Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
DOI: 10.22034/jscit.2025.220288
Volume 14, Number 1, Spring 2025, pp. 45-56
Received: 12/06/2024, Revised: 04/20/2025, Accepted: 05/04/2025

Providing an Intelligent Method for Intrusion Detection in Cloud Computing Environment by Optimizing Neural Network Using Genetic Algorithm

Wahab Aminiazar[1*], Rasoul Farahi[2] and Fatemeh Shateri Aliabad[3]

1*- Department of Computer Engineering and Information Technology, Mah.C., Islamic Azad University, Mahabad, Iran.
2- Department of Computer Engineering and Information Technology, Mah.C., Islamic Azad University, Mahabad, Iran.
3- Department of Computer Engineering and Information Technology, Khal.C., Islamic Azad University, Khalkhal, Iran.

[1*]Aminiazar@iau-mahabad.ac.ir, [2] Stu.rasoul-farahi@iaut.ac.ir, and F.Shateri@iau.ir [3]

Corresponding author's address: Wahab Aminiazar, Faculty of Electrical and Computer Engineering, Islamic Azad University Mahabad Branch, Mahabad, Iran.

Abstract: Given the increasing expansion of cloud computing applications in recent decades, the emergence of new challenges in this field is a given. Among these, security has always been a fundamental requirement for utilizing these environments. One of the significant issues in the realm of cloud computing security is intrusion detection, which, if appropriate measures and sufficient accuracy are not taken, can lead to risks such as data breaches and potential misuse. Therefore, accurate intrusion detection and enhancing the efficiency of protective systems are of special importance. Over the years, researchers have made extensive efforts to optimize this challenge. However, many existing methods have faced the problem of relatively low detection accuracy. One of the contributing factors to this issue is the heterogeneity among irrelevant and redundant information features in the datasets. This article presents an innovative solution for intrusion detection in cloud infrastructure, relying on an improved neural network classification based on genetic algorithms. This approach aims to improve detection accuracy. In this regard, the principal component analysis (PCA) algorithm has been used initially to reduce the dimensions of the dataset. Then, by utilizing artificial neural network classification, normal traffic is separated from malicious traffic. Subsequently, the parameters of the artificial neural network are optimized using a genetic algorithm. Finally, the performance of the proposed method is evaluated and tested on the standard and comprehensive NSL-KDD dataset, as an indicator for assessing intrusion detection systems, in the MATLAB environment, and compared with the results obtained from three other prominent methods. The results of this comparison show that the proposed method has demonstrated better performance with a detection accuracy of 99.41% compared to other methods and is capable of identifying attacks with higher precision.

Keywords- Intrusion Detection, Cloud Computing, Attacks, Neural Network, Genetic Algorithm

مجله علمی رایانش نرم و فناوری اطلاعات
دانشگاه صنعتی نوشیروانی بابل
صفحه مجله: jscit.nit.ac.ir
شناسه دیجیتال: 10.22034/jscit.2025.220288
جلد ۱۴، شماره ۱، بهار ۱۴۰۴ ، صفحه ۴۵-۵۶
دریافت: ۱۴۰۳/۰۹/۱۶، بازنگری: ۱۴۰۴/۰۱/۳۱، پذیرش: ۱۴۰۴/۰۲/۱۴

ارائه یک روش هوشمند جهت تشخیص نفوذ در محیط رایانش ابری با بهینه‌سازی شبکه عصبی توسط الگوریتم ژنتیک

وهاب‌امینی‌آذر[1*]، رسول فرحی[2]، فاطمه شاطری علی‌آباد[3]
۱*- گروه مهندسی برق و کامپیوتر، واحد مهاباد، دانشگاه آزاد اسلامی، مهاباد، ایران.
۲- گروه مهندسی برق و کامپیوتر، واحد مهاباد، دانشگاه آزاد اسلامی، مهاباد، ایران.
۳- گروه مهندسی برق و کامپیوتر، واحد خلخال، دانشگاه آزاد اسلامی، خلخال، ایران.
[1*]Aminiazar@iau-mahabad.ac.ir, [2] Stu.rasoul-farahi@iaut.ac.ir, and [3] F.Shateri@iau.ir

* نشانی نویسنده مسئول: وهاب امینی آذر، مهاباد، خیابان دانشگاه، دانشگاه آزاد اسلامی واحد مهاباد، دانشکده مهندسی برق و کامپیوتر.

چکیده- با توجه به گسترش فزاینده‌ی کاربرد رایانش ابری در دهه‌های اخیر، ظهور چالش‌های جدید در این عرصه، امری بدیهی است. در این میان، امنیت به عنوان یکی از الزامات اساسی بهره‌برداری از این محیط‌ها، همواره مورد توجه بوده است. از جمله مسائل حائز اهمیت در حوزه امنیت رایانش ابری، شناسایی نفوذ به سیستم است که در صورت عدم اتخاذ تدابیر مناسب و دقت کافی، می‌تواند منجر به مخاطراتی همچون افشای اطلاعات و سوءاستفاده‌های احتمالی شود. بنابراین، تشخیص دقیق نفوذ و ارتقای کارایی سیستم‌های حفاظتی، از اهمیت ویژه‌ای برخوردار است. در طول سالیان متمادی، پژوهشگران تلاش‌های گسترده‌ای در جهت بهینه‌سازی این چالش معطوف داشته‌اند. با این حال، بسیاری از روش‌های موجود، با مشکل دقت تشخیص نسبتاً پایین مواجه بوده‌اند. یکی از عوامل موثر در این امر، ناهمگونی میان ویژگی‌های اطلاعاتی نامرتبط و زائد در مجموعه داده‌ها است. در این مقاله، یک راهکار نوین برای شناسایی نفوذ در زیرساخت ابری ارائه شده است که متکی بر دسته‌بندی شبکه‌ی عصبی بهبودیافته مبتنی بر الگوریتم ژنتیک است. هدف از این رویکرد، ارتقای دقت تشخیص است. در این راستا، ابتدا به منظور کاهش ابعاد مجموعه داده، از الگوریتم تحلیل مولفه‌های اصلی (PCA) استفاده شده است. سپس، با بهره‌گیری از دسته‌بندی شبکه‌ی عصبی مصنوعی، ترافیک نرمال از ترافیک‌های تهاجمی تفکیک داده می‌شود. در ادامه، پارامترهای شبکه‌ی عصبی مصنوعی با به‌کارگیری الگوریتم ژنتیک، بهینه‌سازی می‌شود. در نهایت، عملکرد روش پیشنهادی با آزمایش و ارزیابی بر روی مجموعه داده‌ی استاندارد و جامع NSL-KDD، به عنوان شاخصی برای ارزیابی سیستم‌های تشخیص نفوذ، در محیط متلب مورد سنجش قرار گرفته و با نتایج حاصل از سه روش مطرح دیگر مقایسه شده است. نتایج این مقایسه نشان می‌دهد که روش پیشنهادی با دقت تشخیص ۹۹,۴۱٪، در مقایسه با سایر روش‌ها، عملکرد بهتری از خود نشان داده و قادر است حملات را با دقت بالاتری شناسایی نماید.

واژه‌های کلیدی: تشخیص نفوذ، رایانش ابری، حملات، شبکه عصبی مصنوعی، الگوریتم ژنتیک

۱- مقدمه

در عصر کنونی که فناوری اطلاعات به طور فزاینده‌ای در تار و پود زندگی بشر تنیده شده است، نیاز به پردازش‌های محاسباتی در هر زمان و مکانی بیش از پیش احساس می‌شود. این امر، ضرورت دسترسی به منابع محاسباتی قدرتمند بدون نیاز به سرمایه‌گذاری‌های کلان در سخت‌افزار و نرم‌افزار را برجسته می‌سازد [۱]. در این راستا، رایانش ابری[1] به عنوان پاسخی نوین و

راهگشا، امکان دسترسی سهل و منعطف به مجموعه‌ای وسیع از منابع محاسباتی پیکربندی‌پذیر، نظیر شبکه‌ها، سرورها، فضای ذخیره‌سازی، برنامه‌های کاربردی و سرویس‌ها را فراهم آورده است. [۲]. این دسترسی، با حداقل نیاز به مدیریت منابع و دخالت مستقیم ارائه‌دهنده سرویس، به سرعت قابل استقرار و آزادسازی است. به این ترتیب، رایانش ابری امکان دسترسی آسان، توزیع‌شده و فراگیر به منابع محاسباتی تجمیعی و مشترک را میسر می‌سازد. با این حال، این فناوری نوظهور، علی‌رغم مزایای انکارناپذیر، با چالش‌های امنیتی قابل توجهی نیز مواجه است. حفظ امنیت و حریم خصوصی داده‌ها، مستلزم اتخاذ سیاست‌ها و راهکارهای اطمینان‌بخش است که این امر، به عنوان سدی اساسی در برابر پذیرش گسترده این فناوری عمل می‌کند [۳]. در واقع، چالش بنیادین در این حوزه، تضمین امنیت و حفاظت از حریم خصوصی کاربران است. همزمان با گسترش روزافزون تبادل اطلاعات و استفاده از سیستم‌های برخط، میزان حملات سایبری و نفوذ به سیستم‌های اطلاعاتی نیز افزایش یافته است. به منظور پیشگیری از سوء استفاده از اطلاعات و رخنه در سیستم‌های حساس و حیاتی، تامین امنیت این سیستم‌ها امری ضروری است [۲و۳]. از آنجا که ایجاد سیستم‌های کامپیوتری عاری از هرگونه آسیب‌پذیری و نقص امنیتی، از نظر فنی عملاً غیرممکن است، تشخیص نفوذ به عنوان یک حوزه تحقیقاتی حائز اهمیت در سیستم‌های کامپیوتری مورد توجه قرار گرفته است [۳]. سیستم تشخیص نفوذ[۲]، مجموعه‌ای از سخت‌افزار و نرم‌افزار است که به منظور نظارت بر شبکه‌های کامپیوتری و شناسایی فعالیت‌های مخرب یا نقض سیاست‌های مدیریتی و امنیتی طراحی شده است. این سیستم، گزارش‌های حاصله را به بخش مدیریت شبکه ارائه می‌دهد و به این ترتیب، وظیفه شناسایی و تشخیص هرگونه استفاده غیرمجاز از سیستم، سوء استفاده و یا آسیب‌رسانی توسط کاربران داخلی و خارجی را بر عهده دارد. شایان ذکر است که هدف اصلی سیستم تشخیص نفوذ، جلوگیری از حمله نیست، بلکه صرفاً کشف و شناسایی حملات و تشخیص آسیب‌پذیری‌های امنیتی در سیستم‌ها یا شبکه‌های کامپیوتری و اطلاع‌رسانی به مدیر سیستم است. این سیستم‌ها، عموماً در کنار دیواره آتش و به عنوان یک مکمل امنیتی برای آن مورد استفاده قرار می‌گیرند [۴].

با توجه به تعاریف و مفاهیم مطرح شده، رویکردهای سنتی متعددی برای دسته‌بندی روش‌های تشخیص نفوذ در رایانش ابری ارائه شده است که هر یک، مزایا و معایب خاص خود را دارند. پژوهش‌های نوین، به طور فزاینده‌ای بر استفاده از الگوریتم‌های یادگیری ماشین متمرکز شده‌اند. در همین راستا، هدف از این مقاله، ارائه یک سیستم تشخیص نفوذ با انتخاب ویژگی بهینه و روش دسته‌بندی مناسب برای فضای ابری است، به گونه‌ای که نرخ تشخیص بالاتری نسبت به روش‌های موجود داشته باشد. بر این اساس، در این مقاله، به عنوان جنبه نوآورانه، از یک شبکه عصبی بهبود یافته مبتنی بر الگوریتم ژنتیک به منظور شناسایی حملات در فضای ابری استفاده خواهد شد. در ابتدا، پس از انجام پیش‌پردازش‌های لازم، شامل حذف داده‌های ناموجود و نرمال‌سازی داده‌ها، در مرحله کاهش ابعاد ویژگی‌ها، از الگوریتم تحلیل مؤلفه‌های اصلی استفاده خواهد شد تا ابعادی که دارای واریانس یا اطلاعات کمی هستند، حذف شده و ابعاد با اطلاعات بیشتر انتخاب شوند. در گام بعدی، وزن‌ها و بایاس‌های شبکه عصبی با استفاده از الگوریتم ژنتیک بهینه‌سازی خواهند شد. از آنجا که الگوریتم‌های یادگیری شبکه عصبی، نظیر الگوریتم پس انتشار خطا، وزن‌ها و بایاس‌های شبکه را با محاسبه خطا تنظیم می‌کنند، این شبکه‌ها معمولاً با انتخاب تصادفی وزن‌های اولیه آموزش می‌بینند که این امر، منجر به کاهش سرعت همگرایی و گیر افتادن در مینیمم محلی می‌شود. در نتیجه، شبکه عصبی غالباً در یافتن یک راه‌حل مناسب ناتوان است. بنابراین، در این مقاله، به جای استفاده از الگوریتم پس انتشار سنتی برای آموزش شبکه عصبی، مقدار بهینه وزن‌ها و بایاس‌های شبکه عصبی با استفاده از الگوریتم ژنتیک تعیین خواهد شد. این الگوریتم، برخلاف سایر راهکارهای بهینه‌سازی که عملیات را از یک نقطه آغاز می‌کنند، از چندین نقطه در فضای پاسخ به جستجوی جواب بهینه می‌پردازد و به طور مؤثری از مشکل گیر افتادن در مینیمم محلی جلوگیری می‌کند. بر همین اساس، این رفتار الگوریتم می‌تواند باعث افزایش دقت طبقه‌بندی‌کننده شود.

این مقاله از پنج بخش تشکیل شده است. در بخش بعدی، خلاصه‌ای از روش‌های قبلی تشخیص نفوذ در بستر رایانش ابری را بررسی می‌کنیم. در بخش سوم، جزئیات رودیکرد پیشنهادی را ارائه می‌کنیم. نتایج شبیه‌سازی و ارزیابی روش پیشنهادی در بخش چهارم شرح داده شده است و در نهایت بخش پنجم به نتیجه‌گیری و کارهای آتی می‌پردازد.

۲- کارهای مرتبط

لوپز و همکاران در [۵] ویژگی‌های ترافیک مبتنی بر جریان را برای تجزیه و تحلیل واریانس در الگوها میان بسته‌های عادی در مقابل غیرعادی پیشنهاد نمودند. آن‌ها رویکردهای مختلف دسته‌بندی نظارت شده را با استفاده از پارامترهایی مانند نرخ منفی کاذب، صحت، نرخ تشخیص، زمان اجرا و زمان صرف شده برای آموزش ارزیابی نمودند و به این نتیجه رسیدند که الگوریتم جنگل تصادفی مبتنی بر درخت تصمیم رویکرد بهینه شده می‌باشد، که در آن یک

شبکه عصبی متراکم بر روی انواع حملات منع سرویس به خوبی عمل می‌نماید.

میزینی و همکاران [۶] یک رویکرد ترکیبی جدید برای شبکه آنومالی سیستم تشخیص نفوذ با استفاده از الگوریتم‌های کلونی زنبور عسل[۳] و آدابوست برای به دست آوردن نرخ تشخیص بالاتر و نرخ مثبت کاذب پایین‌تر پیشنهاد نمودند. در این رویکرد برای انتخاب ویژگی از الگوریتم کلونی زنبور عسل و برای ارزیابی و دسته‌بندی ویژگی‌ها از الگوریتم آدابوست[۴] استفاده شده است.

بویوویچ و همکارانش [۷] یک رویکرد ترکیبی برای شناسایی حملات منع از سرویس معرفی نمودند، که تشخیص‌های مبتنی بر حجم و ویژگی را ترکیب می‌نماید. این روش برای تصمیم‌گیری به رویکرد میانگین متحرک‌نمایی وابسته است و برای مقادیر آنتروپی و دنباله زمانی تعداد بسته‌ها استفاده می‌شود.

در مرجع [۸] یک سیستم تشخیص نفوذ مبتنی بر یادگیری عمیق برای حملات منع از سرویس بر اساس سه روش، یعنی شبکه عصبی کانولوشن، شبکه عصبی عمیق و شبکه عصبی تکراری پیشنهاد گردید. عملکرد هر روش بر اساس دو نوع دسته‌بندی (چند کلاسی و دودویی)، با استفاده از دو مجموعه، داده ترافیک واقعی - IoT TON و CIC-DDoS2019 تجزیه و تحلیل گردید.

در مقاله [۹] رویکردی ترکیبی برای شناسایی حملات مختلف براساس ماشین یادگیری افراطی با رویکرد تابع کرنل ترکیبی در محیط ابر براساس الگوریتم‌های جستجوی گرانشی و الگوریتم تکامل تفاضلی ارائه شده است. این رویکرد ترکیبی برای بهینه‌سازی پارامترهای ماشین یادگیری افراطی با رویکرد تابع کرنل ترکیبی به کار گرفته شده‌است، که توانایی‌های بهینه‌سازی محلی و سراسری آن را در طول حملات پیش‌بینی بهبود می‌بخشد. همچنین در این رویکرد از الگوریتم تجزیه و تحلیل مولفه اصلی کرنل برای کاهش ابعاد و استخراج ویژگی داده‌های تشخیص نفوذ استفاده شده است. رویکرد پیشنهادی بر روی مجموعه داده‌های KDD99 و UNSW-NB مورد آزمایش و شبیه‌سازی قرار گرفته است. نتایج شبیه‌سازی بر روی مجموعه داده‌های بکار گرفته شده نشان‌دهنده این است که روش پیشنهادی دارای دقت بالا در تشخیص حملات نسبت به روش‌های پیشین می‌باشد و صرفه‌جویی در زمان از مزایای این رویکرد می‌باشد.

سامبانگی و همکارانش [۱۰] یک مدل یادگیری ماشین برای تشخیص نفوذ در محیط ابر را براساس تحلیل رگرسیون خطی چندگانه ارائه داده‌اند. در چارچوب پیشنهادی از رویکرد به دست آوردن اطلاعات برای حمل انتخاب ویژگی‌ها استفاده شده است. این رویکرد یک مدل توسعه یافته در چندین برنامه کاربردی مبتنی بر داده‌کاوی است. بنابراین با استفاده از تکنیک انتخاب ویژگی‌ها، ویژگی‌های مهمی که توسط مدل‌های مختلف پیش‌بینی داده‌کاوی ارائه می‌گردند، برای انجام تحلیل رگرسیون خطی چندگانه در نظر گرفته می‌شوند. سپس رفتار ویژگی‌های مهم انتخاب‌شده و اعمال تحلیل رگرسیون خطی چندگانه برای مجموعه داده CICIDS 2017 مورد تجزیه و تحلیل قرار می‌گیرد.

گووش و همکاران [۱۱] یک سیستم تشخیص نفوذ را در ابر پیشنهاد نمودند که امنیت را بر اساس مفهوم انتخاب ویژگی با استفاده از الگوریتم کرم شب تاب ارائه می‌نماید. رویکرد ترکیبی توسعه یافته بر روی مجموعه داده NSL-KDD مورد ارزیابی قرار گرفت و مشخص گردید فضای ذخیره‌سازی کمتری را به دلیل کاهش تعداد ابعاد از انتخاب ویژگی مصرف می‌نماید. همچنین به زمان آموزش کمتری نیاز دارد. در نتیجه عملکرد دسته‌بندی را بهبود می‌بخشد.

در مرجع [۱۲]، روشی برای شناسایی حملات منع سرویس توزیع شده (DDOS[۵]) با استفاده از چارچوب شبکه عصبی مصنوعی (ANN[۶]) بیان شده است. این چارچوب از طریق الگوریتم رقابتی امپریالیستی تحت آموزش قرار می‌گیرد. روش پیشنهادی با استفاده از مجموعه داده‌های KDDTest و KDDTest-21 مورد ارزیابی قرار گرفته است. نتایج شبیه‌سازی نشان می‌دهد که روش پیشنهادی برای مجموعه‌داده‌های مورد ارزیابی به ترتیب به دقت تشخیص ٪۸۳,۵ و ٪۷۵,۲ دست یافته است و نسبت به تکنیک‌های مبتنی بر یادگیری ماشین پیشین دارای عملکرد بهتری می‌باشد.

در مقاله [۱۳] یک چارچوب نوین به منظور تشخیص نفوذ براساس الگوریتم حافظه کوتاه مدت عمیق جهت تشخیص الگوهای جریان مخرب یا عادی ترافیک شبکه در محیط ابری پیشنهاد شده است. رویکرد پیشنهاد شده با افزایش نرخ تشخیص حملات مخرب و کاهش زمان محاسباتی از حملات مخرب دریافتی جلوگیری می‌نماید. نتایج ارزیابی نشان داد که سیستم پیشنهادی به منظور تشخیص و پیشگیری حمله مؤثر در عملیاتی با محدودیت منابع در محیط‌های رایانش ابری مناسب می‌باشد.

در مقاله [۱۴] نویسندگان یک سیستم تشخیص نفوذ مؤثر را طراحی کردند که برای محیط‌های ابری از طریق انتخاب ویژگی و استراتژی‌های طبقه‌بندی مجموعه طراحی شده است. این روش از یک رویکرد انتخاب ویژگی تک متغیره برای شناسایی مجموعه ویژگی‌های کاهش‌یافته قابل توجه از مجموعه داده‌های نفوذی استفاده می‌کند. علاوه بر این، در این رویکرد پیشنهادی طبقه‌کننده‌های منفرد به طرز نوآورانه‌ی طبقه‌بندی‌کننده‌های

گروهی را برای ایجاد یک طبقه‌بندی قوی از طریق مکانیزم رأی‌گیری یکپارچه می‌کنند. همچنین رویکرد پیشنهادی بر مجموعه داده‌ها به خوبی تعیین می‌کند که آیا ترافیک شبکه در حالت عادی یا حمله قرار دارد؟ ارزیابی رویکرد پیشنهادی با استفاده از معیارهای مانند ROC[۷]، AUC[۸] در طبقه‌بندی‌کننده‌های مختلف ارزیابی شد. ارزیابی یافته‌ها افزایش قابل توجهی در عملکرد نسبت به روش‌های موجود نشان می‌دهد.

مرجع [۱۵] یک سیستم مبتنی بر یادگیری ماشین جدید را برای تشخیص حملات DDoS در رایانش ابری، با استفاده از یک ماشین یادگیری افراطی تکاملی خود-تطبیقی (SaE-ELM-Ca) معرفی می‌کند. این مدل به طور مستقل استراتژی‌های جهش، نرخ متقاطع و تعداد نورون‌های پنهان را بهینه می‌کند که منجر به بهبود کارایی یادگیری می‌شود. ارزیابی روش پیشنهادی بر روی مجموعه داده‌های NSL-KDD، ISCX IDS 2012، CICIDS 2017 و UNSW-NB15 صورت گرفته است. نتایج ارزیابی نشان می‌دهد که روش پیشنهادی دارای عملکرد بهتری نسبت به مدل‌های مانند شبکه عصبی مصنوعی، درخت تصمیم و الگوریتم ماشین بردار پشتیبان می‌باشد.

مقاله [۱۶] به ارائه یک سیستم تشخیص نفوذ براساس الگوریتم کرم شب‌تاب و الگوریتم جنگل تصادفی پرداخته است. که توانایی تشخیص نفوذ با میزان دقت بالا در رایانش ابری را دارا می‌باشد. علاوه بر این، نتایج جامع به‌دست‌آمده از آزمایش دقیق و ارزیابی کامل مدل نه تنها کارایی و مناسب بودن آن را نشان می‌دهد، بلکه به طور قانع‌کننده شایستگی و اثربخشی رویکرد پیشنهادی را نشان می‌دهد.

مقاله [۱۷] یک روش ترکیبی تشخیص نفوذ را در بستر رایانش ابری مبتنی بر ماشین بردار پشتیبان بهبود یافته معرفی می‌کند. در رویکرد پیشنهادی، فرآیند بهینه‌سازی پارامترهای الگوریتم ماشین بردار پشتیبان با استفاده از یک رویکرد ترکیبی بهینه‌سازی مبتنی بر الگوریتم شاهین هریس و ازدحام ذرات انجام می‌شود. رویکرد پیشنهادی با استفاده از مجموعه داده NSL-KDD مورد ارزیابی قرار گرفته است. نتایج ارزیابی دقت و کارایی بالای رویکرد پیشنهادی را در شناسایی حملات منع سرویس توزیعی (DDOS) در مقایسه با روش‌های پیشین نشان می‌دهد.

مقاله [۱۸] یک روش ترکیبی مبتنی بر الگوریتم‌های تراکم هسته گاوسی و یادگیری عمیق را به منظور بهبود و افزایش دقت تشخیص حملات منع سرویس (DDOS) در بستر رایانش ابری معرفی می‌کند. این روش پیشنهادی از مجموعه داده‌های استاندارد و جامع تشخیص نفوذ NSL-KDD و CICIDS2017 برای ارزیابی استفاده می‌کند. نتایج ارزیابی نشان‌دهنده دقت و کارایی بالای رویکرد پیشنهادی در شناسایی حملات منع سرویس (DDOS) نسبت به سایر روش‌های مورد مقایسه پیشین می‌باشد. همچنین این روش علیرغم این که دارای کارایی بالا می‌باشد با محدویت‌هایی مانند نیازهای محاسباتی بالا مواجه است.

در مقاله [۱۹] یک مدل ترکیبی از شبکه‌های باور عمیق و LSTM[۹] بهینه‌سازی شده به منظور تشخیص و شناسایی حملات منع سرویس (DDOS) در محیط ابری ارئه شده است. در این چارچوب پیشنهادی فرایند بهینه‌سازی پارامترهای مدل ترکیبی با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات انجام می‌گیرد. ارزیابی چارچوب پیشنهادی با استفاده از مجموعه داده NSL-KDD مورد ارزیابی و آزمایش قرار گرفته است. نتایج تجربی ارزیابی نشان می‌دهد که این چارچوب پیشنهادی از نظر معیارهایی مانند نرخ دقت و تشخیص نسبت به مدل‌های دیگر دارای عملکرد بهتری می‌باشد.

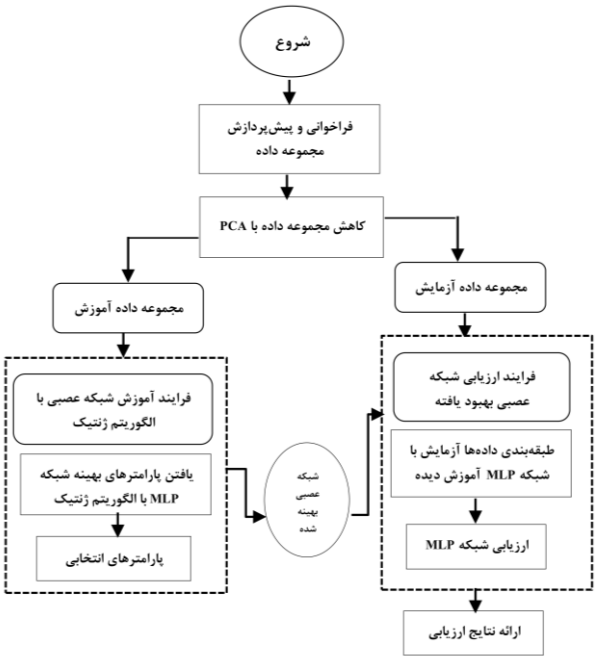
۳- روش پیشنهادی

در این بخش، چارچوب رویکرد پیشنهادی با رویکردی جامع و تفصیلی تبیین می‌گردد. گام نخست، شامل فرآیند آماده‌سازی داده‌ها از طریق پیش‌پردازش است. در ادامه، پس از اتمام مرحله پیش‌پردازش، گام دوم با بهره‌گیری از تکنیک تحلیل مؤلفه‌های اصلی PCA[۱۰] به انجام می‌رسد. در این مرحله، از طریق کاهش ابعاد داده‌ها، مؤلفه‌های کلیدی و مؤثر شناسایی و استخراج می‌گردند که این امر منجر به ارتقاء کارایی مدل و تقلیل پیچیدگی‌های محاسباتی می‌گردد. متعاقباً، داده‌ها پس از طی این دو مرحله، به دو زیرمجموعه مجزای آموزش و آزمایش تقسیم می‌شوند. سپس، فرآیند آموزش شبکه عصبی با استفاده از مجموعه داده آموزشی و بهره‌گیری از الگوریتم ژنتیک به منظور بهینه‌سازی پارامترهای شبکه عصبی صورت می‌پذیرد. در نهایت، عملکرد شبکه عصبی بهینه‌شده با استفاده از مجموعه داده آزمایش مورد ارزیابی قرار می‌گیرد تا میزان تعمیم‌پذیری و دقت پیش‌بینی آن سنجیده شود. در ادامه فلوچارت مراحل روش پیشنهادی در شکل (۱) نشان داده شده است.

۳-۱- گام اول: پیش‌پردازش داده‌ها

در بسیاری از کاربردها تغییرات ویژگی‌ها یکسان نمی‌باشند برای مثال اگر تعداد تلاش‌ها برای دریافت یک پاسخ و شماره درخواست‌های تایید شده کاربر به عنوان دو ویژگی در زیر ساخت ابری در نظر گرفته شوند، این دو ویژگی دارای بازه تغییرات متفاوتی خواهند بود. که این کار باعث می‌شود ویژگی‌های با مقادیر بزرگ

تاثیر بیشتری



شکل ۱: فلوچارت مراحل روش پیشنهادی

بر تابع هزینه بگذارند که لزوماً به معنی مهمتر بودن آن‌ها در مساله نمی‌باشند و بنابراین اثری نامطلوب تلقی می‌گردند. برای رفع این مشکل باید اثر تمام ورودی‌ها با یکدیگر یکسان شود تا شبکه به صورت صحیح آموزش ببیند. در ادامه عمل، چگونکی نرمال‌سازی شرح داده می‌شود. در این روش با یک تبدیل خطی یا غیرخطی، داده‌ها در بازه‌ای که به وسیله کاربر انتخاب می‌شود نرمال می‌شوند. این بازه در کاربردهای شبکه عصبی معمولاً [۱و۱-] انتخاب می‌شوند. برای این کار از رابطه (۱) استفاده می‌شود:

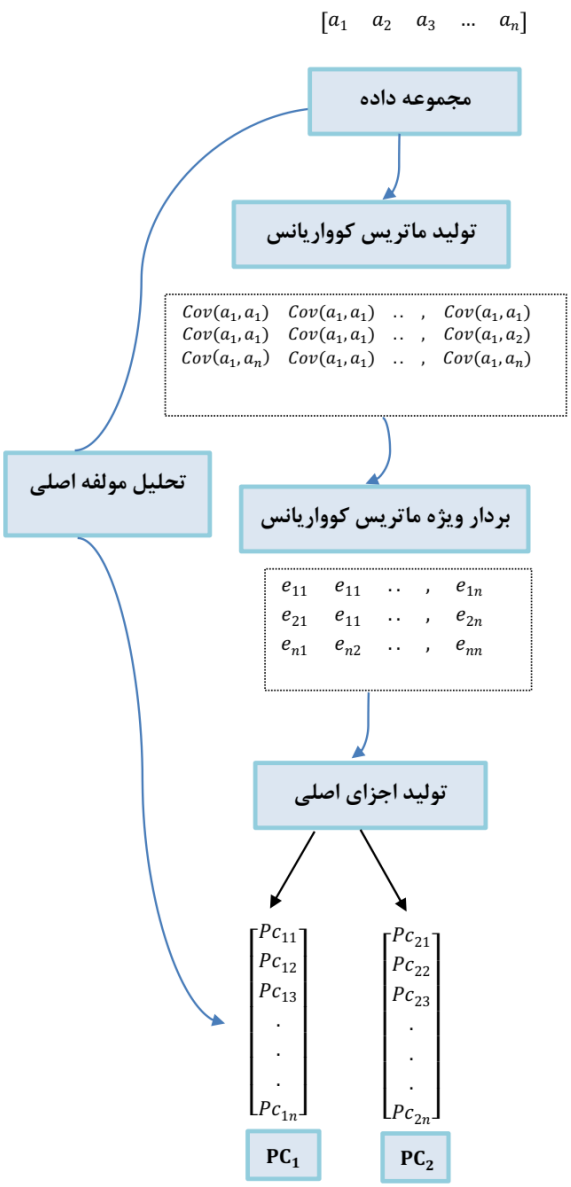
$$X_{normal} = \left[\frac{X_0 - X_{min}}{X_{max} - X_{min}}\right] \times 2 - 1 \qquad (۱)$$

که در رابطه فوق X_{normal} مقدار نرمال شده، X_0 مقدار مشاهده شده و X_{max} و X_{min} به ترتیب مقادیر ماکزیمم و مینیمم داده‌ها می‌باشند.

۳-۲- گام دوم: کاهش ویژگی مجموعه داده

با توجه به ابعاد بالا و تعداد زیاد ویژگی‌های مجموعه داده، از روش تحلیل مولفه‌های اصلی (PCA) برای کاهش بعد و استخراج ویژگی‌های مؤثر استفاده شده است. در ادامه، کاربرد PCA در روش پیشنهادی شرح داده خواهد شد. PCA به عنوان یک روش خطی محبوب برای کاهش بعد، با ترکیب خطی متغیرها، مشاهدات مشابه را گروه‌بندی و خلاصه می‌کند [۲۰]. این روش، ناشی از کاربرد جبر خطی است و در طیف وسیعی از تحلیل‌ها، از علوم شبکه‌های عصبی تا نمودارهای رایانه‌ای، به طور گسترده به کار می‌رود؛ زیرا روشی ساده و غیرپارامتریک برای استخراج اطلاعات مرتبط از داده‌های پیچیده است [۲۰]. هدف اصلی PCA، کاهش تعداد متغیرهای ورودی است. اساساً PCA در چهار مرحله کاهش ابعاد را انجام می‌دهد که در زیر به این چهار مرحله اشاره شده است:

- **مرحله اول:** اولین قدم نرمال‌سازی مقادیر در ویژگی‌های مجموعه داده از بین بردن نتایج مغرضانه است. به عنوان مثال، در یک مجموعه داده می‌توان مقادیر دامنه بزرگتر (۰ تا ۱۰۰) را به عنوان دامنه کوچکتر (۰ تا ۱۰) کنترل کرد. با محاسبه میانگین و واریانس هر صفت حاصل خواهد شد. این کار ممکن است منجر به نتایج مغرضانه شود. برای جلوگیری از آن، PCA تمام مقادیر صفات را در یک مقیاس یکسان نشان می‌دهد [۲۱].
- **مرحله دوم**: دومین قدم در تحلیل مولفه‌های اصلی(PCA) ، محاسبه‌ی ماتریس کوواریانس است. این ماتریس، اندازه‌گیری دقیق همبستگی بین متغیرهای موجود در مجموعه داده را فراهم می‌آورد. عناصر این ماتریس، که می‌توانند مقادیر مثبت یا منفی داشته باشند، میزان و جهت همبستگی بین دو متغیر را مشخص می‌کنند. مقدار مثبت نشان‌دهنده‌ی همبستگی مستقیم و مقدار منفی، نشان‌دهنده‌ی همبستگی معکوس می‌باشد [۲۱].
- **مرحله سوم:** در سومین مرحله مقدار ویژه و بردارهای ویژه جهات را برای محاسبه واریانس بین ویژگی‌های مجموعه داده از ماتریس کوواریانس بدست می‌آوریم.
- **مرحله چهارم:** مرحله‌ی پایانی و چهارم در فرایند استخراج مؤلفه‌های اصلی (PCA)، تولید اجزای اصلی است. هر مؤلفه اصلی، میزان واریانس موجود در متغیرهای اولیه‌ی مجموعه داده را منعکس می‌کند. بهینه‌سازی این فرایند موجب می‌شود که بیشترین میزان واریانس در مؤلفه اصلی اول، بیشترین واریانس باقیمانده در مؤلفه دوم، و به همین ترتیب در مؤلفه‌های بعدی ذخیره شود. این رویکرد، به منظور افزایش کارایی و سرعت پردازش داده‌ها به‌کار گرفته می‌شود [۲۱]. بر همین اساس در شکل (۲) ساختار کلی روش PCA در رویکرد پیشنهادی نشان داده شده است.



شکل۲: ساختار کلی روش PCA در رویکرد پیشنهادی

۳-۳- گام سوم: تعیین داده‌های آموزش و آزمایش

در این مرحله داده‌ها به دو دسته تقسیم می‌شوند: ۸۰ درصد داده‌ها به عنوان داده‌های آموزش و ۲۰ درصد باقیمانده به عنوان داده‌های آزمایش انتخاب می‌شوند. داده‌های آموزش برای آموزش و یادگیری شبکه عصبی و داده‌های آزمایش برای ارزیابی مدل بهبود یافته استفاده می‌شود. در این پژوهش از شبکه عصبی MLP[۱۱] به عنوان دسته‌بندی استفاده شده است و برای بهینه‌سازی وزن‌ها و بایاس‌های شبکه عصبی که پارامترهای مؤثر در طبقه‌بندی هستند، از الگوریتم ژنتیک استفاده شده است که در ادامه به تشریح روش پرداخته شده است.

۳-۴- گام چهارم: ساختار شبکه عصبی پیشنهادی

در این مرحله ساختار شبکه عصبی به منظور طبقه‌بندی حملات ایجاد می‌شود. در این پژوهش از شبکه پرسپترون چند لایه با یک لایه ورودی دو لایه نهان و یک لایه خروجی استفاده شده است. در شبکه عصبی پیشنهادی تعداد نرون‌های لایه ورودی به تعداد مولفه‌های ورودی‌ها می‌باشد و تعداد نرون‌های لایه پنهان مناسب برای هر یک از مجموعه ورودی‌ها انتخاب می‌شوند. همچنین برای بدست آوردن تعداد بهینه نرون‌های لایه‌های نهان تعداد آن‌ها به صورت افزایشی از پایین به بالا افزایش داده می‌شوند تا تعداد نرون‌های بهینه بدست آید. بر همین اساس تعداد نرون‌های مورد استفاده در شبکه عصبی پیشنهادی در لایه پنهان به ترتیب (لایه نهان اول ۴۵ نرون و لایه نهان دوم ۳۵ نرون) می‌باشد، این تعداد نرون در لایه پنهان از یک طرف باعث تسریع محاسبات شده و صحّت قابل قبول نتیجه را ارائه می‌دهند، از طرف دیگر بیشتر شدن تعداد نرون‌های لایه پنهان موجب بیش برازش شبکه و در نتیجه ایجاد جواب اشتباه می‌شود و در نهایت از دو نرون در لایه خروجی برای نمایش نتایج جهت دسته‌بندی (یک کلاس حمله و یک کلاس غیرتهاجمی) استفاده می‌شود. همچنین توابع مورد استفاده در لایه‌ها تابع سیگموئیدی هستند.

۳-۵- کاربرد الگوریتم ژنتیک در روش پیشنهادی

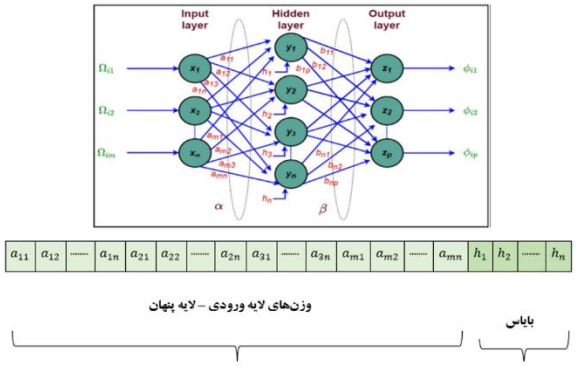
در واقع در اکثر پژوهش‌های انجام شده سیستم‌های تشخیص نفوذ قادر به شناسایی حملات با استفاده از راهکارهای با نظارت هستند. اما در پژوهش پیش رو، قصد بر این است که یک رویکرد هوشمند و بدون نظارت مبتنی بر تکنیک‌های شبکه عصبی مصنوعی و الگوریتم ژنتیک جهت بهبود تشخیص نفوذ در محیط رایانش ابری ارائه شود. سیستم پیشنهادی به دنبال بهبود فرایند دسته‌بندی ترافیک نرمال و غیر عادی می‌باشد. همانطور که گفته شد فرآیند آموزش شبکه عصبی مصنوعی به معنای بهینه کردن پارامترهای آن که شامل وزن‌ها و بایاس‌ها می‌باشند در جهت کمینه شدن MSE[۱۲] می‌باشد. با توجه به اینکه که شبکه‌های عصبی پرسپترون چند لایه از الگوریتم پس انتشار خطا به منظور تنظیم وزن‌ها استفاده می‌کنند که در هر دور اجرا مقدار خطای شبکه ارزیابی می‌گردد، در صورتی که مقدار خطا بالا باشد خطا به عقب انتشار پیدا می‌کند و مقدار وزن‌ها دوباره می‌بایست تنظیم گردد. حال در این مقاله فرآیند آموزش شبکه عصبی و بهینه سازی پارامترهای شبکه با استفاده از الگوریتم ژنتیک انجام می‌شود. این الگوریتم به دلیل قابلیت جستجوی سراسری و اکتشاف در بهینه‌های محلی گیر نکرده و می‌تواند مقدار مناسب وزن و بایاس را با دقت زیادی تعیین نماید.

در ادامه روند بهینه‌سازی پارمتراهای شبکه عصبی با استفاده از الگوریتم ژنتیک که در جهت بهبود روند آموزش شبکه عصبی شرح داده می‌شود.

۳-۶- فرمول‌بندی راهکار با الگوریتم ژنتیک

- **تعیین ساختار کروموزم**

قدم اول در حل مسئله تشخیص نفوذ به کمک الگوریتم ژنتیک، تعیین ساختار و رمزگذاری[۱۳] کروموزوم است. همانطور که می‌دانیم، هر کروموزم در الگوریتم ژنتیک، معادل یک وضعیت از حالات ممکن برای فضای حالت مسئله است. روش‌های زیادی برای رمزگذاری کروموزم‌ها وجود دارد که با توجه به ماهیت مسئله انتخاب می‌شوند. در ساختار کروموزم پیشنهادی، وزن‌هایی که بر روی یال‌های لایه ورودی به لایه پنهان و همچنین وزن‌هایی که بر روی یال‌های لایه پنهان به لایه خروجی همراه با بایاس‌ها که در لایه‌های پنهان وجود دارند، ژن‌های کروموزم پیشنهادی را تشکیل می‌دهند بر همین اساس از کدگذاری مقدار[۱۴] استفاده می‌کنیم. در شکل (۳) ساختار یک کرموزوم در روش پیشنهادی آورده شده است.



$$C_i = \sum^{n} f(a_i.h_i.\Omega_i)b_i \quad for\ i = 1.2.3.\ldots.n$$

شکل ۳: ساختار کروموزم پیشنهادی

برای بدست آوردن تعداد ژن‌های کرموزوم پیشنهادی می‌توان از رابطه (۲) استفاده می‌شود:

$$\text{Total Gens} = kn^2 + kn \qquad (۲)$$

که در رابطه بالا Total Gens نشان‌دهنده تعداد کل ژن‌های موجود در کروموزم، k تعداد لایه‌های پنهان و n تعداد ژن‌های هر کروموزم می‌باشد.

- **جمعیت S**

جمعیت را با pop و اندازه جمعیت را با $pop^{(s)}$ نمایش می‌دهیم.

اعضای تشکیل دهنده جمعیت در این مسئله، C_iها (i=1, 2 , …, ∀) $pop^{(s)}$ هستند. برای ایجاد جمعیت اولیه[۱۵] از اختصاص اعداد تصادفی اعشاری در بازه [1 .. 0] بدون در نظرگیری معیار خاصی به C_iها استفاده می‌کنیم.

- **تابع ارزیابی[۱۶]**

در ادامه تابعی ایجاد می‌کنیم که بتوانیم توسط آن به هر C_i، عددی متناسب با میزان خوب بودن آن، اختصاص دهیم. این تابع را Func می‌نامیم که مقدار بازگشتی آن برای کروموزم C_i، برابر f_i است.

$$f_i = 1 - Accuracy = 1 - \frac{True\ Detection}{Total\ Detection} \qquad (۳)$$

در رابطه فوق Total Detection برابر کل داده‌های آموزشی اولیه می‌باشد. True Detection تعداد نمونه‌هایی که به صورت صحیح تشخیص داده شده‌اند که آیا در آن‌ها نفوذ رخ داده است یا خیر. با توجه به تعریف تابع برازندگی مشخص است که مسئله یک مسئله بیشینه‌سازی می‌باشد هرچه مقدار f_i بیشتر باشد، احتمال مشارکت C_i برای تولید نسل بعدی افزایش می‌یابد. با استفاده از تابع ارزیابی، برای هر کروموزم C_i، ارزش f_i محاسبه می‌شود. سپس بر اساس این ارزش برای همه اشیاء یک جایابی بهینه ایجاد می‌شود.

- **عملگرهای الگوریتم ژنتیک برای مسئله جایابی**
  - **انتخاب[۱۷]**

برای اعمال عملگرهای ژنتیک برروی راه‌حل‌ها (C_iها)، باید ابتدا C_i-هایی به‌عنوان والد انتخاب شوند تا از روی این راه‌حل‌ها، راه‌حل‌های فرزند به وجود آیند. برای انتخاب راه‌حل‌های والد در الگوریتم ژنتیک انتخاب‌شده از روش انتخاب مسابقه‌ای[۱۸] استفاده می‌شود [۲۲]. در شکل (۴) شبه کد این الگوریتم نشان داده شده است.

  - **ترکیب[۱۹]**

ترکیب با تعویض ژن‌ها، بین دو کروموزم انجام می‌گیرد و هر کروموزم خصوصیاتی از خود را به فرزندان می‌دهد. تعداد انتخاب کروموزم‌ها از pop به عنوان والد، طبق پارامتر احتمال ترکیب Pc[۲۰] انجام می‌گیرد. کروموزم‌هایی که دارای f_i بیشتری هستند، شانس بیشتری برای آمیزش و ترکیب دارند. در این قسمت از ترکیب حسابی استفاده می‌کنیم. اگر از نسل فعلی، دو کروموزم C_i و C_j به عنوان والد انتخاب شوند، طبق آمیزش حسابی از آن‌ها دو فرزند ch_1 و ch_2 به صورت معادله‌های (۴) و (۵) بوجود می‌آید.

$$ch_1 = \alpha\ C_i + (1-\alpha)C_j \qquad (۴)$$

$$ch_2 = \alpha\ C_j + (1-\alpha)C_i \qquad (۵)$$

پارامتر α یک مقدار در بازه [1.. 0] می‌باشد که در هر ترکیب می‌تواند مقدار مختلفی داشته باشد. فرزندان بدست‌آمده از این بخش را در جمعیت pop2 قرار می‌دهیم. در شکل (۵) شبه کد این عملگر نشان داده شده است.

```
Input: Size of Tournament population, SIZE
and opulation, Pop
     Output: an individual chromosome
procedure SELECTION
bestScore ←0
bestInd ← null
for i = 1 to SIZE  do
randomly select an individual
chromosome from Pop, newInd
calculate its fitness score, newScore
if bestScore ≤  newScore then
bestScore ← newScore
bestInd ← newInd
end if
end for
return bestInd
end procedur
```

شکل ۴: شبه کد عملگر انتخاب روش پیشنهادی

– جهش[۲۱]

مهم‌ترین وظیفه جهش اجتناب از همگرایی به بهینه محلی است. در این قسمت تعدادی از فرزندان بدست‌آمده از مرحله ترکیب، بر طبق پارامتر احتمال جهش Pm[۲۲] انتخاب شده و جهش می‌یابند. با توجه به آنکه در کدگذاری کروموزم‌ها از روش مقدار اعشاری استفاده می‌شود، پس باید از روش جهش حقیقی[۲۳] استفاده کنیم. اگر C_i با ژن‌های w_{i1} و w_{i2} تحت عمل جهش حقیقی انجام گیرد، آنگاه یکی از ژن‌ها (یکی از ژن‌های w_{i1} و w_{i2} به‌صورت تصادفی انتخاب می‌شود) توسط یک عدد اعشاری تصادفی با توزیع یکنواخت پیوسته در محدوده [1...0] جایگزین می‌شود. C_iهای جهش‌یافته در $pop^{(S)}$ جایگزین می‌شوند. در شکل (۶) شبه کد این عملگر نشان داده شده است. در ادامه به ترتیب در شکل (۷) و شکل (۸) شبه کد چارچوب روش پیشنهادی و فلوچارت بهینه‌سازی پارامترهای شبکه عصبی با الگوریتم ژنتیک نشان داده شده است.

Algorithm 1: Crossover Operation

```
Input: Two parent chromosomes: ch1 and ch2
Output: One offspring chromosome: Cnew
 1: procedure CROSSOVER
 2:          for i = 1 to Pop do
 3:               randomly create a      between 0 and 1, r;
 4:               if r ≤ Arithmetic Rate then
 5:                    set gene i of Cnew as gene i of ch1
 6:               else
 7:                    set gene i of Cnew as gene i of ch2
 8:               end if
 9:          end for
10: end procedure
```

شکل ۵: شبه کد عملگر ترکیب روش پیشنهادی

Algorithm2: Mutation Operation

```
     Input: One chromosome: Ch1
     Output: One mutated chromosome: Ch2
1.   procedure MUTATION
2.   where 1≤ ch ≤ Pop
3.   randomly select a Ch1
4.   randomly select a Ch2
5.   Ch2← Ch1
     Return Ch2
6.   end procedure
```

شکل ۶: شبه کد عملگر جهش روش پیشنهادی

Algorithm 3: Multilayer Perceptron Optimized by Genetic Algorithm

```
Input: Training data set (NSL KDD)
Output: Best parameters
InitPopulation (Pop): Initializes the population;
EvalPopulation (Pop): Evaluates the fitness of each
                      individual in the population;
1: While not Termination do

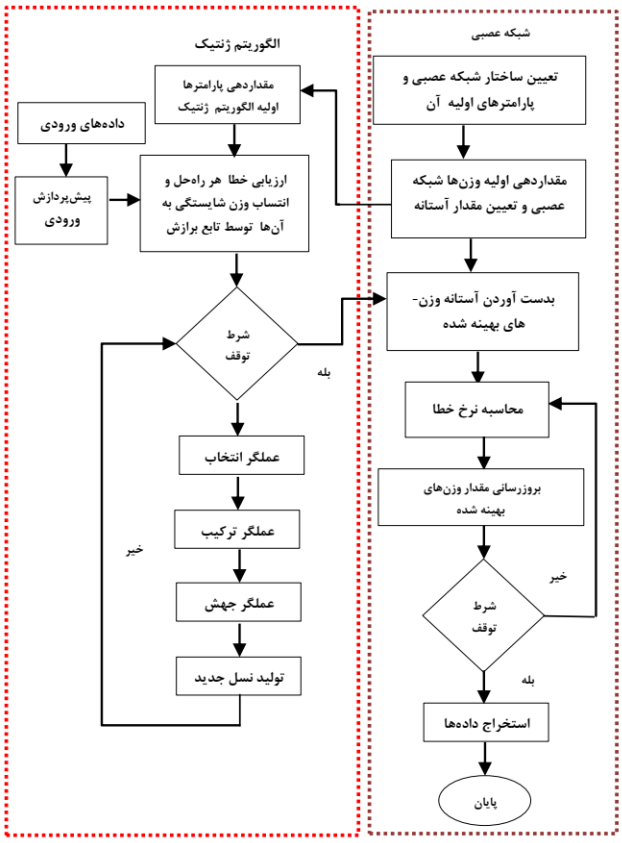
//Elitism based selection
2: Select best-ranking individuals from Pop and delete
others.

//Crossover
3: for i=1 to (Pop.Length)/2 do
4: Randomly select two solutions Xa and Xb from
population;
5: Generate Xc and Xd by crossover to Xa and Xb; 6: Save
Xc and Xd in Pop;
7: end for

//Mutation
8: for i=1 to Pop.Length do
9: Select Solution Xi from Pop;
10: Mutate a set of bits randomly and generate Xi';
11: Update X1 by Xi' in Pop;
12: end for

//Evaluation
13: for i=1 to Pop. Length do
14: Train MLP with using solution X1 from Pop;
15: Evaluate MLP model and calculate accuracy;
16: endfor
17: end while
```

شکل ۷: شبه کد چارچوب روش پیشنهادی



شکل ۸: فلوچارت بهینه‌سازی پارامترهای شبکه عصبی توسط الگوریتم ژنتیک

۴- ارزیابی روش پیشنهادی

۴-۱- بستر شبیه‌سازی

برای تایید عملکرد روش پیشنهادی آن را با روش‌های مراجع [۱۷]، [۱۸] و [۱۹] که از مجموعه داده‌های NSL-KDD استفاده کرده‌اند مقایسه کردیم. آزمایش با استفاده از برنامه متلب در محیط یک سیستم عامل ویندوز ۱۰- ۶۴ بیتی با پردازنده اینتل Core i5 2410 و با ۴ گیگابایت حافظه انجام شد. در ادامه به ترتیب مجموعه داده، معیارهای ارزیابی و نتایج ارزیابی بررسی خواهد شد.

۴-۲- مجموعه داده NSL-KDD

برای استخراج نتایج تجربی، از مجموعه داده واقعی NSL- KDD [۲۳] استفاده شد. این مجموعه، شامل ۱۲۵،۹۷۳ اتصال با ۴۱ ویژگی برای هر اتصال است. ویژگی‌های هر اتصال، به دو دسته امن و غیرامن برچسب‌گذاری شده‌اند. همچنین، ویژگی‌های این پایگاه داده به چهار گروه مختلف طبقه‌بندی شده‌اند:

- **ویژگی‌های پایه**[۲۴]: این ویژگی‌ها، مستقل از محتوای بار، می‌توانند در تشخیص ترافیک مشکوک مؤثر باشند.
- **ویژگی‌های محتوی**[۲۵]: این دسته از ویژگی‌ها، شامل شمارش تلاش‌های ناموفق ورود به سیستم هستند.
- **ویژگی‌های ترافیک براساس زمان**[۲۶]: این ویژگی‌ها، با هدف ثبت تغییرات ترافیکی در پنجره‌های زمانی دو دقیقه‌ای، طراحی شده‌اند .مثال‌هایی از این ویژگی‌ها، میزان اتصال به یک میزبان در بازه‌های دو ثانیه‌ای هستند.
- **ویژگی ترافیک براساس میزبان**[۲۷]: این ویژگی‌ها، با بهره‌گیری از پنجره‌های تاریخی تخمین‌زننده، برای ارزیابی حملات با طول بیش از دو ثانیه، به کار گرفته شده‌اند. در ادامه و در جدول (۱) ویژگی‌های مجموعه داده NSL-KDD آمده است.

جدول ۱ : ویژگی‌های مجموعه داده NSL-KDD

ردیف	نام ویژگی	ردیف	نام ویژگی
۱	Duration	۲۲	Is_urgent_login
۲	Protocol_type	۲۳	Count
۳	Service	۲۴	srv_count
۴	Flag	۲۵	serror_rate
۵	Src_bytes	۲۶	srv-serror-rate
۶	dst_byte	۲۷	serror_rate
۷	Land	۲۸	srv_error_rate
۸	Wrong_fragment	۲۹	same_srv_rate
۹	Urgent	۳۰	diff_srv_rate
۱۰	Hot	۳۱	srv_diff_host_rate
۱۱	num_faild_login	۳۲	dst_host_count
۱۲	Logged_in	۳۳	dst_host_srv_count
۱۳	Num-compromised	۳۴	dst_host_same_srv_count
۱۴	Root_shell	۳۵	dst_host_diff_srv_count
۱۵	Su_attempted	۳۶	dst_host_same_src_port_
۱۶	Num_root	۳۷	dst_host_srv_diff_host _rate
۱۷	Num_file_creations	۳۸	dst_host_serroe_rate
۱۸	Num_shells	۳۹	dst_host_srv_serroe_rate
۱۹	Num_access_files	۴۰	dst_host_rerroe_rate
۲۰	Num_outbound_cmds	۴۱	dst_host_srv_rerroe_rate
۲۱	Is_hot_login		

۴-۳- معیارهای ارزیابی

برای ارزیابی رویکرد پیشنهادی معیارهای متعددی مورد استفاده قرار می‌گیرد؛ معیارهای Accuracy، Precision، Recall ،

F-Measure و ROC مهم‌ترین معیارهایی هستند که برای ارزیابی مدل‌های دسته‌بندی مورد استفاده قرار می‌گیرند. درادامه به تشریح هر کدام از این معیارهای پرداخته شده است.

- **Accuracy:** دقت نسبت کلاس‌های مثبت پیش‌بینی شده درست به همه موارد مثبت پیش‌بینی شده است. به طور عینی، این معیار نشان می‌دهد که پیش‌بینی‌های رویکرد پیشنهادی چقدر درست یا دقیق هستند. دقت زمانی مهم خواهد بود که در یک روش دسته بندی مثبت کاذب مهم‌تر از منفی‌های کاذب باشد. رابطه (۶) نحوه محاسبه این معیار ارزیابی را نشان می‌دهد.

$$\text{Accuracy} = \frac{T_P}{T_P + F_P + T_N + F_N} \qquad (۶)$$

- **Precision**: این معیار نشان می‌دهد که چند درصد از پیشنهادهای الگوریتم درست بوده است. حداکثر مقدار این معیار یک و یا ۱۰۰ درصد و حداقل مقدار آن صفر است و هرچه مواردی که الگوریتم به غلط پیش‌بینی کرده است که به آن منفی کاذب می‌گوییم نسبت به پیش‌بینی‌های درست یا مثبت کاذب بیشتر باشد مقدار Precision کمتر خواهد شد. رابطه (۷) نحوه محاسبه این معیار ارزیابی را نشان می‌دهد که در آن TP پیشنهادهای درست و FP پیشنهادهای غلط را نشان می‌دهند:

$$\text{Precision} = \frac{T_P}{T_P + F_P} \qquad (۷)$$

- **Recall:** Recall برخلاف معیار Precision به دنبال پوشش بر روی کل داده‌هاست. این معیار به دنبال میزان شناسایی درست کلاس‌های حمله، نسبت به کل حملات موجود در مجموعه داده را نشان می‌دهد که حداکثر مقدار این معیار نیز یک و یا ۱۰۰ درصد و حداقل مقدار آن صفر است. هرچه مواردی که حمله بودند ولی الگوریتم پیش‌بینی نکرده است یعنی منفی کاذب، نسبت به پیش‌بینی‌های درست یعنی مثبت کاذب بیشتر باشد مقدار Recall کمتر خواهد شد. رابطه (۸) نحوه محاسبه این معیار ارزیابی را نشان می‌دهد که در آن TP پیشنهادهای درست و FN مواردی که حمله بودند ولی الگوریتم پیش بینی نکرده را نشان می‌دهند.

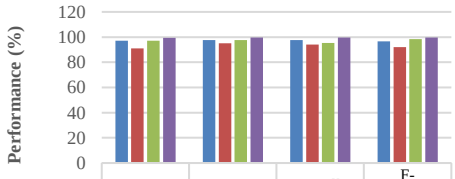
$$\text{Recall} = \frac{T_P}{T_P + F_N} \qquad (۸)$$

- **F-Measure**: این معیار نیز ترکیب متعادلی از دو معیار Precision و Recall است. رابطه (۹) نحوه محاسبه این معیار را نشان می‌دهد.

$$\text{F} - \text{Measure} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \qquad (۹)$$

۴-۴- نتایج ارزیابی

مطالعه حاضر به ارزیابی و تحلیل کارایی راهکار پیشنهادی با اتکا به مقایسه نتایج حاصل با سه رویکرد برجسته دیگر می‌پردازد. برای ارزیابی روش‌ها، از مجموعه داده استاندارد و در دسترس NSL-KDD استفاده شده است. این مجموعه داده به عنوان یک معیار ارزیابی معتبر، امکان سنجش توانایی و دقت روش پیشنهادی را فراهم می‌آورد. شکل ۹ به مقایسه نتایج حاصل از به کارگیری روش پیشنهادی و روش‌های تشخیص نفوذ مورد اشاره در مراجع [۱۷]، [۱۸] و [۱۹] می‌پردازد. دقت، به عنوان مهم‌ترین شاخص ارزیابی عملکرد، بر اساس رابطه (۶)، نسبت تشخیص‌های صحیح ترافیک عادی و نفوذ به کل موارد بررسی شده را نشان می‌دهد. در بین روش‌های بررسی شده، روش پیشنهادی (Proposed Method) با دقت ۹۹,۴۱٪، عملکرد برتری را به نمایش گذاشته است. این برتری ناشی از اعمال تغییرات در مراحل الگوریتم شبکه عصبی بهبودیافته و ایجاد یک فرایند آموزش دقیق‌تر، و همچنین اصلاح معیارهای آماری مورد استفاده در روش مبتنی بر الگوریتم ژنتیک بوده است.



	Accurcay	Precision	Recall	F-Measured
SVM+HHO+PSO [17]	97.05	97.62	97.62	96.62
GKDPCAs + ACDBNs [18]	91	95	94	92
PSO + LSTM [19]	97.05	97.62	95.32	98.32
Proposed Method	99.41	99.67	99.66	99.66

شکل ۹: مقایسه نتایج حاصل از روش پیشنهادی با سایر روش‌های تشخیص نفوذ

علاوه بر این، با توجه به کارایی بالاتر الگوریتم ژنتیک در مقایسه با شبکه عصبی در فرایند آموزش داده، برتری روش پیشنهادی مورد تأیید قرار می‌گیرد. همچنین، روش پیشنهادی بر اساس معیارهای Precision و Recall، که به ترتیب نرخ تشخیص صحیح حملات از بین کل حملات و نرخ تشخیص صحیح ارتباطات عادی از بین کل این نوع ارتباطات را نشان می‌دهند (روابط ۸ و ۹)، مورد ارزیابی قرار

گرفته است. همانطور که در شکل ۸ نشان داده شده، روش پیشنهادی بر اساس این دو معیار نیز عملکرد بهتری نسبت به سایر روش‌ها دارد. درمجموع، روش پیشنهادی با دستیابی به نرخ دقت تشخیص بالا و ویژگی‌های موفق، توانسته است بالاترین دقت تشخیص را در میان روش‌های مورد مطالعه کسب کند.

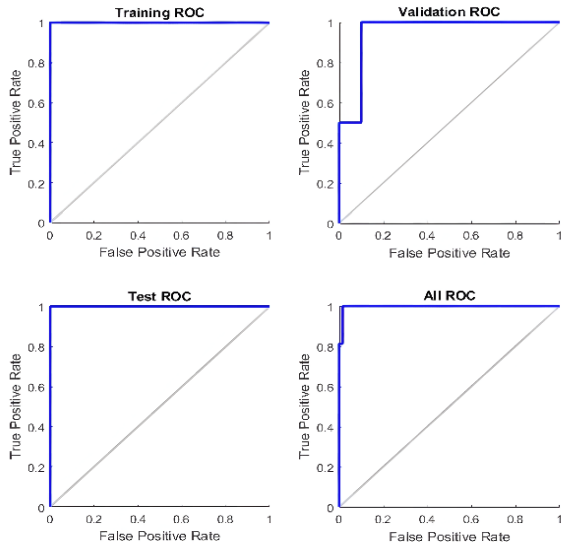
در ارزیابی دیگر نمودار ROC روش پیشنهادی مورد بحث و بررسی قرار گرفته است. نمودار ROC ابزار مفیدی برای ارزیابی عملکرد مدل‌های یادگیری ماشین به شمار می‌رود. در این نمودار محور افقی مربوط به نرخ مثبت اشتباه یا FPR بوده و بیانگر نسبت تعداد نمونه‌های مثبتی که به اشتباه تشخیص داده شده‌اند، به کل نمونه‌های منفی است. محور عمودی بیانگر نرخ مثبت درست یا TPR بوده و بیانگر نسبت تعداد نمونه‌های مثبتی که به درستی تشخیص داده شده‌اند، به کل نمونه‌های مثبت است. با توجه به این دو نرخ، می‌توان در نمودار ROC یک منحنی رسم کرد که نشان می‌دهد با تغییر آستانه چگونه نرخ تشخیص نادرست و درست تغییر می‌کنند. هر نقطه در نمودار ROC به یک آستانه مشخص در تصمیم‌گیری مدل برمی‌گردد. این نقاط وابسته به مقادیر TPR و FPR محاسبه شده برای هر آستانه خاص هستند. خط یکنواخت از (۰،۰) به (۱،۱) در نمودار، نشان‌دهنده عملکرد تصادفی مدل است. همچنین مساحت زیر نمودار ROC معیار (AUC) نیز می‌تواند به عنوان یک معیار جامع از عملکرد مدل استفاده شود که مقدار بین ۰ و ۱ دارد. مقدار AUC بزرگتر از ۰/۵ نشان می‌دهد که مدل به طور متوسط بهتر از یک دسته‌بندی تصادفی تصمیم می‌گیرد.

در شکل (۱۰) خلاصه‌ای از عملکرد روش پیشنهادی در قالب نمودار ROC مشاهده می‌شود که همواره نرخ مثبت واقعی بیشتر نرخ مثبت کاذب است. بر همین اساس نمودار ROC به طرف بالا و راست متمایل بوده که این نشان دهنده سطح زیرنمودار بیشتر، مقادیر TPR بالاتر و در نتیجه عملکرد خوب روش پیشنهادی است.

۵- نتیجه‌گیری

هدف اصلی در انجام این مقاله، ارائه سیستمی جهت تشخیص نفوذ در محیط رایانش ابری با استفاده از شبکه عصبی بهبود یافته مبتنی بر الگوریتم ژنتیک است. هدف اساسی برای طراحی رودیکرد پیشنهادی تعریف شد که شامل بکارگیری یک الگوریتم بهینه‌سازی تکاملی مناسب جهت بهینه‌کردن پارامترهای شبکه عصبی به منظور تشخیص بهتر ناهنجاری‌ها است. برای نیل به این هدف در ابتدای کار مجموعه داده ورودیNSL-KDD نرمال‌سازی شده و پس از آن ابعاد ویژگی‌های مجموعه داده با الگوریتم PCA کاهش می‌یابند و سپس نمونه‌ها به دو دسته آموزش و آزمایش تقسیم می‌شوند. در ادامه آموزش شبکه عصبی با مجموعه داده آموزش و با استفاده از الگوریتم ژنتیک آغاز می‌شود، بر همین اساس پارامترهای شبکه عصبی مورد بهینگی و بازبینی قرار می‌گیرند و در نتیجه عملکرد آن بهبود می‌یابد.

کاهش پیچیدگی بهینه‌سازی پارامترهای شبکه عصبی و همچنین دلیل قابلیت جستجوی سراسری و عدم افتادن در بهینه‌سازی محلی از مهمترین علل استفاده از الگوریتم ژنتیک می‌باشد. به منظور ارزیابی و تحلیل نتایج شبکه عصبی بهبود یافته، نتایج حاصل از روش پیشنهادی با سه روش مطرح دیگر مراجع [۱۷]، [۱۸] و [۱۹] مورد مقایسه قرار گرفت. بررسی نتایج نشان می‌دهد که در روش پیشنهادی، میزان نرخ تشخیص اشتباه به اندازه چشمگیری کاهش و نرخ دقت روش نسبت به سه رویکرد دیگر تشخیص نفوذ افزایش پیدا کرده است. دلیل این بهبود عملکرد را می‌توان تنظیم دقیق وزن و بایاس شبکه عصبی توسط الگوریتم ژنتیک نیز دانست. لازم به ذکر است که استفاده از تکنیک‌های یادگیری عمیق و یادگیری تقویت شده برای بهینه‌سازی پارامترهای شبکه عصبی و همچنین استفاده از سایر روش‌های کاهش ابعاد ویژگی می‌تواند به عنوان کارهای آتی این تحقیق مدنظر قرار بگیرند.



شکل ۱۰: منحنی ROC مربوط به یادگیری سیستم پیشنهادی

مراجع

[1] H. Abusaimeh, "Distributed denial of service attacks in cloud computing," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 6, pp. 163–168, 2020, doi: 10.14569/IJACSA.2020.0110621.

[2] I. T. Aziz, I. H. Abdulqadder, and T. A. Jawad, "Distributed denial of service attacks on cloud computing environment," *Cihan Univ.-Erbil Sci. J.*, vol. 6, no. 1, pp. 47–52, 2022, doi: 10.24086/cuesj.v6n1y2022.pp47-52.

[3] G. S. Kushwah and V. Ranga, "Voting extreme learning machine based distributed denial of service attack detection in cloud computing," *J. Inf. Secur. Appl.*, vol. 53, Art. no. 102532, 2020, doi: 10.1016/j.jisa.2020.102532.

[4] A. Singh and B. B. Gupta, "Distributed denial-of-service (DDoS) attacks and defense mechanisms in various web-enabled computing platforms: Issues, challenges, and future research directions," *Int. J. Semantic Web Inf. Syst.*, vol. 18, no. 1, pp. 1–43, 2022, doi: 10.4018/IJSWIS.297143.

[5] A. D. Lopez, A. P. Mohan, and S. Nair, "Network traffic behavioral analytics for detection of DDoS attacks," *SMU Data Sci. Rev.*, vol. 2, no. 1, Art. no. 14, 2019.

[6] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 31, no. 4, pp. 541–553, 2019, doi: 10.1016/j.jksuci.2018.03.011.

[7] P. D. Bojović, I. Bašičević, S. Ocovaj, and M. Popović, "A practical approach to detection of distributed denial-of-service attacks using a hybrid detection method," *Comput. Electr. Eng.*, vol. 73, pp. 84–96, 2019, doi: 10.1016/j.compeleceng.2018.11.017.

[8] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, "CNN-based network intrusion detection against denial-of-service attacks," *Electronics*, vol. 9, no. 6, Art. no. 916, 2020, doi: 10.3390/electronics9060916.

[9] L. Lv, W. Wang, Z. Zhang, and X. Liu, "A novel intrusion detection system based on an optimal hybrid kernel extreme learning machine," *Knowl.-Based Syst.*, vol. 195, Art. no. 105648, 2020, doi: 10.1016/j.knosys.2020.105648.

[10] S. Sambangi and L. Gondi, "A machine learning approach for DDoS (distributed denial of service) attack detection using multiple linear regression," *Proceedings*, vol. 63, no. 1, Art. no. 51, 2020, doi: 10.3390/proceedings2020063051.

[11] P. Ghosh, D. Sarkar, J. Sharma, and S. Phadikar, "An intrusion detection system using modified-firefly algorithm in cloud environment," *Int. J. Digit. Crime Forensics*, vol. 13, no. 2, pp. 77–93, 2021, doi: 10.4018/IJDCF.2021030105.

[12] G. S. Kushwah and V. Ranga, "DDoS attacks detection in cloud computing using ANN and imperialistic competitive algorithm," in *Proc. Int. Conf. Artif. Intell. Sustain. Comput. (ICSISCET)*, Singapore: Springer, 2022, pp. 253–263, doi: 10.1007/978-981-19-1653-3_20.

[13] S. Mani, B. Sundan, A. Thangasamy, and L. Govindaraj, "A new intrusion detection and prevention system using a hybrid deep neural network in cloud environment," in *Proc. Int. Conf. Comput. Netw.*, Big Data IoT (ICCBI), Singapore: Springer, 2022, pp. 981–994, doi: 10.1007/978-981-19-0898-9_77.

[14] S. Krishnaveni, S. Sivamohan, S. S. Sridhar, and S. Prabakaran, "Efficient feature selection and classification through ensemble method for network intrusion detection on cloud computing," *Cluster Comput.*, vol. 24, no. 3, pp. 1761–1779, 2021, doi: 10.1007/s10586-020-03222-7.

[15] G. S. Kushwah and V. Ranga, "Optimized extreme learning machine for detecting DDoS attacks in cloud computing," *Comput. Secur.*, vol. 105, Art. no. 102260, 2021, doi: 10.1016/j.cose.2021.102260.

[16] E. Moharamkhani, M. Y. F. Hendi, E. Bandar, A. Izadkhasti, and R. S. Raza, "Intrusion detection system based firefly algorithm-random forest for cloud computing," *Concurrency Comput. Pract. Exp.*, vol. 34, no. 24, Art. no. e7220, 2022, doi: 10.1002/cpe.7220.

[17] S. Sokkalingam and R. Ramakrishnan, "An intelligent intrusion detection system for distributed denial of service attacks: A support vector machine with hybrid optimization algorithm based approach," *Concurrency Comput. Pract. Exp.*, vol. 34, no. 27, Art. no. e7334, 2022, doi: 10.1002/cpe.7334.

[18] S. Sureshkumar, G. P. Venkatesan, and R. Santhosh, "Detection of DDoS attacks on cloud computing environment using altered convolutional deep belief networks," *Int. J. Comput. Netw. Inf. Secur.*, vol. 15, no. 5, pp. 63–72, 2023, doi: 10.5815/ijcnis.2023.05.06.

[19] A. Thangasamy, B. Sundan, and L. Govindaraj, "A novel framework for DDoS attacks detection using hybrid LSTM techniques," *Comput. Syst. Sci. Eng.*, vol. 45, no. 3, pp. 2553–2567, 2023, doi: 10.32604/csse.2023.032078.

[20] T. Kurita, "Principal component analysis (PCA)," in *Computer Vision: A Reference Guide*, *Cham*, Switzerland: Springer, 2021, pp. 1013–1016, doi: 10.1007/978-3-030-63416-2_346.

[21] M. Greenacre, P. J. Groenen, T. Hastie, A. I. d'Enza, A. Markos, and E. Tuzhilina, "Principal component analysis," *Nat. Rev. Methods Primers*, vol. 2, no. 1, Art. no. 100, 2022, doi: 10.1038/s43586-022-00184-w.

[22] B. Alhijawi and A. Awajan, "Genetic algorithms: Theory, genetic operators, solutions, and applications," *Evol. Intell.*, vol. 17, no. 3, pp. 1245–1256, 2024, doi: 10.1007/s12065-023-00822-5.

[23] M. S. Pervez and D. M. Farid, "Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs," in *Proc. 8th Int. Conf. Softw., Knowl., Inf. Manag. Appl. (SKIMA)*, Dhaka, Bangladesh, 2014, pp. 1–6, doi: 10.1109/SKIMA.2014.7083560.

پاورقی‌ها:

[1] Cloud Computing
[2] Intrusion Detection System (IDS)
[3] Artificial Bee Colony Algorithm (ABC)
[4] AdaBoost
[5] Distributed Denial of Service
[6] Artificial Neural Networks
[7] Receiver Operating Characteristic
[8] Area under the curve
[9] Long Short-Term Memory (LSTM)
[10] Principal Component Analysis
[11] Multilayer Perceptron
[12] Mean squared error
[13] Encoding
[14] Value Encoding
[15] Initial population
[16] Fitness function
[17] Selection
[18] Tournament Selection
[19] Crossover
[20] Probability Crossover
[21] Mutation
[22] Probability Mutation
[23] Real Mutation
[24] Basic Features
[25] Content features
[26] Time-Based Traffic Features
[27] Host-Based Traffic Features