نشریه علمی‌ترویجی محاسبات نرم
سال ششم/ شمارهٔ اول/ بهار و تابستان ۱۳۹۶/ صفحهٔ ۴۸ـ۶۵

دریافت مقاله: ۱۳۹۴/۸/۸
پذیرش مقاله: ۱۳۹۶/۳/۲۹

راهکار ترکیبی نوین برای تشخیص نفوذ در شبکه‌های کامپیوتری با استفاده از الگوریتم‌های هوش محاسباتی

منصور شیخان[۱*]، عطیه عباسی[۲]

[۱] دانشیار، گروه مهندسی مخابرات، دانشگاه آزاد اسلامی، واحد تهران جنوب، تهران، ایران
msheikhn@azad.ac.ir
[۲] کارشناس‌ارشد، گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد تهران جنوب، تهران، ایران
st_a_abbasi@azad.ac.ir

چکیده

در این مقاله، راهکاری ترکیبی و نوین برای تشخیص نفوذ در شبکه‌های کامپیوتری معرفی شده است که از مزایای هر دو روش شناسایی سوءاستفاده و شناسایی ناهنجاری بهره می‌برد. در راهکار پیشنهادی، سامانه‌های شناسایی ناهنجاری و شناسایی سوءاستفاده به‌منظور بهبود عملکرد شناسایی نفوذ با هم ترکیب شده‌اند. رویکرد پیشنهادی از مجموعه‌ای از الگوریتم‌ها و مدل‌ها برای تحقق سامانهٔ تشخیص نفوذ بهره می‌جوید. برای انتخاب ویژگی‌های ورودی بهینه به سامانه، از الگوریتم جهش قورباغه استفاده شده است. بخش شناسایی سوءاستفاده در این سامانه، درخت تصمیم‌گیری را بدین‌منظور به کار می‌گیرد. برای شناسایی ناهنجاری در این سامانه، از مدل‌های شبکهٔ عصبی پایه- شعاعی یا ماشین بردار پشتیبان استفاده شده است. الگوریتم‌های بهینه‌سازی ازدحام ذرات یا وراثتی نیز در فرایند آموزش شبکهٔ عصبی به کار گرفته شده‌اند. نتایج تجربی به‌دست‌آمده با استفاده از مجموعه‌دادهٔ NSL-KDD گزارش شده است. این نتایج نشان می‌دهند که رویکرد پیشنهادی می‌تواند کارایی شناسایی نفوذ در شبکه را در مقایسه با استفادهٔ صرف از شناسایی ناهنجاری یا سوءاستفاده بهبود ببخشد. همچنین مدلی با انتخاب ویژگی به‌کمک الگوریتم جهش قورباغه و دسته‌بندی با ترکیب روش‌های درخت تصمیم‌گیری و ماشین بردار پشتیبان، با ۱۰ ویژگی انتخابی ورودی به نرخ آشکارسازی ۹۷/۴ درصد می‌رسد. این در حالی است که سامانه‌های آموزش‌دیده با دادگان مشابه در سایر پژوهش‌ها با تعداد ۳۳ و ۱۴ ویژگی انتخابی ورودی به‌ترتیب به نرخ آشکارسازی ۸۲/۳ درصد و ۸۳/۱ درصد رسیده‌اند. همچنین با حفظ نرخ آشکارسازی نفوذ در تراز سایر روش‌های رقیب شبیه‌سازی‌شده در این مقاله، سرعت اجرای الگوریتم تا ۲۸ برابر نسبت به روش‌های مذکور بهبود پیدا می‌کند.

واژه‌های کلیدی: شناسایی نفوذ به شبکه، سامانهٔ آمیختار، درخت تصمیم‌گیری، شبکهٔ عصبی پایه- شعاعی، الگوریتم جهش قورباغه.

۱. مقدمه

نفوذ در سامانه‌های کاربران و انجام اعمال نامطلوب، از جمله مواردی است که کامپیوترهای متصل به اینترنت را تهدید می‌کند. نفوذ به شبکه، معمولاً یک حمله قلمداد می‌شود [۱ و ۲]. امروزه سامانهٔ تشخیص نفوذ به‌عنوان یک بخش استاندارد از زیرساخت‌های امنیتی درآمده است. اهداف سامانهٔ تشخیص نفوذ را می‌توان چنین برشمرد: الف. پیشگیری از مشکلات رفتارهایی که سامانه را مورد حمله یا سوءاستفاده قرار می‌دهند؛ ب. تشخیص حملات و برخورد با آن‌ها؛ پ. مستندسازی حملات موجود؛ ت. کنترل کیفی و فراهم کردن اطلاعات مفید دربارهٔ نفوذ برای مدیران امنیت [۲].

در گزارش «بررسی امنیت و جرایم رایانه‌ای CSI/FBI» آمده است که استفاده از سامانهٔ تشخیص نفوذ در سال ۱۹۹۹، ۴۲ درصد بوده است که این نسبت در سال ۲۰۱۰ به ۶۲ درصد رسیده و تا سال ۲۰۱۴ نیز سهم ۶۲ درصدی خود را حفظ کرده است. این آمار نشان می‌دهد که این سامانه‌ها در فناوری‌های امنیتی بسیار مهم‌اند [۳ـ۵].

سامانه‌های شناسایی نفوذ به شبکه، حملات را از طریق مشاهدهٔ فعالیت‌های شبکه شناسایی می‌کنند. روش‌های اصلی تشخیص نفوذ عبارت است از: شناسایی سوءاستفاده[1] و شناسایی ناهنجاری[2]. شناسایی سوءاستفاده، نفوذها را از طریق الگوها و امضاها[3] که نشان‌دهندهٔ حملات هستند شناسایی می‌کند، اما قادر به شناسایی حملات ناشناخته نیست. مزیت روش شناسایی سوءاستفاده، سرعت بسیار بالای شناسایی ناشی از به‌کارگیری الگوریتم‌های شناسایی با پیچیدگی کمتر است [۶ و ۷]. در روش شناسایی ناهنجاری، از طریق ایجاد پروفایل‌های استفادهٔ نرمال، فعالیت‌هایی که نسبت به کارکرد نرمال انحراف دارند، به‌عنوان نفوذ تشخیص داده می‌شوند. در نتیجه، سامانهٔ شناسایی ناهنجاری قادر به شناسایی نفوذهای ناشناخته‌ای است که رویکرد شناسایی سوءاستفاده، توانایی آن را ندارد. از ایرادهای روش شناسایی ناهنجاری، نرخ بالای هشدارهای اشتباه در آن است [۸].

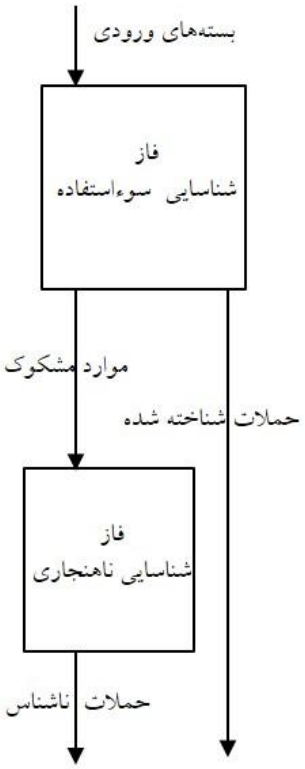
به‌منظور رفع مشکلات این دو روش، سامانه‌های شناسایی نفوذ با ترکیب هر دو روش نیز معرفی شده‌اند که به سه طریق به ترکیب راهکارهای اصلی می‌پردازند: الف. شناسایی ناهنجاری و سپس شناسایی سوءاستفاده؛ ب. رویکرد موازی؛ پ. شناسایی سوءاستفاده و سپس شناسایی ناهنجاری [۸].

ازآنجاکه در رویکرد موازی ترافیک ورودی توسط هر دو روش (شناسایی سوءاستفاده و شناسایی ناهنجاری) به‌صورت جداگانه مورد بررسی قرار می‌گیرد، نرخ تشخیص انواع حملات (شناخته‌شده و ناشناخته) افزایش می‌یابد. هرچند که نرخ بالای اعلان‌های مثبت اشتباه ناشی از روش تشخیص ناهنجاری باقی می‌ماند؛ زیرا اگر مدل تشخیص، ارتباط را به‌عنوان یک حمله دسته‌بندی کند، این روش هم ارتباط ورودی را به‌شکل حمله در نظر می‌گیرد. همچنین مسئلهٔ هزینهٔ محاسباتی تشخیص هم وجود دارد؛ زیرا هر ارتباطی باید با استفاده از هر دوی مدل‌های تشخیص ناهنجاری و تشخیص سوءاستفاده بررسی شود و این می‌تواند باعث افزایش هزینهٔ محاسباتی تشخیص شود [۹].

در رویکرد ترکیبی مورد استفاده در این مقاله، سعی شده که با استفاده از الگوریتم‌ها و مدل‌های موجود و ترکیب کردن آن‌ها با هم، به طرح بهینه‌ای دست یابیم. در این راستا، به‌کارگیری الگوریتم بهینه‌سازی مبتنی بر جهش قورباغه، امکان کاهش تعداد ویژگی‌های ورودی به سامانه را از ۴۱ به ۱۰ فراهم کرده است. سامانهٔ ترکیبی نیز خود از مزیت‌های این طرح در مقایسه با بسیاری از پژوهش‌های ارائه‌شده در این حوزه است. بدین‌ترتیب که ابتدا ترافیک ورودی به سامانهٔ شناسایی سوءاستفاده وارد می‌شود (شکل ۱). موارد خروجی فاز شناسایی سوءاستفاده که تطابقی با الگوهای نفوذ ندارند، به‌عنوان ورودی به سامانهٔ شناسایی ناهنجاری تزریق می‌شوند تا نفوذهای ناشناس را تشخیص دهند. کارایی شناسایی ناهنجاری با افزایش تعداد حملات کاهش می‌یابد، که به‌منظور غلبه بر این مشکل، ابتدا شناسایی سوءاستفاده به‌کار گرفته شده است. سامانهٔ شناسایی سوءاستفاده می‌تواند حملات شناخته‌شده را تشخیص

1. Misuse detection
2. Anomaly detection
3. Signatures

دهد. با استفاده از حذف حملات شناخته‌شده، تعداد حملات به‌طور قابل ملاحظه‌ای برای شناسایی ناهنجاری کاهش می‌یابد. مزیت دیگر، آن است که سامانهٔ ترکیبی پیشنهادی می‌تواند نفوذهای شناخته‌شده را بی‌درنگ شناسایی کند؛ زیرا شناسایی سوءاستفاده با استفاده از روش‌هایی چون الگوریتم درخت تصمیم‌گیری سرعت بالایی دارد [۸ و ۹].



شکل (۱): سامانهٔ شناسایی نفوذ ترکیبی با تقدم شناسایی سوءاستفاده

بخش شناسایی ناهنجاری این طرح نیز از مدل‌های شناخته‌شدهٔ موفق در این مورد (مانند شبکهٔ عصبی مصنوعی) بهره می‌جوید که توانایی شناسایی حملات ناشناس (الگوهای جدید) را دارد [۶]. برای آموزش شبکهٔ عصبی نیز به‌جای به‌کارگیری روش‌های متداول، از الگوریتم‌های هوش محاسباتی که برتری خود را در دستیابی به عملکرد بهتر از خود نشان داده‌اند [۱۰]، بهره گرفته شده است. دراین‌باره، به‌کارگیری شبکهٔ عصبی پایه- شعاعی[1] (RBF) در فاز شناسایی ناهنجاری در روش پیشنهادی نیز سبب بهبود چشمگیر سرعت اجرای الگوریتم در مقایسه با روش‌های ترکیبی مشابه شده است.

1. Radial-Basis Function

مقایسهٔ عملکرد سامانهٔ ترکیبی پیشنهادی با برخی از طرح‌های مشابه که در همین زمینه و روی دادگان مشابه آزمون شده‌اند، نشان از دستیابی طرح پیشنهادی به نرخ آشکارسازی ترجیحی است (جدول ۷ در بخش ۵).

۲. پیشینهٔ تحقیق

در این بخش، ابتدا برخی از کارهای گذشته درخصوص سامانه‌های ترکیبی مرور می‌شوند؛ سپس مبانی الگوریتم‌ها و مدل‌های به‌کاررفته در این مقاله معرفی خواهند شد.

۲.۱. کارهای گذشته

در این زیربخش، برخی از پژوهش‌های انجام‌شده دربارهٔ سامانه‌های ترکیبی مرور شده‌اند. نمونه‌ای از سامانه‌های ترکیبی، معماری سه‌لایهٔ سامانهٔ تشخیص نفوذ است که توسط هوانگ[2] و همکاران [۱۱] طراحی و توسعه داده شده بود. این سامانه شامل یک فهرست سیاه، یک فهرست سفید، و یک دسته‌بند ماشین بردار پشتیبان چندکلاسه مبتنی بر داده‌های استاندارد KDD Cup'99 بود. حملات شناخته‌شده توسط فهرست سیاه از ترافیک شبکه فیلتر می‌شدند و توسط فهرست سفید، ترافیک نرمال کشف می‌شد.

باربارا[3] و همکاران [۱۲] روش کاوش و تحلیل دادهٔ حسابرسی[4] (ADAM) را ارائه کردند که در آن تشخیص سوءاستفاده پس از تشخیص ناهنجاری اعمال شده است. ADAM ترکیبی از قاعده‌کاوی انجمنی و یک روش دسته‌بندی برای تشخیص حملات بود. ابتدا مدل تشخیص ناهنجاری که قاعده‌کاوی انجمنی را استفاده می‌کند، ترافیک مشکوک را تعیین کرده و آن را به مدل تشخیص سوءاستفاده انتقال می‌دهد. سپس مدل تشخیص سوءاستفاده ارتباطات مشکوک را به‌عنوان «نرمال»، «حملات شناخته‌شده» و «حملات ناشناخته»، دسته‌بندی می‌کند (هشدار نادرست مدل تشخیص ناهنجاری).

اینکه ADAM از یک روش تشخیص سوءاستفاده برای تشخیص ارتباطات از نوع حملات ناشناخته استفاده کند،

2. Hwang
3. Barbara
4. Audit Data Analysis and Mining

غیرمعمول است. در روش ADAM، ارتباط‌هایی که نمی‌توانند به‌عنوان الگوی نرمال یا حملات شناخته‌شده دسته‌بندی شوند، به‌عنوان حملات ناشناخته دسته‌بندی می‌شوند. در صورت استفاده از تشخیص ناهنجاری و به‌دنبال آن تشخیص سوءاستفاده، مدل تشخیص ناهنجاری بهتر است نرخ تشخیص بالایی داشته باشد و مدل تشخیص سوءاستفاده بایستی هشدارهای نادرست مدل تشخیص ناهنجاری را با تمیز حملات شناخته‌شده از ناشناخته، حذف کند. به‌هرحال، اغلب سامانه‌های تشخیص سوءاستفاده برای کاهش هشدارهای نادرست مناسب نیستند. دِپرن و همکاران [۱۳]، سامانهٔ تشخیص نفوذ ترکیبی هوشمندانه‌ای پیشنهاد کردند که شامل یک مدل تشخیص ناهنجاری، یک مدل تشخیص سوءاستفاده و یک سامانهٔ پشتیبان تصمیم‌گیری می‌شد. آن‌ها مدل تشخیص ناهنجاری را با یک شبکهٔ عصبی از نوع نقشهٔ خود-سازمانده[1] (SOM) و مدل تشخیص سوءاستفاده را با یک درخت تصمیم‌گیری مدل‌سازی کردند. هر مدل به‌طور مستقل آموزش داده شده و سپس سامانهٔ پشتیبان تصمیم‌گیری، نتایج دسته‌بندی هر دو مدل را ترکیب می‌کرد.

هوانگ و همکاران [۱۴] روش تشخیص سوءاستفاده و به‌دنبال آن روش تشخیص ناهنجاری را برای طراحی یک سامانهٔ تشخیص نفوذ ترکیبی استفاده کردند. مدل تشخیص سوءاستفاده می‌تواند حملات شناخته‌شده را با نرخ پایین اعلان اشتباه‌های مثبت تشخیص دهد و سریع‌تر از مدل تشخیص ناهنجاری عمل کند. ابتدا مدل تشخیص سوءاستفاده به‌منظور تشخیص حملات شناخته‌شده استفاده شده و سپس مدل تشخیص ناهنجاری فقط برای ارتباط‌های غیرقطعی باقی‌مانده به‌کار برده شده است. مدل تشخیص ناهنجاری، داده‌های دورافتاده[2] را که از الگوهای دادهٔ نرمال جدا هستند، تشخیص داده و آن‌ها را به‌عنوان حملات شناخته‌شده دسته‌بندی می‌کند. اما مانند روش ترکیبی موازی، مدل‌های تشخیص ناهنجاری و تشخیص سوءاستفاده هم به‌طور مستقل آموزش داده شده که این مسئله منجر می‌شود نرخ اعلان اشتباه مثبت بالایی در نتایج مشاهده شود.

کیم و همکاران [۹] با رویکردی ترکیبی، ابتدا ترافیک را طی فاز شناسایی سوءاستفاده با الگوریتم درخت تصمیم‌گیری C4.5 مورد بررسی قرار دادند. پس از مرحلهٔ شناسایی سوءاستفاده، در فاز شناسایی ناهنجاری، به‌ازای هر زیرمجموعه از داده‌ها که توسط مدل شناسایی نفوذ به‌عنوان نرمال شناسایی شده‌اند، ماشین‌های بردار پشتیبان[3] (SVM) جداگانه‌ای را برای شناسایی نفوذهای ناشناس به‌کار بردند. چون هر زیرمجموعه شامل داده‌های متمرکزتری است، کارایی آن در ایجاد پروفایل‌های نرمال بیشتر و در نتیجه در شناسایی ناهنجاری، موفق‌تر خواهد بود.

در بسیاری از پژوهش‌ها در حوزهٔ امنیت شبکه‌های کامپیوتری، از الگوریتم‌های هوش محاسباتی برای انتخاب ویژگی و درخت تصمیم‌گیری برای دسته‌بندی استفاده شده است. برای نمونه، ژنگ و همکاران [۱۵] از نسخهٔ دودویی الگوریتم بهینه‌سازی ازدحام ذرات[4] (BPSO) برای انتخاب ویژگی و از درخت تصمیم‌گیری با الگوریتم آموزش C4.5 برای شناسایی موارد هرز[5] استفاده کرده‌اند. همچنین صبری‌عیسی و همکاران [۱۶] از بهینه‌یابی مبتنی بر CFA[6] برای انتخاب زیرمجموعهٔ بهینه از ویژگی‌ها بهره جسته و دسته‌بندی‌کنندهٔ مبتنی بر درخت تصمیم‌گیری را برای قضاوت دربارهٔ ویژگی‌های انتخابی به‌کار گرفتند. از خصوصیات مثبت درخت تصمیم‌گیری، تخمین و انتخاب ویژگی‌های مناسب و منتخب برای دسته‌بندی داده‌ها در حین فرایند آموزش درخت است. از سویی دیگر، مطالعات نشان می‌دهند که افزودن یک مرحله انتخاب ویژگی قبل از آموزش درخت تصمیم‌گیری منجر به افزایش چشمگیر دقت دسته‌بندی درخت تصمیم‌گیری می‌شود [۱۷]. لذا در راهکار پیشنهادی این مقاله نیز، یک مرحله انتخاب ویژگی توسط الگوریتم جهش قورباغه قبل از آموزش درخت تصمیم‌گیری اضافه شد.

در مقایسه با دیگر الگوریتم‌های بهینه‌سازی موجود، الگوریتم جهش قورباغه از دقت و کارایی بیشتری برخوردار است [۱۷ و ۱۸]. از جمله مزایای الگوریتم این است که قابلیت جستجوی

1. Self-Organizing Map
2. Outlier
3. Support Vector Machine
4. Binary Particle Swarm Optimization
5. Spam
6. Cuttlefish Algorithm

محلی آن منجر به کارایی بیشتر می‌شود. همچنین در حجم بالای تعداد ویژگی‌های مجموعه‌داده‌های آموزشی، قابلیت جستجوی تصادفی این الگوریتم از جستجوی تمام فضای ویژگی جلوگیری می‌کند که همین موضوع منجر به افزایش کارایی آن می‌شود [۱۸]. تعداد ویژگی‌های منتخب توسط این الگوریتم از ویژگی‌های منتخب توسط دیگر الگوریتم‌ها نظیر الگوریتم‌های بهینه‌سازی ازدحام ذرات[1] (PSO) و وراثتی[2] (GA) کمتر است [۱۷].

از سوی دیگر، استفاده از گروه دسته‌بندها[3] [۱۹] در حوزهٔ شناسایی نفوذ در شبکه‌های کامپیوتری سابقه دارد [۲۰]. بدین‌ترتیب که هر دسته‌بند برای کار روی مجموعه‌ای از پروتکل‌ها یا خدمات مشابه در شبکه کار کند. برای نمونه، جیاسینتو و همکاران [۲۱] سامانهٔ شناسایی ناهنجاری را مبتنی بر چندین دسته‌بندی‌کننده ارائه دادند. مطالعهٔ مروری از انواع روش‌های گروهی به‌کاررفته در سامانه‌های تشخیص نفوذ نیز توسط فولینو و ساباتینو [۲۲] انجام شده است. با اینکه روش‌های ترکیب مختلفی از دسته‌بندها وجود دارد، یافتن پیکربندی مناسب برای این ترکیب کار مشکلی است [۲۳]. بسیاری از گروه دسته‌بندها تنها به شناسایی سوءاستفاده یا شناسایی ناهنجاری پرداخته‌اند [۲۱ و ۲۲]، حال‌آنکه طرح پیشنهادی در این مقاله به هر دو شناسایی پرداخته و از این لحاظ، علاوه‌بر اینکه از چند دسته‌بند (درخت تصمیم‌گیری، شبکهٔ عصبی مصنوعی و ماشین بردار پشتیبان) استفاده می‌کند، به‌کمک آن‌ها هر دو نوع شناسایی را انجام می‌دهد.

در ادامه، به معرفی مختصر الگوریتم‌ها و مدل‌های به‌کاررفته در این مقاله پرداخته شده است تا در بخش‌های آتی، چگونگی ترکیب آن‌ها با یکدیگر و نتایج حاصل از شبیه‌سازی‌ها بیان شود.

۲.۲. الگوریتم جهش قورباغه

الگوریتم‌های تکاملی بسیاری در راستای کاهش زمان پردازش و بهبود کیفیت راه‌حل‌ها طی دهه‌های اخیر پیشنهاد شده‌اند که الگوریتم ترکیبی بهینه‌سازی مبتنی بر جهش قورباغه[4] (SFLO) یکی از این الگوریتم‌هاست [۲۴].

این الگوریتم یک روش مکاشفه‌ای تقلیدی [۲۵] است که برای یافتن راه حل بهینهٔ کلی، از طریق جستجوی مکاشفه‌ای طراحی شده است. این الگوریتم بر روی مسائل ترکیبی مختلف به‌منظور یافتن راه‌حل‌های کلی کارآمد آزمون شده است. الگوریتم جهش قورباغه شامل جمعیتی از راه‌حل‌های ممکن تعریف‌شده از طریق مجموعه‌ای از قورباغه‌ها (راه‌حل‌ها) است که به زیرمجموعه‌هایی تقسیم شده‌اند. زیرمجموعه‌های مختلف به‌عنوان گونه‌های مختلف قورباغه محسوب می‌شوند که هریک جستجوی محلی جداگانه‌ای انجام می‌دهند. در داخل زیرمجموعه‌ها، هریک از قورباغه‌ها ایده‌هایی دارند که تحت تأثیر ایده‌های قورباغه‌های دیگر قرار دارند و توسط فرایند تکامل تقلیدی متحول می‌شوند. پس از طی مراحل تکامل تقلیدی، ایده‌ها در میان زیرمجموعه‌ها از طریق فرایند درهم‌سازی و بُرزدن جریان می‌یابند. جستجوی محلی و فرایندهای بُرزدن ادامه می‌یابند تا ضابطه همگرایی برآورده شود [۲۴].

۳.۲. درخت تصمیم‌گیری

درخت تصمیم‌گیری یکی از مشهورترین روش‌های ساخت مدل دسته‌بندی است. در الگوریتم‌های دسته‌بندی مبتنی بر درخت تصمیم‌گیری، دانش خروجی به‌صورت یک درخت از حالات مختلف مقادیر ویژگی‌ها ارائه می‌شود.

یک استراتژی حریصانه برای ساخت درخت تصمیم‌گیری این است که رکوردها همیشه بر پایهٔ یک ویژگی کاندید که یک معیار خاص را بهینه می‌کند، شکسته می‌شوند. ویژگی که با توجه به این معیار، بهترین بهبود را برای درخت به ارمغان می‌آورد، شایسته‌ترین ویژگی خواهد بود. بسته به چگونگی انتخاب ویژگی‌های مجموعه‌داده برای قرار گرفتن در درخت تصمیم‌گیری و نیز زمان توقف ساخت درخت، انواع متفاوتی از درخت‌های تصمیم‌گیری وجود دارد. شکستی که توزیع دسته‌ها در گره‌های حاصل از آن همگن باشد، نسبت به سایر شکست‌ها بهتر است. منظور از همگن‌بودن گره این است که همهٔ رکوردهای موجود در آن، متعلق به یک دستهٔ خاص باشند. چون در این‌صورت آن گره

1. Particle Swarm Optimization
2. Genetic Algorithm
3. Ensemble of classifiers
4. Shuffled Frog-Leaping Optimization

به برگ تبدیل می‌شود. در واقع، گره همگن گرهی است که کمترین میزان ناخالصی را داشته باشد.

پس از محاسبهٔ میزان ناخالصی حاصل از هریک از شکست‌ها، آن‌ها را در رابطهٔ بهره قرار داده و میزان بهرهٔ حاصل از هر شکست محاسبه می‌شود. در رابطهٔ بهره، ناخالصی حاصل از شکست ایجادشده از ناخالصی گره والد کم می‌شود. هر شکستی که بهرهٔ بالاتری ایجاد کند، شکست بهتری خواهد بود و در نهایت آن شکست انتخاب می‌شود [۲۶]. درخت تصمیم‌گیری به‌کاررفته در این مقاله درخت C4.5 است که برای محاسبهٔ میزان ناخالصی آن از روش آنتروپی براساس رابطهٔ (۱) استفاده می‌شود:

$$Entropy(t) = -\sum_i p(j|t) logp(j|t) \quad (۱)$$

که در رابطهٔ (۱) p(j|t) نشانگر تعداد رکوردهای دستهٔ jام موجود در گره t، در مقایسه با کل رکوردهای موجود در گره t است. پس از محاسبهٔ آنتروپی برای تک‌تک گره‌ها و سپس محاسبهٔ آنتروپی برای کل شکست، بهرهٔ شکست محاسبه می‌شود. هرچه میزان بهرهٔ یک شکست بیشتر باشد، شکست بهتری محسوب خواهد شد. رابطهٔ (۲) چگونگی محاسبهٔ بهرهٔ یک شکست را نشان می‌دهد:

$$GAIN_{split} = Entropy(p) - \sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \quad (۲)$$

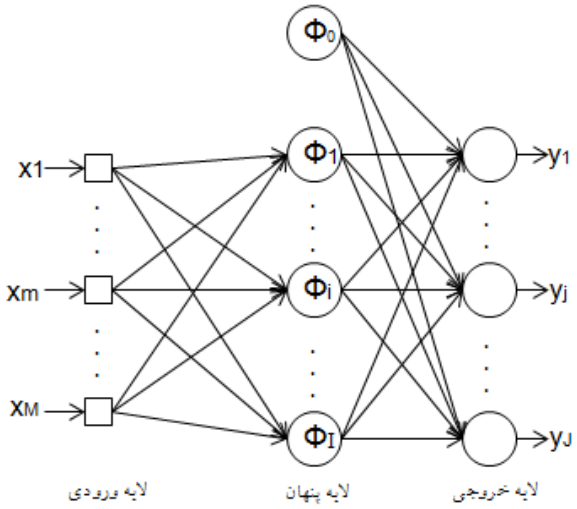
که در این رابطه، n نشانگر تعداد کل رکوردهای موجود در گره والد، n_i مبین تعداد رکوردهای وارد به فرزند iام، Entropy(p) نشانگر آنتروپی گره والد و Entropy(i) نیز نشانگر آنتروپی گره iام است [۲۶]. پس از ساخت مدل دسته‌بندی مبتنی بر درخت تصمیم‌گیری، می‌توان آن را بر روی مجموعه‌داده‌های آزمایشی اعمال کرد. منظور از اعمال‌کردن مدل، پیش‌بینی مقدار ویژگی دسته برای رکورد آزمایشی براساس مدل ساخته شده است.

۴.۲. شبکهٔ عصبی پایه- شعاعی (RBF)

مزیت شبکهٔ عصبی مصنوعی که آن را از طبقه‌بندی‌کننده‌های دیگر متمایز می‌کند، قابلیت تعمیم آن است. شبکه‌های با تابع پایه- شعاعی به‌طور گسترده‌ای برای تخمین ناپارامتری توابع چندبعدی از طریق مجموعه‌ای محدود از اطلاعات آموزشی به کار می‌روند. شبکهٔ RBF به‌واسطهٔ آموزش سریع و فراگیر، بسیار مفید است؛ به‌طوری که با داشتن تعداد نرون‌های کافی در لایهٔ مخفی، قادر به تقریب هر تابع پیوسته و با هر درجه از دقت است.

۱.۴.۲. ساختار RBF

معماری اصلی RBF متشکل از یک شبکهٔ سه‌لایه به شکل (۲) است:



شکل (۲): ساختار یک شبکهٔ RBF [۲۷]

نرون‌های لایهٔ میانی (پنهان) همان توابع پایه- شعاعی هستند [۲۷]. لایهٔ سوم نیز برای تقریب، جمع وزنی خروجی نرون‌های لایهٔ میانی را براساس رابطهٔ (۳) تولید می‌کند:

$$F(x) = \sum_{j=1}^{p} w_j \emptyset(\|x - u_j\|) \quad (۳)$$

درصورتی‌که از RBF برای تقریب تابع استفاده شود، این خروجی مفید خواهد بود. ولی درصورتی‌که نیاز به طبقه‌بندی الگوها باشد، آنگاه یک محدودکنندهٔ سخت را می‌توان به نرون‌های خروجی اعمال کرد تا مقادیر خروجی ۰ یا ۱ تولید شوند. براساس رابطهٔ (۳) برای تقریب تابع F از p تابع پایه- شعاعی که دارای مراکز u_j هستند، استفاده می‌شود. نماد ||.|| مبین تابع فاصله در فضای R_n است که معمولاً فاصلهٔ اقلیدسی انتخاب می‌شود. توابع پایه- شعاعی مختلفی پیشنهاد شده‌اند که معروف‌ترین آن‌ها در شبکه‌های RBF همان تابع گوسی است. دلیل انتخاب تابع گوسی به‌عنوان تابع پاسخ نرون‌ها در

شبکه‌های RBF این است که تابع نمایی از گروه توابعی است که بهترین خواص تقریب‌سازی را دارد. این موضوع تضمین می‌کند که مجموعه‌ای از وزن‌ها وجود دارند که رابطهٔ بین ورودی‌ها و بردارهای هدف را بهتر از هر مجموعهٔ دیگر تقریب می‌زنند و این خاصیت در تابع سیگموئید که در طراحی شبکه‌های پس‌انتشار خطا به کار برده می‌شود، وجود ندارد.

۲.۴.۲. تعیین موقعیت مرکز توابع گوسی

در شبکه‌هایRBF، محاسبهٔ مراکز برای یک مجموعهٔ بزرگ آموزشی می‌تواند بار محاسباتی سنگین و درعین‌حال غیرضروری را به سیستم تحمیل کند. یافتن مراکز بهینهٔ توابع پایه- شعاعی خود می‌تواند به‌عنوان یک مسئله بهینه‌سازی مطرح شود. برای این منظور می‌توان از الگوریتم‌های هوش محاسباتی نیز استفاده کرد که در این مقاله، آموزش شبکهٔ عصبی RBF با PSO و GA به‌منظور مقایسه صورت گرفته است [۲۸].

۲.۵. الگوریتم بهینه‌سازی ازدحام ذرات

الگوریتم PSO یک الگوریتم بهینه‌سازی تصادفی براساس جمعیت است که از شبیه‌سازی رفتار اجتماعی گروه پرندگان مدل‌سازی شده است. گروهی از پرندگان در فضایی به‌صورت تصادفی به‌دنبال غذا می‌گردند. یکی از بهترین استراتژی‌ها می‌تواند دنبال‌کردن پرنده‌ای باشد که کمترین فاصله را تا غذا دارد. این استراتژی ایدهٔ اصلی الگوریتم PSO است [۲۹ و ۳۰].

فضای مسئلهٔ مورد جستجو در الگوریتم PSO معادل فضای مورد جستجو در الگوی حرکت پرندگان است. هر راه‌حل که به آن یک ذره گفته می‌شود، در الگوریتم PSO معادل یک پرنده است و تعداد ذرات (راه‌حل‌ها) معادل تعداد پرندگان است. هر ذره، یک مقدار شایستگی دارد که توسط یک تابع شایستگی محاسبه می‌شود و هرچه ذره در فضای جستجو به هدف، یعنی غذا در مدل حرکت پرندگان نزدیک‌تر باشد، شایستگی بیشتری دارد. همچنین هر ذره دارای یک جابه‌جایی است که هدایت حرکت ذره را بر عهده دارد و به‌کمک آن مکان بعدی ذره مشخص می‌شود. هر ذره با دنبال‌کردن ذرات بهینه در حالت فعلی، به حرکت خود در فضای جستجو ادامه می‌دهد تا اینکه نهایتاً به جواب بهینه دست بیابد [۲۹ و ۳۰].

در هر مرحله از تکرار الگوریتم ازدحام ذرات، بردار سرعت ذره (V_i) و موقعیت ذره (X_i) در تکرار ($t+1$)ام از روابط (۴) و (۵) تعیین می‌شوند:

$$V_i(t+1) = w \times V_i(t) + C_1 \times rand_1 \times [pbest_i(t) - X_i(t)] + C_2 \times rand_2 \times [gbest_i(t) - X_i(t)] \quad (۴)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (۵)$$

اجزای روابط فوق به شرح زیر است:

- $X_i(t)$: مکان فعلی ذرهٔ iام در تکرار tام
- $V_i(t)$: بردار سرعت فعلی ذرهٔ iام در تکرار tام
- $pbest_i(t)$: بهترین مکان تجربه‌شده توسط ذرهٔ iام تا تکرار tام
- $gbest_i(t)$: بهترین مکان به‌دست‌آمده میان همسایگان ذرهٔ i ام تا تکرار tام
- $rand_1$ و $rand_2$: اعداد تصادفی بین 0 و 1
- ضریب یادگیری شخصی C_1 و ضریب یادگیری جمعی C_2. ضرایب C_1 و C_2 تأثیر اجزای تجارب شخصی و گروهی را روی جستجوی تصادفی تعیین می‌کنند. این پارامترها به نام پارامترهای اعتماد نیز شناخته می‌شوند؛ زیرا ضریب C_1 تعیین می‌کند که یک ذره تا چه حد به تجربه‌های خود وابسته باشد، درحالی‌که ضریب C_2 تعیین‌کنندهٔ این امر است که یک ذره تا چه حد به تجربه‌های همسایگان خود تکیه دارد.
- پارامتر وزن اینرسی w: وزن اینرسی در واقع ضریبی از سرعت فعلی ذره است که در تعیین جابه‌جایی در گام بعد مورد استفاده قرار می‌گیرد.

۲.۶. الگوریتم وراثتی

الگوریتم ژنتیک توسط هالند در سال ۱۹۶۵ ارائه شد. به‌طور کلی، الگوریتم‌های وراثتی از اجزای زیر تشکیل می‌شوند [۳۱ و ۳۲]:

الف. کروموزوم: در الگوریتم وراثتی، هر کروموزوم نشان‌دهندهٔ یک نقطه در فضای جستجو و یک راه‌حل ممکن برای مسئله مورد نظر است. کروموزوم‌ها از تعداد ثابتی ژن (متغیر)

تشکیل شده‌اند. برای نمایش کروموزوم‌ها معمولاً از کدگذاری‌های دودویی (رشته‌های بیتی) استفاده می‌شود.

ب. جمعیت: مجموعه‌ای از کروموزوم‌ها یک جمعیت را تشکیل می‌دهند. با تأثیر عملگرهای وراثتی روی هر جمعیت، جمعیت جدیدی با همان تعداد کروموزوم تشکیل می‌شود.

پ. تابع برازندگی: به‌منظور حل هر مسئله با استفاده از الگوریتم‌های وراثتی، ابتدا باید یک تابع برازندگی برای آن مسئله ارائه شود. برای هر کروموزوم، این تابع عددی غیرمنفی را بازمی‌گرداند که نشان‌دهندهٔ شایستگی یا توانایی فردی آن کروموزوم است.

ت. عملگرهای الگوریتم وراثتی: در الگوریتم‌های وراثتی، در طی مرحلهٔ تولیدمثل از عملگرهای وراثتی استفاده می‌شود. با تأثیر این عملگرها بر روی یک جمعیت، نسل بعدی آن جمعیت تولید می‌شود. عملگرهای انتخاب، تقاطع و جهش معمولاً بیشترین کاربرد را در الگوریتم‌های وراثتی دارند.

عملگر انتخاب از بین کروموزوم‌های موجود در یک جمعیت، تعدادی کروموزوم را برای تولیدمثل انتخاب می‌کند. کروموزوم‌های برازنده‌تر شانس بیشتری دارند تا برای تولیدمثل انتخاب شوند. در جریان عمل تقاطع به‌صورت اتفاقی بخش‌هایی از کروموزوم‌ها با یکدیگر تعویض می‌شوند. این عمل باعث می‌شود که فرزندان، ترکیبی از خصوصیات والدین خود را داشته باشند و دقیقاً مشابه یکی از والدین نباشند. تقاطع ممکن است به‌صورت تک‌نقطه‌ای، دونقطه‌ای یا یکنواخت صورت گیرد.

در تقاطع یکنواخت، تمام نقاط کروموزوم شانس یکسان برای ترکیب دارند. در این وضعیت، ژن‌های کروموزوم فرزند می‌تواند از هر والدی انتخاب شود. روابط (۶) و (۷) برای انجام تقاطع یکنواخت به کار می‌روند:

$$y_{1i} = \alpha_i x_{1i} + (1 - \alpha_i) x_{2i} \quad (۶)$$

$$y_{2i} = \alpha_i x_{2i} + (1 - \alpha_i) x_{1i} \quad (۷)$$

که در این روابط، y_1 و y_2 به‌ترتیب مبین فرزندان اول و دوم و x_1 و x_2 والدین هستند. همچنین در مسائل دودویی، مقادیر α_i برابر صفر یا یک و در مسائل پیوسته، مقادیر α_i در بازهٔ $[0,1]$ است. i بیان‌گر ابعاد فضای راه‌حل (ابعاد ورودی‌ها) است.

پس از اتمام عمل تقاطع، عملگر جهش بر روی کروموزوم‌ها اثر داده می‌شود. این عملگر یک ژن از یک کروموزوم را به‌طور تصادفی انتخاب کرده و سپس محتوای آن ژن را تغییر می‌دهد. اگر ژن از جنس اعداد دودویی باشد، آن را به وارونش تبدیل می‌کند و چنانچه متعلق به یک مجموعه باشد، مقدار یا عنصر دیگری از آن مجموعه را به‌جای آن ژن قرار می‌دهد. پس از اتمام عمل جهش، کروموزوم‌های تولیدشده به‌عنوان نسل جدید شناخته شده و برای دور بعدیِ اجرای الگوریتم ارسال می‌شوند.

۳. روش پیشنهادی

اصول کار در روش پیشنهادی در شکل (۱) ارائه شد. در این بخش، به بیان جزئیات بخش‌های شناسایی سوءاستفاده و ناهنجاری پرداخته خواهد شد. در روش ترکیبی پیشنهادی، قبل از اعمال داده‌ها به مدل ابتدا به‌منظور پیش‌پردازش داده‌ها با الگوریتم هوش محاسباتی جهش قورباغه، ویژگی‌های مهم آن استخراج شد تا از این رهگذر عملکرد کلی سامانه و به‌خصوص کارایی درخت تصمیم‌گیری افزایش یابد.

در ادامه، طی مرحلهٔ شناسایی سوءاستفاده، داده‌های آموزشی نرمال به زیرمجموعه‌هایی که الگوهای اتصالات آن‌ها تنوع کمتری نسبت به کل داده‌های نرمال دارند، تقسیم می‌شوند. سپس در مرحلهٔ شناسایی ناهنجاری، به‌ازای هر زیرمجموعه مدل شناسایی ناهنجاری جداگانه‌ای به کار می‌رود و چون هر زیرمجموعه شامل داده‌های متمرکزتری است، کارایی آن در ایجاد پروفایل‌های نرمال بیشتر و در نتیجه در شناسایی ناهنجاری، موفق‌تر خواهد بود [۹].

اگر به هنگام آموزش مجدد مدل در محیط عملیاتی، درخت تصمیم‌گیری نتواند داده‌ها را به‌صورت کارا به زیرمجموعه‌ها تقسیم کند، کیفیت داده‌های ورودی به مرحلهٔ شناسایی ناهنجاری به‌قدر کافی خوب نبوده و تأثیر زیادی بر روی دقت شبکهٔ SVM می‌گذارد و این امر کارایی شبکهٔ SVM را کاهش می‌دهد؛ در مقابل در شبکهٔ عصبی RBF این محدودیت وجود ندارد. گرچه آموزش شبکه عصبی RBF از SVM کندتر است، ازآنجاکه آموزش مدل، فرایندی برون‌خط است، تأثیری بر روی کارایی مدل‌ها ندارد [۳۳].

با توجه به موارد فوق و برای جلوگیری از کاهش دقت شبکهٔ SVM، در مواردی که خروجی درخت تصمیم‌گیری کارا نیست؛ از شبکهٔ عصبی RBF در فاز شناسایی ناهنجاری استفاده شده و به مقایسهٔ عملکرد SVM و RBF در فاز شناسایی ناهنجاری پرداخته شد. شایان ذکر است که آموزش RBF با الگوریتم‌های هوش محاسباتی انجام شده است. به‌کارگیری RBF به‌جای SVM و انتخاب ویژگی قبل از مرحلهٔ شناسایی سوءاستفاده به‌منظور کاهش ابعاد مسئله تغییراتی بودند که منجر به بهبود چشمگیری در کارایی روش پیشنهادی جدید شدند. همچنین برای شبیه‌سازی‌های انجام‌شده، از نرم‌افزار Matlab نسخهٔ 2013a استفاده شده است.

به‌طور خلاصه، شناسایی نفوذ در رویکرد پیشنهادی به‌ترتیب از مراحل زیر تشکیل شده است:

۱. پیش‌پردازش داده‌ها شامل گام‌های زیر:

الف. همگن‌سازی داده‌ها (داده‌های آموزش و آزمون) از طریق جای‌گذاری نویسه‌های حرفی مجموعه‌داده با مقادیر عددی

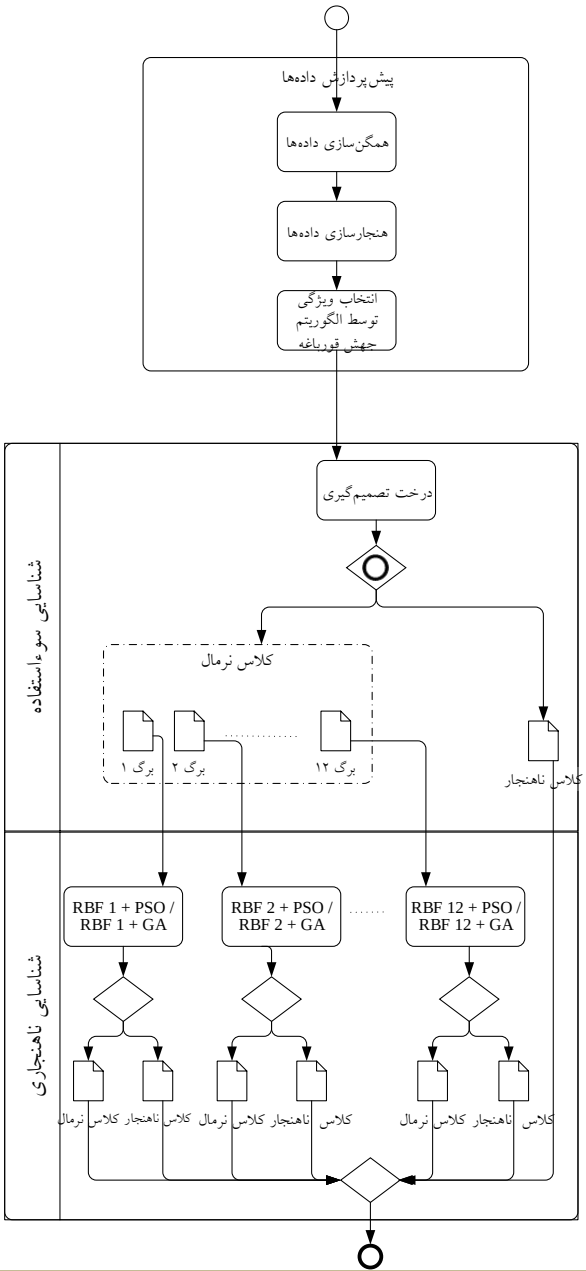
ب. هنجارسازی داده‌ها (داده‌های آموزش و آزمون): در این مرحله مقادیر ویژگی‌های پیوسته طبق رابطهٔ (۸) به بازهٔ [-1,1] هنجارسازی می‌شوند.

$$X = 2 \times \frac{x - \min(x)}{\max(x) - \min(x)} - 1 \qquad (۸)$$

پ. کاهش ابعاد مسئله با انتخاب ویژگی توسط الگوریتم هوش محاسباتی جهش قورباغه

باید یادآوری شود که هنجارسازی (یا تغییر مقیاس ویژگی‌ها)، عدم توازن بین داده‌ها را از بین می‌برد [۳۴ـ۳۶]. در دادگان مورد استفاده در این مقاله (NSL-KDD)، برخی از ویژگی‌ها دارای مقادیر عددی بزرگ هستند که می‌توانند بر دیگر ویژگی‌ها چیره شوند. برای نمونه، ویژگی dst_bytes می‌تواند مقادیر صفر تا حدود $۱/۳\times۱۰^{۹}$ را داشته باشد، حال‌آنکه ویژگی same_srv_rate مقادیر بین صفر تا یک را دارد [۳۷].

۲. به‌کارگیری روش‌های گوناگون در مدل ترکیبی شناسایی نفوذ (ابتدا فاز شناسایی سوءاستفاده و سپس فاز شناسایی ناهنجاری) شامل درخت تصمیم‌گیری C4.5 به‌منظور شناسایی سوءاستفاده و سپس به‌کارگیری RBFهای جداگانه در فاز تشخیص ناهنجاری به‌ازای دسته‌هایی که نرمال شناسایی شده‌اند. یادآوری می‌شود که پارامترهای آموزش در شبکهٔ عصبی RBF به‌کمک الگوریتم وراثتی یا الگوریتم ازدحام ذرات تعیین شده‌اند. شمای بلوکی ارائه‌شده در شکل (۳)، مراحل کار را در روش تشخیص نفوذ پیشنهادی با جزئیات بیشتر نمایش می‌دهد.



شکل (۳): روندنمای روش تشخیص نفوذ پیشنهادی

۱.۳. مرحلهٔ شناسایی سوءاستفاده

پس از پیش‌پردازش داده‌ها و انتخاب ویژگی، در فرایند داده‌کاوی، داده آمادهٔ اعمال به مرحلهٔ یادگیری مدل می‌شود. برای این منظور، از درخت تصمیم‌گیری C4.5 استفاده شده است. در مرحلهٔ یادگیری، نظم حاکم بر داده‌های پیش‌پردازش‌شده با توجه به روش کاوش داده‌ای که انتخاب می‌شود، شناسایی شده و مدل تولیدشده برای ارزیابی به مرحلهٔ بعد (یعنی ارزیابی و تفسیر مدل) منتقل خواهد شد. پس از آموزش درخت تصمیم‌گیری و ذخیرهٔ آن، نوبت به هرس آن رسید، تا جایی که در برگ‌های باقی‌مانده تنها ۱۲ برگ با برچسب نرمال باقی ماندند. ازآنجاکه تقسیم داده‌های آموزشی به زیرمجموعه‌های مجزا بر مبنای قواعد مختلف، فرایندی زمان‌بر است، ایجاد مجموعه‌های بیش از حد کوچک منجر به کندی بیش از حد این عملیات می‌شود. از سوی دیگر، محدودیت و عدم پراکندگی بیش از حد ورودی‌های شبکهٔ عصبی در فاز آموزش، منجر به کاهش قدرت تعمیم آن خواهد شد؛ در نتیجه بنا به این دلایل درخت هرس شد.

۲.۳. مرحلهٔ شناسایی ناهنجاری

در این مرحله، به‌ازای هرکدام از برگ‌های باقی‌مانده با برچسب نرمال، RBFهای جداگانه‌ای به کار گرفته شد که داده‌های ورودی به هرکدام از آن‌ها برابر با داده‌هایی بودند که طبق دسته‌بندی صورت‌گرفته توسط درخت تصمیم‌گیری به برگ مربوط به آن شبکهٔ عصبی رسیده بودند. برای به‌دست‌آوردن داده‌های موجود در هرکدام از برگ‌ها، به‌ازای همهٔ برگ‌های با برچسب نرمال شروط سازندهٔ آن‌ها را به دست آورده و سپس بر مبنای آن شروط، مجموعه‌داده‌های آموزشی به زیرمجموعه‌های ورودی مجزا تقسیم شدند که به‌ازای هرکدام از آن‌ها شبکه‌های عصبی مجزا به کار رفتند. این کار باعث می‌شود تا دقت شبکه به‌دلیل استفاده از مجموعه‌داده‌های همگن‌تر افزایش یابد و سامانهٔ شناسایی ناهنجاری به‌دلیل آشنایی با الگوهای نرمال که پراکندگی کمتری دارند، در مقابله با ناهنجاری‌هایی که از الگوی نرمال تبعیت نمی‌کنند، بهتر قادر به شناسایی آن‌ها باشد [۳۸].

۴. نتایج ارزیابی راهکار پیشنهادی

۱.۴. مجموعه‌دادهٔ NSL-KDD

برای تجزیه و تحلیل و نیز مقایسهٔ عملکرد راهکار پیشنهادی از مجموعه‌دادهٔ NSL-KDD استفاده شده است [۳۹]. دادگان NSL-KDD نسخهٔ ویرایش‌شدهٔ مجموعه‌دادهٔ KDD'99 است که به‌دلیل مشکلات KDD'99 درخصوص وجود نمونه‌های تکراری زیاد در داده‌های آموزشی و آزمون و با برطرف کردن این مشکلات ارائه شده است. این مجموعه‌داده ۴۱ ویژگی دارد که از انواع پیوسته، گسسته و نمادین با گسترهٔ وسیعی از مقادیر عددی می‌باشند. همان‌گونه که بیان شد، پیش‌پردازش داده‌ها برای هنجارسازی عدم توازن در کل دادگان و نیز حذف اثر اختلاف مقیاس در چنین پایگاه داده‌ای ضروری بوده و در کارهای مشابه نیز اجرا شده است [۳۴ـ۳۷]. به بیان دیگر، با هدف جلوگیری از غلبهٔ ویژگی‌های با مقادیر عددی بزرگ بر سایر ویژگی‌ها، هنجارسازی در چنین دادگانی بر روی تمامی داده‌ها (که در ادامه، برخی از آن‌ها به‌عنوان داده‌های آموزشی و برخی دیگر به‌عنوان داده‌های آزمون انتخاب می‌شوند) صورت می‌پذیرد [۴۰ـ۴۲].

۲.۴. نتایج ارزیابی کلی

به‌منظور کاهش ابعاد مسئله، تعداد ۱۰ ویژگی به‌شکل هوشمند از میان آن‌ها با به‌کارگیری الگوریتم جهش قورباغه انتخاب شدند. انتخاب ویژگی (حذف متغیر) به درک داده‌ها، کاهش نیاز محاسباتی، کاهش اثر مسئلهٔ ابعاد و بهبود عملکرد پیش‌گو کمک می‌کند. تمرکز انتخاب ویژگی بر این است که زیرمجموعه‌ای از متغیرهایی را از ورودی انتخاب کند که بتواند به‌طور کارا دادهٔ ورودی را توصیف کند، درحالی‌که اثر متغیرهای غیرضروری را کاهش داده و نتایج پیش‌بینی خوبی را فراهم کند [۴۳].

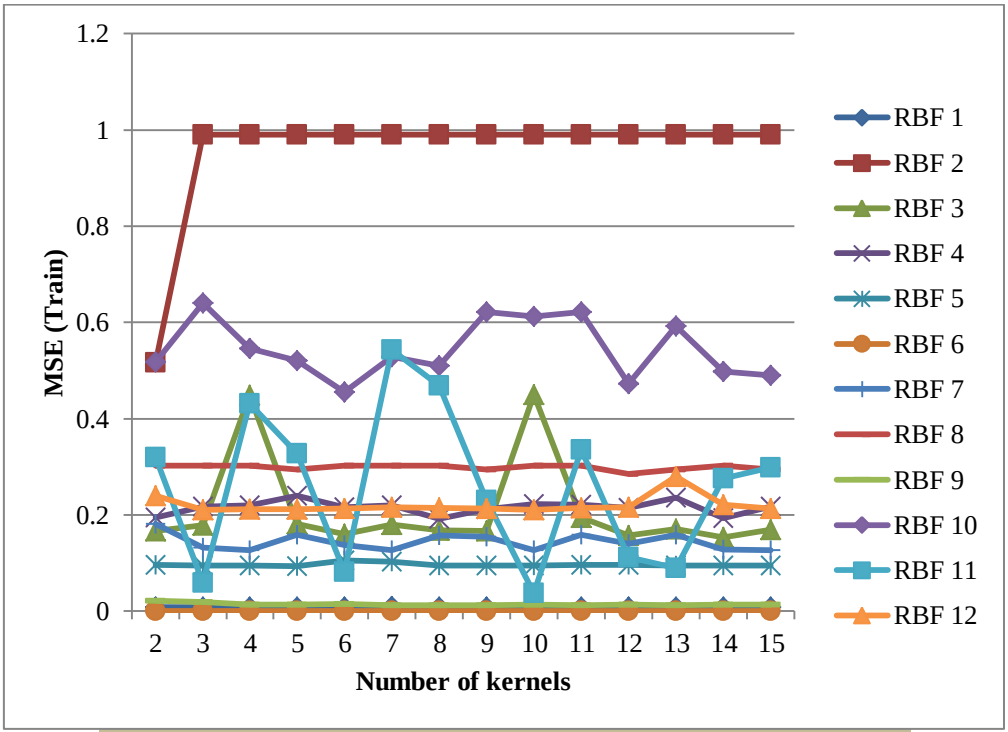
در این مرحله، به‌منظور ارزیابی مدل، ابتدا کلیهٔ داده‌های ورودی در فاز شناسایی سوءاستفاده (تشخیص امضا) به درخت تصمیم‌گیری آموزش‌دیده داده شدند. دربارهٔ داده‌هایی

که درخت تصمیم‌گیری، آن‌ها را به‌عنوان بسته‌های نرمال شناسایی کرده بود، با فرض اینکه حملاتی ناشناس هستند که امضای آن‌ها یا همان ویژگی‌های بستهٔ داده برای مدل شناسایی سوءاستفاده قابل تشخیص نبوده است، بار دیگر توسط قسمت شناسایی ناهنجاری (RBF آموزش‌دیده) بررسی شدند تا تصمیم نهایی درخصوص کلاس آن‌ها در این فاز گرفته شود. شیوهٔ کار در فاز شناسایی ناهنجاری به این شکل است که ورودی‌های این فاز ابتدا طبق شروط سازندهٔ برگ‌های درخت تصمیم‌گیری بار دیگر مورد بررسی قرار می‌گیرند تا مشخص شود که به‌عنوان ورودی به کدام‌یک از RBFها باید داده شوند. پس از آن، RBF انتخابی دربارهٔ کلاس نهایی بستهٔ داده تصمیم‌گیری می‌کند.

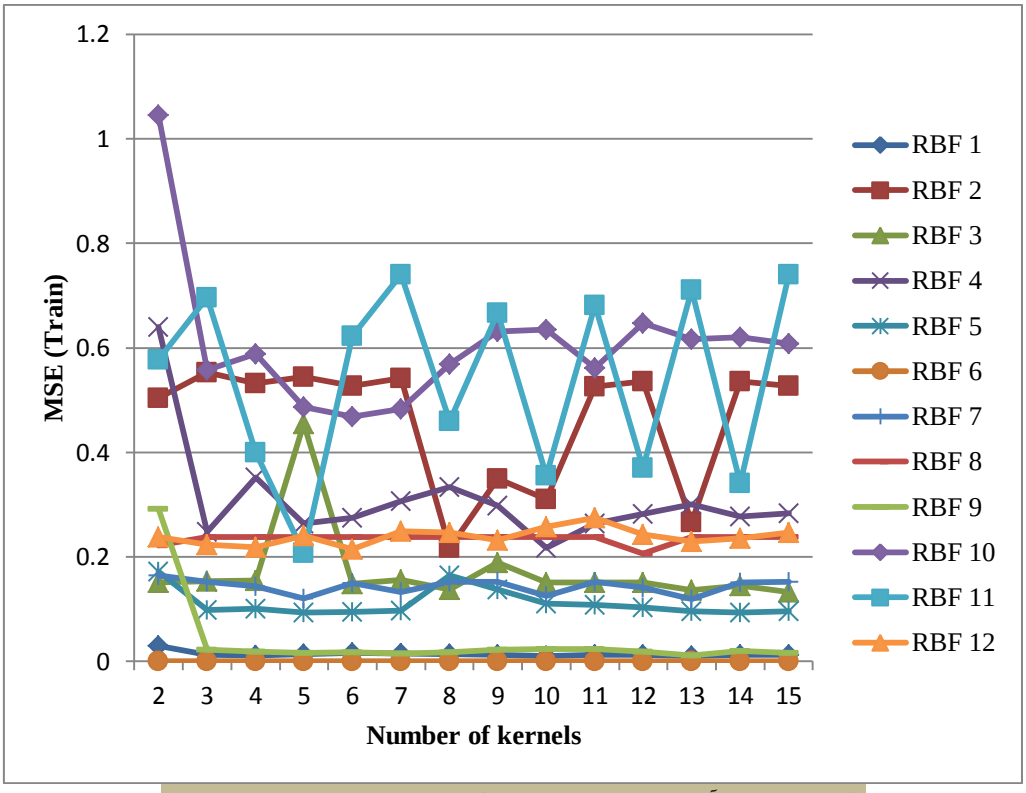
پس از بررسی نتایج مشاهده شد که در صورت به‌کارگیری RBF به‌جای SVM با کاهش چشمگیر زمان لازم برای اجرای مدل مواجهیم، به‌گونه‌ای که دربارهٔ زمان لازم برای آزمون، این زمان به‌طور متوسط ۲۶ مرتبه کمتر شد (در صورت استفاده از PSO، ۲۴ مرتبه بهبود و در صورت استفاده از GA بیش از ۲۸ مرتبه بهبود در زمان اجرا خواهیم داشت)؛ درحالی‌که خطای مدل نیز با اندکی کاهش مواجه بود.

گفتنی است که طی فرایند آموزش RBF تعداد کرنل بهینهٔ هرکدام از RBFها به‌صورت جداگانه مشخص شد. تعداد هسته‌های بهینه به‌ازای هرکدام از RBFها در صورت به‌کارگیری الگوریتم‌های PSO و وراثتی به‌ترتیب از شکل‌های (۴) و (۵) قابل استخراج است. هرکدام از خطوط موجود در نمودارها بیانگر روند تغییرات خطای MSE یکی از RBFها به‌ازای تعداد کرنل‌های گوناگون است و تعداد کرنل بهینه هر RBF برابر با تعداد کرنلی است که منجر به کمینه شدن خطای MSE شود.

همان‌طور که قبلاً نیز اشاره شد، آموزش شبکهٔ عصبی RBF به‌کمک الگوریتم‌های PSO و وراثتی (GA) به‌شکل مجزا انجام شد و نتایج این دو با یکدیگر مقایسه شدند. درخصوص کارایی به‌کارگیری PSO یا GA در آموزش RBF، نتایج تقریباً مشابه بود و تفاوت چشمگیری مشاهده نشد.



شکل (۴): خطای آموزش تمامی RBFها به‌کمک الگوریتم PSO به‌ازای تعداد کرنل‌های متفاوت



شکل (۵): خطای آموزش تمامی RBFها به‌کمک GA به‌ازای تعداد کرنل‌های متفاوت

برای ارزیابی مدل‌ها از معیار میانگین مربع خطا (MSE)، جذر میانگین مربع خطا (RMSE)، میانگین قدرمطلق خطا (MAE) و مجموع مربعات خطا (SSE) طبق روابط (۹) تا (۱۲) استفاده شد:

$$\mathrm{MSE} = \sum_{i=1}^{n} \frac{(t_i - y_i)^2}{n} \quad (۹)$$

$$\mathrm{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(t_i - y_i)^2}{n}} \quad (۱۰)$$

$$\mathrm{MAE} = \sum_{i=1}^{n} \frac{|t_i - y_i|}{n} \quad (۱۱)$$

$$\mathrm{SSE} = \sum_{i=1}^{n} (t_i - y_i)^2 \quad (۱۲)$$

در روابط فوق، n مبین تعداد نمونه‌ها، t_i نشانگر مقدار خروجی هدف و y_i مقدار خروجی کلاس تخمین زده‌شده توسط مدل است [۴۴]. مراحل آموزش و آزمون مدل‌ها با استفاده از رایانه‌ای با پردازندهٔ Intel Core i7 با سرعت ۲/۱ گیگاهرتز و حافظهٔ دسترسی تصادفی ۶ گیگابایت انجام شده است. تعداد داده‌های به‌کاررفته برای آموزش مدل‌ها برابر با ۶۲۸۶۴ داده و برای آزمایش مدل‌ها برابر با ۸۵۳۴۷ داده است. در ادامه، نتایج حاصل از به‌کارگیری مدل‌ها ارائه شده است.

۳.۴. نتایج آموزش درخت تصمیم‌گیری

در جدول (۱) نتایج خطا در شناسایی سوءاستفاده آورده شده است. مدت زمان لازم برای آموزش درخت نیز برابر با ۰/۱۸ ثانیه بود.

جدول (۱): مقادیر خطای آموزش و آزمون در فاز شناسایی سوءاستفاده توسط درخت تصمیم‌گیری

فاز	MSE	MAE	SSE
آموزش	۰/۰۰۴۰۱	۰/۰۰۲۰۰	۲۵۲
آزمون	۰/۲۰۳۴	۰/۱۰۱۷	۱۷۳۶۰

۴.۴. نتایج مدل ترکیبی درخت تصمیم‌گیری و RBF با آموزش توسط الگوریتم PSO

در جدول (۲)، نتایج به‌کارگیری مدل ترکیبی تشخیص نفوذ در

شرایطی که آموزش مدل RBF با الگوریتم PSO انجام شده باشد، ارائه شده است. خطاهای ارائه‌شده در جدول (۲) به‌ازای این تنظیم مقادیر برای پارامترهای الگوریتم PSO به دست آمدند: تعداد اعضای جمعیت: ۶۰، ضریب یادگیری شخصی (C_1): ۱/۴۹۶۲، ضریب یادگیری جمعی (C_2): ۱/۴۹۶۲، ضریب w: ۱ و ضریب کاهشی w: ۰/۹.

۵.۴. نتایج مدل ترکیبی درخت تصمیم‌گیری و RBF با آموزش توسط GA

در جدول (۳) نیز نتایج به‌کارگیری مدل ترکیبی تشخیص نفوذ در شرایطی که آموزش مدل RBF با الگوریتم وراثتی انجام شده باشد، ارائه شده است. خطاهای ارائه‌شده در جدول (۳) به‌ازای این تنظیم مقادیر برای پارامترهای الگوریتم وراثتی به دست آمدند: تعداد اعضای جمعیت: ۲۰، احتمال تقاطع: ۰/۹، احتمال وقوع جهش: ۰/۳ و نرخ جهش: ۰/۶.

جدول (۲): مقادیر خطای آموزش RBFهای مجزا ـ آموزش توسط الگوریتم PSO

مدل	MSE	RMSE	MAE	SSE
RBF 1	۰/۰۰۷۵۳	۰/۰۸۶۷۷	۰/۰۰۳۷۶	۱۹۶
RBF 2	۰/۵۱۶۹۲	۰/۷۱۸۹۷	۰/۲۵۸۴۶	۱۲۶۳۶
RBF 3	۰/۱۵۳۵۹	۰/۳۹۱۹۰	۰/۰۷۶۸۰	۸۲۸
RBF 4	۰/۱۹۲۶۲	۰/۴۳۸۸۸	۰/۰۹۶۳۱	۴۲۲۸
RBF 5	۰/۰۹۳۹۲	۰/۳۰۶۴۶	۰/۰۴۶۹۶	۷۶۰
RBF 6	۰/۰۰۰۲۲	۰/۰۱۴۹۹	۰/۰۰۰۱۱	۴
RBF 7	۰/۱۲۶۹۰	۰/۳۵۶۲۲	۰/۰۶۳۴۵	۲۷۴۴
RBF 8	۰/۲۸۵۷۱	۰/۵۳۴۵۲	۰/۱۴۲۸۶	۱۳۶
RBF 9	۰/۰۱۱۸۲	۰/۱۰۸۷۲	۰/۰۰۵۹۱	۶۸۰
RBF 10	۰/۴۵۵۳۵	۰/۶۷۴۸۰	۰/۲۲۷۶۷	۹۲۸
RBF 11	۰/۰۳۷۲۴	۰/۱۹۲۹۹	۰/۰۱۸۶۲	۲۰
RBF 12	۰/۲۱۰۰۱	۰/۴۵۸۲۷	۰/۱۰۵۰۱	۷۶

جدول (۳): مقادیر خطای آموزش RBFهای مجزا - آموزش توسط GA

مدل	MSE	RMSE	MAE	SSE
RBF 1	۰/۰۱۰۷۱	۰/۱۰۳۵۰	۰/۰۰۵۳۶	۱۴۰
RBF 2	۰/۲۱۷۴۵	۰/۴۶۶۳۲	۰/۱۰۸۷۳	۲۶۶۴
RBF 3	۰/۱۳۲۸۹	۰/۳۶۴۵۴	۰/۰۶۶۴۵	۳۶۰
RBF 4	۰/۲۱۷۱۹	۰/۴۶۶۰۴	۰/۱۰۸۵۹	۲۳۸۸
RBF 5	۰/۰۹۲۹۳	۰/۳۰۴۸۵	۰/۰۴۶۴۷	۳۷۶
RBF 6	۰/۰۰۰۴۵	۰/۰۲۱۲۳	۰/۰۰۰۲۳	۴
RBF 7	۰/۱۱۸۶۵	۰/۳۴۴۴۶	۰/۳۴۴۴۶	۱۲۷۶
RBF 8	۰/۲۰۶۳۵	۰/۴۵۴۲۶	۰/۱۰۳۱۷	۵۲
RBF 9	۰/۰۱۱۹۶	۰/۱۰۹۳۸	۰/۰۰۵۹۸	۳۴۴
RBF 10	۰/۴۶۷۸۴	۰/۶۸۳۹۹	۰/۲۳۳۹۲	۴۸۰
RBF 11	۰/۲۰۷۴۱	۰/۴۵۵۴۲	۰/۱۰۳۷۰	۵۶
RBF 12	۰/۲۱۴۴۸	۰/۴۶۳۱۲	۰/۱۰۷۲۴	۴۰۰

همچنین زمان لازم برای آموزش مدل‌های مورد شبیه‌سازی در جدول (۴) آورده شده است. اختلاف زمان آموزش RBF با PSO و GA به‌دلیل تفاوت تعداد اعضای جمعیت PSO و GA است. ازآنجاکه در الگوریتم‌های بهینه‌سازی مبتنی بر جمعیت، افزایش تعداد اعضای جمعیت منجر به افزایش زمان آموزش به‌دلیل تکرارهای بیشتر برای اعضای جمعیت بیشتر می‌شود، هرکدام از مدل‌ها با حداقل اعضای جمعیت‌شان که منجر به میزان خطای کمینه شود، آموزش داده شده‌اند. همان‌طور که مشاهده می‌شود، به‌کارگیری SVM در قیاس با به‌کارگیری RBF باعث کاهش زمان آموزش در مرحلهٔ شناسایی ناهنجاری می‌شود، چون آموزش RBF با الگوریتم‌های بهینه‌یابی هوشمند مذکور به‌منظور یافتن مقادیر بهینهٔ وزن‌های شبکه صورت گرفته است و تکرارهای انجام‌شده برای یافتن این مقادیر بهینه بخش قابل‌توجهی از زمان را به‌خود اختصاص می‌دهد. اما ازآنجاکه فرایند آموزش برون‌خط است، طولانی‌تر شدن آن در شرایطی که عملکرد اجرای برخط مدل بهبود یابد، قابل‌قبول است.

جدول (۴): زمان آموزش مدل‌ها

مدل پیشنهادی	زمان کل (sec)	زمان آموزش مدل شناسایی سوء استفاده (sec)	زمان لازم برای فراهم کردن داده برای مدل شناسایی ناهنجاری (sec)	زمان آموزش مدل شناسایی ناهنجاری (sec)
C4.5+ SVM	۴۷/۴۶۷۴۱	۰/۱۸۴۸۲	۱۷/۷۹۹۴۵	۲۹/۴۸۳۱۴
C4.5+ RBF-PSO	۹۶/۷۵۱۸۵	۰/۱۸۴۸۲	۱۷/۷۹۹۴۵	۷۸/۷۶۷۵۸
C4.5+ RBF-GA	۷۷/۲۰۳۶۰	۰/۱۸۴۸۲	۱۷/۷۹۹۴۵	۵۹/۲۱۹۳۳

۶.۴. نتایج حاصل از آزمایش مدل‌ها با داده‌های آزمون

نتایج تجربی نمایانگر این موضوع‌اند که به‌کارگیری SVM در فاز شناسایی ناهنجاری با حجم بالای داده‌های ورودی به مدل، باعث عملکرد ضعیف مدل و در نهایت به ناکارآمدی مدل در استفاده‌های برخط منجر می‌شود (جدول ۵). همان‌گونه که مشاهده می‌شود، از لحاظ زمان اجرای برخط مدل، بهترین گزینه به‌کارگیری ترکیبی درخت تصمیم‌گیری و شبکهٔ عصبی RBF (با آموزش توسط الگوریتم PSO) است. از لحاظ میزان معیارهای مختلف خطا نیز گزینهٔ مذکور مقادیر قابل رقابت و بسیار نزدیک به مدل ترکیبی درخت تصمیم‌گیری و شبکهٔ عصبی RBF (با آموزش توسط GA) ارائه می‌کند (جدول ۶).

جدول (۵): زمان آزمایش مدل‌ها با داده‌های آزمون

مدل پیشنهادی	زمان کل (sec)
C4.5+SVM	۷۷/۲۱۶۳۲۶
C4.5+RBF-PSO	۲/۷۰۱۰۵۷
C4.5+RBF-GA	۳/۲۲۹۶۹۷

جدول (۶): خطای آزمایش مدل‌های پیشنهادی با داده‌های آزمون

مدل پیشنهادی	MSE	RMSE	MAE	SSE
C4.5+SVM	۰/۳۴۹۴	۰/۵۹۱۱۰	۰/۱۷۴۷۰	۲۹۸۲۰
C4.5+RBF-PSO	۰/۳۳۳۷	۰/۵۷۷۶۶	۰/۱۶۶۸۵	۲۸۴۸۰
C4.5+RBF-GA	۰/۳۳۲۹	۰/۵۷۶۹۷	۰/۱۶۶۴۵	۲۸۴۱۲

۵. نتیجه‌گیری

در این مقاله، راهکاری آمیختار برای تشخیص نفوذ در شبکه‌ها ارائه شد. در این بخش به مقایسهٔ عملکرد سامانهٔ آمیختار پیشنهادی، که از چندین الگوریتم و مدل در بخش‌های مختلف خود بهره می‌جست، با سامانه‌های مشابهی که از الگوریتم‌های انتخاب ویژگی و نیز ابزار دسته‌بندی از انواع مدل‌ها مانند درخت‌های تصمیم‌گیری، شبکه‌های عصبی مصنوعی، SVM و Naïve Bayes برای شناسایی نفوذ بهره برده‌اند، پرداخته می‌شود. نتایج ارائه‌شده حاصل از آزمون سامانه با دادگان KDD'99 یا NSL-KDD است (جدول ۷).

یادآوری می‌شود که دادگان NSL-KDD نسخهٔ فشرده‌ای از دادگان اصلی KDD'99 با همان تعداد ویژگی است. در جریان این فشرده‌سازی، رکوردهای افزونه در مجموعهٔ آموزشی حذف شده؛ لذا دسته‌بندی‌کننده‌ها، گرایشی به داده‌های با تکرار بیشتر ندارند. البته اغلب نرخ‌های آشکارسازی گزارش‌شده روی KDD'99 بهتر از نرخ‌های گزارش‌شده روی NSL-KDD است. برای نمونه در مرجع [۵۲]، یک سامانهٔ تشخیص ناهنجاری به‌کمک SOM پیاده‌سازی شده است که نرخ آشکارسازی روی دادگان KDD'99 و NSL-KDD در آن به‌ترتیب ۹۲/۳۷ و ۷۵/۴۹ درصد است.

همان‌گونه که در جدول (۷) مشاهده می‌شود، سامانهٔ آمیختار پیشنهادی در این مقاله، بهترین نرخ آشکارسازی را در مقایسه با سایر مدل‌های آزمایش‌شده روی دادگان NSL-KDD در سالیان اخیر ارائه می‌دهد. این در حالی است که تعداد ویژگی‌های به‌کاررفته در مدل‌های پیشنهادی نیز کمتر از تعداد ویژگی‌های مورد استفاده در سایر مدل‌هاست. لذا می‌توان نتیجه گرفت که سامانهٔ پیشنهادی ترکیب مناسبی از روش‌های انتخاب ویژگی و دسته‌بندی را ارائه کرده و نتایج عملکرد آن در تراز سامانه‌های موفقی است که روی دادگان KDD'99 و NSL-KDD آزمون شده‌اند.

علاوه‌براین، نتایج تجربی نشان دادند که گرچه ایدهٔ به‌کارگیری SVMهای موازی که ورودی‌های آن‌ها توسط برگ‌های درخت از داده‌هایی با حجم کمتر و منسجم‌ترند، منجر به کاهش زمان آموزش مدل می‌شود، زمان اجرای مدل در قیاس با به‌کارگیری

RBF به‌جای SVM به‌شکل قابل‌توجهی بالاست (حدود ۲۸ برابر زمان اجرای مدل با به‌کارگیری RBF) و همین امر منجر به ناکارآمدی مدلی که از SVM استفاده می‌کند، به‌ویژه در شبکه‌های با سرعت بالا می‌شود؛ در شرایطی که خطاهای اجرا و نرخ هشدارهای اشتباه سیستم در هر دو مدل مشابه‌اند. با توجه به اینکه آموزش مدل، فرایندی برون‌خط و آزمون و اجرای آن فرایندی برخط است، می‌توان نتیجه گرفت که کاربرد RBF به‌جای SVM موجب بهبودی چشمگیر در کارایی مدل ترکیبی شده است.

برای آن دسته افراد که مشتاق تحقیق بیشتر و کارهای نو در این زمینه هستند، پیشنهاد می‌شود که به‌منظور کاهش هشدارهای اشتباه در مرحلهٔ شناسایی ناهنجاری، به‌دنبال راهکاری برای کاهش نرخ هشدارهای منفی اشتباه مرحلهٔ شناسایی سوءاستفاده باشند؛ زیرا خروجی‌های مرحلهٔ شناسایی سوءاستفاده که تحت عنوان نرمال دسته‌بندی شدند، به‌عنوان ورودی مرحلهٔ شناسایی ناهنجاری به کار می‌روند و مدل شناسایی ناهنجاری برای آموزش بهتر و تشخیص تخلف از الگوی نرمال نیاز به داده‌های نرمال به‌عنوان ورودی خود دارد و با کاهش نرخ منفی اشتباه در مرحلهٔ شناسایی سوءاستفاده، می‌توان به این مهم دست یافت.

جدول (۷): مقایسهٔ نرخ آشکارسازی سامانهٔ پیشنهادی با سامانه‌های مشابه

نام اختصاری مدل آمیختار	الگوریتم انتخاب ویژگی	ابزار دسته‌بندی	نام دادگان	تعداد ویژگی‌های انتخابی	نرخ آشکارسازی (٪)
FGBARM[1]+GSA[2]-Fuzzy ARTMAP [۴۵]	FGBARM	GSA-Fuzzy ARTMAP	KDD'99	۲۶	۹۳/۷۴
FGBARM+PSO-Fuzzy ARTMAP [۴۶]	FGBARM	PSO-Fuzzy ARTMAP	KDD'99	۲۹	۹۷/۲۵
FGBARM+GA-Fuzzy ARTMAP [۴۶]	FGBARM	GA-Fuzzy ARTMAP	KDD'99	۲۹	۹۷/۹۰
DBN[3]+SVM [۴۷]	DBN	SVM	NSL-KDD	۱۴	۸۳/۱۴
RST[4]+SVM [۴۸]	RST	SVM	KDD'99	۲۹	۸۹/۳۶
Info-Gain+J48 [۴۹]	Info-Gain	درخت تصمیم‌گیری J48	NSL-KDD	۳۳	۸۱/۹۴
Info-Gain+Naïve Bayes [۴۹]	Info-Gain	Naïve Bayes	NSL-KDD	۳۳	۷۵/۷۹
Info-Gain+MLP[5] [۴۹]	Info-Gain	MLP	NSL-KDD	۳۳	۷۳/۵۵
Info-Gain+SVM [۴۹]	Info-Gain	SVM	NSL-KDD	۳۳	۷۱/۰۲
Info-Gain+CART[6] [۴۹]	Info-Gain	CART	NSL-KDD	۳۳	۸۲/۳۲
Ensemble+Bayesian [۵۰]	اعمال ۹ ویژگی پایه، ۱۳ ویژگی محتوایی و ۱۹ ویژگی ترافیکی به سه دسته‌بند مجزا	گروه دسته‌بندهای k-NN فازی، MLP و Naïve Bayse و روش ادغام بیزی	KDD'99	مجموع ۴۱ ویژگی در سه دسته‌بند استفاده شده‌اند	۹۳/۳۵
Ensemble(RBF+SVM) [۵۱]	Best First Search	RBF+SVM	NSL-KDD	۹	۸۵/۱۹
SFLO+C4.5-SVM (مدل پیشنهادی)	SFLO	C4.5-SVM	NSL-KDD	۱۰	۹۷/۴
SFLO+C4.5-PSO-RBF (مدل پیشنهادی)	SFLO	C4.5-PSO-RBF	NSL-KDD	۱۰	۹۶/۹
SFLO+C4.5-GA-RBF (مدل پیشنهادی)	SFLO	C4.5-GA-RBF	NSL-KDD	۱۰	۹۶/۹

1. Fuzzy Grid-Based Association Rule Mining
2. Gravitational Search Algorithm
3. Deep Belief Network
4. Rough Set Theory
5. Multi-Layer Perceptron
6. Classification And Regression Tree

مراجع

[۱] زندی م.، فتحی م.، شرایعی ا.، «**بررسی کاربردهای داده‌کاوی در زمینه تشخیص نفوذ**»، پنجمین کنفرانس ملی فرمانـدهی و کنترل ایران، دانشگاه تهران، آذر ماه ۱۳۹۰.

[2] Gowrison G., Ramar K., Muneeswaran K., Revathi T., "*Minimal complexity attack classification intrusion detection system*", Applied Soft Computing, Vol. 13, No. 2, pp. 921-927, 2013.

[3] Aydin M., Zaim A., Ceylan K., "*A hybrid intrusion detection system design for computer network security*", Computers and Electrical Engineering, Vol. 35, No. 3, pp. 517-526, 2009.

[4] *Computer Crime and Security Survey* (CSI/FBI 2010), 15th Annual, 2010-2011, www.GoCSI.com.

[5]*US cybercrime: Rising risks, reduced readiness (Key findings from the 2014 US State of Cybercrime Survey)*, 2014, www.pwc.com/cybersecurity.

[6] Sheikhan M., "*Artificial neural network models for intrusion detection*", In: Encyclopedia of Information Assurance, Taylor & Francis, New York, 2014 (DOI: 10.1081/E-EIA-120051983).

[7] Sheikhan M., "*Fuzzy models for intrusion detection*", In: Encyclopedia of Information Assurance, Taylor & Francis, New York, 2015 (DOI: 10.1081/E-EIA-120051982).

[8] Zhang J., Zulkernine M., "*A hybrid network intrusion detection technique using random forest*", In: Proceedings of the 1st International Conference on Availability, Reliability and Security, pp. 262-269, 2006.

[9] Kim G., Lee S., Kim S., "*A novel hybrid intrusion detection method integrating anomaly detection with misuse detection*", Expert System with Applications, Vol. 41, No. 4, pp. 1690-1700, 2014.

[10] Sheikhan M., Abbasnezhad Arabi M., Gharavian D., "*Structure and weights optimisation of a modified Elman network emotion classifier using hybrid computational intelligence algorithms: A comparative study*", Connection Science, Vol. 27, No. 4, pp. 340-357, 2015.

[11] Hwang T.S., Lee T.-J., Lee Y.-J., "*A three-tier IDS via data mining approach*", In: Proceedings of ACM Conference MineNet, pp. 713-722, 2007.

[12] Barbara D., Couto J., Jajodia S., Popyack L., Wu N., "*ADAM: Detecting intrusions by data mining*", In: Proceedings of the IEEE Workshop on Information Assurance and Security, pp. 11-16, 2001.

[13] Depren O., Topallar M., Anarim E., Ciliz M.K., "*An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks*", Expert Systems with Applications, Vol. 29, No. 4, pp. 713-722, 2005.

[14] Hwang K., Cai M., Chen Y., Qin M., *"Hybrid intrusion detection with weighted signature generation over anomalous internet episodes*", IEEE Transactions on Dependable and Secure Computing, Vol. 4, No. 1, pp. 41-55, 2007.

[15] Zhang Y., Wang S., Phillips P., Ji G., "*Binary PSO with mutation operator for feature selection using decision tree applied to spam detection*", Knowledge-Based Systems, Vol. 64, pp. 22-31, Jul. 2014.

[16] Sabry Eesa A., Orman Z., Abdulazeez Brifcani A.M., "*A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems*", Expert Systems with Applications, Vol. 42, No. 5, pp. 2670-2679, 2015.

[17] Pirgazi J., Khanteymoori A.R., "*SFLA based gene selection approach for improving cancer classification accuracy*", Amirkabir International Journal of Science & Research (Modeling, Identification, Simulation & Control), Vol. 47, No. 1, pp. 1-8, 2015.

[18] Dai Y., Hu B., Su Y., Mao C., Chen J., Zhang X., Moore P., Xu L., Cai H., "*Feature selection of high-dimensional biomedical data using improved SFLA for disease diagnosis*", In: Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine, pp. 458-463, 2015.

[19] Kuncheva L.I., *Combining Pattern Classifiers: Methods and Algorithms*, John Wiley & Sons, 2nd Edition, 2014.

[20] Giacinto G., Roli F., Didaci L., "*Fusion of multiple classifiers for intrusion detection in computer networks*", Pattern Recognition Letters, Vol. 24, No. 12, pp. 1795-1803, 2003.

[21] Giacinto G., Perdisci R., Del Rio M., Roli F., "*Intrusion detection in computer networks by a modular ensemble of one-class classifiers*", Information Fusion, Vol. 9, No. 1, pp. 69-82, 2008.

[22] Folino G., Sabatino P., "*Ensemble based collaborative and distributed intrusion detection systems: A survey*", Journal of Network and Computer Applications, Vol. 66, pp. 1-16, May 2016.

[23] Aburomman A.A., Ibne Reaz M.B., "*A novel SVM-kNN-PSO ensemble method for intrusion detection system*", Applied Soft Computing, Vol. 38, pp. 360-372, Jan. 2016.

[24] Elbeltagi E., Hegazy T., Grierson D., "*A modified shuffled frog-leaping optimization algorithm: Application to project management*", Structure and Infrastructure Engineering, Vol. 3, No. 1, pp. 53-60, 2007.

[۲۵] اعیانزاده ر.، تشنه‌لب م.، «**تکامل تدریجی رفتار در الگوریتم‌های مِمِتیک با استفاده از سازگاری در تقلید افراد**»، هشتمین کنفرانس سیستم‌های هوشمند، دانشگاه فردوسی مشهد، شهریور ۱۳۸۶.

[۲۶] صنیعی آباده م.، محمودی س.، طاهرپرور م.، **داده‌کاوی کاربردی**، چاپ اول، انتشارات نیاز دانش، ۱۳۹۱.

[27] Tong X., Wang Z., Yu H., "*A research using hybrid RBF/Elman neural networks for intrusion detection system secure model*", Computer Physics Communications, Vol. 180, No. 10, pp. 1795-1801, 2009.

[28] Shao Y., Chen Q., Jiang H., "*RBF neural network based on particle swarm optimization*", Lecture Notes in Computer Science, Vol. 6063, pp. 169-176, 2010.

[29] Bai Q., "*Analysis of particle swarm optimization algorithm*", Computer and Information Science, Vol. 3, No. 1, pp. 180-184, 2010.

[30] Eberhart R.C., Kennedy J., "*A new optimizer using particle swarm theorem*", In: Proceedings of the 6th International Symposium on Micro Machine and Human Science, pp. 39-43, 1995.

[31] Mitchell M., *Genetic Algorithms: An Overview*, Complexity, Vol. 1, No. 1, pp. 31-39, 1995.

[32] Alberts A., Rvira N., Aguayo H., Maier T., "*Optimization with genetic algorithms and splines as a way for computer aided innovation*", IFIP for Information Processing, Vol. 277, pp. 7-18, 2008.

[33] Xiong K., Jiang D.X., Ding Y.S., Li K., "*Comparison of RBF neural network and support vector machine on aero-engine vibration fault diagnosis*", Key Engineering Materials, Vol. 347, pp. 323-328, Sep. 2007.

[34] Raj Kumar P.A., Selvakumar S., "*Distributed denial of service attack detection using an ensemble of neural classifier*", Computer Communications, Vol. 34, No. 11, pp. 1328-1341, 2011.

[35] Raj Kumar P.A., Selvakumar S., "*Detection of distributed denial of service attacks using an ensemble of adaptive and hybrid neuro-fuzzy systems*", Computer Communications, Vol. 36, No. 3, pp. 303-319, 2013.

[36] Ippoliti D., Zhou X., "*A-GHSOM: An adaptive growing hierarchical self organizing map for network anomaly detection*", Journal of Parallel and Distributed Computing, Vol. 72, No. 12, pp. 1576-1590, 2012.

[37] Wang W., Zhang X., Gombault S., Knapskog S.J., "*Attribute normalization in network intrusion detection*", In: Proceeding of 10th International Symposium on Pervasive Systems, Algorithms, and Networks, Kaohsiung, Taiwan, 2009.

[38] Chang F., Guo C.Y., Lin X.R., Lu C.J., "*Tree decomposition for large-scale SVM problems*", Journal of Machine Learning Research, Vol. 11, No. 1, pp. 2935-2972, 2010.

[39] Tavallaee M., Bagheri E., Wei L., Ghorbani A., NSL-KDD Data Set (Available on http://nsl.cs.unb.ca/NSL-KDD), [Accessed on 20 Feb. 2015].

[40] Davis J.J., Clark A.J., "*Data preprocessing for anomaly based network intrusion detection: A review*", Computers & Security, Vol. 30, No. 6-7, pp. 353-375, 2011.

[41] Ikram S.T., Cherukuri A.K., "*Intrusion detection model using fusion of chi-square feature selection and multi class SVM*", Journal of King Saud University-Computer and Information Sciences, {Article in Press}, 2016 (DOI: 10.1016/j.jksuci.2015.12.004).

[42] Bostani H., Sheikhan M., "*Modification of supervised OPF-based intrusion detection systems using unsupervised learning and social network concept*", Pattern Recognition, Vol. 62, pp. 56-72, Feb. 2017.

[43] Chandrashekar G., Sahin F., "*A survey on feature selection methods*", Computers and Electrical Engineering, Vol. 40, No. 1, pp. 16-28, 2014.

[44] Negnevitsky M., *Artificial Intelligence, A Guide to Intelligent Systems*, Addison Wesley, 2nd Edition, 2005.

[45] Sheikhan M., Sharifi Rad M., "*Gravitational search algorithm-optimized neural misuse detector with selected features by fuzzy grids based association rules mining*", Neural Computing and Applications, Vol. 23, No. 7, pp. 2451-2463, 2013.

[46] Sheikhan M., Sharifi Rad M., "*Using particle swarm optimization in fuzzy association rules-based feature selection and fuzzy ARTMAP-based attack recognition*", Security and Communication Networks, Vol. 6, No. 7, pp. 797-811, 2013.

[۴۷] رشیدی س.، **سیستم نیمه‌نظارتی تشخیص ناهنجاری در**

شبکه‌های ارتباطی مبتنی بر ماشین بردار پشتیبان و یادگیری عمیق، پایان‌نامهٔ کارشناسی‌ارشد مهندسی مخابرات، دانشگاه آزاد اسلامی – واحد تهران جنوب، ۱۳۹۴.

[48] Shrivastava S.K., Jain P., "*Effective anomaly based intrusion detection using rough set theory and support vector machine*", International Journal of Computer Applications, Vol. 18, No. 3, pp. 35-41, 2011.

[49] Bajaj K., Arora A., "*Improving the intrusion detection using discriminative machine learning approach and improve the time complexity by data mining feature selection methods*", International Journal of Computer Applications, Vol. 67, No. 1, pp. 5-11, 2013.

[50] Chou T.S., Fan J., Fan S., Makki K., "*Ensemble of machine learning algorithms for intrusion detection*", In: Proceedings of the IEEE International Conference on System, Man and Cybernetics, pp. 3976-3980, 2009.

[51] Govindarajan M., Chandrasekaran R.M., "*Intrusion detection using an ensemble of classification methods*", In: Proceedings of the World Congress on Engineering and Computer Science, Vol. 1, pp. 1-6, 2012.

[52] Ibrahim L.M., Basheer D.T., Mahmod M.S., "*A comparison study for intrusion database (KDD99, NSL-KDD) based on self organizing map (SOM) artificial neural network*", Journal of Engineering Science and Technology, Vol. 8, No. 1, pp. 107-119, 2013.