow$ remove specific sample x_k from P
- 16: end for
- 17: end for

آموزشی آموزش داده می‌شود و اهمیت هر ویژگی بر اساس میزان کاهش ناخالصی در گره‌های درخت محاسبه می‌گردد. برای هر گره تقسیم t که در آن ویژگی f به کار رفته است، کاهش ناخالصی طبق فرمول (۱) و در خط ۳ از الگوریتم ۲ تعریف می‌شود:

$$\Delta i_t = I_t - \frac{N_{left}}{N_t} \cdot I_{left} - \frac{N_{right}}{N_t} \cdot I_{right} \quad (1)$$

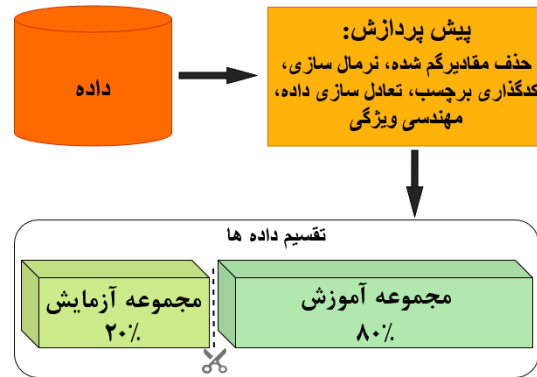
در این رابطه I_t میزان ناخالصی (مانند Gini یا Entropy) در گره t ، N_t تعداد نمونه‌های آن گره، و I_{left}, I_{right} مقادیر ناخالصی در گره‌های فرزند چپ و راست هستند.

اهمیت کلی ویژگی f از تجمیع کاهش ناخالصی‌ها در تمام گره‌هایی که ویژگی در آن‌ها ظاهر شده است به دست می‌آید و طبق فرمول (۲) و در ادامه خط ۳ الگوریتم ۲ به صورت زیر محاسبه می‌شود:

$$FI(f) = \frac{\sum_{t \in splits(f)} w_t \cdot \Delta i_t}{\sum_j \sum_{t \in splits(j)} w_t \cdot \Delta i_t} \quad (2)$$

در اینجا w_t وزن گره متناسب با تعداد نمونه‌های آن و $splits(f)$ مجموعه‌ی گره‌هایی است که در آن ویژگی f ظاهر شده است. پس از محاسبه‌ی اهمیت تمام ویژگی‌ها در خط ۴ الگوریتم ۲، مجموعه‌ی از مقادیر آستانه (τ) استخراج و به صورت نزولی مرتب می‌شود. این خروجی شامل دقت پایه، اهمیت ویژگی‌ها و لیست آستانه‌ها است و به عنوان ورودی الگوریتم ۳ مورد استفاده قرار می‌گیرد.

- **پاکسازی داده‌ها:** حذف مقادیر گمشده و ورودی‌های ناسازگار از مجموعه داده‌ها. نرمال سازی: مقیاس‌بندی ویژگی‌های عددی به یک محدوده یکنواخت برای اطمینان از آموزش پایدار.



شکل ۲. مرحله پیش‌پردازش داده

- **کدگذاری برچسب:** تبدیل متغیرهای دسته‌بندی شده و برچسب‌های کلاس به قالب عددی.

- **متعادل سازی داده‌ها:** برای اصلاح عدم تعادل کلاس در مجموعه داده‌های نفوذ، از نسخه‌ی بهبودیافته‌ی روش SMOTE¹ استفاده شده است **الگوریتم ۱**. در این الگوریتم، نمونه‌های کلاس اقلیت (M_T)، درصد نمونه‌برداری بیش از حد ($N\%$)، تعداد نزدیکترین همسایه (k) و پارامتر دانه‌بندی (α) به عنوان ورودی در نظر گرفته می‌شوند.

ابتدا مجموعه‌ی اقلیت (S^*)، مکانیزم رتبه‌بندی ویژگی‌ها (y)، پارامتر دانه‌بندی (c) و مقدار بیشینه‌ی (r) مقداردهی می‌شوند. سپس برای هر نمونه‌ی اقلیت در مجموعه‌ی (T) فواصل وزنی میان نمونه‌ها با استفاده از عملگر IMOWAD ($x_i, x_{j'}, w$) محاسبه می‌گردد تا همسایگان مناسب شناسایی شوند. با درونیابی بین نمونه‌ی انتخاب‌شده و همسایه‌ی آن، نمونه‌های مصنوعی جدید x'_k تولید شده و به مجموعه‌ی اقلیت افزوده می‌شوند. در صورت نیاز، برخی از نمونه‌های قدیمی برای حفظ تعادل حذف می‌گردند.

این فرآیند تا زمانی ادامه می‌یابد که مجموعه‌ی داده‌ی مصنوعی S_D به تعادل کامل برسد. نتیجه‌ی نهایی، مجموعه‌ای از نمونه‌های مصنوعی واقع‌گرایانه است که باعث بهبود عملکرد مدل در مواجهه با داده‌های نامتوازن می‌شود.

- **تقسیم داده‌ها:** پس از پیش‌پردازش و متعادل‌سازی داده‌ها، مجموعه داده به دو بخش آموزشی و آزمایشی تقسیم می‌شود تا ارزیابی مدل به صورت مستقل انجام گیرد. این تقسیم‌بندی، ارزیابی

¹ Synthetic Minority Over-sampling Technique

الگوریتم ۳: انتخاب τ با توجه به δ

Input $\leftarrow (X_{train}, Y_{train}), (X_{test}, Y_{test}), baseline_acc, \tau_list, FI(f), allowed \delta$

- 1: $best_tau \leftarrow None$
- 2: $best_acc \leftarrow 0$
- 3: **for** each τ in τ_list :

$selected_features \leftarrow \{f \mid FI(f) > \tau\}$

Train DT_model2 on selected features

$acc \leftarrow$ Accuracy on $(X_{test_selected}, Y_{test})$

If $(baseline_acc - acc) \leq \delta$ **then**:

$best_tau \leftarrow \tau$

$best_acc \leftarrow acc$

break
- 4: **If** $best_tau = None$ **then**:

$best_tau \leftarrow 0$

$best_acc \leftarrow baseline_acc$

Output $\leftarrow best_tau, best_acc, selected_features$

الگوریتم ۲: محاسبه اهمیت ویژگی و تولید لیست τ

Input $\leftarrow$ Training set (X_{train}, Y_{train}) , Testing set, Decision Tree parameter (max – depth)

- 1: Train initial Decision Tree DT_{model} on all features,
- 2: Compute baseline accuracy $baseline_acc$
- 3: Compute feature importance for each feature:

Compute feature importance (f) **for** each feature :

$$\Delta i_t = I_t - \frac{N_{left}}{N_t} \cdot I_{left} - \frac{N_{right}}{N_t} \cdot I_{right}$$

$$FI(f) = \frac{\sum_{t \in (f) splits} w_t \cdot \Delta i_t}{\sum_j \sum_{t \in splits(j)} w_t \cdot \Delta i_t}$$

- 4: τ_list (sorted descending) $\leftarrow$ Extract unique importance values

Output $\leftarrow baseline_acc, FI(f)$ for all features, τ_list

کاهش یافته و کارایی آن در محیط‌های واقعی IoT به‌طور قابل توجهی بهبود می‌یابد.

۵-۳- لایه اول سیستم تشخیص نفوذ: درخت تصمیم

در لایه‌ی نخست مدل پیشنهادی، از یک طبقه‌بندی‌کننده‌ی درخت تصمیم‌گیری دودویی بهره گرفته شده است که به‌طور خودکار و با سرعت بالا، نمونه‌های ترافیک به دو دسته‌ی عادی و حمله تفکیک می‌شوند. در این پژوهش، از درخت تصمیم‌گیری استاندارد CART با معیار ناخالصی جینی بهره گرفته شده است که بر روی برچسب‌های دودویی کل مجموعه‌ی آموزشی آموزش داده شده است. استفاده می‌شود. این درخت دارای عمق متوسط (عمق ۱۴) بوده و به‌منظور جلوگیری از بیش‌برازش، هرس شده است. انتخاب درخت تصمیم برای این لایه بر اساس ویژگی‌های کلیدی آن از جمله توضیح‌پذیری بالا و هزینه‌ی محاسباتی پایین آن و امکان استقرار در دستگاه‌های لبه یا دروازه‌های اینترنت اشیا صورت گرفته است.

- ورودی: بردار ویژگی ۱۶ بعدی یک جریان شبکه.

- خروجی: یک برچسب دودویی (عادی یا حمله)

مزیت رویکرد دولایه از همین مرحله قابل مشاهده است؛ درخت تصمیم قادر است بخش عمده‌ای از ترافیک عادی را با سرعت زیاد و کمترین هزینه‌ی محاسباتی فیلتر کرده و تنها جریان‌هایی را که به‌عنوان حمله شناسایی شده‌اند، به لایه‌ی دوم ارسال کند. این فرآیند منجر به کاهش بار محاسباتی مدل یادگیری عمیق در لایه‌ی دوم می‌شود.

در طراحی این بخش، پارامترهای درخت تصمیم به گونه‌ای تنظیم شده‌اند که فراخوانی برای نمونه‌های حمله در سطح بالایی حفظ شود. به‌منظور اطمینان از شناسایی تمامی فعالیت‌های مخرب، وجود

الگوریتم ۳ در این بخش که مرحله‌ی دوم مهندسی ویژگی است، مقدار آستانه‌ی بهینه τ بر اساس کنترل افت دقت تعیین می‌شود. برای هر مقدار آستانه τ مجموعه‌ی ویژگی‌های منتخب طبق فرمول (۳) و در خط ۳ الگوریتم به‌صورت زیر تعریف می‌شود:

$$S(\tau) = \{f \mid FI(f) > \tau\} \quad (3)$$

مدل درخت تصمیم جدید با استفاده از این ویژگی‌ها آموزش داده می‌شود و دقت حاصل از اعتبارسنجی K-بخشی^۱ طبق فرمول (۴) محاسبه می‌گردد:

$$Acc(\tau) = \frac{1}{K_{cv}} \sum_{i=1}^{K_{cv}} Acc_i \quad (4)$$

اگر اختلاف دقت نسبت به مقدار پایه کمتر از حد مجاز δ باشد، آستانه‌ی τ پذیرفته می‌شود. شرط پذیرش τ در خط ۳ الگوریتم طبق فرمول (۵) به‌صورت زیر تعریف می‌شود:

$$BaselineAcc - Acc(\tau) \leq \delta \quad (5)$$

که در آن $BaselineAcc$ همان دقت پایه است. بالاترین مقدار τ که این شرط را برآورده سازد، به‌عنوان τ بهینه انتخاب می‌شود. در صورتی که هیچ آستانه‌ای این شرط را برآورده نکند، مقدار صفر در نظر گرفته می‌شود (خط ۴). در نهایت، خروجی این مرحله شامل τ ، مجموعه ویژگی‌های منتخب و دقت حاصل روی داده‌های آزمون است.

این طراحی دو مرحله‌ای ضمن حفظ توضیح‌پذیری و کنترل افت دقت، باعث انتخاب مجموعه‌ای بهینه و فشرده از ویژگی‌ها می‌شود. چرا که مرحله اول اهمیت نسبی ویژگی‌ها را آشکار می‌سازد و مرحله دوم با کنترل افت دقت، مجموعه‌ای بهینه و کوچک‌تری از ویژگی‌ها را برای مدل انتخاب می‌کند. در نتیجه، پیچیدگی مدل

¹ K-fold cross-validation

padding='same')

این لایه الگوهای محلی کوتاه مدت مانند پیک ها و پرش های ناگهانی را استخراج می کند.

• BatchNormalization()

این لایه به پایداری و تسریع فرآیند آموزش کمک می کند.

• Conv1D(64, kernel_size=3, activation='relu', padding='same')

این لایه وظیفه استخراج ویژگی های سطح بالاتر را دارد.

• MaxPooling1D(pool_size=2)

با کاهش طول دنباله، این لایه الگوهای مهم را برجسته می کند.

• Dropout(0.3)

برای جلوگیری از بیش برآزش (Overfitting) در این مرحله استفاده می شود.

❖ بخش BiLSTM (مدل سازی زمینه زمانی)

این بخش به مدل سازی وابستگی های زمانی در هر دو جهت (پیش رو و پس رو) اختصاص یافته است.

پیکربندی شامل موارد زیر است:

• Bidirectional(LSTM(64,activation='relu',return_sequences=True))

• این لایه وابستگی های زمانی را در هر دو جهت (گذشته به آینده و آینده به گذشته) مدل می کند و دنباله را برای لایه بعدی حفظ می کند.

• Bidirectional(LSTM(32,return_sequences=False))

• این لایه زمینه زمانی را به یک بردار خلاصه فشرده می کند.

• Dropout(0.3)

• برای کاهش بیش برآزش پس از مدل سازی بازگشتی به کار می رود.

❖ سر خروجی (طبقه بندی)

• Dense(64) → ReLU → Dropout(0.3)

• این لایه ها ویژگی های استخراج شده را برای تصمیم گیری نهایی ترکیب و آماده می کنند.

• Dense(num_classes) with softmax

• این لایه نهایی توزیع احتمالات نهایی را برای کلاس های مختلف مانند (نرمال، حمله سیاه چاله، حمله شماره نسخه، حمله سیل آسا، حمله کاهش رتبه) تولید می کند.

❖ دلایل طراحی معماری ترکیبی

• لایه های کانولوشنی وظیفه استخراج الگوهای محلی و گذرا (مانند پیک های ناگهانی در شمارش بسته ها) را بر عهده دارند و با فشرده سازی طول دنباله، هزینه محاسباتی کاهش می یابد.

• BiLSTM با مدل سازی وابستگی های زمانی در دو جهت،

تعداد محدودی مثبت کاذب مجاز در نظر گرفته شده است.

پس از تصمیم گیری درخت، روند پردازش به صورت زیر ادامه می یابد:

• اگر نمونه در گروه «عادی» قرار گیرد، به عنوان بی خطر

در نظر گرفته شده و پردازش بیشتری بر روی آن انجام نمی شود.

• اگر نمونه در گروه «حمله» قرار گیرد، یک پیام هشدار

تولید می شود و ویژگی های همان نمونه یا دنباله ای از بردارهای ویژگی اخیر برای تشخیص نوع حمله به لایه دوم منتقل می شوند.

این سازوکار موجب افزایش کارایی کلی سیستم می شود؛ زیرا تمرکز درخت تصمیم بر روی نمونه های مشکوک باعث صرفه جویی در منابع محاسباتی شده و نرخ منفی های کاذب را کاهش می دهد. حتی در شرایطی که درخت بیش از حد حساس عمل کند و برخی نمونه های عادی را به عنوان حمله برچسب گذاری کند، این موارد در لایه دوم توسط طبقه بندی کننده ی دقیق تر بازنسازایی و اصلاح می شوند.

۳-۶- لایه دوم سیستم تشخیص نفوذ: طبقه بندی کننده

ترکیبی CNN-BiLSTM

در لایه دوم، از یک طبقه بندی کننده ی سبک وزن مبتنی بر ترکیب شبکه ی عصبی کانولوشنی (CNN) و حافظه ی طولانی مدت دوطرفه (BiLSTM) برای طبقه بندی ناهنجاری ها در ترافیک شبکه استفاده شده است. این مدل به صورت اختصاصی برای تمایز میان چهار نوع حمله ی RPL و ترافیک عادی طراحی شده و با هدف استخراج الگوهای محلی و مدل سازی وابستگی های زمانی در هر دو جهت (گذشته و آینده) پیکربندی شده است.

۱-۳-۶ پیکربندی و نقش هر لایه

این مدل از چند بخش اصلی تشکیل شده است که هر یک وظیفه ی مشخصی در فرآیند یادگیری و طبقه بندی دارند:

❖ نمایش ورودی

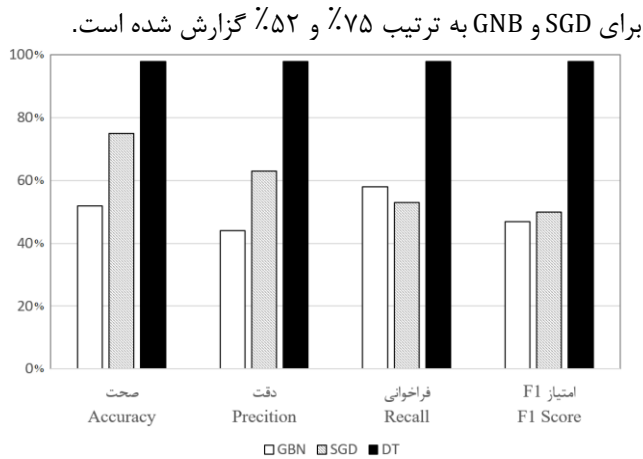
دنباله کوتاه از بردارهای ویژگی اخیر: ورودی مدل، یک دنباله از بردارهای ویژگی اخیر است که به صورت (طول_دنباله × تعداد_ویژگی ها) شکل دهی شده اند. این ورودی به عنوان یک سیگنال ۱ بعدی به لایه Conv1D منتقل می شود.

❖ بخش کانولوشن (استخراج ویژگی های موضعی)

هدف این بخش، شناسایی الگوهای محلی و کوتاه مدت در داده های ترافیکی است.

در این بخش از ساختار زیر استفاده شده است:

• Conv1D(128, kernel_size=3, activation='relu',



شکل ۳. نتایج پارامترهای ارزیابی برای سه مدل کلاس دودویی.

GNB: بیز ساده گاوسی، SGD: نزول گرادیان تصادفی، DT: درخت

تصمیم

جدول ۳ معیارهای عملکرد طبقه‌بندی دودویی مدل درخت تصمیم را نشان می‌دهد. همان‌طور که مشاهده می‌شود، الگوریتم درخت تصمیم توانایی قابل توجهی در شناسایی الگوهای پیچیده و غیرخطی ناهنجاری‌ها ارائه کرده است، قابلیت‌هایی که در مدل‌های خطی یا ساده‌تر مشاهده نمی‌شود. برتری عملکردی ذکر شده، انتخاب درخت تصمیم به‌عنوان مکانیزم فیلترینگ سریع در لایه اول را توجیه می‌کند؛ لایه‌ای که مسئولیت شناسایی سریع و قابل اعتماد ترافیک مشکوک و در عین حال کاهش نرخ خطا را بر عهده دارد.

۲-۴- ارزیابی لایه دوم: طبقه‌بندی حمله چندکلاسه با

استفاده از CNN-BiLSTM

در لایه دوم، تنها نمونه‌هایی که توسط لایه اول به عنوان «غیرعادی» شناسایی شده‌اند، پردازش شده‌اند و در چهار دسته حمله شامل سیاه‌چاله، سیل‌آسا، شماره نسخه و کاهش رتبه طبقه‌بندی شده‌اند. همچنین هشدارهای کاذب تولید شده در لایه اول بازشناسی و به رده ترافیک عادی تخصیص داده شده‌اند.

جدول ۴. عملکرد طبقه‌بندی چندکلاسه مدل-DT+CNN

BiLSTM در پنج کلاس روی مجموعه داده

	صحت	فراخوانی	امتیاز F1
سیاه‌چاله	۰.۸۶	۰.۸۸	۰.۸۷
شماره نسخه	۱.۰۰	۱.۰۰	۱.۰۰
سیل‌آسا	۱.۰۰	۱.۰۰	۱.۰۰
نرمال	۰.۹۸	۰.۹۸	۰.۹۸
کاهش رتبه	۰.۹۶	۰.۹۳	۰.۹۴

نتایج حاصل از ارزیابی عملکرد طبقه‌بندی چندکلاسه مدل

امکان تمایز میان نوسانات (نویز) گذرا از الگوی حمله را فراهم می‌کند.

- انتخاب ساختار ترکیبی: CNN-BiLSTM مبتنی بر نیاز به یک طبقه‌بند سبک‌وزن، سریع و مقاوم در برابر نویز انجام شد. این ترکیب به کاهش تعداد پارامترها و هزینه‌ی محاسباتی نسبت به معماری‌های عمیق‌تر منجر شده و در نتیجه زمان آموزش کوتاه‌تر و سرعت استنتاج بالاتر حاصل می‌شود. از این‌رو، استفاده از این مدل در محیط‌های اینترنت اشیا (IoT) با منابع محاسباتی محدود، رویکردی کارآمد و بهینه محسوب می‌شود.

۴- نتایج تجربی

در این بخش، عملکرد مدل پیشنهادی دولایه شامل یک درخت تصمیم دودویی و مدل CNN-BiLSTM بر روی مجموعه داده‌ی ROUT-4-2023، که شامل ترافیک عادی و چهار نوع حمله‌ی RPL است، ارزیابی شده است. عملکرد مدل با استفاده از معیارهای استاندارد صحت، دقت تشخیص، فراخوانی و امتیاز F1 سنجیده شده است و تحلیل ماتریس درهم‌ریختگی برای بررسی دقیق عملکرد انجام شده است. علاوه بر این، زمان آموزش و سرعت استنتاج را برای نشان دادن مناسب بودن مدل برای برنامه‌های

جدول ۳. معیارهای عملکرد طبقه‌بندی دودویی مدل درخت

تصمیم	صحت	فراخوانی	امتیاز F1
نرمال (۰)	۱.۰۰	۰.۹۷	۰.۹۹
حمله (۱)	۰.۹	۱.۰۰	۰.۹۷
صحت	۰.۹۸		
میانگین وزن دار	۰.۹۸	۰.۹۹	۰.۹۸
میانگین ماکرو	۰.۹۸	۰.۹۸	۰.۹۸

کاربرد اینترنت اشیا در زمان واقعی اندازه‌گیری شده است.

۱-۴- ارزیابی لایه اول: تشخیص نفوذ دودویی با

استفاده از مدل‌های کلاسیک یادگیری ماشین

در گام نخست، هدف، تفکیک ترافیک عادی از ترافیک غیرعادی بوده است. برای این منظور، عملکرد سه الگوریتم کلاسیک یادگیری ماشین شامل بیز ساده گاوسی (GNB^۱)، نزول گرادیان تصادفی (SGD^۲) و درخت تصمیم (DT) مقایسه شده است. همان‌طور که در شکل ۳ نشان داده شده است، مدل درخت تصمیم در تمامی معیارها عملکردی برتر نسبت به دو روش دیگر ارائه کرده است. به طور مشخص، صحت تشخیص DT بیش از ۹۸٪ بوده، در حالی که

^۲ Stochastic Gradient Descent

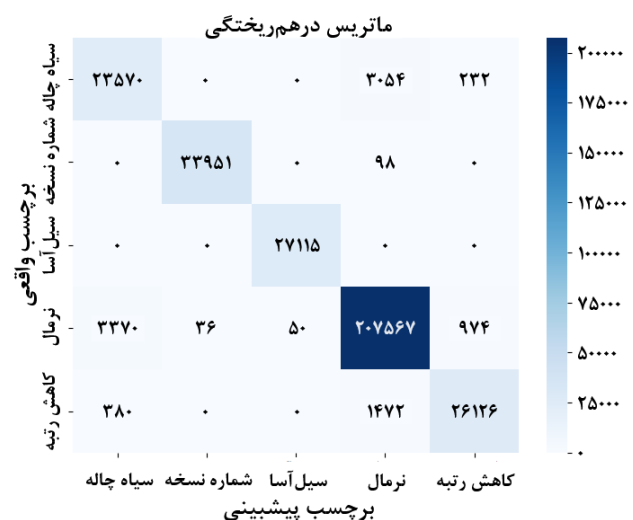
^۱ Gaussian Naive Bayes

DT+CNN-BiLSTM در پنج کلاس روی مجموعه داده در جدول ۴ ارائه شده است. همان طور که مشاهده می شود، مدل پیشنهادی توانسته است حملات سیل آسا و شماره نسخه را با صحت و یادآوری کامل شناسایی کند. همچنین، حملات کاهش رتبه و سیاه چاله با امتیاز F1 به ترتیب برابر ۰.۹۴ و ۰.۸۷ طبقه بندی شده اند، به طوری که تنها سردرگمی جزئی ناشی از شباهت های توپولوژیکی مشاهده می شود. علاوه بر این، برای ترافیک عادی، F1 برابر ۰.۹۸ حفظ شده است، که نشان دهنده توانایی بازیابی مؤثر نمونه های خوش خیم طبقه بندی نشده از لایه اول است.

این نتایج بیانگر آن هستند که معماری CNN-BiLSTM در لایه دوم توانایی تحلیل عمیق و دقیق الگوهای حمله را ارائه کرده است و عملکرد مبتنی بر قواعد درخت تصمیم در لایه اول را تکمیل می کند. ترکیب این دو لایه یک خط لوله قوی برای تشخیص سریع ناهنجاری ها و طبقه بندی دقیق انواع حمله ایجاد می کند.

۳-۴- تحلیل ماتریس درهم ریختگی

به منظور درک دقیق تر از نحوه عملکرد مدل در فرآیند طبقه بندی، ماتریس درهم ریختگی پیش بینی های آن بر روی داده های آزمایشی مورد بررسی قرار گرفته است. شکل ۴. این ماتریس بر اساس مقایسه برچسب های واقعی و مقادیر پیش بینی شده ساخته شده و عملکرد مدل را در تفکیک پنج کلاس زیر نشان می دهد:



شکل ۴. ماتریس درهم ریختگی مدل پیشنهادی روی داده های تست.

- کلاس ۰: حمله سیاه چاله
- کلاس ۱: حمله شماره نسخه
- کلاس ۲: حمله سیل آسا
- کلاس ۳: ترافیک عادی

• کلاس ۴: حمله کاهش رتبه

همان طور که در شکل ۴ مشاهده می شود، بالاترین مقادیر در قطر اصلی ماتریس قرار دارند که نشان دهنده دقت بالای مدل در طبقه بندی صحیح نمونه ها است. به طور خاص:

- از مجموع نمونه های واقعی کلاس ۰ سیاه چاله ، تعداد ۲۳۵۷۰ مورد به درستی شناسایی شدند. تنها ۳۳۷۰ نمونه به اشتباه به عنوان نرمال و ۳۸۰ نمونه به عنوان کاهش رتبه طبقه بندی شدند.

- کلاس ۱ شماره نسخه ، با دقت بسیار بالا شناسایی شده و تقریباً تمام نمونه ها ۳۳۹۵۱ به درستی طبقه بندی شدند. تنها ۳۶ نمونه به اشتباه در کلاس نرمال قرار گرفتند.

- کلاس ۲ سیل آسا ، نیز عملکرد عالی داشته و تمامی ۲۷۱۱۵ نمونه به طور صحیح شناسایی شده اند، بدون هیچ خطای قابل توجه.

- در کلاس ۳ نرمال، تعداد ۲۰۷۵۶۷ نمونه درست طبقه بندی شدند. با این حال ۳۰۵۴ نمونه به اشتباه به عنوان سیاه چاله، ۹۸ نمونه به عنوان شماره نسخه ، ۰ نمونه به عنوان سیل آسا و ۱۴۷۲ نمونه به عنوان کاهش رتبه شناسایی شدند.

- کلاس ۴ کاهش رتبه ، نیز با موفقیت بالا تشخیص داده شد؛ ۲۶۱۲۶ نمونه به درستی شناسایی شدند، اما ۹۷۴ نمونه به اشتباه به عنوان نرمال و ۲۳۲ نمونه به عنوان سیاه چاله طبقه بندی شدند.

مدل دولایه پیشنهادی (DT + CNN-BiLSTM) توانسته است ترافیک عادی و انواع حملات RPL را با صحت بالا و نرخ خطای بسیار پایین طبقه بندی کند. اکثر خطاها میان کلاس هایی رخ داده اند که از نظر ویژگی های رفتاری به هم نزدیک هستند، اما حتی این خطاها نیز در سطح بسیار محدود باقی مانده اند. طراحی دو مرحله ای مدل کمک کرده است تا خطاهای بحرانی، به ویژه اشتباه گرفتن ترافیک عادی با حملات، به حداقل برسد؛ موضوعی که برای کاربرد در محیط های واقعی اینترنت اشیا اهمیت زیادی دارد.

۴-۴- مقایسه با سایر مدل ها

به منظور ارزیابی اثربخشی رویکرد پیشنهادی، یک تحلیل مقایسه ای جامع در برابر مجموعه ای از روش های پیشرفته مطرح در مقالات انجام شد. شکل ۵ عملکرد پنج مدل CNN، LSTM، FFNN، و BiLSTM را در چهار معیار کلیدی ارزیابی شامل صحت^۱، دقت پیش بینی^۲، فراخوانی^۳ و امتیاز

³ Recall

¹ Accuracy

² Precision

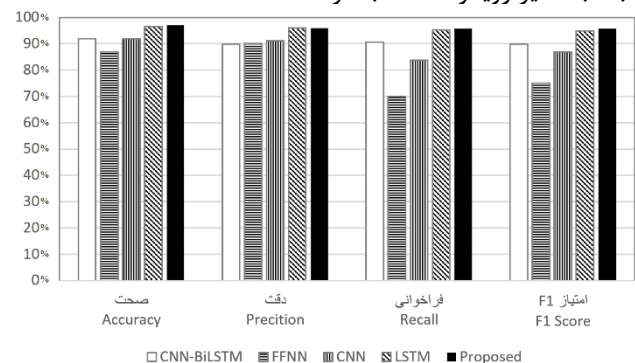
جدول ۵. مقایسه معیارهای عملکرد برای مدل‌های اخیر روی مجموعه داده‌های تشخیص نفوذ مبتنی بر اینترنت اشیا و RPL

مدل	دیتاست	حوزه	صحت	امتیاز F1	دقت	فراخوانی	دولایه
TLBO-EIM[11]	NSL-KDD	Network	88.21	-	-	-	X
SS-DeepID[7]	CIC-IDS	Network	88.33	88.85	88.78	88.91	X
RNN-BPTT[6]	BoT-IoT	IoT	87.20	-	-	-	X
CNN-BiLSTM[8]	IoTID20	IoT	91.87	89.98	89.91	90.70	X
CNN-BiLSTM[8]	N-BaIoT	IoT	90.12	89.97	89.87	90.21	X
DT-NB-ANN-SGD[9]	ROUT-4-2023	RPL-based IoT	86.21	78.21	-	69.54	X
FFNN	ROUT-4-2023	RPL-based IoT	87.00	75.00	90.00	70.00	X
CNN	ROUT-4-2023	RPL-based IoT	92.00	87.00	91.20	83.80	X
LSTM	ROUT-4-2023	RPL-based IoT	97.00	95.35	95.50	95.20	X
Proposed	ROUT-4-2023	RPL-based IoT	97.00	95.80	96.00	95.90	✓

F1^۱ نشان می‌دهد. همان‌گونه که در شکل مشاهده می‌شود، مدل

پیشنهادی ما در تمامی معیارهای ارزیابی، عملکردی برتر نسبت به سایر مدل‌های پایه ارائه می‌دهد.

در حالی که مدل‌های LSTM و CNN-BiLSTM نیز عملکرد قابل توجهی در طبقه‌بندی نشان می‌دهند، استفاده از معماری ترکیبی درخت تصمیم و CNN-BiLSTM منجر به دستیابی به تعادلی بهینه میان صحت و فراخوانی شده و در نتیجه بالاترین مقدار امتیاز F1 را نسبت به سایر رویکردها کسب کرده است.



شکل ۵. مقایسه عملکرد مدل‌های پایه و پیشنهادی بر اساس چهار

معیار کلیدی

جدول ۵ مقایسه جامعتری میان مدل پیشنهادی و چندین مدل مختلف مبتنی بر یادگیری عمیق را که بر روی مجموعه داده‌های مرتبط با شبکه و اینترنت اشیا ارزیابی شده‌اند، ارائه می‌دهد. این مقایسه شامل مدل‌هایی است که به‌طور خاص بر شناسایی حملات مبتنی بر RPL تمرکز داشته‌اند. نتایج نشان می‌دهد که مدل پیشنهادی در مقایسه با سایر روش‌ها از نظر صحت تشخیص، نرخ هشدار کاذب و توانایی شناسایی حملات پیچیده عملکرد بهتری ارائه می‌دهد و می‌تواند به‌عنوان یک چارچوب کارآمد و بهینه در حوزه تشخیص حملات در شبکه‌های اینترنت اشیا مورد استفاده قرار گیرد.

این نتایج تأیید می‌کند که روش پیشنهادی به بهترین تعادل کلی

^۱ F1-Score

میان صحت، یادآوری و کارایی محاسباتی دست می‌یابد، به‌ویژه در سناریوهای اینترنت اشیا مبتنی بر RPL طراحی سلسله مراتبی ارائه شده، نقش شبکه CNN-BiLSTM را ساده‌تر کرده و موجب می‌شود که این مدل با تمرکز بر نمونه‌های پیچیده‌تر، صحت کلی و یادآوری مختص هر کلاس را به‌طور محسوسی بهبود بخشد، بدون آن که هزینه محاسباتی اضافی بر سیستم تحمیل شود.

این استراتژی نه تنها یک رویکرد کارآمد و مقیاس‌پذیر برای امنیت اینترنت اشیا ارائه می‌دهد، بلکه یک الگوی کلی برای طراحی سیستم‌های تشخیص نفوذ محسوب می‌شود: انتقال تصمیمات دودویی سریع به طبقه‌بندی‌کننده‌های سبک و اختصاص مدل‌های پیچیده به تحلیل عمیق‌تر. در نتیجه، روش پیشنهادی نه تنها با روش‌های پیشرفته موجود قابل مقایسه است، بلکه در بسیاری از معیارها، به‌ویژه در شاخص‌های امتیاز F1 و استحکام عملکرد از آن‌ها پیشی می‌گیرد. این یافته‌ها پتانسیل بالای این رویکرد را به‌عنوان یک راه‌حل عملی، سبک و قابل استقرار برای شناسایی حملات در محیط‌های واقعی اینترنت اشیا برجسته می‌کند.

۴-۵- زمان آموزش و کارایی

یکی از انگیزه‌های کلیدی پشت معماری دولایه پیشنهادی، کارایی محاسباتی است. همانطور که در جدول ۶ نشان داده شده است، مدل پیشنهادی تقریباً به ۸.۲ ثانیه برای آموزش درخت تصمیم روی کل مجموعه داده‌ها و حدود ۱۴۵ دقیقه برای آموزش مدل روی زیرمجموعه برجسب‌گذاری شده با حمله (حدود ۲۵٪ از داده‌ها) نیاز دارد که در مجموع تقریباً ۱۴۶ دقیقه می‌شود. این زمان به‌طور قابل توجهی سریع‌تر از آموزش یک مدل یادگیری عمیق معادل روی کل مجموعه داده‌ها است. درخت تصمیم سبک وزن به‌طور مؤثر تشخیص ناهنجاری دودویی را مدیریت می‌کند و به CNN-BiLSTM پیچیده‌تر اجازه می‌دهد تا در طبقه‌بندی حمله چند کلاسی

می‌دهد که رویکرد پیشنهادی علاوه بر دقت بالاتر، زمان شناسایی حمله را به حدود ۰.۰۶۲ ثانیه کاهش داده است، که این امر آن را برای کاربردهای زمان واقعی و محیط‌های با منابع محدود مناسب می‌سازد. ترکیب دو لایه، ضمن کاهش بار محاسباتی، امکان تمرکز بر نمونه‌های مشکوک و پیچیده را فراهم می‌کند و عملکرد کلی سیستم را بهبود می‌بخشد.

مراجع

- [1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.
- [2] Alfriehat N., Anbar M., Aladaileh M., Hasbullah I., et al., "RPL-based attack detection approaches in IoT networks: review and taxonomy," Artificial Intelligence Review, vol. 57, art. 248, Aug. 2024.
- [3] T. Winter et al., RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks, RFC 6550, IETF, 2012.
- [4] M. Emeç and M. H. Özcanhan, "ROUT-4-2023: RPL based routing attack dataset for IoT," IEEE Dataport, doi: 10.21227/3mbe-5j70, 2023.
- [5] K. O. A. Alimi, K. Ouahada, A. M. M. Abu-Mahfouz, S. Rimer, and O. A. Alimi, "Refined LSTM Based Intrusion Detection for Denial-of-Service Attack in Internet of Things," Journal of Sensor and Actuator Networks (JSAN), vol. 11, no. 3, article 32, July 2022. DOI: 10.3390/jsan11030032
- [6] Abdelouahid Derhab, Arwa Aldweesh, Ahmed Z Emam, and Farrukh Aslam Khan. 2020. Intrusion detection system for internet of things based on temporal convolution neural network and efficient feature engineering. Wireless Communications and Mobile Computing 2020 (2020), 1–16.
- [7] Mohamed Abdel-Basset, Hossam Hawash, Ripon K Chakraborty, and Michael J Ryan. 2021. Semi-supervised spatiotemporal deep learning for intrusions detection in IoT networks. IEEE Internet of Things Journal 8, 15 (2021), 12251–12265.
- [8] Y. Guan, M. Noferesti, and N. Ezzati-Jivan, "CNN-BiLSTM-Based Classification of RPL Attacks in IoT Smart Grid Networks (Industry Track)," in Proc. ACM/IFIP/USENIX Middleware Conf., Bologna, Italy, Dec. 2023.
- [9] A. Etheridge and V. Anu, RPL Attack Detection in IoT Environments: An Ensemble Approach, in Proc. IEMTRONICS 2024 – International IoT, Electronics and Mechatronics Conference, Lecture Notes in Electrical Engineering, vol. 1228, Springer, 2025, pp. 113–122.
- [10] M. Osman, J. He, N. Zhu, and F. M. M. Mokbal, "An ensemble learning framework for the detection of RPL attacks in IoT networks based on the genetic feature selection approach," Ad Hoc Netw., vol. 152, 103331, Oct. 2023.
- [11] Mustafa Qahatan Alsudani, Salah H. Abbdal Reflish, Kohbalan Moorthy, and Myasar Mundher Adnan. 2023. A new hybrid teaching learning based Optimization -Extreme learning Machine model based Intrusion-Detection system. Materials Today: Proceedings 80 (2023),
- [12] A. Chougule, R. Mane, K. Bhattacharjee, and S. Mehta, "Ensemble Learning Based Intrusion Detection System for RPL-Based IoT Networks," in Demystifying AI and ML for Cyber-Threat Intelligence, A. Chougule and R. Mane, Eds. Cham: Springer, 2025, pp. 27–37, doi: 10.1007/978-3-031-90723-4_3.

تخصیص یابد.

جدول ۲ زمان آموزش و آزمایش مدل‌های پیشین و مدل پیشنهادی را نشان می‌دهد. همان‌طور که در این جدول مشاهده می‌شود، در زمان آزمایش، درخت تصمیم قادر است بیشتر ترافیک را تنها در کسری از ثانیه تشخیص دهد و تنها نمونه‌های مشکوک به حمله به لایه دوم منتقل می‌شوند که هرکدام تقریباً در چند ثانیه پردازش می‌شوند. این فعال‌سازی انتخابی، توان عملیاتی و مقیاس‌پذیری بالای سیستم را حتی در شرایط حمله بالا تضمین می‌کند. علاوه بر این، اندازه مدل سبک‌وزن — فقط چند کیلوبایت برای درخت تصمیم و چند مگابایت برای CNN-BiLSTM آن را برای استقرار در دستگاه‌های اینترنت اشیا با منابع محدود مناسب می‌سازد. در مقایسه با معماری‌های سنگین‌تر مانند یادگیری عمیق یا یادگیری گروهی که ممکن است به ده‌ها مگابایت حافظه نیاز داشته باشند، طرح پیشنهادی تعادل عملی میان عملکرد و امکان استقرار ارائه می‌دهد.

جدول ۶. مقایسه زمان آموزش و آزمایش مدل‌های پیشین و

مدل پیشنهادی

مدل	دوره‌ها	میانگین زمان در هر دوره (ثانیه)	کل زمان آموزش (دقیقه)	زمان آزمایش (ثانیه)
CNN-BiLSTM	۶۰۰	-	-	-
CNN	۵۰	۸۶.۷۴	۶۳.۶۸	۶۲.۳
LSTM	۲۵	۲۲۳.۸	۹۳.۲۵	۱۲۹.۱۲
روش پیشنهادی (تشخیص وقوع حمله)	-	-	۰.۰۸	۰.۹۱۶
روش پیشنهادی (تشخیص نوع حمله)	۲۰	۴۲۲.۸۲	۱۴۵.۲	۱۰۷

۵- نتیجه گیری

در این مقاله، یک چارچوب تشخیص نفوذ دولایه‌ای سبک‌وزن برای شبکه‌های اینترنت اشیا مبتنی بر RPL ارائه شده است. در لایه نخست، وقوع حمله با استفاده از مدل درخت تصمیم به‌صورت سریع و کم‌هزینه شناسایی می‌شود و در لایه دوم، نوع حمله با بهره‌گیری از ترکیب CNN-BiLSTM طبقه‌بندی می‌شود. ارزیابی‌های انجام‌شده روی مجموعه داده ROUT-4-2023 نشان داد که صحت کلی مدل به ۹۷٪ و F1 به ۹۵.۸٪ رسیده است، در حالی که نرخ مثبت کاذب پایین نگه داشته شده است. مقایسه با روش‌های موجود نشان