

## An Effective and Lightweight Intrusion Detection for IoT based on Fog and Cloud using KNN Classification

Ali Kaffash<sup>1</sup>, Seyed Reza Kamel<sup>2\*</sup>, Maryam Kheirabadi<sup>3</sup>

1. Ph.D Student, Department of Computer Engineering, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran.
2. Associate Professor, Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.\*Corresponding Author, [rezakamel@ieee.org](mailto:rezakamel@ieee.org)
3. Assistant Professor, Department of Computer Engineering, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran.

### Abstract

**Introduction:** In today's ever-evolving landscape of technology, the Internet of Things (IoT) has emerged as a transformative force, interconnecting countless smart devices that permeate our daily lives. From smart homes and cities to industrial automation and healthcare, IoT has brought about unprecedented convenience and efficiency. However, this rapid proliferation of IoT devices has also given rise to significant security challenges. The IoT ecosystem encompasses a diverse array of devices, ranging from wearable fitness trackers to critical infrastructure components, all of which are susceptible to cyber threats. Unauthorized access, data breaches, and malicious attacks on IoT networks pose severe risks to data privacy, infrastructure stability, and public safety. As a result, the need for robust security measures, such as IoT Intrusion Detection Systems, has become increasingly evident. The importance of these systems cannot be overstated, as they serve as the first line of defense against a myriad of IoT-related threats. By identifying and responding to potential security breaches, IoT Intrusion Detection Systems help maintain the integrity of data, ensure the functionality of IoT devices, and preserve the trustworthiness of the entire IoT network. The limitation of resources in electronic devices of the Internet of Things has caused less attention to the security. Today, Intrusion detection systems (IDSs) are one of the most important solutions to identify all types of attacks and threats and adopt appropriate solutions to deal with them. In addition, due to the openness of the environment, the placement of devices based on the Internet of Things makes this environment more vulnerable. For this reason, providing an effective and efficient intrusion detection system can be a suitable solution for this environment.

**Method:** In this article, we have presented a two-layer intrusion detection system based on KNN classification to separate malicious traffic from normal mode and multilayer perceptron artificial network to detect the type of attack. The data set used is KDD-CUP 99 data set.

**Results:** The experiment results show 99.743% accuracy for the data set as well as the improvement of Accuracy, Recall, Precision, F-measure, TPR and FPR parameters. In addition, the delay time of the proposed method is improved 40% compared to the MLP-MLP method and has a 139% lower delay than the fog-free state.

**Discussion:** The present study aims to propose two-layer hierarchical IDS based on machine learning, which detects attacks by considering the limitations of IoT resources. In order to create an efficient and accurate IDS, the combination of two improved K-nearest neighbor (KNN) algorithms and multi-layer perceptron (MLP) neural network applied in the fog and cloud to separate the attacks from normal traffic, respectively. we evaluated our proposed method using KDD-CUP 99 dataset. The results prove the improvement in accuracy, compared to the previous methods.

**Keywords:** IoT security, intrusion detection system, fog, cloud, KNN, MLP.



انجمن علمی تجارت الکترونیکی ایران

سامانه‌های پردازشی و ارتباطی چندرسانه‌ای هوشمند

Intelligent Multimedia Processing and Communication Systems (IMPCS)



واحد نیشابور

## ارائه یک سیستم مؤثر و سبک برای تشخیص نفوذ در محیط اینترنت اشیا مبتنی بر محاسبات مه و ابر بر اساس طبقه‌بندی KNN

دوره پنجم، تابستان ۱۴۰۳  
شماره دوم، صص: ۵۵-۶۴

تاریخ دریافت: ۱۴۰۳/۰۱/۱۴  
تاریخ پذیرش: ۱۴۰۳/۰۲/۲۹

علی کفاش<sup>۱</sup>، سیدرضا کامل<sup>۲</sup>، مریم خیرآبادی<sup>۳</sup>

۱. دانشجوی دکتری، گروه مهندسی کامپیوتر، واحد نیشابور، دانشگاه آزاد اسلامی، نیشابور، ایران. [alikaffash@yahoo.com](mailto:alikaffash@yahoo.com)
۲. دانشیار، گروه مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران. (نویسنده مسئول) [rezakamel@ieee.org](mailto:rezakamel@ieee.org)
۳. استادیار، گروه مهندسی کامپیوتر، واحد نیشابور، دانشگاه آزاد اسلامی، نیشابور، ایران. [m.kheirabadi@iau-neyshabur.ac.ir](mailto:m.kheirabadi@iau-neyshabur.ac.ir)

**چکیده:** محدودیت منابع در وسایل الکترونیکی اینترنت اشیا باعث شده است کمتر به مقوله امنیت در آن توجه شود. راهکارهای تشخیص نفوذ امروزه یکی از مهم‌ترین و اساسی‌ترین راه‌حل‌ها برای شناسایی انواع حملات و تهدیدات و اتخاذ راهکار مناسب برای مقابله با آن‌ها می‌باشند. همچنین با توجه به باز بودن محیط قرار گرفتن وسایل مبتنی بر اینترنت اشیا باعث آسیب‌پذیری بیشتر این محیط می‌شود. به همین علت ارائه یک سیستم تشخیص نفوذ مؤثر و کارا می‌تواند راهکار مناسبی برای این محیط باشد. در این مقاله به ارائه یک سیستم تشخیص نفوذ دولایه مبتنی بر طبقه‌بندی KNN برای جداسازی ترافیک عادی از حمله و شبکه مصنوعی پرسپترون چندلایه برای تشخیص نوع حمله پرداخته‌ایم. مجموعه داده استفاده شده مجموعه داده معروف KDD-CUP 99 می‌باشد. نتایج آزمایش بیانگر دقت ۹۹٫۷۴۳٪ برای مجموعه داده و همچنین بهبود پارامترهای Accuracy، Recall، Precision، F-measure، TPR و FPR می‌باشد. همچنین زمان تأخیر روش پیشنهادی نسبت به روش MLP-MLP میزان ۴۰٪ بهبود یافته است و ۱۳۹٪ تأخیر کمتری نسبت به حالت بدون مه دارد.

**واژه‌های کلیدی:** ابر، مه، KNN, MLP, KDD-CUP 99, Intrusion Detection System

## ۱. مقدمه

اینترنت اشیا شبکه‌ای از اشیا با قابلیت شناسایی واضح عناصر است [1]. یکی از چالش‌های مهم در اینترنت اشیا، محافظت از دستگاه‌های IOT در مقابل حملات به دلیل ناهمگونی دستگاه‌ها و پروتکل‌ها، دسترسی مستقیم به دستگاه‌ها از طریق اینترنت و محدودیت منابع در دستگاه‌ها می‌باشد [2]. به‌طور کلی، دو نوع سیستم تشخیص نفوذ وجود دارد: سیستم‌های شناسایی سوء استفاده و رفتارهای غیرعادی. در سیستم‌های شناسایی سوء استفاده، سیستم از ساختار کلی حملات اطلاع دارد و الگوریتم خاصی را برای اشکال مختلف حملات در اختیار دارد اما در سیستم‌های شناسایی رفتارهای غیرعادی، تنها رفتار درست و معمولی کاربر است که اطلاعات و مشخصات آن در سیستم قرار دارد [3]. محیط‌های هوشمند از طریق اینترنت اشیا و محاسبات مه در حال واقعی شدن هستند، اما هیچگاه از تهدیدات و آسیب‌پذیری‌های امنیتی در امان نمی‌باشند؛ تکنیک‌ها برای بهره‌برداری از آسیب‌پذیری‌های رایانه‌ای دائماً در حال به‌روزرشدن و بهبود یافتن هستند. از جمله اهداف اصلی فراهم‌نمودن امکان دسترسی به سیستم‌ها، به‌دست‌آوردن و استفاده نامناسب و غیرقانونی از اطلاعات محرمانه و از دسترس خارج کردن دسترسی به منابع می‌باشد [4]. برای مثال حادثه‌ای که در سال ۲۰۱۶ میلادی برای دستگاه‌های IOT رخ داد، به صورتی که حمله بات نت Mirai به ارائه‌دهنده سرویس Dyn باعث شد که صدها سایت نظیر نتفلیکس، گیت‌هاب برای ساعت‌ها از دسترس خارج شوند [5] [4]. صنعت ۴ یک حوزه کاربردی مهم برای بحث اینترنت اشیا می‌باشد. سیستم‌های صنعتی از پتانسیل اینترنت اشیا بهره‌می‌گیرند تا هزینه‌های عملیاتی غیرضروری خود را کاهش و قابلیت استفاده و اطمینان‌داری‌های صنعتی را افزایش دهند. اینترنت اشیا صنعتی یا به‌اختصار IIOT وظیفه اتصال ماشین‌ها، حسگرها و محرک‌ها را در صنایع تولیدی به‌عهده‌دارد تا بتواند کلیه مراحل زنجیره تولید را به صورت خودکار درآورده و بر آن نظارت داشته‌باشد [6]. تهدیدات امنیتی در این حوزه نیز موجب نگرانی‌های گسترده‌ای شده‌است، زیرا تهدیدات امنیتی در این حوزه می‌تواند خسارات سنگینی به دارایی‌ها وارد نماید یا زندگی بشر را با تهدیدهای جدی مواجه کند. این مهم اهمیت اتخاذ راهکارهای امنیتی برای IOT را آشکارتر می‌نماید و این تنوع حملات اهمیت سرمایه‌گذاری مالی و فکری در حوزه امنیت اینترنت اشیا را آشکار می‌سازد [7]. امنیت در این محیط‌ها به‌عنوان اینترنت اشیا امری حیاتی و اجتناب‌ناپذیر است؛ زیرا دستگاه‌های اینترنت اشیا امروزه به عنوان جز لاینفک در زندگی روزمره قرار گرفته‌اند که با اطلاعات حساس سروکار دارند. از طرفی بعضی سیستم‌های اقدامات حیاتی و نظارتی را انجام می‌دهند که مستلزم یک عملیات بدون وقفه می‌باشد. اینترنت اشیا و محاسبات مه از یک سری تکنولوژی‌ها و خدمات و استانداردها تشکیل شده‌اند که هر کدام از آن‌ها خود دارای مسایل امنیتی و الزامات حریم شخصی می‌باشد [8].

سیستم‌های پیشگیری و تشخیص نفوذ به‌طور گسترده‌ای رویدادهای مرتبط با حملات سایبری را بررسی و نظارت می‌کنند. با وجود این که ممکن است سیستم‌های IDS در نوع تکنیک استفاده‌شده تفاوت‌هایی داشته‌باشند، اما همگی از یک سری اصول مشترک تبعیت می‌کنند [9]:

۱. کسب اطمینان از اینکه رکوردهای اطلاعات مرتبط با رویدادهای واقعی هستند. اطلاعات می‌توانند ذخیره‌شوند و در دسترس قرارگیرند، حتی برای سیستم‌هایی که ارتباطشان قطع شده‌است.
۲. این مورد که مدیران امنیتی باید از وقایع مهم باخبر شوند امری ضروری می‌باشد. سیستم می‌تواند هشدارهای از طریق کانال‌های متفاوت ارسال کند: پیام‌هایی از طریق کنسول سیستم، ایمیل‌ها، پیام کوتاه، اسکریپت‌های تریگر تعریف‌شده توسط کاربر و ...، محتوای متن پیام باید عمدتاً اطلاع‌رسانی باشد. جزئیات کامل در IDS ذخیره می‌شود.
۳. تولید گزارش: چنین خروجی‌هایی وقایع کشف‌شده را خلاصه کرده یا جزئیات مرتبط با آن را ارائه نماید.
۴. یک IDS باید بتواند در هنگام شناسایی تهدید جدید، اقدامات دفاعی را آغاز کند به‌عنوان مثال، ممنوع کردن آدرس‌های IP خاص یا محدوده IP مرتبط با مبدأ فعالیت‌های غیرعادی. همچنین ممکن است از دسترسی به منابع خاصی که نیاز به محافظت دارند به‌وسیله پیکره بندی مجدد دستگاه‌های شبکه مثل مسیریاب‌ها و دیوارهای آتش به منظور جلوگیری از دسترسی به آن منابع محافظت کند.
۵. یک IDS ممکن است به‌منظور مسدود کردن محتوای مخرب، ترافیک را تغییر دهد؛ مانند پیوست‌های ایمیل که دارای برنامه‌های مخرب هستند و قسمتی که ایمن تلقی می‌شود را ارسال نماید.
۶. فناوری‌های IDS به‌طور کامل و ۱۰۰٪ دقیق نیستند. بنابراین، فعالیت‌های بی‌ضرر ممکن است به‌عنوان مخرب (مثبت کاذب) یا بالعکس (منفی کاذب) طبقه‌بندی شوند. درحالی‌که نرخ تشخیص می‌تواند بهبود یابد، خطاها را نمی‌توان به‌طور کامل ریشه‌کن کرد.

خلاصه روش پیشنهادی به شرح زیر است:

- در این مقاله به ارائه یک ساختار سیستم تشخیص نفوذ دولایه پرداخته‌ایم که با استفاده از مفاهیم مه و ابر می‌تواند طیف وسیعی از حملات در محیط اینترنت اشیا را با استفاده از طبقه‌بندی بهبود یافته KNN در لایه مه و الگوریتم شبکه عصبی MLP در لایه ابر به ترتیب شناسایی و نوع آن را تشخیص دهد.

همچنین عدم استفاده از دیتا ست‌های جدیدتر در کنار دیتا ست‌های قدیمی می‌تواند از نقدها وارد به این روش باشد.

با توجه به تحقیق انجام‌شده، محققان [15] یک IDS سبک با نام Sample Selected که یک ماشین یادگیری بی‌نهایت (SS-ELM) برای غلبه بر محدودیت فضای گره‌های مه ارائه کردند. همچون روش قبلی در این مدل نیز مجموعه داده KKD-CUP 99 استفاده شده است و نشان می‌دهد از نظر دقت تشخیص و زمان آموزشی کارایی بهتری نسبت به الگوریتم انتشار برگشتی کلاسیک دارد.

محققان در مقاله خود [16] یک روش توزیع‌شده برای تشخیص نفوذ ارائه‌نموده‌اند که بر اساس یادگیری عمیق برای محیط شبکه اینترنت اشیا / مه قابل استفاده است. نویسندگان در این روش از یک دستگاه مه به‌عنوان هماهنگ‌کننده اصلی در اجرای اشتراک‌گذاری مشترک و بهینه‌سازی پارامترهای مدل استفاده می‌کنند. این گره اولیه می‌تواند به‌عنوان یک نقطه شکست واحد (SPOF) در نظر گرفته شود که نسبت به یک رویکرد به‌روزرسانی پارامتر مبتنی بر ابر راحت‌تر به خطر می‌افتد. عملکرد مدل عمیق با روش‌های سنتی یادگیری ماشین مقایسه می‌شود، و کشف حمله توزیع‌شده در برابر سیستم تشخیص متمرکز ارزیابی می‌شود. نتایج تجربی، اثر بخشی مدل پیشنهادشده را در تشخیص حملات سایبری نشان می‌دهد (دقت بالا و یادآوری).

در مقاله دیگری محققان [17] یک تکنیک جدید تشخیص نفوذ مبتنی بر محاسبات مه در کار خود پیشنهاد داده‌اند. این روش با استفاده از ماشین یادگیری افراطی متوالی آنالین (OSELM) که قادر است به‌طور هوشمند به تفسیر حملات ترافیک اینترنت اشیا بپردازد. در روش ارائه شده پیاده‌سازی سیستم تشخیص نفوذ در گره‌های مه و خلاصه‌سازی در سرورس‌دهنده ابر انجام می‌شود. در گره‌های مه پارامترهای ورودی تعداد لایه‌های پنهان نرون‌ها به همراه تابع فعال‌ساز و تعداد کلاس‌ها می‌باشند، سپس در مرحله مقدار دهی با استفاده از داده‌های آموزشی اقدام به تخصیص تصادفی وزن‌ها و بایاس می‌کند و در ادامه کار، اقدام به محاسبه ماتریس لایه پنهان اولیه و اوزان خروجی می‌نماید. سپس در مرحله یادگیری متوالی برای سایر داده‌های اضافی ورودی اقدام به محاسبه ماتریس لایه پنهان اولیه و اوزان خروجی اولیه می‌کند و در مرحله یادگیری متوالی برای سایر داده‌های اضافی ورودی، اقدام به محاسبه ماتریس آخرین لایه پنهان می‌کند و وزن خروجی نهایی را بر اساس الگوریتم کمترین مربع بازگشتی محاسبه خواهد کرد. در نهایت نیز الگوریتم ارائه‌شده با الگوریتم‌های ANN، naïve bayes و ELM مقایسه شده و در بعضی حملات بر اساس فاکتورهای ارزیابی دقت، نرخ کشف، نرخ هشدار بهتر عمل نموده و در مواردی هم طبق آزمایش‌های انجام‌شده سایر الگوریتم‌ها بهتر عمل کرده‌اند. همچنین مجموعه داده استفاده شده نیز KDD-CUP می‌باشد که برخلاف عنوان مقاله که مدعی ارائه روش برای اینترنت اشیا می‌باشد، مجموعه داده اینترنت اشیا نمی‌باشد و از طرفی فقط با همین یک مجموعه داده ارزیابی شده است.

• توجه به تشخیص حملات در بخش مه و الگوریتم به‌کاررفته در آن تأخیر سیستم تشخیص نفوذ به‌طور مؤثری کاهش می‌یابد و فقط ترافیکی که حمله تشخیص داده شده است برای شناسایی نوع حملات به ابر ارسال می‌شود و شبکه عصبی MLP در ابر به‌روزرسانی خواهد شد. ما عملکرد مدل پیشنهادی را با استفاده از مجموعه داده KDD-CUP 99 ارزیابی می‌کنیم. نتایج نشان می‌دهد که رویکرد پیشنهادی در معیارهای عملکرد چندگانه از رویکردهای موجود بهتر عمل می‌کند.

در بخش ۲ پیشینه پژوهش و رویکردهای مختلف موجود را برای انواع مختلف سیستم‌های تشخیص نفوذ بررسی می‌کنیم، در بخش ۳ به تشریح کامل روش پیشنهادی پرداخته و تحلیل و ارزیابی روش خود را در بخش ۴ ارائه خواهیم کرد و در انتها نتیجه‌گیری را در بخش ۵ ارائه می‌کنیم.

## ۲. پیشینه پژوهش

در مقاله [10] به معرفی یک سیستم تشخیص نفوذ سبک و توزیع شده برای سیستم ایمنی مصنوعی (AIS) پرداخته‌اند. سیستم IDS مذکور در ساختار اینترنت اشیا سه‌لایه‌ای توزیع شده است که شامل ابر، مه و لایه‌های لبه می‌باشند. این روش مبتنی بر AIS بر روی مجموعه داده KDD-CUP 99 [11] مورد آزمایش قرار گرفت و نشان داد می‌تواند در برابر حملات کم‌تکرار نظیر R2L و U2R مؤثر باشد. علاوه بر این مدل خود را بر روی SSH Brute Force بر روی مجموعه داده ISCX تست کرده‌اند [12]. یکی از نقاط ضعف این IDS قدیمی بودن مجموعه داده استفاده‌شده در آن می‌باشد. از سوی دیگری مدل‌های IDS که از مجموعه داده‌های مدرن استفاده می‌کردند برای پلتفرم‌های ابری طراحی شدند [13].

محققان [14] در مقاله خود به ارزیابی کارایی شبکه‌های عمیق و کم‌عمق متعدد برای سیستم‌های تشخیص نفوذ تحت شبکه (NIDS) می‌پردازند. تحقیقات نشان می‌دهد شبکه‌های عمیق در مقایسه با شبکه‌های کم‌عمق در بیشتر آزمایش‌ها از اجرای بهتری برخوردارند. دلیل اصلی این امر می‌تواند عبور اطلاعات از چند لایه مختلف به منظور یادگیری الگوهای پنهان زیرلایه‌ها و رکوردها یا سوابق متصل حملات شبکه باشد که در نهایت این ویژگی‌های آموخته‌شده هر لایه با یکدیگر جمع می‌شوند تا حالت عادی را از حملات متنوع جدا کنند. به علاوه شبکه‌های عمیق نه تنها در بخش اجرا خوب عمل می‌کنند بلکه در شناسایی و طبقه‌بندی حملات شناخته‌شده و همچنین حمله‌های ناشناخته نیز عملکرد بهتری دارند. به‌منظور رسیدن به نرخ کشف قابل قبول در این مقاله پیکره‌بندی‌های متنوع از تنظیمات شبکه و پارامترهای آن در شبکه‌های عمیق اعمال شده است. همه پیکره‌بندی‌های شبکه عمیق تا ۱۰۰۰ دوره (epochs) در مرحله آموزش با یک نرخ آموزش در بازه ۰.۵ - ۰.۰۱ به منظور دستیابی به الگوهای متنوع زمانی از حالت عادی و حمله، اجرا شده‌اند. دقت پایین برای برخی از حملات و

است. جهت بهنجارسازی داده‌ها در ویژگی‌های که نیاز به به-هنجارسازی دارند از معادله (۱) استفاده می‌کنیم:

$$n(m, n) = \frac{c(m+n) - \min(n)}{\max(n) - \min(n)}$$

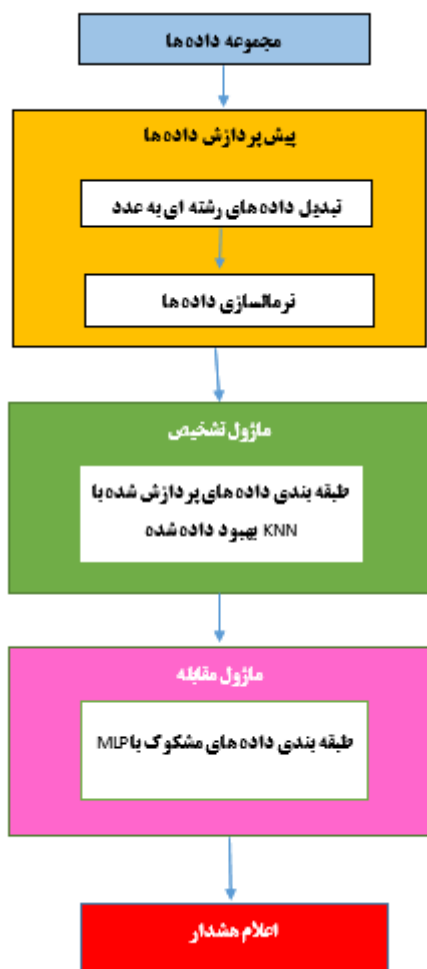
Max(n): مقدار بیشینه مربوط به ستون m

Min(n): مقدار کمینه مربوط به ستون m

C(m,n): عدد صحیح مربوط به سطر m و ستون n

N(m,n): مقدار نرمالیزه سطر m از ویژگی n

پس از انجام پیش پردازش مقادیر کلیه نشانه‌ها در گستره (۰ و ۱) قرار می‌گیرند و این داده‌ها قابل استفاده هستند.



شکل ۲: کلیت کار الگوریتم پیشنهادی

ماژول تشخیص و مقابله: اساساً باید تأخیر بسیار کمی داشته‌باشد تا امکان پاسخ سریع برای کاهش آسیب احتمالی که یک ترافیک غیرعادی می‌تواند ایجاد کند را فراهم سازد [13]. بنابراین ما از یک مدل KNN بهبود یافته در لایه مه استفاده می‌کنیم تا سرعت تشخیص را بیشتر کنیم. الگوریتم KNN بهبود یافته‌ای که در این مقاله استفاده شده است،

محققان [18] یک سیستم سبک تشخیص نفوذ مبتنی بر درخت تصمیم برای محیط محاسبات مه ارائه کرده‌اند که سیستم ارائه شده بر محدودیت‌های گره مه می‌تواند غلبه کند. مجموعه داده KDD-CUP 99 برای آزمایش کارایی سیستم تشخیص نفوذ ارائه شده استفاده شده است. کارایی سیستم تشخیص نفوذ ارائه شده برای ترافیک عادی ۹۸,۶۷٪ و برای ترافیک غیرعادی ۹۶,۶۵٪ تشخیص داده شد که در مقایسه با کارایی طبقه‌بندی‌های Naïve Bayes و KNN بهتر می‌باشد. همچنین IDS ارائه شده زمان تشخیص را برای هر کدام از روش‌های طبقه‌بندی دودویی و چندکلاسی مقایسه می‌کند. یکی از معایب سیستم ارائه شده تأخیر زمان شناسایی می‌باشد.

### ۳. روش پیشنهادی

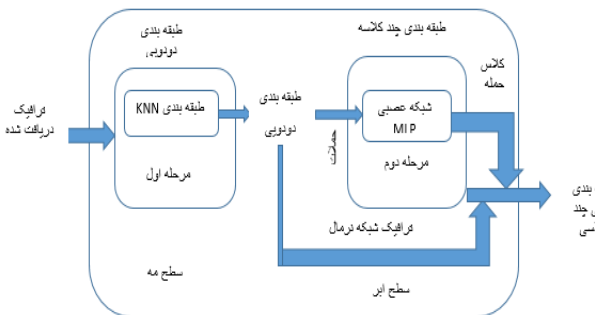
هر دستگاه اینترنت اشیا دارای چند ماژول واقع در لایه مه است که ترافیک شبکه را تجزیه و تحلیل و طبقه‌بندی می‌کند. هنگامی که یک گره مه ترافیک شبکه را دریافت می‌کند، آن را روی ماژول‌های مختلف از جمله ماژول پیش پردازش، ماژول تشخیص و ماژول مقابله، پردازش می‌کند. این ماژول بدون تعامل با لایه ابر عمل می‌کند و به‌طور مستقل می‌تواند ترافیک ورودی را به دو دسته نرمال و حمله طبقه‌بندی کند. ماژول تشخیص، وظیفه تجزیه و تحلیل ویژگی‌های ترافیک ضبط شده و طبقه‌بندی آن‌ها را در ترافیک نرمال یا حمله برعهده می‌گیرد. ماژول مقابله، چنانچه ترافیک ورودی توسط ماژول تشخیص، حمله طبقه‌بندی شود این ترافیک به محیط ابر ارسال می‌شود تا نوع حمله را تشخیص داده و مدل شبکه عصبی واقع در لایه ابر به‌روزرسانی شود. سطح تشخیص در لایه ابر، امکان تصحیح مثبت کاذب سطح تشخیص لایه مه را فراهم می‌کند. به عبارتی، رکوردهایی که حمله نیستند ولی در لایه مه، به اشتباه حمله تشخیص داده شده‌اند. وقتی این رکوردها توسط لایه ابر، یک رکورد نرمال تشخیص داده شود، مشخصات رکورد به لایه مه ارسال شده تا ماژول تشخیص به‌روزرسانی شود و این رکورد به دسته نرمال طبقه‌بندی شود تا در تشخیص‌های بعدی نرخ مثبت کاذب کاهش یابد. شکل ۲ چارت روش پیشنهادی مقاله را نشان می‌دهد.

### ۳.۱. ماژول پیش پردازش

ماژول پیش پردازش، داده‌های رشته‌ای را به عدد صحیح تبدیل می‌کند و سپس کل داده‌ها را نرمال سازی می‌کند تا کارایی تشخیص نفوذ بهبود یابد. از آنجاکه طبقه‌بندی KNN تنها از داده‌های عددی استفاده می‌کند. بنابراین ویژگی‌های متن نیاز به تبدیل به مقادیر عددی دارد. بنابراین، برخی از مقادیر عددی برای ویژگی‌های متنی مختلف به ترتیب با عدد صحیح مقداردهی می‌شوند.

در مرحله دوم پیش پردازش هر کدام از ویژگی‌ها به‌هنجار می‌شود. یعنی در گستره (۰ و ۱) مقیاس گذاری می‌شوند. گستره مقادیر صحیح مربوط به ویژگی‌ها، متفاوت است. برخی از ویژگی‌ها دارای گستره‌ای بسیار بزرگ از اعداد صحیح هستند و برخی دیگر مقادیر ۰ یا ۱ دارند و یا پیوسته هستند بنابراین به هنجارسازی برای این ویژگی‌ها ضروری

در لایه ابر از شبکه عصبی چندلایه برای تشخیص نوع حمله استفاده شده است. هدف شبکه عصبی، تولید یک مدل بر اساس داده‌های آموزشی است به بیانی دیگر مشکل سیستم‌های تشخیص نفوذ در طبقه بندی الگو را بیان می‌کند. الگوهای ذخیره شده باید به طور مداوم به روز شوند. شکل 3 ساختار روش پیشنهادی در مرحله طبقه بندی را نشان می‌دهد. جهت بهبود دقت شبکه عصبی در لایه ابر، رکوردهایی که در لایه مه به عنوان نرمال تشخیص داده شده‌اند (منفی صحیح یا True Negative) با برجسب نرمال وارد لایه ابر شده و برای یادگیری شبکه عصبی استفاده می‌شوند.



شکل 3- ساختار روش پیشنهادی در مرحله طبقه‌بندی [13]

در این پژوهش، معماری MLP شامل یک شبکه عصبی پیشخور دو لایه است. به عبارتی یک شبکه عصبی با یک لایه مخفی و یک لایه خروجی استفاده شده است (لایه ورودی شمرده نمی‌شود، زیرا لایه ورودی شبیه بافر عمل می‌کند و پردازش در آن صورت نمی‌گیرد). ساختار این شبکه دو لایه، به صورت  $(X_1 \ 10 \ X_2)$  در نظر گرفته شده است که  $X_1$  تعداد ویژگی‌های ترافیک ورودی می‌باشد. یعنی  $X_1$  ورودی، ۱۰ گر در لایه میانی و  $X_2$  گر در لایه خروجی وجود دارد و مقدار  $X_1$  در مجموعه داده kdd برابر با و مقدار  $X_2$  در مجموعه داده kdd برابر با ۵ است. گرهای خروجی دارای مقادیر باینری خواهند بود. به این مفهوم که برای هر کدام از کلاس‌های خروجی، تنها یکی از گرهای خروجی یک شده و بقیه صفر خواهند بود. بنابراین برای کلاس‌های مختلف، خروجی‌های متفاوت خواهیم داشت که برای مجموعه داده kdd عبارتند از: طبیعی (۰۰۰۰۱)، انکار سرویس (۰۰۰۱۰)، حملات دور دست (۰۰۱۰۰)، کاربر به ریشه (۰۱۰۰۰)، حملات پویش (۱۰۰۰۰).

مجموعه داده آموزشی به دست آمده برای تنظیم دقیق طبقه بندی کننده MLP استفاده می‌شود. در این تحقیق از تابع trainseg به عنوان تابع آموزش شبکه عصبی استفاده شده است. همچنین تعداد دوره آموزشی ۱۰۰۰ و میزان خطای مطلوب ۰,۰۰۱ تعریف شده است. در نهایت، ویژگی‌های آزمایشی به طبقه بندی کننده MLP آموزش دیده برای شناسایی حملات معرفی می‌شوند.

تنها گرهایی را که در فضا نزدیک یکدیگر هستند مقایسه می‌کند و از محاسبات غیر ضروری اجتناب می‌کند [19].

الگوریتم بهبود یافته گرها را به یک شبکه فضایی اختصاص می‌دهد که به عنوان یک شاخص عمل می‌کند. در نهایت فواصل مورد نیاز بین گرها تعیین می‌شود تا لبه‌های مربوط به هر گر اضافه شود. برای انجام این کار، الگوریتم از طریق سلول‌های شبکه و گرهای مربوط به آن‌ها تکرار می‌شود. برای هر سلول شبکه I، فاصله تا تمام گرهای خود سلول شبکه  $(r = 0)$ ، تا گرهای سلول‌های شبکه مجاور  $(r = 1)$ ، و گرهای سلول‌های شبکه بعدی  $(r = 2)$ ، و غیره را در نظر می‌گیرد. تا زمانی که نزدیکترین گر برای هر گر از سلول شبکه اصلی پیدا شود. سپس از این حداقل فاصله‌ها برای یافتن تمام لبه‌هایی که از یک گر از سلول شبکه در نظر گرفته شده شروع می‌شوند، استفاده می‌شود، که نیاز به بررسی همه گرها در سلول‌های شبکه ابرمکعب مربوطه دارد [19].

اندازه  $\mu$  شبکه فضایی بر زمان اجرا تأثیر دارد. در حالت ایده‌آل، می‌توان با انتخاب یک  $\mu$  مناسب، به زمان اجرای  $O(n)$  برسیم، به این معنی که تعداد سلول‌های شبکه‌ای را که باید در هر مرحله تکرار در نظر گرفته شود ثابت نگه داریم (تعیین نزدیکترین همسایه برای هر گر و اضافه کردن لبه‌ها). بنابراین تعداد گرها در هر سلول باید مستقل از تعداد کلی گر  $n$  باشد. اگر قرار باشد این عدد در مقدار  $C$  ثابت بماند، طول ضلع  $\mu$  برابر است با  $\sqrt[4]{n/2}$ . این انتخاب تضمین می‌کند که میانگین تعداد سلول‌های شبکه‌ای که باید در هنگام یافتن نزدیکترین گرهای همسایه در نظر گرفته شود، مستقل از تعداد کلی گرهای ثابت باقی‌ماند [19].  $n$  تعداد کل داده‌ها،  $d$  تعداد بعد یا ویژگی‌های داده،  $p$  چگالی شبکه داده‌ای است. طبق [19] مقدار چگالی شبکه  $p$  برابر با ۱,۶ باید باشد.

ابتدا داده‌ها را به  $\mu^d$  سلول شبکه تقسیم کرده و شناسه سلول شبکه-ای شامل این داده تعیین می‌شود. در مرحله بعدی فاصله بین داده‌ها و لبه‌ها محاسبه می‌شود. برای این کار، برای هر سلول شبکه I فاصله همه داده‌های سلول شبکه خودش به داده‌های سلول‌های شبکه همسایه مقایسه شده تا وقتی که یک داده نزدیک برای هر داده سلول شبکه اصلی پیدا شود. سپس با استفاده از این کمترین فاصله همه لبه‌های سلول شبکه پیدامی‌شوند.

با انتخاب یک اندازه سلول مناسب، پیچیدگی کلی الگوریتم را می‌توان به  $O(n)$  کاهش داد و این مقدار در مقابل پیچیدگی الگوریتم KNN که برابر با  $O(n^2)$  است بسیار مناسب‌تر است.

با ایجاد روش‌هایی برای نمایه‌سازی گرها توسط یک شبکه فضایی و تعیین همسایگی‌ها در شبکه، می‌توانیم الگوریتم تولید یک مدل موکنیک را بهبود بخشیم. استفاده از شبکه فضایی که در بخش قبل مورد بحث قرار گرفت، الگوریتم را در مقایسه با الگوریتم ساده کارآمدتر می‌کند، زیرا امکان بهره‌برداری از موقعیت شبکه حاصل را فراهم می‌کند [19].

#### ۴. ارزیابی و کارایی سیستم تشخیص نفوذ

ارزیابی مدل طبقه‌بندی یکی از مهم‌ترین بخش‌هایی است که در دسته‌بندی باید به آن توجه نمود. جهت ارزیابی مدل طبقه‌بندی باید بر اساس نمونه‌های آموزشی و تست صورت گیرد. وقوع حالات مختلف برای دسته‌ها با توجه به مجموعه داده‌های ورودی برای دسته‌بندی با مقادیر TP, FP, FN, TN برای دو دسته مثبت و منفی در (جدول ۴-۱) نشان داده شده است.

جدول ۴-۱ ماتریس درهم ریختگی برای یک مسأله دسته‌بندی

##### دودسته‌ای

نوع رکورد	رکوردهای تخمینی (Predicated Records)		
	نوع دسته	دسته -	دسته +
رکوردهای واقعی	دسته -	TN	FP
	دسته +	FN	TP

معیار (TN): رکوردهایی که حمله نیستند و به درستی نیز نرمال تشخیص داده شده است.

معیار (TP): رکوردهایی که حمله هستند و به درستی، حمله تشخیص داده شده است.

معیار (FP): رکوردهایی که حمله نیستند ولی به اشتباه حمله قرار گرفتند.

معیار FN: رکوردهایی که حمله هستند ولی به اشتباه به عنوان فعالیت درست قرار گرفتند.

با توجه به پارامترهای مطرح شده در رابطه‌های زیر معیارهای ارزیابی مختلفی ارائه شده است که از جمله مهم‌ترین آن‌ها می‌توان به معیار درستی، دقت، فراخوانی و معیار F-measure اشاره کرد.

مهم‌ترین معیار برای تعیین کارایی یک الگوریتم دسته‌بندی معیار Accuracy می‌باشد. این معیار دقت کل یک دسته‌بند را محاسبه می‌کند. این معیار نشان‌دهنده این موضوع است که چند درصد از کل مجموعه داده‌ها به درستی دسته‌بندی شده است. رابطه (2) نحوه محاسبه معیار درستی را نشان می‌دهد.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (2)$$

دو مقدار TP و TN مهم‌ترین مقادیری هستند که باید بیشینه شوند تا کارایی دسته‌بندی به حداکثر برسد.

معیار Precision درصدی را نشان می‌دهد که از میان تمامی دسته‌ها که توسط دسته‌بند به آن دسته نسبت داده شده‌اند، درست دسته‌بندی شده‌اند. به عبارتی دقت دسته‌بندی دسته i را با توجه به کل مواردی

نشان می‌دهد که برچسب i برای نمونه مورد بررسی توسط دسته‌بند پیشنهاد شده است. نحوه محاسبه این معیار در رابطه (3) نشان داده شده است. اندیس i در این پارامترها به این مفهوم است که پارامترها باید برای هر دسته i محاسبه شوند.

$$Precision_i = \frac{TP_i}{TP_i + FP_i} \quad (3)$$

معیار Recall برای یک دسته، که از میان تمامی دسته‌های حملات متعلق به آن دسته، به درستی دسته‌بندی شده است. به عبارتی دقت دسته‌بندی دسته i را با توجه به کل نمونه‌های با برچسب i نشان می‌دهد. نحوه محاسبه این معیار در رابطه (4) نشان داده شده است.

$$Recall_i = \frac{TP_i}{TP_i + FN_i} \quad (4)$$

نکته قابل توجه این است که معیار Recall کارایی دسته‌بند را با توجه به تعداد رخداد دسته i نشان می‌دهد در حالیکه معیار Precision اساساً مبتنی بر دقت پیش‌بینی دسته می‌باشد و بیانگر آن است که به چه میزان می‌توانیم به خروجی دسته‌بند اعتماد کنیم.

معیار F-measure از ترکیب معیارهای Precision و Recall به دست می‌آید و در مواردی استفاده می‌شود که نتوان اهمیت ویژه‌ای را برای هر یک از دو معیار Precision و Recall نسبت به یکدیگر قائل شد رابطه (5) نحوه محاسبه این معیار را نشان می‌دهد.

$$F - measure = \frac{2 * Precision_i * Recall_i}{Precision_i + Recall_i} \quad (5)$$

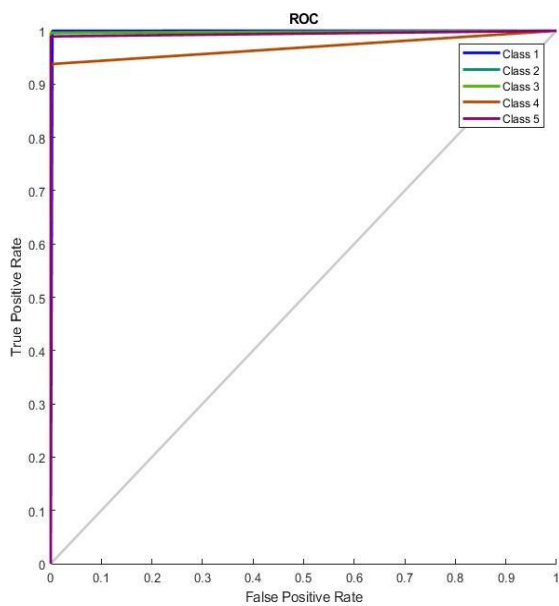
همچنین در مسائل واقعی معیارهای دیگری نظیر TPR و FPR وجود دارد که از اهمیت ویژه‌ای برخوردارند. این معیارها که توجه بیشتری به دسته‌بند مثبت نشان می‌دهند توانایی دسته‌بند را در تشخیص دسته مثبت مشخص می‌کنند. معیار TPR نشان می‌دهد که دقت تشخیص دسته مثبت چه مقدار است و معیار FPR نرخ هشدار غلط را با توجه به دسته منفی بیان می‌کند. رابطه (6) معیار FPR را نشان می‌دهد که نرخ هشدار غلط را با توجه به دسته منفی بیان می‌کند. رابطه ۷ معیار TPR را نشان می‌دهد.

$$FPR = \frac{fp}{tn + fp} \quad (6)$$

$$TPR = \frac{tp}{fn + tp} \quad (7)$$



معیار مهمی که برای تعیین میزان کارایی یک دسته‌بندی استفاده می‌شود معیار AUC است. این معیار نشان‌دهنده سطح زیر نمودار ROC می‌باشد و هرچه مقدار این عدد مربوط به یک دسته‌بندی بزرگتر باشد کارایی نهایی دسته‌بند مطلوب‌تر ارزیابی می‌شود. در واقع منحنی‌های ROC منحنی‌های دوبعدی هستند که در آن‌ها DR یا همان تشخیص صحیح دسته‌بند روی محور  $y$  و به‌طور مشابه FAR یا همان نرخ تشخیص غلط دسته منفی روی محور  $x$  رسم می‌شوند. باید توجه داشت که منحنی مذکور رفتار یک دسته‌بند را بدون توجه به توزیع دسته‌ها یا هزینه خطا نشان می‌دهد و بنابراین زمانه که یک دسته‌بند در کل فضای کارایی به وضوح بر دسته دیگری تسلط یابد می‌توان گفت که بهتر از دیگری است. در این نمودار وضعیت class1، Normal وضعیت class2، Dos، وضعیت class3، وضعیت class4، R2L، وضعیت class5 و U2R وضعیت Probe را نشان می‌دهد. شکل ۵ نمودار ROC را نشان می‌دهد.



شکل ۵- نمودار ROC برای کلاس‌های مجموعه داده KDD-CUP

الگوریتم پیشنهادی توسط زبان برنامه‌نویسی MATLAB بر روی سیستمی با مشخصات پردازنده ۶۴ بیتی، رم ۸ گیگابایت و سی پی یو ۷ هسته‌ای پیاده‌سازی شده است که نتایج روش پیشنهادی برای مجموعه داده KDD-CUP 99 در جدول زیر آمده است.

جدول ۴- پیاده‌سازی روش پیشنهادی بر روی مجموعه داده KDD-CUP 99

	F-measure	Recall	Precision	FPR	TPR
Normal	99.86 %	99.99%	99.72%	0.28%	99.99%
Dos	99.88 %	99.77%	99.99%	0.01%	99.77%
R2L	99.83 %	100.00%	99.66%	0.34%	100.00%
U2R	45.45 %	99.99%	29.41%	41.38%	99.99%
Probe	99.95 %	99.89%	100.00%	0.00%	99.89%

از نتایج جدول بالا می‌توان متوجه شد که حمله Probe بهترین مقدار اف-امتیاز را دارد بنابراین این حمله با دقت و صحت بهتری قابل تشخیص است. و حمله U2R کمترین مقدار تشخیص را داراست زیرا تعداد رکوردهای این نوع حمله، نسبت به بقیه حملات بسیار کمتر است. هرچه مقدار معیار نرخ هشدار غلط کمتر باشد بهتر است بنابراین به ترتیب کلاس‌های Normal, Dos, Probe, R2L و U2R هشدار غلط کمتری را دارند. در بین کلاس‌های حمله، U2R بدترین نرخ هشدار غلط را دارد اما نرخ هشدار درست بهتری دارد.

به‌طور کلی، دقت تشخیص روش پیشنهادی برای مجموعه داده kdd، 99.743% است. همچنین ماتریس درهم‌ریختگی به دست آمده از روش پیشنهادی برای مجموعه داده kdd در نمودار زیر آمده است.

		Confusion Matrix						
		1	2	3	4	5		
Actual Class	1	20335 53.8%	57 0.2%	0 0.0%	1 0.0%	0 0.0%	99.7%	0.3%
	2	1 0.0%	13584 35.9%	1 0.0%	0 0.0%	0 0.0%	100.0%	0.0%
	3	0 0.0%	0 0.0%	291 0.8%	0 0.0%	1 0.0%	99.7%	0.3%
	4	0 0.0%	0 0.0%	0 0.0%	15 0.0%	36 0.1%	29.4%	70.6%
	5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3469 9.2%	100%	0.0%
		1	2	3	4	5		
		100.0%	99.6%	99.7%	93.8%	98.9%	99.7%	0.3%
		0.0%	0.4%	0.3%	6.3%	1.1%		
		Target Class						

شکل ۴: ماتریس درهم‌ریختگی مجموعه داده KDD-CUP



## تحلیل تأخیر:

ما ۵۵۰ گره اینترنت اشیا را در یک محیط شبیه‌سازی شده مستقر کردیم. از میان ۵۵۰ گره اینترنت اشیا، ۵۰ دستگاه مه و ۵۰۰ دستگاه لبه هستند. فرض می‌کنیم که هر دستگاه لبه حداقل به یک دستگاه مه متصل است. فاصله بین شبکه اینترنت اشیا تا ابر راه دور حدود ۵۰۰ مایل و فاصله بین دستگاه اینترنت اشیا و گره‌های مه محلی آن در ۱۰ متر است. ما ترافیک اینترنت اشیا را با استفاده از توزیع پواسون که به‌طور گسترده برای مدل‌سازی ترافیک شبکه استفاده می‌شود، تولید کردیم. در فرآیند پواسون، تعداد بسته‌های ورودی و/یا طول بسته‌ها به‌صورت توزیع نمایی مدل‌سازی می‌شوند. تأخیر در صف و نرخ افت بسته متناسب با تراکم ترافیک است. آزمایش شبیه‌سازی ۲۰۰ نمونه زمانی به طول می‌انجامد. سرعت انتشار داده‌ها برابر با  $2 \times 10^8$  متر بر ثانیه تنظیم شده است. انتقال دستگاه به ابر با ظرفیت تئوری نرخ بیت خالص تا ۱۰۰ مگابیت بر ثانیه در لینک پایین و ۵۰ مگابیت بر ثانیه در لینک بالا استفاده می‌کند [20].

## تخمین تأخیر:

تأخیر به‌عنوان میانگین زمان صرف‌شده برای پاسخ به درخواست خدمات تعریف می‌شود، در اینجا، زمان مورد نیاز برای شناسایی یک فعالیت عادی یا غیرعادی مدنظر است. این را می‌توان به‌صورت زیر بیان کرد:

$$D_i = \frac{\sum_{i=1}^N (D_{Propagation\ i} + D_{Transmission\ i} + D_{Queueing\ i} + D_{Processing\ i})}{N} \quad (8)$$

در معادله 8،  $i$  نشان‌دهنده  $i$ امین دستگاه است.  $D$  نشان‌دهنده تأخیر است،  $t$  نشان‌دهنده زمان و  $N$  تعداد کل دستگاه‌های فعال است. تأخیر انتشار، زمان انتقال یک بیت داده از فرستنده به گیرنده است که به‌صورت زیر تعریف می‌شود.

$D_{propagation} = d/S$   
(که در آن  $d$  نشان‌دهنده فاصله بین دو طرف ارتباط از طریق پیوند ارتباطی، و  $S$  نشان‌دهنده سرعت انتقال است). داده‌های منتشرشده روی پیوندی به شکل سیگنال‌های الکترومغناطیسی با سرعتی معین (که توسط رسانه‌ای که داده‌ها از آن عبور می‌کنند تعیین می‌شود)، حرکت می‌کنند.

تأخیر انتقال به زمان مورد نیاز برای ارسال تمام داده‌های یک بسته به لینک انتقال اشاره دارد.

$$D_{Transmission} = L/R$$

(که در آن  $L$  طول بسته و  $R$  نرخ انتقال است).  $D_{Transmission}$  بر اساس اندازه داده‌ها و پهنای باند پیوند انتقال (بر حسب bps) تعیین می‌شود. تأخیر در صف،  $D_{Queueing}$ ، زمان انتظاری است که بسته به آن نیاز دارد تا در بافر روتر بماند تا پردازش شود. تأخیر در صف توسط سه عامل تعیین می‌شود، از جمله نرخ داده‌هایی که به روتر می‌آیند، پهنای باند پیوند خروجی روتر و وضعیت ترافیک شبکه.

تأخیر پردازش،  $D_{Processing}$ ، مدت زمانی است که روتر برای پردازش بسته صرف می‌کند. تأخیر بستگی به سرعت پردازش روتر دارد. ما فرض می‌کنیم که  $D_{Queueing}$  و  $D_{Processing}$  در اینجا ثابت هستند.

زمان تأخیر در سه حالت مقایسه شده است.

(۱) سیستم تشخیص نفوذ دو سطحی KNN-MLP (روش

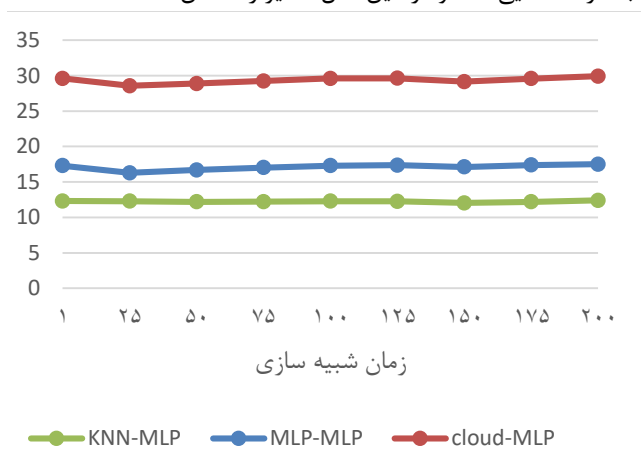
پیشنهادی)

(۲) سیستم تشخیص نفوذ دو سطحی MLP-MLP

(۳) سیستم تشخیص نفوذ در ابر بدون مه

ما شبکه IOT را برای آزمایش میانگین تأخیر آن در یک دوره ۲۰۰ باری شبیه‌سازی کردیم. شکل ۶ متوسط تأخیر را در دو مجموعه داده مورد بررسی نشان می‌دهد.

همان‌طور که شکل ۶ نشان می‌دهد زمان تأخیر در روش پیشنهادی نسبت به بقیه روش‌ها کمتر است سپس روش MLP-MLP و در نهایت روش تشخیص نفوذ در ابر بدون مه بیشترین تأخیر را دارد. روش پیشنهادی تقریباً ۴۰٪ تأخیر کمتری نسبت به روش MLP-MLP دارد و ۱۳۹٪ تأخیر کمتری نسبت به تشخیص نفوذ در ابر بدون مه دارد. به‌طور خلاصه، این IDSهای دولایه می‌توانند به‌طور مؤثر نفوذهای شبکه را شناسایی کنند و در عین حال تأخیر را کاهش دهند.



شکل ۶ - مقایسه زمان تأخیر در مجموعه داده KDD-CUP 99

## ۵. نتیجه‌گیری و پیشنهاد کارهای آینده

امنیت در اینترنت اشیا یکی از چالش‌های مهم در این زمینه می‌باشد. سیستم‌های تشخیص نفوذ سنتی بایستی خود را با ویژگی‌ها و محدودیت‌های مخصوص به این شبکه‌ها مطابقت دهند. در این مقاله یک روش جدید، قدرتمند، سبک و دولایه بر اساس محاسبات مه در محیط ابری ارائه داده‌ایم. روش پیشنهادی یک روش سبک وزن بر اساس الگوریتم نزدیکترین بهبود یافته است [18]. این روش تنها گره‌هایی را که در فضا نزدیک یکدیگر هستند مقایسه می‌کند و از محاسبات غیر ضروری اجتناب می‌کند. در نهایت پیچیدگی کلی الگوریتم را می‌توان به  $O(n)$  کاهش داد و این مقدار در مقابل پیچیدگی الگوریتم

- [8] C. B. W. ., R. B. M. B. M. S. ., G. d. S. V. Cristiano Antonio de Souza, "Hybrid approach to intrusion detection in fog-based IoT environments," *Computer Networks*, vol. 180, 2020.
- [9] R. M. C. K. S. d. A. B.B. Zarpelão, "A survey of intrusion detection in Internet of Things," *Journal of Network and Computer Applications*, vol. 84, pp. 25-37, 2017.
- [10] L. G. P. M. Q. A. X. L. M. T. D. K. H. S. B. A. K. Victor Chang, "A Survey on Intrusion Detection Systems for Fog and Cloud Computing," *future internet*, vol. 14, no. 89, 2022.
- [11] F. V. A. P. P. J. H. T. & T. H. Hosseinpour, "An intrusion detection system for fog computing and IoT based logistic systems using a smart data approach," *International Journal of Digital Content Technology and its Applications*, p. 10, 2016.
- [12] "http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html," [Online].
- [13] "https://www.unb.ca/cic/datasets/ids.html," [Online].
- [14] A. A. K. N. G. Yasmine Labiod, "Fog Computing-Based Intrusion Detection Architecture to Protect IoT Networks," *Wireless Personal Communications*, vol. 125, pp. 231-259, 2022.
- [15] K. P. S. P. P. R. Vinayakumar, "Evaluating effectiveness of shallow and deep networks to intrusion detection system," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017.
- [16] X. Z. X. L. X. L. F. & Y. L. An, "Sample selected extreme learning machine based intrusion detection in fog computing and MEC," *Wireless Communications and Mobile Computing*, 2018.
- [17] "Distributed attack detection scheme using deep learning approach for internet of things," *Future Generation Computer Systems*, vol. 82, pp. 761-768, 2018.
- [18] K. ., S. S. S. Prabavathy, "Design of Cognitive Fog Computing for Intrusion Detection in Internet of Things," *Journal of Communications and Networks*, pp. 291-298, 2018.
- [19] L. Z. S. W. C. H. T. L. Victor C. M. Leung, "Intrusion detection system based on decision tree over big data in fog environment," *Big IoT Data Analytics in Fog Computing*, 2018.
- [20] F.-B. Mocnik, "An improved algorithm for dynamic nearest-neighbour models," *Journal of Spatial Science*, vol. 67, no. 3, pp. 411-438, 2022.
- [21] J. L. ., B. Souradip Roy, "A Two-layer Fog-Cloud Intrusion Detection Model for IoT Networks," *Internet of Things*, vol. 19, 2022.

نزدیکترین همسایگی که از مرتبه  $O(n^2)$  می‌باشد بسیار مناسب‌تر می‌باشد. همچنین برای پیاده‌سازی در محیط‌های اینترنت اشیا که دارای محدودیت‌های منابع است، مناسب می‌باشد. روش پیشنهادی به کمک یکی از جدیدترین دیتاست‌های اینترنت اشیا با نام KDD-CUP [10] ارزیابی شده‌است. به کمک ارزیابی‌ها نشان دادیم که روش پیشنهادی موجب بهبود در Precision, Recall, Accuracy, F-measure و FPR و TPR می‌شود. طبق محاسبات انجام‌شده، روش پیشنهادی از دیدگاه پیچیدگی زمانی، بهبود قابل‌ملاحظه‌ای را در لایه مه و ابر نشان می‌دهد. روش را در سه حالت: الف- استفاده از الگوریتم نزدیکترین همسایگی بهبودیافته در لایه مه و MLP neural network در لایه ابر، ب- روش پایه (استفاده از الگوریتم MLP neural network در لایه مه و ابر)، ج- روشی که لایه مه وجود ندارد و فقط لایه ابر موجود است، مورد ارزیابی قرار دادیم. نتایج ارزیابی بیانگر تأخیر کمتر روش پیشنهادی (روش الف) به میزان ۴۰٪ نسبت به روش حالت (ب) و ۱۳۹٪ تأخیر کمتر نسبت به روش (حالت ج) می‌باشد. به عنوان کار آینده می‌توان روش پیشنهادی را بر روی سایر دیتاست‌ها پیاده‌سازی و با سایر روش‌های یادگیری ماشین مقایسه کرد. همچنین ما قصد داریم تا IDS پیشنهادی را روی شبکه‌های واقعی IoT مستقر کنیم تا عملکرد آن را آزمایش کنیم. ترکیب سایر روش‌های یادگیری ماشین با الگوریتم ارائه‌شده در لایه مه نیز می‌تواند کار آینده در راستای این تحقیق باشد.

#### References :

- [1] M. T. S. MohammadJavad Zand, "Improvement of IOT Security in ZigBee Network Using AES256 Algorithm," *Intelligent Multimedia Processing and Communication Systems (IMPCS)*, no. 2, p. 53, 2020.
- [2] W. H. Hassan, "Current research on Internet of Things (IoT) security: A survey.," *Computer Networks*, vol. 148, pp. 283-294, 2019.
- [3] A. G. Rozbeh Hosseinzhad, "Intrusion Detection System in The Cloud Computing Using Heterogeneity Detection Technique," *Intelligent Multimedia Processing and Communication Systems (IMPCS)*, no. 1, p. 39, 2021.
- [4] R. B. M. ., L. C. M. W. ., G. A. G. Cristiano Antonio de Souza Carlos Becker Westphall, "Intrusion detection and prevention in fog based IoT environments: A Systematic Literature Review," *Computer Networks*, 2022.
- [5] G. K. A. S. J. V. C. Koliass, "DDoS in the IoT: Mirai and Other Botnets," *Computer*, vol. 50, no. 7, pp. 80-84, 2017.
- [6] S. Y. H. Tanaka, "On modeling and simulation of the behavior of," in *2017 IEEE International Symposium on*, 2017.
- [7] V. K. P. M. A.C. Panchal, "Security issues in IIoT: a comprehensive survey of attacks on IIoT and its countermeasures," in *2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN)*, 2018.