بهبود سیستم های تشخیص نفوذ باکاهش ویژگی مبتنی بر الگوریتم ژنتیک و تکنیک‌های داده‌کاوی

مهدی کشاورزی[۱*]، حسین مومن زاده حقیقی[۲]

۱- دانشجو کارشناسی ارشد mehdikeshavarzi2000@gmail.com

۲- عضو هیئت علمی momenzadeh.hossein@gmail.com

چکیده

امروزه سیستم های کامپیوتری مبتنی بر شبکه، نقش حیاتی در جامعه مدرن امروزی دارند و به همین علت ممکن است هدف دشمنی و یا نفوذ قرار گیرند. به منظور ایجاد امنیت کامل در یک سیستم کامپیوتری متصل به شبکه، استفاده از دیوار آتش و سایر مکانیزم های جلوگیری از نفوذ همیشه کافی نیست و این نیاز احساس می شود تا از سیستم های دیگری به نام سیستم های تشخیص نفوذ استفاده شود. سیستم تشخیص نفوذ را می توان مجموعه ای از ابزارها، روش ها و مدارکی در نظر گرفت که به شناسایی، تعیین و گزارش فعالیت های غیرمجاز یا تائید نشده تحت شبکه، کمک میکند. سیستم های تشخیص نفوذ به صورت سیستم های نرم افزاری و سخت افزاری ایجاد شده و هر کدام مزایا و معایب خاص خود را دارند. به دلیل وجود مشخصه های زیاد در داده های مربوط به سیستم های تشخیص نفوذ در این تحقیق ما مشخصه های مطلوب و موثر را با استفاده از الگوریتم ژنتیک بهبود یافته انتخاب می کنیم. سپس با استفاده از تکنیک های داده کاوی استاندارد، مدلی برای طبقه بندی داده ها ارائه می دهیم. برای ارزیابی عملکرد روش پیشنهادی از پایگاه داده ***NSL-KDD*** که نسبت به سایر داده های تشخیص نفوذ از رکوردهای واقعی تری برخورد دار است، استفاده خواهیم کرد.

واژه های کلیدی: سیستم تشخیص نفوذ، داده کاوی، انتخاب ویژگی، الگوریتم ژنتیک، پایگاه داده ***NSL-KDD***

۱- مقدمه

نفوذ به مجموعه اقدامات غیرقانونی که محرمانگی و یا دسترسی به یک منبع را به خطر می اندازد ***[۱]***، اطلاق می گردد. سیستم تشخیص نفوذ را می توان مجموعه ای از ابزارها، روش ها و مدارکی در نظر گرفت که به شناسایی، تعیین و گزارش فعالیت های غیرمجاز یا تائید نشده تحت شبکه، کمک میکند. سیستم های تشخیص نفوذ به صورت سیستم های نرم افزاری و سخت افزاری ایجاد شده و هر کدام مزایا و معایب خاص خود را دارند. سرعت و دقت از مزایای سیستم های سخت افزاری است و عدم شکست امنیتی آن ها توسط نفوذگران، قابلیت دیگر این گونه سیستم ها می باشد. اما استفاده آسان از نرم افزار، قابلیت انطباق پذیری در شرایط نرم افزاری و تفاوت سیستم های عامل مختلف، عمومیت بیشتری را به سیستم های نرم افزاری می دهد و عموماً این گونه سیستم ها انتخاب مناسب تری هستند. افزایش دسترسی به داده ها و پردازش سریعتر آنها و در عین حال افزایش حجم داده ها و نیاز به فراهم آوردن داده ها از منابع مختلف از طریق شبکه های کامپیوتری، منجر به پدید آمدن منابع تهدید آمیزی می گردد که از طریق نقاط ضعف موجود در سیستم ها، به استثمار سیستم ها و ایجاد اختلال در آن ها می پردازد. تشخیص و جلوگیری از نفوذ (***IDS***) امروزه به عنوان یکی از مکانیزم های اصلی در برآوردن امنیت شبکه ها و سیستم های کامپیوتری مطرح است. سیستم‌های تشخیص نفوذ

متعددی برای کشف حملات وجود دارند که چالش اصلی آنها بالا بردن کارایی می‌باشد ***[۲]***. بیشتر سیستم های تشخیص نفوذ کنونی از تمامی پارامترهای موجود در بسته‌های شبکه برای ارزیابی و کشف الگوهای حملات استفاده می‌نمایند، در صورتی که برخی از این پارامترها غیرمرتبط و زائد می‌باشند. استفاده از تمامی پارامترها باعث می شود که فرآیند تشخیص طولانی و کارایی سیستم تشخیص نفوذ تنزل یابد. در واقع چالش اساسی در سیستم تشخیص نفوذ حجم عظیم داده‌هاست ***[۳]***. از طرفی با توجه به ترافیک بالا کاهش نرخ هشدار غلط در سیستم تشخیص نفوذ نیز از اهمیت خاصی برخوردار است. تمام سیستم‌های تشخیص نفوذ قادر به تولید هشدار در مورد وقوع نفوذ در شبکه می‌باشند. ولی به علت حجم بالای هشدارهای تولید شده توسط این سیستم ها و همچنین تولید هشدارهای اشتباه، این سیستم‌ها قادر به مدیریت و آنالیز هشدارهای تولید شده نمی باشند. امروزه بیشتر رویکردها در تشخیص نفوذ مربوط به مساله انتخاب و یا استخراج ویژگی‌های مهم متمرکز شده است. تعداد ویژگی‌هایی که بهترین دقت را در تشخیص نفوذ دارند، در مجموعه داده‌های مختلف بیشتر به صورت تجربی بدست می‌آید. اما انتخاب ویژگی‌ها ممکن است باعث از دست دادن قسمتی از داده‌ها شود.

بنابراین با توجه به تفاوت داده‌های مربوط به شبکه، تخمین تعداد بهینه ویژگی‌ها یکی از چالش‌های مهم می‌باشد. در این تحقیق سعی می‌شود ویژگی‌هایی را به عنوان صفات اساسی انتخاب کنیم که صحت تشخیص نفوذ را بالا ببرد و ثانیا تاثیر ویژگی‌هایی که انتخاب نشده‌اند را نیز در سیستم تشخیص نفوذ لحاظ نماییم. ما با استفاده از الگوریتم‌های ژنتیک و تکنیک‌های داده کاوی مدلی برای تشخیص نفوذ به شبکه ارائه دهیم.

۲- مروری بر کارهای انجام شده

بعد از سال ۱۹۷۰ و با افزایش تعداد کامپیوترها نیاز به امنیت کامپیوتری بیش از پیش آشکار شد. در همان زمان وزارت دفاع آمریکا، متوجه افزایش استفاده از کامپیوترها در سیستم های نظامی و در نتیجه آن اهمیت امنیت این سیستم ها شد. دنینگ و نیومن تحلیل و طراحی سیستم ***IDES*** که یک سیستم تشخیص نفوذ بلادرنگ و خبره بود را در سال ۱۹۸۵ به انجام رسانیدند ***[٤]***. تحقیق انجام شده به عنوان پایه ای برای بسیاری از تحقیقات در زمینه تشخیص نفوذ در آن دهه قرار گرفت. تحقیقات بسیاری در زمینه کاهش ویژگی های سیستم های تشخیص نفوذ انجام شده است، که از بین آنها، برخی روش ها، مبتنی بر هوش مصنوعی و برخی براساس روش های داده کاوی می باشند. خدایار و همکارانش یک روش ترکیبی یادگیری ماشین به منظور تشخیص نفوذ در شبکه ارائه دادند ***[۵]***. روش ترکیبی ارائه شده در این تحقیق مبتنی بر مفهوم کاهش ابعاد و الگوریتم های درخت تصمیم و روش های ترکیبی بوده و از دقت ۹۱/۷۹ درصدی برخوردار است. در تحقیقی دیگر در سال ۲۰۰۸ یک روش مبتنی بر الگوریتم ژنتیک برای تشخیص نفوذ روی پایگاه داده ***KDDCUP99*** ارائه شد ***[٦]***. این روش از یک رویکرد یادگیری ماشین به همراه الگوریتم ژنتیک برای استخراج ویژگی استفاده می کند. با تولید یک سری قوانین و بر اساس هر قاعده یک نوع خاص نفوذ شناسایی می شود. رویکرد یادگیری ترکیبی خوشه بندی ***K-means*** و طبقه بندی بیز ساده در ***[۷]*** ارائه شد. خوشه بندی تمام داده ها را به گروه های مربوطه قبل از استفاده طبقه بندی نسبت می دهد. در این پژوهش از داده های ***KDDCUP99*** برای آموزش استفاده شده است و نتایج عملکرد بهتری از جهت دقت و سرعت کشف نفوذ با نرخ هشدار غلط مناسب را نشان می دهد. سریپارنا ساها و همکارانش یک سیستم تشخیص نفوذ مبتنی بر یادگیری ماشین توسعه دادند ***[۸]***. آنها الگوریتم ژنتیک را به همراه ***SVM*** برای تعیین اتوماتیک مجموعه مناسب از ویژگی ها به کار بردند. این مسئله از کاربرد الگوریتم ژنتیک برای انتخاب ویژگی و کلاس بندی مبتنی بر ***SVM*** حمایت می کند. در این تحقیق یک دیکشنری از پیش تعریف شده از انواع حملات وجود دارد کهاین روش متمایزترین ویژگی ها برای هر نوع حمله را در دسترس قرار می دهد. در ***[۹]*** با ترکیب ویژگی های نوع ***filter*** و ***wrapper***، یک نوع روش ترکیبی برای انتخاب ویژگی با استفاده از یک الگوریتم ژنتیک توسعه یافته شامل مکانیزم تنبیه و پاداش ارائه شد. بنایچا و همکارانش نیز سیستم تشخیص نفوذی با استفاده از

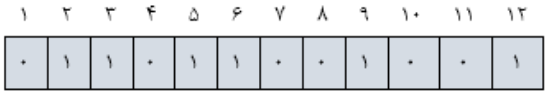
الگوریتم ژنتیک توسعه دادند که در آن ایده ای برای بهبود بخش های ایجاد جمعیت اولیه و اپراتور انتخاب در الگوریتم ژنتیک ارائه شده است [۱۰]. یک روش جدید برای انتخاب ویژگی های موثر در ساخت مدل تشخیص نفوذ در تحقیق [۱۱] ارائه شد. روش انتخاب ویژگی پیشنهادی با توجه به میانگین ویژگی ها از هر کلاس نسبت به کلاس کل محاسبه می شود. در این تحقیق ار روش کلاس بندی درخت تصمیم استفاده می شود. در تحقیقی دیگر از الگوریتم کلونی مورچه ها، به منظور استخراج ویژگی ها در سیستم های تشخیص نفوذ استفاده کرده است [۱۲]. این روش بدلیل استفاده از زیر مجموعه های ساده برای کلاس بندی قابلیت اجرای سریع و پیچیدگی محاسباتی کمی می باشد. کاهش تعداد ویژگی ها با استفاده از نمایش گراف و اطلاعات اکتشافی برای به روز رسانی فرمون ها باعث ارائه دقت بالاتر در تشخیص نفوذ و هشدار اشتباه پایین تر می شود. به منظور مقابله با نفوذکنندگان به سیستم‌ها و شبکه‌های کامپیوتری توسط هر دو دسته کاربران داخلی و حمله‌کنندگان خارجی، ما در این تحقیق به بررسی رویکردی نوینی مبتنی بر الگوریتم ژنتیک در تشخیص و ممانعت از نفوذ و آینده این تکنولوژی می پردازیم.

۳- سیستم تشخیص نفوذ پیشنهادی

با توجه به اینکه در پردازش داده ها با اعداد سروکار دارم و برخی از مشخصه های این مجموعه داده دارای مقادیر رشته ای هستند، لذا تغییر آنها به مقادیر عددی برای انجام پردازش الزامی می باشد. برای تبدیل نوع های رشته ای به عدد نیز به این صورت عمل می کنیم که به ازای هر نوع یک عدد منحصر به فرد در نظر می گیریم.گام اول در ایجاد هر مدلی مبتنی بر تکنیک های داده کاوی، مرحله پیش پردازش داده ها می باشد. پیش پردازش جهت آماده سازی داده ها جهت پردازش و همچنین بهبود کیفیت داده های واقعی انجام می شود. این مرحله شامل نرمال سازی و در هم ریختن داده ها می باشد. در مرحله بعد با استفاده از الگوریتم های ژنتیک موازی ویژگی های مطلوب استخراج می شود. در برخی از مجموعه داده ها تعداد ویژگی ها زیاد می باشند و ممکن است برخی از این ویژگی ها در طبقه بندی داده ها نقشی نداشته باشند. بنابراین نیاز است که زیر مجموعه ای از بهترین ویژگی ها انتخاب شوند. روشی که برای انتخاب ویژگی ها در این پژوهش در نظر گرفته شده، براساس اندازه گیری توزیع ویژگی ها عمل می کند. به این صورت که در تکرارهای متوالی ویژگی های هنجیده استفاده شده در کروموزوم های الگوریتم ژنتیک شناسایی شده و از آنها در ساخت راه حل های جدید استفاده می شود. روش ارائه شده می تواند روی مجموعه داده هایی با ابعاد متفاوت کار کند. بعد از انتخاب ویژگی ها برای ارزیابی دقت کلاس بندی از دو طبقه بند k نزدیک ترین همسایه(KNN) و درخت تصمیم(DT) استفاده می کنیم. در مرحله آخر، طبقه بندی داده ها با ویژگی های استخراج شده محاسبه می شود. منظور از تعداد ویژگی ها مشخص کردن تعداد ویژگی های انتخابی از کل ویژگی ها در مجموعه داده می باشد.. بعد از تعیین ویژگی های مطلوب نهایی، دقت طبقه بندی ویژگی های انتخاب شده را روی بهترین روش از بین KNN و DT نشان می دهیم.

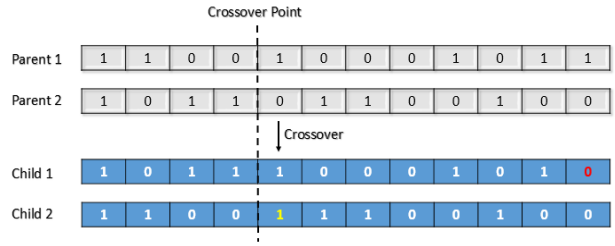
۳-۱- الگوریتم ژنتیک بهبود یافته برای انتخاب ویژگی ها

اولین مرحله در الگوریتم ژنتیک مشخص کردن ساختار کروموزوم ها می باشد. در واقع کد گذاری و نمایش کروموزوم ها به نوع مسئله و کاربرد آن بستگی دارد. در این مسئله شماره هر ویژگی را به عنوان معیار انتخاب آن ویژگی در نظر می گیریم. در این صورت ویژگی هایی که شماره آن در کروموزم وجود داشته باشد در طبقه بندی داده ها شرکت خواهند داشت و بقیه در نظر گرفته نمی شوند. طول کروموزم با توجه به تعداد کل ویژگی ها تعیین می شود و تعداد ژن های یک کروموزوم با توجه به تعداد ویژگی های مطلوب(DNF) مشخص می شود. در شکل ۱ نمونه ای از کروموزم پیشنهادی را مشاهده می کنید.



شکل ۱: ساختار کروموزوم پیشنهادی

الگوریتم ژنتیک کار خود را با جمعیت اولیه از کروموزوم ها آغاز می کند. جمعیت اولیه بـه دو صـورت تصـادفی و هیوریسـتیکی از فضای جستجو تولید می شود. ما در این تحقیق کروموزم ها را با هر دو روش از فضای جستجو انتخاب می کنیم. الگوریتم ژنتیک در ابتدا کار خود را با تولید جمعیت اولیه تصادفی شروع می کند. سپس با توجه به معیارهایی (در بخش های بعدی معرفی خواهد شـد) ویژگی های هنجیده و ناهنجیده را از فضای جستجو استخراج کرده و از این پس تولید جمعیت جدید با توجه به این ویژگی ها انجام می پذیرد. در اینجا ویژگی های هنجیده، برداری از ویژگی ها می باشد که در تولید جمعیت اولیه باید از آنها استفاده شـود و ویژگـی های ناهنجیده برداری از ویژگی ها است که استفاده از آنها در تولید جمعیت های پیشین نتـایج مطلـوبی نداشـته و از آنهـا در تولیـد جمعیت جدید استفاده نخواهد شد. با توجه با اینکه در روش پیشنهادی الگوریتم ژنتیک به صورت متوالی اجرا می شـود، لـذا در هـر تکرار نیاز به تولید جمعیت اولیه از ویژگی ها می باشد، که در تکرار اول این جمعیت بـه صـورت تصـادفی و در سـایر تکرارهـا بـه صورت هیوریستیکی از فضای جستجو تولید می شود. برای نمایش تعداد اعضای جمعیت از پارامتر ***NP*** استفاده خواهیم کـرد. تـابع برازندگی(***fitness***) در مسئله انتخاب ویژگی ها، نرخ خطا کلاس بندی داده ها می باشد. بنابراین نمونه های آموزشی را با توجـه بـه هر کروموزوم (ویژگی های انتخاب شده) کلاس بندی کرده و خطا محاسبه شده را به عنوان معیار خوبی آن کرومـوزوم در نظـر مـی گیریم. بنابراین مسئله ما مینیمم کردن تابع هدف می باشد. در این تحقیق از دو طبقـه بنـد معـروف ***KNN*** [۱۳] و ***DT*** [۱٤] جهـت کلاس بندی داده ها و محاسبه معیار برازندگی کروموزوم ها استفاده شده است. دلیل استفاده از این دو طبقه بند سرعت بالای آنهـا در طبقه بندی پایگاه داده بزرگ می باشد. با توجه به اینکه در روش انتخاب ویژگی، نیاز به محاسبه متوالی برازندگی ویژگی های انتخاب شده وجود دارد، لذا استفاده از روش هایی که با سرعت بالا (نسبت به دقت) بتوانند طبقه بندی داده هـا را انجـام دهنـد، در اولویـت است. به منظور انتخاب کروموزوم ها جهت تولید مثل از روش انتخاب تورنمنت استفاده مـی کنـیم. مهمتـرین عملگـر در الگـوریتم ژنتیک، عملگر ترکیب (آمیزش) است. جفت هایی که در قسمت انتخاب، به عنوان والـد در نظـر گرفتـه شـدند، در ایـن قسـمت ژن هایشان را با هم مبادله می کنند و اعضایی جدیدی بوجود می آورند. شکل های مختلفی از عملگر ترکیب وجود دارد که از مهمتـرین آنها می توان به ترکیب تک نقطه ای، ترکیب دو نقطه ای، ترکیب ***n*** نقطه ای و ترکیب یکنواخت اشاره کـرد. لازم بـه ذکـر اسـت کـه آمیزش معمولاً بر روی همه زوج کروموزوم‌های انتخاب شده برای جفت‌گیری به کار برده نمی‌شود. معمولاً احتمال آمیزش بـرای هـر زوج کروموزوم بین ۰/٦ تا ۰/۹۵ در نظر گرفته می‌شود که به این عدد نرخ آمیزش (***Pc***) گفته می‌شود. در صورتی کـه بـر روی یـک زوج کروموزوم عمل آمیزش صورت نگیرد، فرزندان با تکرار نمودن والدین تولید می‌شوند. در این تحقیق از اپراتور ترکیب تک نقطه ای استفاده شده که نمونه ای از آن را در شکل ۲ مشاهده می کنید.



شکل ۲ : اپراتور ترکیب تک نقطه ای بر روی کروموزوم پیشنهادی

همانطور که مشاهده می کنید دو مورد از ژن های فرزندان تغییر کرده است. در این مثال تعداد کل ویژگی ها (طول کروموزوم) ۱۲ و تعداد ویژگی های انتخاب شده (تعداد یک ها در کروموزوم) ٦ می باشد. تعداد ویژگی های انتخاب شده در تمامی کروموزوم ها برابر است. همانطور که در مثال بالا مشاهده می کنید، با نقطه برش انتخاب شده و با اعمال اپراتور ترکیب تک نقطه ای تعداد ویژگی ها در فرزند اول ۷ و در فرزند دوم ۵ می باشد. با توجه به موارد گفته شده باید تعداد ویژگی های انتخابی (ژن های یک کروموزوم) در فرزندان تولید شده برابر ٦ شود. لذا در فرزند اول که تعداد ویژگی های انتخابی بیشتر است یک مورد را به تصادف حذف می کنیم. در فرزند دوم نیز یک ویژگی استفاده نشده از والد اول به تصادف انتخاب کرده و در ژن متناظر قرار می دهیم. اپراتور ترکیب در روش پیشنهادی دو فرزند تولید می کند.

جهش یک فرایند تصادفی است که در آن محتوای یک ژن جهت تولید یک ساختار ژنتیک جدید جایگزین می شود. نقش جهش در الگوریتم ژنتیک بازگرداندن مواد ژنتیکی گم شده و یا پیدا نشده داخل جمعیت است. مزیت عملگر جهش اینست که توان دسترسی به همه فضای جستجو را به ما می دهد. در این قسمت از دو فرزند تولید شده توسط اپراتور ترکیب، یک مورد به تصادف انتخاب شده و جهش می یابد. در این تحقیق از اپراتور جهش تغییر بیت استفاده می کنیم. نمونه ای از این اپراتور در شکل ۳ نشان داده شده است. ایده اصلی این روش استفاده از ویژگی هایی می باشد که از آنها در راه حل پیشنهادی استفاده نشده است. در این اپراتور ابتدا یک ویژگی از کروموزوم (ژن با محتوای یک) انتخاب شده و حذف می گردد، سپس ویژگی جدید به کروموزم اضافه می شود. در اپراتور ترکیب فقط ویژگی های بکار رفته در دو والد انتخابی در فرزند استفاده می شود. لذا بکار گیری ویژگی های مشاهده نشده در بهبود راه حل پیشنهادی لازم و ضروری می باشد. به یاد داشته باشید که ساخت جمعیت اولیه با توجه به ویژگی های هنجیده و ناهنجیده استخراج شده انجام می گرفت. لذا در این اپراتور نباید ویژگی هایی اضافه یا حذف شود که معیار ساخت جمعیت اولیه را نقض کند. بنابراین از بین ویژگی های بکار رفته در کروموزوم فقط ویژگی هایی می توانند برای حذف انتخاب شوند که در لیست ویژگی های هنجیده نباشند. همچنین برای اضافه کردن ویژگی جدید، نباید از لیست ویژگی های ناهنجیده موردی انتخاب شود.

شکل ۳ : اپراتور جهش تغییر بیت بر روی یکی از فرزندان تولید شده

در این تحقیق به اندازه تعداد کروموزوم های جمعیت اولیه (***NP***) فرزند تولید می شود. سپس از میان جمعیت نسل قبل و جمعیت نسل فعلی تعدادی انتخاب شده و به نسل بعد می روند. برای انتخاب جمعیت نسل بعد، ابتدا کروموزوم های جمعیت نسل قبل و جمعیت نسل جاری بر اساس معیار برازندگی به صورت نزولی مرتب می شوند. سپس $1/4$ ابتدایی لیست(بهترین ها) به صورت مستقیم به نسل بعد می روند. کروموزوم های $1/4$ انتهایی لیست(بدترین ها) حذف می گردند و در نهایت سایر اعضای جمعیت بین کروموزوم های باقیمانده لیست، به صورت تصادفی انتخاب می شوند. یکی از پدیده‌های جالب در الگوریتم های ژنتیک این است که نسل‌های میانی کروموزوم هایی تولید می کنند که از نظر تابع ارزش و خوب بودن بسیار مناسب هستند. این کروموزوم ها ممکن است در اثر انجام اپراتورهای ترکیب و جهش از بین رفته و دیگر تولید نشوند. یک روش این است که اینگونه موارد را شناسایی کرده و در نسل های بعدی نیز از آن‌ها استفاده کنیم. به این تکنیک نخبه‌گرایی می‌گویند که عملاً تأثیر بسزایی در رسیدن به جواب مسئله دارد، چون این روش باعث می شود کروموزوم های خوب مستقیما و بدون واسطه به نسل های بعدی بروند. در این تحقیق در هر

نسل تنها یک کروموزوم که بهترین راه حل را از نظر طبقه بندی پیشنهاد داده، مستقیما به نسل بعدی منتقل می شود. نکته‌ای که در طراحی هر الگوریتمی بایستی به آن توجه شود، شرط خاتمه الگوریتم می باشد. در این تحقیق از تعداد مشخصی نسل برای شرط خاتمه الگوریتم ژنتیک استفاده میکنیم.

۳-۲- استفاده از فاکتور توزیع ویژگی ها در تشخیص ویژگی های هنجیده و ناهنجیده

در پایان الگوریتم ژنتیک، جمعیتی از راه حل ها بدست می آید. در اکثر روش های انتخاب ویژگی مبتنی بر الگوریتم ژنتیک، ویژگی ها بکار رفته در بهترین راه حل را به عنوان ویژگی های انتخاب شده در نظر گرفته و داده های آموزشی را بر مبنای این ویژگی ها کلاس بندی می کنند. ساختار الگوریتم ژنتیک بر مبنای جستجوی تصادفی می باشد، به همین دلیل همیشه راه حل بهینه و یکسانی را تولید نمی کند. به عنوان مثال اگر الگوریتم ژنتیک پیشنهادی را روی مجموعه داده تشخیص نفوذ با تعداد ثابتی از ویژگی ها در چندین اجرای مجزا اعمال کنیم، همیشه ویژگی های انتخابی در بهترین راه حل یکسان نیستند. با این شرایط پیدا کردن ویژگی های مطلوبی که بهترین عملکرد را در طبقه بندی داده ها دارند، امکان پذیر نخواهد بود. لذا در این تحقیق رویکردی را پیشنهاد می دهیم که تا حد زیادی به انتخاب بهترین ویژگی ها به ما کمک خواهد کرد. هدف ما در این بخش شناسایی ویژگی های هنجیده و ناهنجیده، با توجه به راه حل های بدست آمده از الگوریتم ژنتیک می باشد. برای شناسایی ویژگی های هنجیده و ناهنجیده از معیار فاکتور توزیع ویژگی ها (***FD***) در جمعیت استفاده می کنیم. توزیع هر ویژگی در جمعیت نشان دهنده میزان تکرار آن ویژگی می باشد. مفهوم توزیع ویژگی های هنجیده در جمعیت در واقع میزان تکرار آن ویژگی در بخش جمعیت هنجیده می باشد. به عنوان مثلا جمعیت هنجیده راه حل هایی می باشند که برازندگی آنها از میانگین برازندگی کل جمعیت بیشتر باشد. همچنین مفهوم توزیع ویژگی های ناهنجیده در جمعیت، میزان تکرار آن ویژگی در قسمت ناهنجیده جمعیت می باشد. به عنوان مثلا جمعیت ناهنجیده راه حل هایی می باشند که برازندگی آنها از میانگین برازندگی کل جمعیت کمتر باشد. فاکتور توزیع یک ویژگی در جمعیت هنجیده، نسبت تکرار آن ویژگی به کل جمعیت هنجیده می باشد و فاکتور توزیع یک ویژگی در جمعیت ناهنجیده، نسبت تکرار آن ویژگی به کل جمعیت ناهنجیده می باشد. شکل ٤ مفهوم فاکتور توزیع ویژگی ها را به خوبی نشان داده است.

Solution	f1	f2	f3	f4	f5	f6	f7	f8	Fitness
1	0	0	1	1	0	1	0	1	**87**
2	1	1	0	1	1	0	0	0	**85**
3	1	0	1	1	0	1	0	0	**78**
4	1	1	0	1	0	0	0	1	**70**
5	0	0	1	0	0	1	1	1	**50**
6	1	0	1	0	1	0	1	0	**45**
7	0	1	0	1	0	0	1	1	**40**
8	1	0	1	0	1	1	0	0	**39**
9	0	0	0	1	1	1	0	1	**37**
10	0	0	0	1	1	1	1	0	**32**

توزیع ویژگی ها در جمعیت خوب

$$FD_1^{best}=\frac{3}{4}\quad FD_2^{best}=\frac{2}{4}$$
$$FD_3^{best}=\frac{2}{4}\quad FD_4^{best}=\frac{4}{4}$$
$$FD_5^{best}=\frac{1}{4}\quad FD_6^{best}=\frac{2}{4}$$
$$FD_7^{best}=\frac{0}{4}\quad FD_8^{best}=\frac{2}{4}$$

توزیع ویژگی ها در جمعیت بد

$$FD_1^{bad}=\frac{2}{6}\quad FD_2^{bad}=\frac{1}{6}$$
$$FD_3^{bad}=\frac{3}{6}\quad FD_4^{bad}=\frac{3}{6}$$
$$FD_5^{bad}=\frac{4}{6}\quad FD_6^{bad}=\frac{4}{6}$$
$$FD_7^{bad}=\frac{4}{6}\quad FD_8^{bad}=\frac{3}{6}$$

شکل ٤ : مثالی از فاکتور توزیع ویژگی ها

همانطور که در مثال بالا مشاهده می کنید، با توجه به میانگین برازندگی جمعیت ٤ راه حل در جمعیت هنجیده و ٦ راه حل در جمعیت ناهنجیده فرض شده است. میزان تکرار ویژگی اول (***f1***) در جمعیت هنجیده ۳ و در جمعیت ناهنجیده ۲ می باشد. بنابراین فاکتور توزیع این ویژگی برای جمعیت هنجیده $FD_1^{best}=\frac{3}{4}$ و برای جمعیت ناهنجیده $FD_1^{bad}=\frac{2}{6}$ می باشد. در اینجا با توجه

به فاکتور توزیع ویژگی ها لیست ویژگی های هنجیده و ناهنجیده معرفی می شوند. به این صورت که بعد از اتمام الگوریتم ژنتیک، فاکتور توزیع را برای همه ویژگی ها محاسبه می کنیم. سپس ویژگی هایی که در جمعیت هنجیده فاکتور توزیع آنها از مقدار ثابتی مثل α بیشتر باشد به لیست ویژگی های هنجیده اضافه می شوند. همچنین ویژگی هایی که در جمعیت ناهنجیده فاکتور توزیع آنها از مقدار ثابتی مثل β کمتر باشد به لیست ویژگی های ناهنجیده اضافه می شوند. پارامتر α و β در واقع میزان تشابه راه حل ها و یا سخت گیری در انتخاب آنها را برای انتخاب یک ویژگی در جمعیت هنجیده و ناهنجیده کنترل می کنند. توجه داشته باشید که تعداد ویژگی های مطلوب در ابتدا به صورت ثابت تعریف شده و به عنوان ورودی به الگوریتم داده می شود. لذا برای تکمیل لیست ویژگی های هنجیده نیاز به تکرار مجدد الگوریتم ژنتیک و یافتن راه حل های جدید می باشد. بنابراین الگوریتم ژنتیک به صورت متوالی برای یافتن ویژگی های هنجیده تکرار شده تا زمانی که تعداد ویژگی های هنجیده استخراج شده به تعداد ویژگی ها مطلوب درخواستی برسد.

همانطور که در بخش های قبل نیز گفته شد، برای کمک به الگوریتم ژنتیک برای یافتن راه حل های بهینه تر، از لیست ویژگی های هنجیده و ناهنجیده در تولید جمعیت اولیه استفاده می کنیم. به طوری که تمامی جمعیت تولیدی شامل ویژگی های هنجیده استخراج شده باشند. همچنین از ویژگی های ناهنجیده استخراج شده در تولید جمعیت جدید استفاده نشود. این روش با ثابت قرار دادن تعدادی از ویژگی ها در هر تکرار، فضای جستجو را به میزان قابل توجهی کاهش می دهد. بعد از یافت ویژگی های مطلوب، نرخ خطا کلاس بندی با دو کلاسیفایر معرفی شده محاسبه می شود.

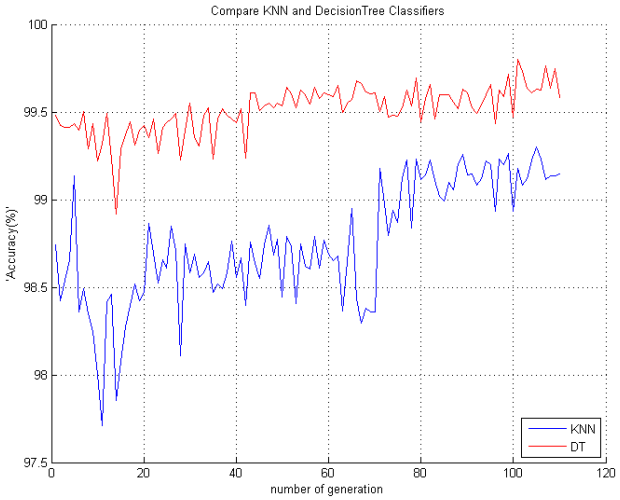
۳-۳-کنترل تطبیقی پارامترها

کنترل تطبیقی پارامترها در واقع روشی در تئوری کنترل به منظور تطبیق سیستم کنترلی با پارامترهایی در سیستم که مقدار آنها در طول فرآیند کاری سیستم تغییر می کند، است. مبنای کنترل تطبیقی براساس تخمین پارامترها می باشد. روش های مختلف تخمین شامل روش بازگشتی حداقل مربعات و روش گرادیان کاهشی می باشد. هر دوی این روش ها قوانین تطبیقی را به صورت آنلاین در سیستم تنظیم می کنند. در این تحقیق مقادیر دو پارامتر نرخ جهش(***Pm***) و میزان تشابه (α و β) در طی اجرای الگوریتم تغییر می کند. پارامتر نرخ جهش در ابتدای کار مقدار نسبتا بالایی دارد و در طی روند اجرای الگوریتم به صورت متوالی از مقدار آن کاسته می شوند. پارامتر میزان تشابه نیز در ابتدا درصد بالایی از فضای انتخابی را شامل می شود. ولی هر قدر به جلو می رویم به دلیل تفاوت میان ویژگی های انتخابی از مقدار آن کاسته می شوند. این پارامتر در تعداد تکراری مشخص، که بهبودی در شناسایی ویژگی های هنجیده نداشته باشیم، به میزانی کاسته می شود. این روش تا حدودی مشکل همگرائی زودرس به نواحی بهینه محلی در الگوریتم ژنتیک با عملگرهای ژنتیکی نرخ ثابت را هم رفع می‌کند.

۴- آزمایشات و نتایج

در این تحقیق برای ارزیابی روش پیشنهادی از مجموعه داده ***NSL-KDD*** استفاده شده است. برای دسترسی کامل به این مجموعه داده می توانید از ***[۱۵]*** استفاده کنید. این مجموعه داده بهبود یافته مجموعه داده ***KDDCUP'99*** است و برای حل برخی از مشکلات ذاتی آن پیشنهاد شده است ***[۱۶]***. مجموعه داده ***NSL-KDD*** از ٤١ ویژگی و ٥ کلاس تشکیل شده است. رکوردها شامل یک کلاس نرمال و ٤ نوع کلاس حمله ***Dos***، ***R2L***، ***U2R*** و ***Probing*** هستند. برای پیاده سازی و آنالیز مجموعه داده ها از نرم افزار متلب ورژن ۲۰۱۳ استفاده شده است. نتایج بدست آمده از آزمایشات به منظور افزایش دقت ارزیابی، میانگین ۳۰ مرتبه تکرار تست می‌باشد. در پیاده سازی انجام شده اندازه جمعیت ۲۵، تعداد نسل ۳۰، نرخ ترکیب ۰/۸۵ و نرخ جهش ۰/۱۵ در نظر گرفته شده است. سخت گیری الگوریتم در انتخاب ویژگی های هنجیده $\alpha = 0.95$ و ناهنجیده $\beta = 0.90$ می باشد. با اعمال الگوریتم پیشنهادی

تعداد ۲۳ ویژگی به صورت خودکار از فضای جستجو انتخاب شد. ما کارایی دو طبقه بند معرفی شده را از لحاظ دقت روی هر کروموزوم تولید شده بررسی کرده و نتایج را در شکل ۵ نشان داده ایم. با توجه به نتایج الگوریتم انتخاب ویژگی، روش طبقه بندی ***DT*** کارایی بهتری نسبت به ***KNN*** داشته و به همین منظور نتایج طبقه بندی بر مبنای این روش است.



شکل ۵ : مقایسه روش های طبقه بندی **KNN** و **DT** از لحاظ دقت تشخیص

۴-۱- معیارهای ارزیابی

ارزیابی یک مدل دسته بندی می تواند بر اساس نمونه های آموزشی و آزمایشی صورت گیرد. برای ارزیابی باید برچسبی که مدل دسته بندی به آن دسته حمله نسبت داده شده، مقایسه شود. وقوع حالات مختلف برای دسته ها با توجه به مجموعه داده های ورودی برای دسته بندی با مقادیر ***FN*** (***False Negative***)، ***TN*** (***True Negative***)، ***FP*** (***False Positive***) و ***TP*** (***True Positive***) برای سیستم تشخیص نفوذ با دو دسته در جدول ۱ نشان داده شده است. این جدول به ماتریس گیجی (***Confusion***) معروف است.

جدول ۱ : ماتریس درهم ریختگی برای داده های سیستم تشخیص نفوذ

Actual Records	***Predicted Normal***	***Predicted Attack***
Normal	***TN***	***FP***
Intrusions	***FN***	***TP***

معیار ***TN*** : درصد رکوردهای معتبر که به درستی طبقه بندی شده است.

معیار ***TP*** : درصد رکورد حمله که به درستی طبقه بندی شده است.

معیار ***FN*** : درصد رکوردهای که به اشتباه به عنوان فعالیت درست قرار گرفتند در حالی که در واقع آنها حمله هستند.

معیار ***FP*** : درصد رکوردهای که به اشتباه به عنوان حمله قرار گرفتند در حالی که در واقع آنها فعالیت معتبر هستند.

مهمترین معیار برای تعیین کارایی یک الگوریتم دسته بندی، معیار ***Accuracy*** می باشد. این معیار دقت کل یک دسته بند را محاسبه می کند. رابطه زیر نحوه محاسبه این معیار را نشان می دهد.

$$Accuracy = \frac{TP + TN}{TP+FP+TN+FN} \qquad (۱)$$

معیار ***Precision*** درصدی را نشان می دهد که از میان تمامی دسته ها که توسط دسته بند به آن دسته نسبت داده شده انـد، درسـت دسته بندی شده اند. به عبارتی دقت دسته بندی دسته ***i*** را با توجه به کل مواردی نشان می دهـد کـه برچسـب ***i*** بـرای نمونـه مـورد بررسی توسط دسته بند پیشنهاد شده است. نحوه محاسبه این معیار در رابطه زیر نشان داده شده است.

$$Precision_i = \frac{TP_i}{TP_i + FP_i} \qquad (۲)$$

معیار ***Recall*** برای یک دسته، که از میان تمامی دسته های حملات متعلق به آن دسته، به درستی دسته بندی شده است. بـه عبـارتی دقت دسته بندی دسته ***i*** را با توجه به کل نمونه های با برچسب ***i*** نشان می دهد. این معیار به صورت زیر محاسبه می شود.

$$Recall_i = \frac{TP_i}{TP_i + FN_i} \qquad (۳)$$

نکته قابل توجه این است که معیار ***Recall*** کارایی دسته بند را با توجه به تعداد رخداد دسـته ***i*** نشـان مـی دهـد، درحالیکـه معیـار ***Precision*** اساساً مبتنی بر دقت پیشبینی دسته می باشد و بیانگر آن است که به چه میزان می توانیم به خروجی دسـته بنـد اعتمـاد کنیم.

معیار ***F-measure*** از ترکیبی معیارهای ***Precision*** و ***Recall*** بدست می آید و در مواردی استفاده می شـود کـه نتـوان اهمیـت ویژهای را برای هر یک از دو معیار ***Precision*** و ***Recall*** نسبت به یکدیگر قائل شد. رابطه زیر نحوه محاسبه این معیار را نشـان می دهد.

$$F - measure = \frac{1 \times Precision_i \times Recall_i}{Precision_i + Recall_i} \qquad (٤)$$

کارآیی سیستم های تشخیص نفوذ را میتوان با معیارهای معرفی شده مورد ارزیابی قرار داد.

۲-۴- ارزیابی روش پیشنهادی

بعد از پیدا کردن ویژگی های مطلوب توسط الگوریتم ژنتیک، اکنون نتایج طبقه بندی مجموعه داده ***NSL-KDD*** را مطابق با ویژگـی های انتخاب شده بررسی می کنیم. جدول ۲ نتایج پیش بینی روش پیشنهادی را برای بخش های آموزش و تست به صورت جداگانـه و در حالت دو کلاسه نشان می‌دهد. میزان دقت هر یک از ٤ دسته ی حملات ***DOS***، ***Probe***، ***U2R*** و ***R2L*** به همراه دسته نرمـال محاسبه شده است. نتایج هر دسته را می توانید در جدول ۳ مشاهده کنید. ماتریس درهم ریختگی داده هـای سیسـتم تشـخیص نفـوذ برای هر یک از ٤ دسته ی حملات به همراه دسته نرمال محاسبه شده و در جـدول ٤ نشـان داده شـده اسـت. در ایـن جـدول تعـداد رکوردها برای هر نوع حمله به همراه تعداد پیش‌بینی‌ها ذکر شده است. در جدول ۵ بهترین نتایج طبقـه بنـدی روش پیشـنهادی را بـا معیارهای متفاوت مشاهده می کنید. نتایج بر مبنای هر کلاس در مقابل سایر کلاس ها محاسبه شده است. در ادامه برای ارزیابی هـر چه بیشتر رویکرد فوق، عملکرد سیستم پیشنهادی را با سایر روش های تشخیص نفوذ که نتایج آزمایشات خود را بـر روی داده هـای ***NSL-KDD*** ارائه کرده اند، مقایسه می‌کنیم. نتایج جدول ٦ نشان می‌دهد که روش پیشنهادی در مقایسه با سایر روش های تشخیص نفوذ و به ازای برخی از حملات دقت بیشتری داشته و در سایر موارد دقت مناسبی را ارائه می‌دهد.

جدول ۲ : نتایج روش پیشنهادی در حالت دو کلاسه

Dataset	*Predicted Normal*	*Predicted Attack*	*Predicted*
Train	*99.94*	*99.89*	*99.92*
Test	*99.87*	*99.78*	*99.81*

جدول ۳ : درصد تشخیص روش پیشنهادی به تفکیک نوع حمله

Dataset	*Normal*	*DOS*	*U2R*	*R2L*	*Probe*
Train	*99.94*	*99.96*	*84.62*	*98.59*	*99.80*
Test	*99.87*	*99.97*	*97.52*	*99.42*	*99.71*

جدول ٤ : ماتریس درهم ریختگی به تفکیک نوع حمله

Actual Records			*Predicted*				
Records Type	*Dataset*	*Number*	*Normal*	*DOS*	*U2R*	*R2L*	*Probe*
Normal	*Train*	*67343*	*67303*	*6*	*7*	*11*	*16*
	Test	*9710*	*9683*	*3*	*2*	*7*	*1*
DOS	*Train*	*45927*	*8*	*45909*	*0*	*0*	*10*
	Test	*7458*	*1*	*7454*	*1*	*0*	*0*
U2R	*Train*	*52*	*7*	*0*	*44*	*1*	*0*
	Test	*200*	*2*	*0*	*197*	*3*	*0*
R2L	*Train*	*995*	*13*	*0*	*1*	*981*	*0*
	Test	*2754*	*9*	*3*	*2*	*2753*	*2*
Probe	*Train*	*11656*	*22*	*0*	*0*	*1*	*11633*
	Test	*2421*	*7*	*0*	*0*	*0*	*2413*

جدول ۵ : ارزیابی دقت روش پیشنهادی با معیارهای مختلف

Records Type		*Accuracy*	*Recall*	*Precision*	*F-measure*
Normal	*Train*	*99.92*	*99.89*	*99.94*	*99.92*
	Test	*99.82*	*99.77*	*99.87*	*99.82*
DOS	*Train*	*99.93*	*99.89*	*99.96*	*99.93*
	Test	*99.85*	*99.73*	*99.97*	*99.85*
U2R	*Train*	*92.27*	*99.92*	*86.66*	*92.82*
	Test	*98.68*	*99.83*	*97.58*	*98.69*
R2L	*Train*	*99.26*	*99.93*	*98.61*	*99.27*
	Test	*99.64*	*99.86*	*99.42*	*99.64*
Probe	*Train*	*99.87*	*99.93*	*99.80*	*99.87*
	Test	*99.77*	*99.82*	*99.71*	*99.77*

جدول ٦ : مقایسه روش پیشنهادی با سایر روش های تشخیص نفوذ به تفکیک نوع حمله

Classifier	*Normal*	*DOS*	*U2R*	*R2L*	*Probe*	*Accuracy*
SIPSO [17]	-	*99.80*	*97.50*	*82.50*	*99.70*	-
Adaptive IDS [6]	*66.51*	*88.64*	*66.51*	*20.88*	*99.15*	*75.15*
GA [10]	-	-	-	-	-	*68.51*
CSM [11]	-	-	-	-	-	*99.79*
AdaBoost+ RandomTree +IG [5]	*99.85*	*100.0*	*88.45*	*98.00*	*99.70*	*97.20*
Ant Colony [12]	*97.41*	*99.78*	*93.51*	*99.17*	*74.65*	*98.9*
MARS [18]	*99.71*	*99.97*	*76.00*	*98.75*	*99.85*	*92.75*
Proposed Method	*99.87*	*99.97*	*97.52*	*99.42*	*99.71*	*99.81*

۵- نتیجه‌گیری

دقت الگوریتم های داده کاوی بستگی به انتخاب ویژگی های مناسب و همچنین تعداد رکوردهای مورد نظر برای یادگیری است. الگوریتم ژنتیک ارائه شده به خوبی ویژگی های مناسب را انتخاب می کند. با تنظیم دقیق و تطبیقی پارامترهای میزان تشابه، مجموعه ویژگی های هنجیده و ناهنجیده با سرعت بیشتری شناسایی شده و کارایی روش پیشنهادی بالا می رود. روش های داده کاوی معرفی شد نیز به خوبی دسته بندی داده ها را انجام می دهند. با این حال استفاده از الگوریتم های داده کاوی در سیستم تشخیص نفوذ دارای چالش هایی می باشند. داده ها و ترافیک تولید شده در شبکه، حجم بسیار بالایی دارند و تعداد ویژگی های آنها زیاد و اغلب ناهمگن هستند و نیاز به الگوریتم های داده کاوی با کارایی بالایی دارند. داده هایی که در شبکه سیر می کنند به صور ذاتی حالت جریانی دارند و نیاز است تا بتوان آنها را به صورت برخط شناسایی کرد. حملاتی که امروزه بر علیه شبکه صورت می پذیرد، عموما از چندین مبدا انجام می گیرد و هدف آنها نیز ممکن است چندین مقصد باشد. بنابراین نیار است تا بتوان داده های شبکه را در چندین نقطه تحلیل و نفوذ حملات توزیعی را تشخیص داد.

۶- کارهای آینده

هدف این تحقیق، یافتن مجموعه ویژگی بهینه برای سیستم های تشخیص نفوذ بوده است. با توجه به اینکه سیستم های تشخیص نفوذ، با داده‌های حجیم برای تحلیل مواجه هستند، کاهش مجموعه داده میتواند راه حل مناسبی برای افزایش دقت تشخیص آنها باشد. نیازمندی دیگری که در سیستم های تشخیص نفوذ مطرح می باشد، دانستن مجموعه ویژگی بهینه برای هر نوع حمله است. چرا که در اینصورت، سیستم تشخیص نفوذ قادر خواهد بود برای تشخیص هر نوع حمله، تنها از مجموعه ویژگی متناسب با آن حمله استفاده کند. همچنین در ادامه راه این تحقیق پیشنهاد می‌شود راه‌حلی به منظور یافتن تعداد ویژگی‌های مطلوب ابتدایی(***DNF***)، به صورت خودکار ارائه شود. به عنوان مثال میتوان الگوریتم ژنتیک را با طول رشته متغیر بهبود بخشید. در این حالت هر کروموزوم در جمعیت دارای تعداد ویژگی‌های متفاوتی خواهد بود.

مراجع

[1] Liao, Hung-Jen, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. (2013), Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications 36, no. 1 : 16-24.

[2] Pan, Shengyi, Thomas Morris, and Uttam Adhikari.(2015). Developing a hybrid intrusion detection system using data mining for power systems. IEEE Transactions on Smart Grid 6.6 : 3104-3113.

[3] Zuech, Richard, Taghi M. Khoshgoftaar, and Randall Wald.(2015). Intrusion detection and big heterogeneous data: a survey. Journal of Big Data 2.1 : 1.

[4] Denning, Dorothy E., and Peter G. Neumann. (1985), Requirements and model for IDES—a real-time intrusion detection expert system. Document A005, SRI International 333.

[۵] خدایار, محمد؛ علیرضا عصاره و منصور امینی لاری، ۱۳۹۳، بکارگیری الگوریتم های ترکیبی یادگیری ماشین در بهبود سیستمهای تشخیص نفوذ، همایش ملی مهندسی رایانه و مدیریت فناوری اطلاعات، تهران، شرکت علم و صنعت طلوع فرزین.

[6] Goyal, Anup, and Chetan Kumar, (2008). GA-NIDS: a genetic algorithm based network intrusion detection system. Northwestern university.

[7] Muda, Z., W. Yassin, M. N. Sulaiman, and N. I. Udzir, (2011). Intrusion detection based on K-Means clustering and Naïve Bayes classification. In Information Technology in Asia (CITA 11), 2011 7th International Conference on, pp. 1-6.

[8] Sriparna Saha, Ashok Singh Sairam, Asif Ekbal,(2012). Genetic Algorithm Combined with Support Vector Machine for Building an Intrusion Detection System, International Conference on Advances in Computing, Communications and Informatics (ICACCI-2012)

[9] Zhu, Shuxin, and Bin Hu, (2013). Hybrid feature selection based on improved GA for the intrusion detection system. Indonesian Journal of Electrical Engineering and Computer Science 11, no. 4 : 1725-1730.

[10] Benaicha, Salah Eddine, Lalia Saoudi, Salah Eddine Bouhouita Guermeche, and Ouarda Lounis, (2014). Intrusion detection system using genetic algorithm. In Science and Information Conference (SAI), 2014, pp. 564-568.

[11] Chae, Hee-su, Byung-oh Jo, Sang-Hyun Choi, and Twaekyung Park, (2015). Feature Selection for Intrusion Detection using NSL-KDD. Recent Advances in Computer Science, ISBN : 978-960.

[12] Aghdam, Mehdi Hosseinzadeh, and Peyman Kabiri, (2016). Feature selection for intrusion detection system using ant colony optimization. International Journal of Network Security 18.3 : 420-432.

[13] Park, Chan Hee, and Seoung Bum Kim. (2015), Sequential random k-nearest neighbor feature selection for high-dimensional data." Expert Systems with Applications 42.5 : 2336-2342.

[14]

[15] Hidayati, R., Kanamori, K., Feng, L., & Ohwada, H. (2016). Combining Feature Selection with Decision Tree Criteria and Neural Network for Corporate Value Classification. In Pacific Rim Knowledge Acquisition Workshop (pp. 31-42). Springer International Publishing.

[16] Nsl-kdd dataset for network based intrusion detection systems. Available on: http://nsl.cs.unb.ca/KDD/NSL-KDD.html, March 2009.

[17] Tavallaee M, Stakhanova N and Ghorbani AA., (2010). Towards credible evaluation of anomaly based intrusion detection methods , IEEE Transaction on System, Man and Cybernetics, Part-c, Applications and Reviews; 40(5):516-524.

[18] Warsi, Sana, Yogesh Rai, and Santosh Kushwaha, (2015). Selective Iteration based Particle Swarm Optimization (SIPSO) for Intrusion Detection System. International Journal of Computer Applications 124.17.

[19] Mubarak, Shaik Liyakhat, (2016). Intrusion Detection System using SVM, SOM & NN.

improving intrusion detection systems by feature reducing based on genetics algorithm and data mining techniques

Mehdi Keshavarzi[1*], hossein Momenzadeh [2*]

[1*] Mehdikeshavarzi2000@gmail
[۲*] Momenzadeh.hossein@gmail

ABSTRACT:

The network-based computer systems play critical role in our modern society; so there is highly chance these systems might be target of intrusion and attacks. In order to implement full-scale security in a computer network, firewalls and other intrusion prevention mechanisms aren't always enough and needs other systems called intrusion detection systems. An Intrusion detection system can be set of tools, algorithms and evidence that help to identify, locate and report illegal or not approved activities by the network. Intrusion detection systems can be established by software or hardware systems and each have their own advantages and disadvantages. Because of various characteristics of intrusion detection data, in this research we select effective characteristics using improved genetic algorithm. Then by means of standard data mining techniques, we present a model for data classification.For performance evaluation of this suggested method, we used NSL-KDD **database that has more realistic records than other intrusion detection data.**

KEYWORDS: system intrusion detection, data mining, feature selection, genetic algorithms, database NSL-KDD