دانشکده مهندسی گروه کامپیوتر

پایان نامه کارشناسی ارشد مهندسی نرم افزار

تشخیص نفوذ در شبکه به روش محاسبات نرم تکاملی با استفاده از طبقه بندی کننده الگوی فازی- عصبی

عادل نجاران طوسی

استاد راهنما: دکتر کاهانی

استاد مشاور: دکتر منصفی

تابستان ۱۳۸۵

فهرست

فهرست جداول

فهرست اشکال

خلاصه

هدف اصلی یک سیستم تشخیص نفوذ طبقه بندی فعالیتهای یک سیستم به دو گروه اصلی است: فعالیتهای نرمال و فعالیتهای نفوذی (مشکوک). سیستم های تشخیص نفوذ به طور معمول نوع حملات را مشخص می کنند یا آنها را در گروه های خاص طبقه بندی می کنند. هدف اصلی در ارائه این پایان نامه ترکیب چند روش محاسبات نرم به عنوان یک سیستم طبقه بندی کننده می باشد که نفوذ ها را بر اساس نوع حمله از فعالیتهای عادی در سطح شبکه تشخیص داده و گزارش می کند. در میان روش های مختلف محاسبات نرم شبکه های فازی- عصبی، سیستم های استنتاج فازی و الگوریتم ژنتیک در این پایاننامه به کار گرفته شده اند. یک مجموعه از طبقه بندی کننده های فازی- عصبی که به صورت موازی عمل می کنند، برای انجام طبقه بندی اولیه مورد استفاده قرار می گیرند. سپس ماژول تصمیم گیری فازی بر اساس خروجی های طبقه بندی کننده های فازی – عصبی یک تصمیم گیری نهایی در مورد اینکه آیا فعالیت جاری نفوذی است یا یک فعالیت عادی انجام می دهد. در نهایت برای رسیدن به بهترین نتایج الگوریتم ژنتیک ساختار موتور تصمیم گیری فازی را بهنیه سازی می کند. آزمایشات و ارزیابی سیستم ارائه شده بر اساس مجموعه داده های ارزیابی KDD انجام گرفته است. نتایج نشان می دهد که روش ارائه شده در مقایسه با روش های دیگر در تشخیص نفوذ موثر باشد.

تقدیر و تشکر

فصل اول: مقدمه

اینترنت به همان نسبت که در تبادل و انتقال اطلاعات انقلاب ایجاد کرده است، شانس بیشتری به خرابکاران و نفوذگران می دهد تا بتوانند به اطلاعات امینتی یا مخفی دسترسی پیدا کنند. مطالعه سیستم های تشخیص نفوذ از این جهت که این سیستم ابزار مهمی در جهت مقابله با نفوذ های احتمالی می باشد حائز اهمیت است. سیستم های تشخیص نفوذ در یک محیط که دائماً در حال تغییر است باید به تشخیص فعالیت های یا رفتار های غیر طبیعی در شبکه بپردازد علاوه بر این در این چنین سیستم هایی مهم است که میزان هشدار های غلط یعنی آن دسته از فعالیتهایی که جز فعالیتهای طبیعی یا معمول شبکه هستند و به عنوان فعالیت های نفوذی شناخته می شوند در حداقل ممکن باشد[۲۴].

به طور کلی یک سیستم تشخیص نفوذ فعالیتهای محیطی را که در آن عمل می کند مانیتور می کند و سپس اطلاعات غیر ضروری را از اطلاعات بدست آمده حذف می کند، معمولا یک سری خصیصه برای این اطلاعات جمع آوری شده استخراج می شود، آنگاه پس از ارزیابی فعالیتها، احتمال وجود حمله بررسی می شود که این عمل توسط تشخیص دهنده انجام می گیرد. پس از تشخیص معمولا سیستم واکنش مناسب را در قبال تهاجم تشخیص داده شده انجام می دهد. اغلب سیستم های تشخیص نفوذ همانطور که از اسم آنها پیدا است، فقط حملات را تشخیص داده و اعلام هشدار می نمایند و معمولا هیچ عمل بازدارنده ای از آنها صادر نمی شود، بعضی از انواع این سیستم در قبال تشخیص نفوذ واکنشهای بازدارنده را نیز انجام می دهند. مهمترین بخش یک سیستم تشخیص نفوذ، تشخیص دهنده است که وظیفه اصلی وارسی اطلاعات جمع آوری شده را بر عهده دارد [۸, ۴].

محاسبات نرم به عنوان مجموعه ای از روشهای ابتکاری، که یک سیستم محاسبانی هوشمند را به وجود می آورند که این سیستم توانایی شگفت آور ذهن انسان برای استدلال کردن و یادگیری در یک محیط نا معلوم و بدون قطعیت را دارا می باشد [۳۹]، به کرات در سیستم های تشخیص نفوذ به کار گرفته شده اند[۳۴ , ۴۰,۱۳ ,۵ ,۲]. در این مطالعه یک سیستم تشخیص نفوذ جدید که از ترکیب چندین روش محاسبات نرم شامل متد های فازی-عصبی، سیستم های استنتاج فازی و الگوریتم های ژنتیک به وجود آمده است، ارائه می شود. اصلیترین نوآوری در این کار استفاده از خروجی های شبکه فازی- عصبی به عنوان یک مجموعه متغیر های زبانی است که مشخص کننده میزان معتبر بودن خروجی فعلی می باشند.

فازی به عنوان یک از روشهای محاسبات نرم قابلیت خود را در سیستم های تشخیص نفوذ به اثبات رسانده است[۱۳ ,۱۱ ,۱۰ ,۵ ,۱] . سیستم های فازی ویژگیهای خاصی دارند که آنها را برای تشخیص نفوذ مناسب می کند [۱۰]. سیستم های فازی به طور معمول از یک فرد خبره برای ایجاد پایگاه داده قوانین فازی خود استفاده می کنند و سپس این قوانین برای تصمیم گیری در مورد ورودیهای سیستم استفاده می شوند. اما کسب دانش از متخصصین به دلایل متعددی سخت، مقارن با اشتباه و یک پروسه وقت گیر و تکراری است. علاوه بر این سیستم های فازی معمول تطبیق پذیر نیستند بدین معنا که پس از آنکه قوانین فازی در پایگاه قوانین قرار داده شدند این قوانین تغییری نخواهند کرد و در صورتیکه سیستم با وضعیت های جدیدی رو به رو شود نه تنها قادر نیست قوانین جدید را به پایگاه قوانین اضافه کند، بلکه قادر به تغییر قوانین موجود نیز نمی

باشد. بنابراین ساخت یک سیستم فازی با قابلیت های یادگیری و تطبیق پذیری اخیراً بسیار مورد توجه قرار گرفته است[۱]. روشهای مختلفی برای تولید و تنظیم خودکار قوانین فازی بدون نیاز به یک فرد خبره ارائه شده است که روشهای فازی- عصبی [۳۰ ,۱۹] و ژنتیک- فازی [۱۷ ,۲۵] دو مورد از معروف ترین این روشها در این مجال هستند.

یکی از روشهای مرسوم در تشخیص نفوذ استفاده از روش های طبقه بندی الگو است. از دیدگاه طبقه بندی هر فرایند طبقه بندی شامل دو فاز است، فاز اول آموزش پارامتر های طبقه بندی کننده با استفاده از داده های آموزشی و فاز دوم استفاده از طبقه بندی کننده برای کلاس بندی داده های آزمایش. در این پایاننامه ما از شبکه فازی- عصبی تطبیق پذیر به عنوان طبقه بندی کننده استفاده کرده ایم که این شبکه قابلیت دارد که سیستم را بر اساس داده نمونه مدل کند.

سیستم تشخیص نفوذ ارائه شده در این پایان نامه دارای چندین لایه است. در لایه اول چندین طبقه بندی کننده فازی- عصبی وجود دارد که از مشخصه های استخراج شده از داده های شبکه استفاده می کنند و یک کلاس بندی اولیه را انجام می دهند. پس از آن یک سیستم استنتاج فازی به عنوان یک موتور تصمیم گیرنده بر اساس خروجی های طبقه بندی کننده های لایه اول یک تصمیم گیری نهایی را در مورد فعالیت جاری انجام که مشخص می کند این فعالیت یک فعالیت نفوذی است یا جزء فعالیتهای عادی شبکه است. در نهایت برای رسیدن به یک ساختار بهینه الگوریتم ژنتیک برای ماژول تصمیم گیری فازی را بهینه سازی می کند.

آزمایشگاه لینکلن در دانشگاه MIT، تحت حمایت آژانس پروژه تحقیقاتی پیشرفته[۱] دفاع (DARPA) و آزمایشگاه تحقیقاتی نیروی هوایی (AFRL/SNHS) ، اولین مجموعه داده برای ارزیابی سیستم های تشخیص را تولید و توزیع کرده است [۲۴ ,۷]. قبل از این رویداد هیچ مجموعه داده معتبری برای مقایسه سیستم های تشخیص نفوذ وجود نداشت و به این ترتیب مبنایی برای مقایسه سیستم ها تشخیص نفوذ به وجود آمد که به آن مجموعه داده های DARPA می گویند[۱۸]. بعد ها پنجمین کنفرانس سراسری کشف دانش و داده کاوی[۲] ACM SIGKDD به منظور برگزاری یک مسابقه در زمینه سیستم های یادگیری ماشین، داده های TCP جمع آوری شده در مجموعه DARPA را به فرم یک مجموعه آموزش و آزمایش[۳] شامل خصیصه[۴] های بدست آمده برای رکورد های اتصال[۵] جمع آوری و تولید کرد. هدف اصلی از این مسابقه انتخاب طبقه بندی کننده[۶] با بیشترین توانایی و کیفیت در تشخیص اتصالات نرمال و نفوذی بود. این مجموعه داده، مجموعه داده های ارزیابی KDD cup 99 یا به اختصار KDD'99 نامیده می شود و در این مطالعه برای ارزیابی و انجام آزمایشات سیستم ارائه شده مورد استفاده قرار گرفته است.

---

[۱] Defense Advanced Research Project Agency

[۲] International Conference on Knowledge Discovery and Data Minig

[۳] Train and Test Set

[۴] Feature

[۵] اتصال (Connection) یک دنباله ار بسته های TCP که در یک زمانهای مشخص شروع می شود و خاتمه می یابد.

[۶] Classifier

بخش های بعدی این پایان نامه به صورت زیر سازمان دهی شده است: ابتدا به بررسی کار مشابه و ارائه پیش زمینه های لازم می پردازیم. این فصل با تعریف تشخیص نفوذ و سیستم های تشخیص نفوذ آغاز می شود. سپس مجموعه داده های KDD که در بالا به آن اشاره کردیم را با جزئیات بیشتر بررسی می کنیم. در ادامه کار های مختلفی که در زمینه تشخیص نفوذ با توجه به داده های KDD انجام گرفته است را به اختصار بررسی می کنیم. سپس به معرفی سیستم های تشخیص نفوذ که بر اساس روش های محاسبات نرم و روشهای یادگیری ماشین شکل گرفته اند، می پردازیم. این فصل با مقدمه برای سیستم های فازی که نقش اساسی را در سیستم ارائه شده در این پایاننامه بازی می کنند ادامه می یابد. پس از معرفی سیستم های استنتاج فازی شبکه فازی- عصبی تطبیق پذیر با جزئیات ساختار ارائه می شود و در انتها این فصل با مقدمه ای بر الگوریتم های ژنتیک پایان می یابد.

فصل سوم به دو قسمت کلی تقسیم می شود. در بخش اول شبکه فازی- عصبی تطبیق پذیر که در اصل یک طبقه بندی کننده الگو نمی باشد و اغلب به عنوان مدل کننده یک سیستم استفاده می شود به صورت یک طبقه بندی کننده ارائه می شود و نتایج ارزیابی روش ارائه شده بر این اساس برای داده های KDD ارائه می شود. در بخش دوم روش ارائه شده در بخش اول به دو صورت طبقه بندی کننده دوگانه و چندگانه مورد مقایسه قرار می گیرد و نتایج هر یک از این دو روش بر روی داده های KDD مورد مقایسه قرار می گیرد که نتایج حاکی از بهتر بودن طبقه بندی کننده دو گانه در این فرایند می باشد که این امر به نوبه خود زمینه سازی ارائه سیستم تشخیص نفوذ مورد نظر ما در فصل بعدی است.

در فصل بعدی سیستم تشخیص نفوذ ارائه شده در این پایاننامه معرفی می شود. این معرفی با ارائه ساختار یا همان معماری کلی سیستم آغاز می شود. سپس در ادامه اجزاء این سیستم با جزئیات در بخش های بعدی مورد بررسی قرار می گیرد. ابتدا منابع داده ها که از مجموعه داده های KDD استخراج شده است معرفی می شوند. در ادامه با ماژولهای طبقه بندی کننده فازی-عصبی در سطح اول معماری سیستم آشنا می شویم. بعد از آن ساختار ماژول تصمیم گیری فازی ارائه می شود. ماژول الگوریتم ژنتیک و ساختار آن در سیستم ارئه شده آخرین مطلبی است که در این بخش آن می پردازیم.

فصل بعدی یا فصل پنجم شامل یک بخش کلی است که در آن نتایج آزمایشات ارائه شده است. که در آن سیستم ارائه شده با چند روش دیگر مقایسه می شود.

فصل آخر به نتیجه گیری کلی و ارائه راهکارهایی برای فعالیتهای آتی در این زمینه می پردازد.

فصل دوم: پیش زمینه و بررسی کارهای مشابه

۱ – ۳ مقدمه

این فصل با تعریف مساله آغاز می شود و تشخیص نفوذ از دیدگاه های مختلف مورد بررسی در این قسمت از پایاننامه مورد بررسی قرار می گیرد. تعاریف اولیه در تشخیص نفوذ، انواع سیستم های تشخیص نفوذ و نوع نگرش آنها به حملات، روشهای برخورد با حملات در شبکه کامپیوتری از جمله مباحثی است که در این بخش به آن می پردازیم. در انتهای بخش اول مجموعه داده های مختلف برای ارزیابی کارایی سیستم های تشخیص نفوذ ارائه می شوند. در بخش بعدی مجموعه داده های KDD cup 99 به عنوان نمونه ای از این داده ها که مرجع اصلی برای انجام آزمایشات و ارزیابی سیستم ارائه شده در این مطالعه است با جزئیات بیشتر معرفی می شود. پس از آن به بررسی مطالعات و فعالیتهای انجام شده در زمینه تشخیص نفوذ که از داده های KDD برای ارزیابی و یا انجام تحقیق در آنها استفاده شده است، می پردازیم. مطالعه و بررسی سیستم های محاسبات نرم و نقش آنها در زمینه تشخیص نفوذ از جمله مباحثی است که در بخش بعد به آن پرداخته شده است. روشهای مختلف محاسبات نرم در زمینه تشخیص نفوذ به اختصار در این بخش معرفی می شوند و توانایی این روشها در این زمینه مورد بررسی قرار می گیرد. بخش ششم از این فصل به معرفی سیستم های استنتاج فازی نمونه ای از روشهای محاسبات نرم به عنوان زمینه ای برای سیستم استنتاج فازی- عصبی تطبیق پذیر معرفی می شوند. ادامه این فصل به معرفی سیستم استنتاج فازی- عصبی استفاده شده در این پایاننامه می پردازد. در نهایت از آن رو که الگوریتم ژنتیک نقش مهمی در چهارچوبه سیستم ارائه شده در این پایاننامه بر عهده دارد، مقدمه ای بر الگوریتم های ژنتیک روش کارکرد و عملگر های موجود در این زمینه در انتهای این فصل ارائه می شود.

۲ – ۳ تشخیص نفوذ

با فراگیر شدن کامپیوتر و گسترش شبکه‌های کامپیوتری در اجتماع امروز و به دلیل حملات و نفوذهایی که به شکل‌های مختلفی به این شبکه‌ها صورت می‌گیرد، امنیت شبکه‌های کامپیوتری از حساسیت خاصی برخوردار گشته است. از آنجا که از نظر تکنیکی ایجاد سیستمهای کامپیوتری (سخت افزار و نرم افزار) بدون نقاط ضعف و شکست امنیتی عملاً غیر ممکن است، کشف نفوذ در تحقیقات سیستمهای کامپیوتری با اهمیت خاصی دنبال می‌شود.

نفوذ به مجموعه اقدامات غیر قانونی که صحت و محرمانگی و یا دسترسی به یک منبع را به خطر می‌اندازد اطلاق می‌گردد. نفوذها می‌توانند به دو دستهٔ داخلی و خارجی تقسیم شوند. نفوذهای خارجی به آن دسته از نفوذهایی گفته می‌شود که توسط افراد مجاز و یا غیر مجاز از خارج شبکه به درون شبکه داخلی صورت می‌گیرد و نفوذهای داخلی توسط افراد مجاز در سیستم و شبکه داخلی و از درون خود شبکه انجام می‌پذیرد.

نفوذگر[۱]ها عموماً از عیوب نرم‌افزاری ، شکستن کلمات رمز، استراق سمع ترافیک شبکه و نقاط ضعف طراحی در شبکه، سرویس‌ها و یا کامپیوترهای شبکه برای نفوذ به سیستم‌ها و شبکه‌های کامپیوتری بهره می‌برند. به عبارت دیگر یک نفوذگر کسی است که تلاش می کند که به سیستم نفوذ کند و از آن سوء استفاده کند . البته کلمه سوء استفاده پهنه وسیعی را در بر می گیرد و می تواند منعکس کننده عملی جدی مانند دزدیدن اطلاعات محرمانه و یا عمل کوچک و ناچیزی مثل دزیدن آدرس e-mail جهت فرستادن spam باشد.

به منظور مقابله با نفوذگران سیستم‌ها و شبکه‌های کامپیوتری، روشهای متعددی تحت عنوان روشهای تشخیص نفوذ ایجاد گردیده که عمل نظارت بر وقایع اتفاق افتاده در یک سیستم یا شبکه کامپیوتری را بر عهده دارند. هدف از تشخیص نفوذ کشف هرگونه استفاده غیر مجاز، سوء استفاده و یا آسیب رساندن به سیستم‌ها و یا شبکه‌های کامپیوتری توسط هر دو دسته کاربران داخلی و خارجی است.

سیستم‌های کشف نفوذ به صورت سیستم‌های نرم افزاری و یا سخت افزاری ایجاد شده و هر کدام مزایا و معایب خاص خود را دارا می‌باشد. سرعت و دقت از مزایای سیستم‌های سخت افزاری است و عدم شکست امنیت آنها توسط نفوذگران قابلیت دیگر اینگونه سیستم‌هاست. اما استفاده آسان از نرم افزار و قابلیت انطباق پذیری در شرایط نرم افزاری و تفاوت سیستم عامل‌های مختلف عمومیت بیشتری را در سیستم‌های نرم افزاری ارائه می‌کند و عموماً اینگونه سیستم‌ها انتخاب مناسب‌تری هستند.

سیستم‌های تشخیص نفوذ به لحاظ تکنیک برخورد با نفوذها به سه دسته تقسیم می‌شوند[۴].

۱. تشخیص سوء استفاده[۲]

در روشهای تشخیص سوء استفاده، الگوهای نفوذ از پیش ساخته شده به صورت قانون نگهداری می‌شوند، به طوری که هر الگو انواع متفاوتی از یک نفوذ خاص را در برگرفته و در صورت بروز چنین الگویی در سیستم وقوع نفوذ اعلام می‌شود. به این روش ها، روش های مبتی بر امضاء[۳]نیز می گویند، زیرا معمولا تشخیص دهنده دارای پایگاه داده ای از امضا یا الگو های رخداد حمله است و سعی میکند با بررسی ترافیک شبکه الگو های مشابه با آنچه را در پایگاه داده خود نگهداری می کنند، بیابد. این دسته از روش‌ها تنها قادر به شناسائی نفوذهای شناخته شده می‌باشند و در صورت بروز حملات جدید در سطح شبکه، نمی توانند آنها را شناسایی کنند و مدیر شبکه همواره باید الگوهای حملات جدید را به سیستم تشخیص نفوذ اضافه کند. از مزایای این روش دقت در تشخیص نفوذهایی است که الگوی آنها عیناً به سیستم داده شده است.

---

[۱] Intruder

[۲] Misuse Detection

[۳] Signature Based

۲. تشخیص رفتار غیر عادی[۱]

در روشهای تشخیص رفتار غیر عادی که بر اساس رفتار عادی سیستم بنا شده است، در یک بازهٔ زمانی خاص و مطمئن از این جهت که هیچگونه نفوذی در این بازه صورت نگرفته، اقدام به ایجاد نماها[۲]ی رفتار عادی کاربران می‌شود و در صورتی که رفتار سیستم در زمان آزمایش از این الگوها تبعیت نکند به صورت یک نفوذ احتمالی در نظر گرفته می‌شود. از مزایای این روش می توان به این نکته اشاره نمود که این روش ها قادر هستند حملات جدید و فعالیتهای نفوذی جدید که هیچ آموزشی برای آنها ندیده اند را شناسایی کنند، زیرا همانطور که گفته شد این روش ها به دنبال الگو هایی می گردند که به اندازه کافی از رفتار های نرمال شبکه متفاوت هستند و بدین ترتیب قادر به تشخیص حملات جدید نیز هستند. با این وجود ایجاد یک سیستم تشخیص نفوذ بر اساس این روش تشخیص آنامولی یا رفتارهای غیر عادی همیشه کاری آسانی نیست و این روش ها از دقت روشهای تشخیص سوء استفاده برخوردار نیستند.

روش های تشخیص نفوذ گوناگونی که بر این اساس کار می کنند تا کنون ارائه شده اند که از معروفترین این روش ها می توان به روش ارائه شده توسط Denning اشاره نمود [۹]،که بر اساس معیار های آماری داده ها کار می کند.

۳. تشخیص ترکیبی[۳]

این تشخیص دهنده ها در اصل ترکیبی از روشهای تشخیص سوء استفاده و روش تشخیص رفتار غیر عادی هستند. به عبارت دیگر تشخیص در این روش هم با استفاده از فعالیتهای نرمال سیستم و فعالیتهای نفوذی نفوذگر انجام می گیرد. از معروفترین این روش ها می توان به روش ارائه شده در [۲۲] اشاره نمود که از قوانین RIPPER برای تشخیص نفوذ استفاده می کند.

علاوه بر تقسیم بندی فوق به طور کلی سیستم‌های تشخیص نفوذ از نقطه نظر منبعی که تشخیص نفوذ روی آن صورت می‌گیرد نیز به دو دسته تقسیم می‌شوند[۸]:

۱. تشخیص نفوذ بر اساس مدل میزبان

در روش مبتنی بر میزبان، تشخیص نفوذ در یک سیستم منفرد مد نظر بوده و معمولا این روش ها بر اساس فعالیتهای کاربر سیستم شامل فراخوانی های سیستمی و غیره می باشد و میتواند به هر دو صورت تشخیص نفوذ مبتنی بر تشخیص سوء استفاده و یا تشخیص رفتار غیر عادی انجام شود.

۲. تشخیص نفوذ بر اساس ترافیک شبکه

---

[۱] Anomaly Detection

[۲] Profile

[۳] Compound Detection

در تشخیص نفوذ مبتنی بر ترافیک شبکه حملات به کل ساختار شبکه و یا هر یک از میزبانهای شبکه، می‌تواند سیستم تشخیص نفوذ رابرای اعلام نفوذ فعال کند. این روش نیز می‌تواند به صورت تشخیص سوءاستفاده و یا تشخیص رفتار غیر عادی اعمال شود.

تفاوت این دو دسته در منبع داده‌ای است که سیستم تشخیص نفوذ برای جمع‌آوری اطلاعات از آن بهره می‌برد. در تشخیص نفوذ مبتنی بر مدل میزبان منبع داده از اطلاعات یک کامپیوتر استفاده می‌کند، به این صورت که یک کارگزار[۱] هوشمند بر روی میزبان نظارت شده نصب می‌گردد و جنبه‌های متفاوتی از امنیت میزبان از قبیل فایلهای رخدادهای سیستم عامل[۲]، فایلهای رخدادهای برنامه‌های کاربردی و غیره را در نظر می‌گیرد. در حالیکه در تشخیص نفوذ مبتنی بر شبکه از ترافیک شبکه به عنوان منبع اصلی اطلاعات استفاده می‌شود.ما در این مطالعه سعی کرده ایم یک سیستم تشخیص نفوذ بر اساس ترافیک شبکه ارائه کنیم.

به طور کلی یک سیستم تشخیص نفوذ فعالیتهای محیطی را که در آن عمل می کند مانیتور می کند و سپس اطلاعات غیر ضروری را از اطلاعات بدست آمده حذف می کند، معمولا یک سری خصیصه برای این اطلاعات جمع آوری شده استخراج می شود، آنگاه پس از ارزیابی فعالیتها، احتمال وجود حمله بررسی می شود که این عمل توسط تشخیص دهنده انجام می گیرد. پس از تشخیص معمولا سیستم واکنش مناسب را در قبال تهاجم تشخیص داده شده انجام می دهد. اغلب سیستم های تشخیص نفوذ همانطور که از اسم آنها پیدا است، فقط حملات را تشخیص داده و اعلام هشدار می نمایند و معمولا هیچ عمل بازدارنده ای از آنها صادر نمی شود، بعضی از انواع این سیستم در قبال تشخیص نفوذ واکنشهای بازدارنده را نیز انجام می دهند. مهمترین بخش یک سیستم تشخیص نفوذ، تشخیص دهنده است که وظیفه اصلی وارسی اطلاعات جمع آوری شده را بر عهده دارد. شکل ۱-۳ شمای کلی یک سیستم تشخیص نفوذ را بر اساس تعاریف ارائه شده نشان می دهد.

تاکنون روشهای بسیار متنوعی در ساخت یک تشخیص دهنده به کار گرفته شده است که پهنه وسیعی از متدهای مختلف را در بر می گیرد، به عنوان مثال روشهای داده کاوی[۲۲]، نمودار های گذر حالات[۱۶]، روشهای خوشه بندی[۳] [۱۴] و تکنیک های طبقه بندی[۴][۱۳] را میتوان در این میان مشاهده کرد. نمونه های اشاره شده نمونه های اندکی از صد ها روش به کار گرفته شده برای سیستم های تشخیص نفوذ است. تنوع این روشها بقدری زیاد است که نوشتن چندین کتاب در این زمینه برای پوشش ها آنها کافی به نظر نمی رسد و ظاهرا کمتر روش علمی برای حل مساله تشیخص نفوذ وجود دارد که تاکنون در این زمینه به کار گرفته نشده است. لازم بذکر است به علت اهمیت موضوع تکنیک ها و روشهای بسیار متنوعی در این زمینه تحقیقاتی به

---

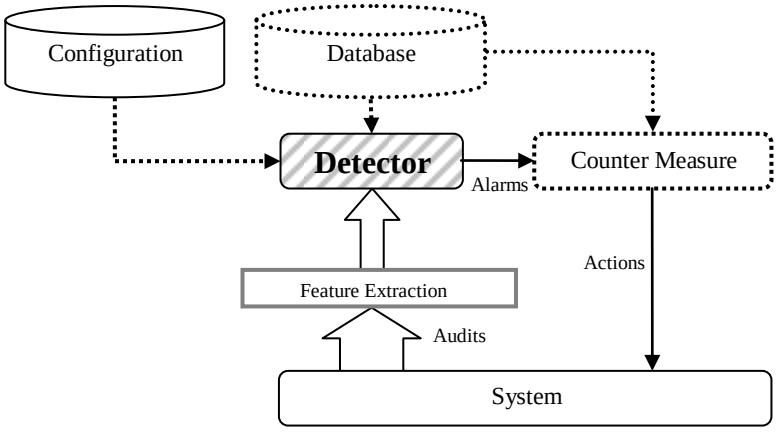
[۱] Agent

[۲] Log Files

[۳] Clustering

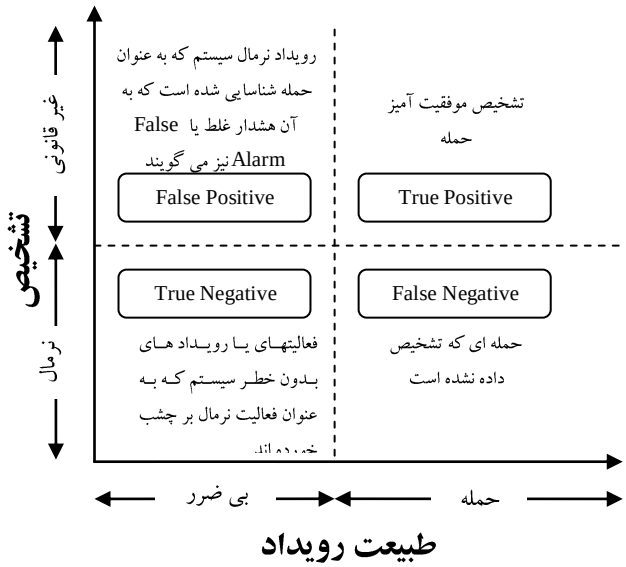
[۴] Classification

کار گرفته شده است. ما در این مطالعه سعی کرده ایم یک روش طبقه بندی الگو را که از تکنیک های مختلف محاسبات نرم استفاده می کند، برای یک سیستم تشخیص نفوذ ارائه کنیم.



شکل ۱-۳: یک سیستم تشخیص نفوذ ساده

از ویژگیهای بسیار مهم یک سیستم تشخیص نفوذ دقت در تشخیص و شناخت نفوذهای اتفاق افتاده و عدم تشخیص غلط رفتارهای نرمال به عنوان یک رفتار غیر عادی می‌باشد. ضعف یک سیستم کشف نفوذ در هر یک از این دو ویژگی باعث می‌شود تأثیر منفی در کارآیی سیستم اصلی داشته باشیم تا جایی که عدم حضور سیستم تشخیص نفوذ در چنین حالتی منطقاً بر حضور آن ترجیح خواهد داشت. چرا که آلارم‌های غلط مسئول امنیتی سیستم را از حضور سیستم کشف نفوذ خسته کرده وهمچنین آلارم‌هایی که به دلیل ضعف سیستم اعلام نمی‌شوند، امنیت سیستم کشف نفوذ را زیر سئوال خواهند برد. به طور کلی واکنشهای یک سیستم تشخیص نفوذ در قبال رویداد های مختلف سیستم را می توان به چهار دسته ارائه شده در شکل ۲-۳ تقسیم نمود، شکل گویای همه مطالب هست و ما از توضیح اضافه در این مورد می پرهیزیم. در فصل های آتی مجددا به این تعاریف رجوع خواهیم کرد.



شکل ۲-۳: وضعیت های ممکن در قبال واکنشهای یک سیستم تشخیص نفوذ

در پایان این بخش اشاره به فعالیتهایی که به منظور مقایسه و ارزیابی سیستم های تشخیص نفوذ انجام گرفته است خالی از لطف نیست. آزمایشگاه لینکلن در دانشگاه MIT، تحت حمایت آژانس پروژه تحقیقاتی پیشرفته[۱] دفاع (DARPA) و آزمایشگاه تحقیقاتی نیروی هوایی (AFRL/SNHS) ، اولین مجموعه داده برای ارزیابی سیستم های تشخیص را تولید و توزیع کرده است[۷,۲۴]. قبل از این رویداد هیچ مجموعه داده معتبری برای مقایسه سیستم های تشخیص نفوذ وجود نداشت و به این ترتیب مبنایی برای مقایسه سیستم ها تشخیص نفوذ به وجود آمد که به آن مجموعه داده های DARPA می گویند[۱۸]. بستری که داده های مورد نظر از آن جمع آوری شد شامل دو بخش داخلی و خارجی است که بخش داخلی در حقیقت بخشی از شبکه LAN پایگاه نیروی هوایی است که شبیه سازی شده است و شامل چهار ماشین می باشد که این ماشینها مقصد حملات مختلف در این ارزیابی هستند. بخش خارجی یک ترافیک اینترنتی شبیه سازی شده شامل صد ها ماشین است. داده ها در این مجموعه حاوی اطلاعات بسته های TCP می باشد که بین میزبانهای مختلف در قسمت داخلی یا خارجی از شبکه رد و بدل شده اند. این داده ها شامل اطلاعات سه هفته ای آموزشی که دو هفته آن ترافیک زمینه بدون هیچگونه حمله و نفوذ است و یک هفته آن ترافیک زمینه به همراه تعداد کمی از حملات است. مکان و موقعیت حملات در داده های آموزشی به وضوح بر چسب گذاری شده است. این مجموعه همین طور شامل اطلاعات دو هفته دیگرمی باشد، که به منظور تست بدون هیچ بر چسبی ارائه شده اند. البته مجموعه داده های تست حاوی برچسب حملات نیز وجود دارد که کارایی سیستم ها به وسیله آن می تواند مورد ارزیابی قرار گیرد.

[۱] Defense Advanced Research Project Agency

بعد ها پنجمین کنفرانس سراسری کشف دانش و داده کاوی[۱] ACM SIGKDD به منظور برگزاری یک مسابقه در زمینه سیستم های یادگیری ماشین، داده های TCP جمع آوری شده در مجموعه DARPA را به فرم یک مجموعه آموزش و آزمایش[۲] شامل خصیصه[۳] های بدست آمده برای رکورد های اتصال[۴] جمع آوری و تولید کرد. هدف اصلی از این مسابقه انتخاب طبقه بندی کننده[۵] با بیشترین توانایی و کیفیت در تشخیص اتصالات نرمال و نفوذی بود. این مجموعه داده، مجموعه داده های ارزیابی KDD cup 99 یا به اختصار KDD'99 نامیده می شود و در این مطالعه برای ارزیابی و انجام آزمایشات سیستم ارائه شده مورد استفاده قرار گرفته است.در بخش بعدی این مجموعه داده ای با جزئیات بیشتر مورد بررسی قرار خواهد گرفت.

۳ – ۳ مجموعه داده های *KDD'99*

مجموعه داده های KDD cup 99 شامل ۴۱ خصیصه استخراج شده برای هر اتصال استخراج می باشد که یک برچسب وضعیت اتصال را که نرمال یا یک حمله از کلاسی خاص است، مشخص می کند. لیست این خصیصه ها در ضمیمه ۱ قابل مشاهده است. این خصیصه ها به طور کلی دارای چهار فرم پیوسته[۶]، گسسته[۷] و سمبولیک[۸] با بازه وسیعی از مقادیر هستند که به طور کلی به چهار دسته تقسیم می شوند[۲۰]:

- دسته اول شامل *خصیصه های ذاتی*[۹] یک اتصال هستند، که به نوبه خود شامل خصیصه های پایه اتصالهای TCP است. مدت یک اتصال، نوع پروتکل (TCP, UDP, ...) و نوع سرویس مورد استفاده (telnet , http, ...) نمونه هایی از این خصیصه ها هستند.
- *خصیصه های محتوایی*[۱۰]، بخشی از خصیصه های یک اتصال هستند که از طریق وارسی بخش داده ای بسته های TCP بدست می آیند. به عنوان مثال تعداد login های شکست خورده[۱۱] نمونه ای از این خصیصه ها است.

---

[۱] International Conference on Knowledge Discovery and Data Minig

[۲] Train and Test Set

[۳] Feature

[۴] اتصال (Connection) یک دنباله ار بسته های TCP که در یک زمانهای مشخص شروع می شود و خاتمه می یابد.

[۵] Classifier

[۶] Continuous

[۷] Discrete

[۸] Symbolic

[۹] Intrinsic

[۱۰] Content Feature

[۱۱] Failed Login Attemp

- *خصیصه های میزبان یکسان* ، اتصالاتی را که در دو ثانیه گذشته دارای مقصد یکسان با اتصال جاری بوده اند بررسی می کند و مقادیر آماری که به رفتار پروتکل، سرویس و غیره وابسته است بدست می آورند. مثالی از این خصیصه ها تعداد اتصالات به یک میزبان خاص در دو ثانیه گذشته است.
- خصیصه های *سرویس یکسان مشابه*[۱]، خصیصه هایی هستند که اتصالاتی که در دو ثانیه گذشته سرویس یکسانی با اتصال جاری دارند را بررسی می کند. و مقادیر آماری خاصی را برای آنها محاسبه می کنند. تعداد اتصالات با سرویس یکسان با اتصال جاری در دو ثانیه گذشته نمونه ای از این خصیصه هاست.

به همین ترتیب حملات در این مجموعه به چهار دسته اصلی تقسیم می شوند[۲۰]:

- حملات جلوگیری از سرویس DoS[۲]: این دسته از حملات سعی می کنند منابع محاسباتی یا حافظه را آنقدر مشغول کنند تا کابران نتوانند از حقوق مشروع خود برای استفاده از این منابع بهره گیرند. Smurf نمونه ای از این حملات است که در آن مهاجم از بسته های درخواست انعکاس ICMP[۳] که به آدرس IP پخشی[۴] ارسال می شود برای ایجاد یک حمله جلوگیری از سرویس استفاده می کند.
- حملات کاربر به هسته سیستم U2R[۵]: در این دسته از حملات، مهاجم با استفاده از حفره های امنیتی سیستم به حقوق کاربر ارشد سیستم دسترسی پیدا می کند و فعالیتهای غیر قانونی اش را انجام می دهد. حمله xterm یک نمونه از حملات این گروه است که از سرریز بافرها در کتابخانه Xaw که توسط سیستم عامل Redhat5 ارائه شده است استفاده می کند و به این ترتیب می تواند دستورات دلخواه با استفاده از حقوق کاربر ارشد اجرا کند.
- حملات سیستم دور به ماشین محلی R2L[۶]: دسترسی غیر مجاز از طریق حفره های امنیتی به یک ماشین با استفاده از یک ماشین راه دور، در این دسته قرار می گیرد، حملات Dictionary نمونه ای است که در آن مهاجم سعی می کند با تکرارهای مختلف و بررسی احتمالات نام کاربری و کلمه عبور یک ماشین دیگر را از راه دور بدست آورد و به آن نفوذ کند.
- حملات پویشی[۷]: در این دسته از حملات مهاجم به بررسی میزبان ها و پرت های مختلف می پردازد تا بتواند حفره های امنیتی آنها را کشف کند و در اصل این دسته از حملات زمینه سازی برای حملات دیگر هستند. Nmap نمونه ای مناسب برای ارائه در مورد این دسته از حملات است که در حقیقت یک ابزار همه منظوره برای انجام بررسی های مختلف در سطح شبکه است.

---

۱ Similar Same Service
۲ Denial of Service
۳ Echo
۴ Broadcast IP
۵ User to Root
۶ Remote to Local
۷ Probe

جدول ۱-۳ حملات موجود در مجموعه داده های KDD را بر اساس هر یک از کلاس های چهارگانه فوق نشان می دهد.

جدول ۱-۳: دسته بندی حملات موجود در مجموعه داده های KDD'99

DOS	**apache2**, back, land, **mailbomb**, neptune, pod, **proccesstable**, smurf, teardrop, **udpstorm**
U2R	buffer_overflow, **httptunnel, ps**, loadmodule, Multihop, perl, rootkit, **sqlattack**, **xterm**
R2L	ftp_write, guess_password, imap, **named**, phf, **sendmail**, **snmpgetattack, snmpguess**, spy, warezclient, warezmaster, **worm, xlock**, **xsnoop**
Probe	ipsweep, **mscan**, nmap, portsweep, **saint**, satan

مجموعه داده های KDD به مجموعه رکوردهای آموزشی و آزمایشی (تست) تقسیم بندی می شوند. تعداد کل رکورد های اتصال در مجموعه داده های آموزشی در حدود پنج میلیون رکورد می باشد. این تعداد رکورد برای اهدافی که ما در این مطالعه دنبال می کنیم، بسیار زیاد است. بنابراین ما مجموعه کوچکتری از داده های KDD را که مجموعه داده های ۱۰٪ نامیده می شود و در اصل خلاصه ای از داده های آموزشی KDD cup 99 است را در اینجا استفاده می کنیم. توزیع رکورد های نرمال و چهار کلاس حمله در این زیر مجموعه ۱۰ درصدی در جدول ۲-۳ ارائه شده است.

همانطور که از اطلاعات جدول پیداست توزیع نمونه ها در کلاس های مختلف به طور قابل ملاحظه ای با هم متفاوت است. به عنوان مثال در حالیکه کلاس حمله Dos تقریبا" حاوی چهارصد هزار نمونه است، کلاس U2R بیش از۵۲ نمونه ندارد. لازم بذکر است که این تفاوت آموزش یک طبقه بندی کننده را بسیار مشکل می کند. یکی از مهمترین تلاش های ما در ارائه پایاننامه حاضر، فائق آمدن بر مشکلات ناشی از این تفاوتهای چشمگیر با استفاده از طبقه بندی کننده های مختلف برای هر کلاس است.

جدول ۲-۳: توزیع الگوها در زیر مجموعه ۱۰٪ داده های KDD'99

کلاس	تعداد الگوها	درصد الگوها
Normal	۹۷۲۷۷	٪۱۹/٦۹
Probe	٤۱۰۷	٪۰/۸۳
DoS	۳۹۱٤۵۸	٪۷۹/۲٤
U2R	۵۲	٪۰/۰۱

R2L	۱۱۲٦	٪۰/۲۳
	٤۹۲۰۲۱	٪۱۰۰

داده های تست در مجموعه KDD در مقایسه با مجموعه آموزشی توزیع متفاوتی دارند، علاوه بر این داده های تست حاوی حملات اضافه ای هستند که در داده های آموزشی وجود ندارند. وجود حملات[۱] جدید و توزیع متفاوت داده های تست عملیات طبقه بندی را بسیار پیچیده می کند. این دسته از حملات جدید در جدول ۱-۳ با حروف پر رنگ نوشته شده اند. جدول ۳-۳ و ٤-۳ به ترتیب توزیع حملات و رکورد های اتصال نرمال را در مجموعه داده های تست KDD'99 و توزیع حملات جدید براساس کلاس اصلی داده های حمله در مجموعه تست را نشان می دهند. داده های جدول ٤-۳ بیان گر توزیع غیر یکنواخت این حملات در کلاس های مختلف حمله است. به عنوان مثال در مجموعه تست ۱۸۹ الگو از ۲۲۸ الگو موجود در کلاس U2R حملاتی هستند که هیچ الگو مشابهی برای این حملات در مجموعه داده های آموزشی وجود ندارد.

جدول ۳-۳: توزیع الگوها در زیر مجموعه تست (آزمایش) دارای برچسب داده های KDD'99

کلاس	تعداد الگوها	در صد الگوها
Normal	٦۰٥۹۳	٪۱۹/۴۸
Probe	٤۱٦٦	٪۱/۳٤
DoS	۲۲۹۸٥۳	٪۷۳/۹۰
U2R	۲۲۸	٪۰/۰۷
R2L	۱٦۱۸۹	٪٥/۲۰
	۳۱۱۰۲۹	٪۱۰۰

جدول ۴-۳: توزیع الگوهای حملات جدید در زیر مجموعه تست (آزمایش) دارای برچسب داده های KDD'99

کلاس	تعداد الگوهای حملات جدید	تعداد کل الگوهای موجود	درصد الگوها
Probe	۱۷۸۹	٤۱٦٦	٪٤۳
DoS	٦٥٥٥	۲۲۹۸٥۳	٪۳
U2R	۱۸۹	۲۲۸	٪۸۳
R2L	۱۰۱۹٦	۱٦۱۸۹	٪٦۳
	۱۸۷۲۹	۲٥۰٤۳٦	٪۷٥

[۱] Novel Attacks

مجموعه داده های KDD'99 که دارای۴۱ خصیصه برای هر رکورد اتصال هستند و برای هر رکورد اتصال یک برچسب که مشخص کننده نوع آن رکورد می باشند ارائه می کند، در این مطالعه مبنایی برای مقایسه و انجام آزمایشات است و تمامی آزمایشات ارائه شده در این مطالعه بر اساس این مجموعه رکوردهای حاوی خصیصه های استخراج شده برای اتصال انجام گرفته است.

۴ – ۳ کارهای مرتبط با مجموعه داده های *KDD*

در مسابقه KDD cup 99، ۲۴ شرکت کننده حضور داشتند. سه شرکت کننده برتر همگی از مدلهای مختلف درخت تصمیم استفاده کرده اند. مقام اول [۳۱] از یک مجموعه درخت C5 برای تشخیص استفاده می کند ، که شامل روش Bagged Boosting حساس به هزینه است. در این فرایند، ۵۰ نمونه آموزشی از مجموعه آموزشی اصلی استخراج شده است. نمونه های انتخابی شامل همه الگو های دو کلاس کوچکتر U2R و R2L و ۴۰۰۰ الگو غیر تکراری از کلاس PROBE، ۸۰۰۰۰ الگو غیر تکراری نرمال و ۴۰۰۰۰۰ الگو غیر تکراری در کلاس DOS می باشد. برای هر نمونه ۱۰ درخت تصمیم که از هزینه خطا و گزینه های boosting استفاده می کند ساخته شده است. در نهایت پیش بینی لازم بر اساس ۱۰×۵۰ درخت ایجاد شده انجام گرفته است. هر درخت C5 در زمانی نزدیک به یک ساعت بر روی یک پردازنده دو هسته (2×300Mhz) با 512MB حافظه اصلی و 9GB دیسک انجام گرفته است.

شرکت کننده که در مکان بعدی قرار گرفت نیز از درخت های تصمیم استفاده کرده است[۲۳]. در روشی که این شرکت کننده استفاده می کند مجموعه ای از درخت های تصمیم ایجاد شدند و سپس از یک روش بهینه سازی ضابطه مند خاص برای انتخاب مجموعه ای بهینه از درختهای اولیه که وظیفه تشخیص و پیشبینی نهایی را بر عهده دارند، استفاده شد. آموزش بر اساس مجموعه داده های ۱۰٪ انجام گرفته است که به بخشهای،که این بخش ها به یکدیگر وابسته نیستند، تقسیم شدند و از یک روش بهینه سازی برای انتخاب زیر مجموعه ای از درختها، که وظیفه پیش بینی نهایی را بر عهده دارند، استفاده می شود. این روش بهینه سازی طوری طراحی شده است که هزینه کلی طبقه بندی نادرست را مینیم می کند در حالیکه قابلیت اطمینان و ثبات پیش بینی را نیز تضمین می کند. سخت افزار استفاده شده در این روش شامل یک کامپیوتر پنتویم دو 350Mhz با 128MB حافظه اصلی بوده است و فرایند تشخیص الگو تقریباً ۲۲ ساعت زمان برده است.

روش وارده مقام سوم، شامل درخت های تصمیم دو لایه است[۳۶]. لایه اول بر اساس رکورد های اتصالی که یک فرد خبره در مسائل حفاظتی با نگاه کردن به آنها نمی تواند به راحتی حمله بودن آنها را تشخیص دهد آموزش می یابد، در حالیکه لایه دوم براساس اتصالاتی که توسط لایه اول طبقه بندی نمی شوند ساخته شده است. مجموعه داده آموزشی مجدداً از زیر مجموعه ۱۰٪ انتخاب شده اند و تعدادی از اتصالات Normal و DOS به صورت تصادفی حذف شده اند. سپس این مجموعه به صورت تصادفی به سه نمونه تقسیم شدند: ۲۵٪ برای تولید درخت، ۲۵٪ برای تنظیم درخت، و ۵۰٪ باقیمانده برای تخمین کیفیت مدل.

کامپیوترهایی که استفاده شد یک پنتیوم رو میزی 150Mhz با 32MB حافظه اصلی و 200MB فضای خالی بر روی دیسک به همراه یک پنتیوم 133Mhz با 40MB حافظه اصلی و در حدود 200MB فضای خالی بر روی دیسک بودند. برای تولید هر درخت ۳۰ دقیقه زمان لازم بود و عملیات چیزی درحدود ۶ ساعت زمان برده است.

بعد ها روشهای طبقه بندی دیگری برای حل مساله ارائه شده در KDD cup 99 پدیدار شدند. یکی از موفقترین این روشها که توسط Lee و Stolfo [۲۲] ارائه شد براساس یک چهارچوبه داده کاوی که از قوانین Ripper استفاده می کند، عمل می کند. الگوریتم های قوانین شرکت پذیری و وقایع تکراری به ترتیب برای استخراج ارتباط بین خصیصه ها و بیان توالی رکورد های اتصال در این روش استفاده شده اند. در این چهار چوبه یک سیستم طبقه بندی کننده با قوانین RIPPER برای طبقه بندی اتصلات به ۵ گروه استفاده شده است. RIPPERمشخصه هایی از ۴۱ مشخصه داده های موجود در KDD را جستجو می کند، که بیشترین تمایز را دارند و از آنها برای ایجاد قوانین استفاده می کند. قوانین شرکت پذیری برای استخراج ارتباط بین ۴۱ خصیصه رکورد های اتصال استفاده می شود و الگوریتم وقایع تکراری توالی رکوردهای اتصال را می یابد.

یکی دیگر از این روشها چهارچوبه ای است که Agarwal و Joshi برای یادگیری یک مدل مبتنی بر قانون به نام PNrule ارائه کرده اند[۳]، که طبقه بندی کننده ای مناسب برای مجموعه داده هایی است که توزیع داده ها در کلاسهای مختلف این داده ها خیلی متفاوت است. چهار چوبه PNrule یک فرایند دو مرحله ای شامل استخراج قوانین از داده های آموزشی با بیشترین پوشش و دقت و سپس بدست آوردن قوانینی برای حذف داده هایی که به اشتباه توسط گروه قوانین اولیه وارد شده اند، می باشد. چندین طبقه بندی کننده دو گانه[۱] هر کدام برای یک کلاس از داده ها آموزش داده می شوند. برای هر کلاس دو نوع از قوانین استفاده می شود قوانین P و قوانین N. قوانین P وجود داده های کلاس مقصد را بیان می کنند در حالیکه قوانین N داده هایی را به کلاس تعلق ندارد می یابد.

علاوه بر روشهای اشاره شده کار هایی وجود دارند که در آنها کارایی الگوریتم های مختلف یادگیری ماشین و تکنیک های طبقه بندی الگو گوناگون را روی مجموعه داده های KDD cup 99 مقایسه کرده اند. Sabhnani و Serpen کارایی مجموعه کاملی از روشهای تشخیص الگو و یادگیری ماشین را بر اساس مجموعه داده های فوق ارزیابی می کنند. نتایج نشان می دهد که بعضی از طبقه کننده ها برای برخی از کلاس ها یا گروه های حملات بهتر عمل می کنند و این در حقیقت انگیزه ای برای آنها می شود تا یک مدل طبقه بندی چند گانه[۲] را ارائه کنند که برای هر کلاس مشخص از حملات داده های KDD از طبقه بندی کننده های مختلفی استفاده می کند[۳۲]. اخیراً روشهای محاسبات نرم برای ایجاد سیستم های تشخیص نفوذ مورد

---

[۱] Binary-Classifier

[۲] Multi-classifier

توجه قرار گرفته اند. بعضی از این روشها به شکل های گوناگون روی داده های KDD مورد ارزیابی قرار گرفته اند. طبقه بندی کننده های مبتی بر پایگاه قوانین فازی، درخت های تصمیم، ماشین های تکیه گاه برداری، برنامه نویسی ژنتیک خطی توسط Abraham و Jain در [۲] مقایسه شده اند که هدف اصلی روشن ساختن اهمیت الگو های محاسبات نرم برای مدل کردن سیستم های تشخیص نفوذ است.

Abadeh و همکارانش یک الگوریتم یادگیری ژنتیک- فازی را در [۱] ارائه کرده اند. نویسندگان این مطلب مطرح می کنند که این الگوریتم برای تشخیص نفوذ در شبکه مناسب است و آزمایشات آنها روی داده های KDD انجام شده است. کار دیگری که از الگوریتم ژنتیک برای اضافه کردن قابلیت یادگیری به قوانین فازی استفاده می کند و سپس این قوانین تطبیق پذیر را برای تشخیص نفوذ استفاده می کند کاری است که توسط Gomez و Dasgupta انجام گرفته است[۱۳]. برنامه سازی ژنتیک بر اساس الگوریتم RSS-DSS برای فیلتر کردن پویا مجموعه داده ها تکنیک دیگری است که نتایج قابل توجهی را ارائه می دهد و در این زمینه کاری انجام گرفته است[۳٤]. البته لازم به ذکر است که این مدل فقط مشخص می کند که رکورد جاری یک حمله است یا نه و مشخص نمی سازد که رکورد های حملات به کدام گروه حمله ای خاص تعلق دارند.

Yeung و Chow افرادی هستند که از یک روش جدید تشخیص، که براساس یک تخمین گر Parzen-window با هسته گوسی و تخمین تراکم غیر پارامتریک ایجاد شده است، برای ساخت یک سیستم کشف نفوذ تشخیص آنامولی استفاده کرده اند[۳۸]. این مدل همانند روش برنامه سازی ژنتیک بر اساس RSS-DSS ارائه شده در [۳٤] به این علت که یک سیستم تشخیص آنامولی است فقط می تواند حمله یا عدم حمله بودن رکوردها را مشخص می سازد.

به نظر می رسد اشاره به نمونه هایی از مقالاتی که از جنبه های مختلف به نقد کردن مجموعه داده های ایجاد شده توسط DARPA پرداخته اند خالی از لطف نباشد[۱۸, ۲٦]. McHugh با توجه به اطلاعات ترافیکی جمع آوری شده توسط DARPA ، از عدم هماهنگی آماری ترافیک، پایین بودن نرخ ترافیک نسبت به شبکه های واقعی، توزیع نسبتا یکنواخت چهار کلاس اصلی حملات، توزیع انحرافی میزبان های قربانی[۱]، و ساختار مسطح شبکه به عنوان ضعف های مجموعه داده های DARPA یاد می کند[۱۸]. تحلیل با جزئیات بیشتر که توسط Mahoney و Chan انجام می شود انتقادات McHugh را تایید می کند و نشان می دهد که این مجموعه داده های ارائه شده توسط DARPA برای ارزیابی سیستم های تشخیص نفوذ به طور محسوسی خصوصیات آماری متفاوتی نسب به ترافیک واقعی دارند. نویسندگان این مقاله روشی را پیشنهاد می کنند که این مشکلات را تا حد زیادی تعدیل می کند[۲٦]. در روش آنها داده های جدید دیگری به مجموعه داده های DARPA تزریق می شود تا ترافیک داده های آموزشی به یک شبکه واقعی نزدیک تر گردد. نتایج آزمایشات نشان می دهد تا حد بسیار زیادی این عمل موثر بوده است و مشکلات موجود را مرتفع ساخته است.

[۱] میزبان هایی که مورد حمله قرار گرفته اند.

در این قسمت ما ادعا می کنیم اگر چه روش Mahoney و Chan [۲۶] مشکلات داده های DARPA را تا حدود زیادی بر طرف می کند، اما اعمال این روش بر روی مجموعه داده های KDD که به صورت مجموعه از رکورد های اتصال بدست آمده از داده های DARPA می باشد، کاری بس مشکل و نا ممکن می نماید. علاوه بر این چون ما برای مقایسه کار جاری نیاز به حفظ شرایط آزمایش با دیگر کار های مقایسه شده داریم، بنابراین در این پایاننامه از مجموعه داده های اصلی KDD با توجه به انتقادات موجود بر آن، استفاده می کنیم و این نقایص را برا ی ارزیابی سیستم در نظر داریم، هر چند که هیچ فعالیت خاصی را در قبال بر خورد با آن انجام نمی دهیم.

۵ – ۳ مقدمه ای بر محاسبات نرم و روش های یادگیری ماشین در سیستم های تشخیص نفوذ

اگر بخواهیم تعریفی مناسب برای محاسبات نرم ارائه دهیم می توان گفت محاسبات نرم مجموعه ای از روشهای ابتکاری هستند که یک سیستم محاسبانی هوشمند را به وجود می آورند که این سیستم توانایی شگفت آور ذهن انسان را برای استدلال کردن و یادگیری در یک محیط نا معلوم و بدون قطعیت را دارا می باشد[۳۹]. محاسبات نرم مجموعه ای از چندین نمونه محاسباتی شامل شبکه های عصبی، مجموعه های فازی و استناج فازی، استدلال تقریبی[۱]، الگوریتم های ژنتیک و آب دادن فولاد شبیه سازی شده[۲] و ... می باشد. هر کدام از این اجزا اصلی توانایی های خاص خود را دارند، جدول ۵-۳ اهم توانایی های هر یک از این روشها را نشان می دهد.

جدول ۵-۳: توانایی های متد های مختلف محاسبات نرم

توانایی	روش
یادگیری و تطبیق پذیری	شبکه عصبی
بیان دانش با استفاده از قوانین فازی if - then	تئوری مجموعه های فازی
جستجوی تصادفی سیستماتیک	الگوریتم های ژنتیک

روشهای محاسبات نرم زیادی در زمینه تشخیص نفوذ تاکنون به کار گرفته شده اند[۲,۵,۱۳,۳۴,۴۰]. شبکه های عصبی در این میان نقش بسزایی دارند. HIDE نمونه‌ای از این سیستم‌هاست که بر اساس پیش پردازش داده‌های ترافیک شبکه و طبقه بندی آن از طریق شبکه‌های عصبی، رفتار نرمال شبکه را بدست می‌آورد[۴۰].پردازش شبکه‌های عصبی شامل دو مرحله است در مرحله اول شبکه عصبی با داده‌های نمونه‌ای

[۱] Approximate Reasoning

[۲] Simulated Anealing

که نشان دهندهٔ رفتار عادی کاربر هستند آموزش داده می‌شود و در مرحله دوم شبکه عصبی داده‌های مربوط به فعالیت کاربر را دریافت کرده و تعیین می‌کند که این رفتار به چه میزان با نمونه‌هایی که توسط آنها آموزش داده شده است، مشابهت دارد.

علاوه بر شبکه های عصبی تئوری مجموعه های فازی به عنوان یک روش محاسبات نرم قدرتمند، توانایی خود را در زمینه سیستم های تشخیص نفوذ به اثبات رسانده است[۱, ۵, ۱۰,۱۱,۱۳]. سیستم های فازی ویژگی های مهمی دارند که آنها را برای تشخیص نفوذ مناسب می سازد. یکی از اصلترین مشخصه ها این است که یک سیستم فازی می تواند به آسانی ورودی های از منابع مختلف دریافت می کند ترکیب کند. علاوه بر این بسیاری از نفوذ ها را نمی توان به طور مطلق تشخیص داد و همچنین درجه هشداری که برای یک نفوذ باید صادر شود اغلب فازی است[۱۰].

اکثر سیستم های فازی از یک فرد خبره برای ایجاد پایگاه داده قوانین فازی خود استفاده می کنند. اغلب در این سیستم ها فردی که آشنایی کاملی با سیستم دارد، مجموعه ای از قوانین حسی خود را با استفاده از قوانین if-then فازی بیان می کند و سپس این قوانین در پایگاه دانش قوانین فازی قرار می گیرد و سیستم فعالیتهای خود را بر اساس این قوانین انجام می دهد. اما کسب دانش از متخصصین به دلایل متعددی سخت، مقارن با اشتباه و یک پروسه وقت گیر و تکراری است. علاوه بر این سیستم های فازی معمول تطبیق پذیر نیستند بدین معنا که پس از انکه قوانین فازی در پایگاه قوانین قرار داده شدند این قوانین تغییری نخواهند کرد و در صورتیکه سیستم با وضعیت های جدیدی رو به رو شود نه تنها قادر نیست قوانین جدید را به پایگاه قوانین اضافه کند، بلکه قادر به تغییر قوانین موجود نیز نمی باشد. بنابراین ساخت یک سیستم فازی با قابلیت های یادگیری و تطبیق پذیری اخیراً بسیار مورد توجه قرار گرفته است[۱]. روشهای مختلفی برای تولید و تنظیم خودکار قوانین فازی بدون نیاز به یک فرد خبره ارائه شده است که روشهای فازی- عصبی [۱۹, ۳۰] و ژنتیک- فازی [۲۵, ۱۷] دو مورد از معروف ترین این روشها در این مجال هستند.

مقالات متعددی وجود دارند که از روش های فوق برای سیستم های تشخیص نفوذ استفاده کرده اند که اغلب این سیستم ها براساس روشهای ژنتیک- فازی شکل گرفته اند و از آن جمله می توان به روشی که Gomez و همکارانش در [۱۳] انجام داده اند اشاره نمود. در این روش سعی شده است با استفاده از الگوریتم ژنتیک مجموعه ای از قوانین فازی مناسب به عنوان یک طبقه بندی کننده فازی ایجاد شود که بتواند داده ها را به خوبی طبقه بندی کند. الگوریتم ژنتیک به کار گرفته شده با استفاده از داده های آموزشی قوانین فازی تصادفی تولید می کند و الگوریتم تا فرارسیدن زمانیکه این قوانین فازی بتوانند عملیات تشخیص را به خوبی انجام دهند قوانین را تغییر می دهد تا در نهایت به مجموعه قوانین مناسب برسد. نمونه دیگری که بر این اساس کار می کند در [۱] ارائه شده است که روش ژنتیک- فازی ارائه شده توسط Ishibuchi و همکارانش [۱۷] را برای یک سیستم تشخیص نفوذ به کار گرفته است. در این روش هر قانون فازی به صورت یک رشته کد می شود و مجموعه ای از پنج نشانه خاص، به عنوان متغیرهای زبانی در نظر گرفته می شود که این سمبول ها

در قوانین فازی کد شده قرار دارند. سیستم به این صورت آموزش می بیند که ابتدا یک سری قوانین اولیه در پایگاه قوانین فازی قرار می گیرد سپس این قوانین بر اساس داده های آموزشی ارزیابی می شوند. با استفاده از عملگر های مختلف ژنتیک قوانین فازی جدیدی ایجاد می شوند و به جمعیت حاضر قوانین اضافه می گردند و به این ترتیب گروهی از قوانین جدید جایگزین قوانین موجود می شوند. در نهایت پس از اینکه الگوریتم به سطح مناسبی از آموزش رسید فعالیت آموزش متوقف می شود و از قوانین حاصله برای تشخیص نفوذ در داده های تست استفاده می شود.

نمونه های کمتری از روشهای فازی- عصبی استفاده نموده اند و از جمله کار هایی که در این زمینه انجام شده است سیستم NFIDS است[۲۸]. این سیستم که سیستم کشف نفوذ بر اساس تشخیص آنامولی (رفتار غیر عادی) است که از شبکه عصبی و منطق فازی برای تشخیص فعالیتهای مخرب در شبکه استفاده می کند. این کار که در دانشگاه تهران انجام شده است بر روی داده های بدست آمده از یک شبکه محلی در دانشگاه ارزیابی شده است. NFIDS یک سیستم سلسله مراتبی است که از سه لایه تشکیل شده است. لایه اول شامل چندین عامل[۱] تشخیص نفوذ است که این عامل ها وظیفه مانیتور کردن فعالیتهای یک میزبان یا یک شبکه را بر عهده دارند و فعالیتهای غیر عادی را به لایه دوم گزارش می کنند. عامل های لایه دوم گزارش دریافتی از لایه اول را با وضعیت کلی ترافیک شبکه ای که آنها بررسی می کنند تطبیق داده و یک گزارش به لایه بالاتر ارسال می کنند. لایه بعدی یک گزارش سطح بالاتر را با داده های مربوطه و اخطاری که به رابط کاربر ارسال می شود ترکیب می کند. قسمت اصلی این سیستم یک ماژول تصمیم گیری است که از شبکه عصبی و منطق فازی برای تشخیص نفوذ استفاده می کند. این سیستم هر چند توسط نویسندگان به عنوان یک سیستم فازی- عصبی معرفی شده است، اما در حقیقت با آنچه مدنظر ما است، یعنی اضافه نمودن قابلیت یادگیری و تطبیق پذیری به یک سیستم فازی تا حد زیادی متفاوت است.

سیستم فازی- عصبی دیگری که در سال ۲۰۰۴ توسط یک گروه ۶ نفره ارائه شده است کاری است که از SNORT[۲] برای تحلیل بلادرنگ ترافیک و بسته های وارده به یک شبکه IP در فاز آموزش سیستم استفاده کرده است. این سیستم یک سیستم تشخیص سو استفاده است که یک سیستم استنتاج فازی از نوع ممدانی را با یک شبکه عصبی ترکیب کرده است. نود های خروجی شبکه عصبی در حقیقت توابع عضویت سیستم فازی را نمایش می دهند و در طول فاز آموزش می توانند تغییر کنند[۵].

همانطور که قبلا اشاره کردیم سیستم های فازی- عصبی کمی برای تشخیص نفوذ در شبکه به کار گرفته شده اند، این در حقیقت انگیزه بود که ما مدل های فازی- عصبی دیگری را برای ایجاد یک سیستم تشخیص نفوذ به کاربگیریم. یکی از معروف ترین مدل های فازی- عصبی که توسط Jang ارائه شده است ANFIS[۳] یا

---

[۱] Agent

[۲] Snort یک سیستم تهاجم‌یاب سبک مبتنی بر UNIX می‌باشد که برای استفاده در شبکه‌های کوچک و متوسط مناسب می‌باشد.

[۳] Adaptive Neuro-Fuzzy Inference System

سیستم استنتاج فازی- عصبی تطبیق پذیر نام دارد[۱۹]. در چهارچوبه ارائه شده در این پایانامه از این مدل فازی- عصبی به عنوان یک طبقه بندی کننده استفاده شده است. در ادامه این فصل و در بخش های بعدی مدل فازی- عصبی فوق را با جزئیات بیشتر بررسی می کنیم.

۶ - ۳ سیستم های فازی

پیش از آنکه به جزئیات سیستم استنتاج فازی- عصبی تطبیق پذیر[۱۹] بپردازیم ضروری به نظر می رسد که مروری بر سیستم های فازی داشته باشیم. سالهای اخیر گواهی بر گسترش سریع برنامه های کاربردی و سیستم های فازی از منظر تعداد و تنوع می باشد. در میان متد های محاسبات نرم منطق فازی بیش از همه به کار گرفته شده است و یک از متدهایی است که بیش از همه مورد توجه است. ساختار اصلی یک سیستم استنتاج فازی شامل مدلی است که مشخصات یک ورودی را به توابع عضویت[۱] ورودی نگاشت می کند. انواع شناخته شده ای از سیستم های استنتاج فازی وجود دارند. مدل فازی ممدانی[۲] [۲۷] از اولین گام هایی بود که تلاش می کرد یک ورودی را بر اساس تجربه یک فرد خبره به یک فضای خروجی نگاشت می کرد. مثالی از مدل فازی ممدانی با دو قانون برای دو ورودی و یک خروجی در ادامه آمده است:

if x *is* A1 *and* y *is* B1 *then* Z *is* C1.
if x *is* A2 *and* y *is* B2 *then* Z *is* C2.

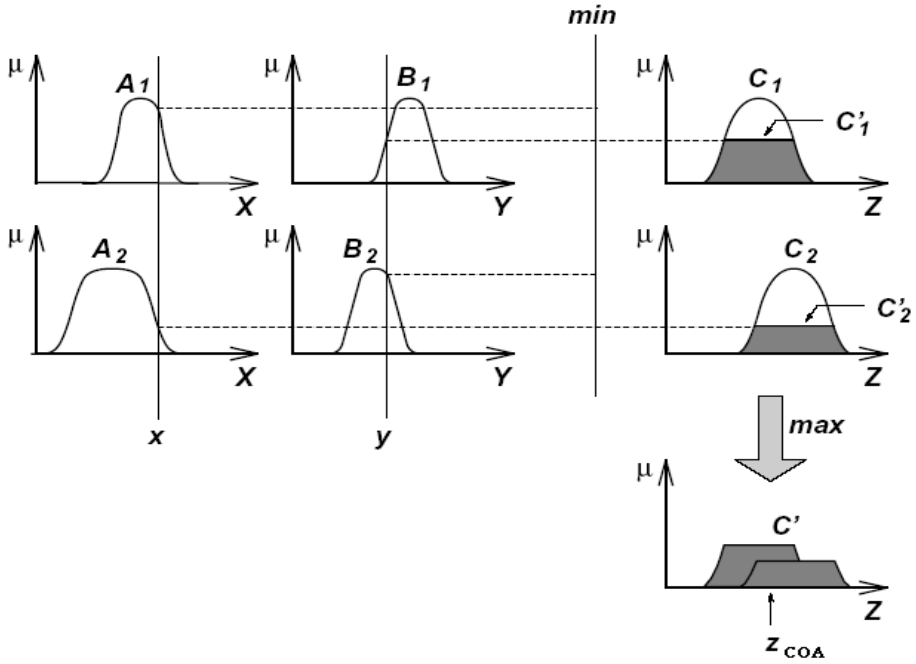
A و B مجومه های فازی ورودی با توابع عضویت A1 ، A2 و B1 ، B2 هستند و C مجموعه فازی خروجی می باشد.

ماکزیمم و مینیمم به ترتیب به عنوان عملگر های T- Norm و T-conorm در نظر گرفته شده اند. سیستم استنتاج فازی حاصل در شکل ۳-۳ نمایش داده شده است. خواننده محترم برای اطلاعات بیشتر در مورد T-Norm و T-conorm می تواند به [۱۹] مراجعه کند.

---

[۱] Membership Functions

[۲] Mamdani Fuzzy Model



شکل ۳-۳ سیستم استنتاج فازی ممدانی با دو ورودی و یک خروجی همراه با دو قانون و min و max به ترتیب به عنوان عملگر T-norm وT-conorm [۱۹]

از آنجایکه سیستم های واقعی از مقادیر دقیق[۱] استفاده می کنند، باید از یک غیر فازی ساز[۲] استفاده کنیم تا مقادیر فازی خروجی سیستم به مقادیر دقیق تبدیل شوند. در حقیقت غیر فازی سازی راهکاری است که یک مقدار دقیق را بر اساس مجموعه فازی خروجی محاسبه می کند[۱۹]. در مثال فوق از غیر فازی ساز مرکز ثقل استفاده شده است. غیر فازی ساز مرکز ثقل به صورت زیر محاسبه می شود.

$$Z_{COA} = \frac{\int_Z \mu_A(z) z dz}{\int_Z \mu_A(z) dz} \qquad (۳-۱)$$

که در آن $\mu_A(z)$ تابع عضویت خروجی بدست آمده است.

لازم به ذکر است که ما از سیستم استنتاج فازی ممدانی به عنوان یک ماژول تصمیم گیرنده نهایی در چارچوبه ارائه شده در این پایاننامه استفاده کرده ایم جزئیات بیشتر در مورد ساختار سیستم و موتور تصمیم در فصل های بعدی ارائه خواهد شد.

تلاش برای ساخت یک روش سیستماتیک به منظور تولید قوانین فازی بر اساس یک مجموعه داده که شامل زوج های ورودی- خروجی می باشد منجر به ایجاد مدل فازی TSK[۳] [۳۵] شد. که به مدل فازی Sugeno مشهور است. قوانین فازی در مدل Sugeno به شکل زیر است:

---

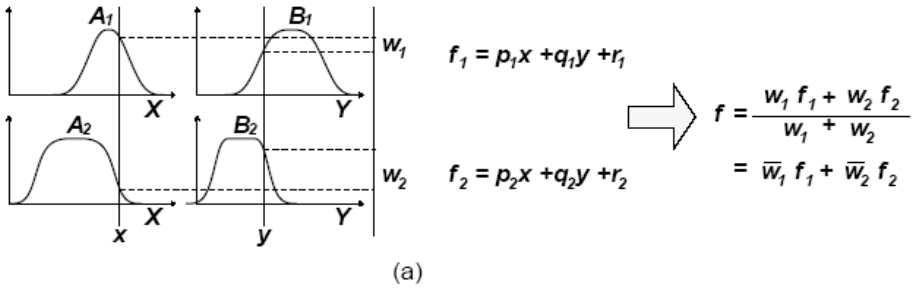
[۱] Crisp

[۲] Defuzzifier

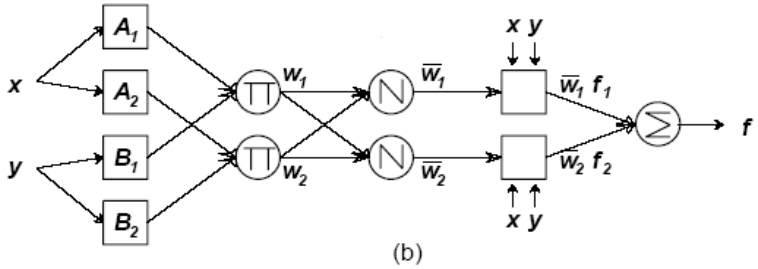
[۳] Takagi-Sugeno-Kang

if x *is* A *and* y *is* B *then* z=f(x, y)

که در آن A و B مجموعه های فازی ورودی و z =f(x, y) یک تابع چند جمله ای درجه یک یا صفر در بخش خروجی می باشد. روال استنتاج برای یک مدل فازی Sugeno از درجه یک در شکل (a)۴-۳ نشان داده شده است. در مدل استنتاج Sugeno برای اجتناب از رویه زمان بر غیر فازی سازی که در مدل ممدانی وجود دارد، این رویه در مدل Sugeno با میانگین گیری وزنی جایگزین شده است، که در شکل قابل مشاهده است.



(a)



(b)

شکل ۴-۳ (a) مدل استنتاج فازی Sugeno (b) ساختار ANFIS معادل. [۱۹]

مدل های دیگری همانند سیستم استنتاج فازی Tsukamoto و چندین مدل دیگر از سیستم های استنتاج فازی نیز وجود دارند که مجالی و فرصتی برای پرداختن به آنها در این پایاننامه وجود ندارد. ANFIS که از آن به عنوان سیستم استنتاج فازی- عصبی تطبیق پذیر یاد کردیم، بر اساس مدل استنتاج Sugeno ایجاد شده است. در بخش بعدی ساختار ANFIS با بررسی جزئیات این مدل شرح داده خواهد شد.

۳ – ۷ سیستم استنتاج فازی- عصبی تطبیق پذیر

معمولا وضعیت های وجود دارند که شخص نمی تواند با نگاه کردن به داده ها تشخیص دهد که توابع عضویت باید چه شکلی داشته باشند. علاوه بر این انتخاب پارامتر هایی توابع عضویت مبهم است و این پارامتر ها هستند که توابع عضویت را برای داده های ورودی و خروجی مناسب می کنند. در حقیقت این همان

موقعیتی است که یادگیری تطبیق پذیر فازی- عصبی که ANFIS با آن آمیخته شده است می تواند بسیار راهگشا باشد.

یک سیستم استنتاج فازی با دو ورودی x,y و یک خروجی z را از نوع مدل فازی Sugeno از درجه یک، در نظر بگیرید. مجموعه قوانین فازی این سیستم با دو قانون فازی در ذیل آمده است:

if x *is* A1 *and* y *is* B1, *then* f1=p1x+q1+r1.
if x *is* A2 *and* y *is* B2, *then* f2=p2x+q2+r2.

شکل (a) ۴-۳ مکانیزم استنتاج را برای این مدل نشان می دهد. همانطور که در شکل (b) ۴-۳ نشان داده شده است، این مکانیزم استنتاج می تواند توسط یک شبکه عصبی پیش خور[۱] با قابلیت یادگیری با سرپرست[۲] که ANFIS نامیده می شود پیاده سازی شود. نود های مربعی، نودهای را نشان می دهند که دارای پارامتر های تطبیق پذیر هستند و نود های دایره نود های با پارامترهای ثابت را نشان می دهند. لایه اول شبکه ANFIS مورد نظر شامل نود های مربعی شکلی است که وظیفه فازی سازی و انتخاب توابع عضویت را بر عهده دارند. پارامتر های این لایه پارامتر های مقدم[۳] نامیده می شوند. در لایه دوم عمل T-norm انجام می شود که انرژی لازم برای آتش کردن هر قانون را فراهم می کند. نسبت میزان آتش[۴] i امین قانون به مجموع میزان آتش تمامی قانون های دیگر که در لایه سوم محاسبه می شود، میزان آتش نرمال شده را تولید می کند. چهارمین لایه شامل نود های مربعی است که ضرب میزان آتش نرمال شده در لایه قبل را در مقدار تابع خروجی برای هر قانون براساس ورودی های مورد نظر انجام می دهد. پارامتر های این لایه پارامتر های تالی نامیده می شوند. خروجی نهایی با جمع تمامی این خروجی های لایه قبلی در لایه پنجم محاسبه می شود. پارامترهای مقدم و تالی برای اساس روشهای آموزش ANFIS قابل تنظیم هستند.

در حقیقت ANFIS روشی را برای یادگیری یک مدل فازی بر اساس مجموعه داده ها ارائه می کند که در آن پارامترهای توابع عضویت ورودی و توابع خروجی به بهترین شکل ممکن بر اساس زوج داده های ورودی- خروجی تنظیم می شوند، به عبارت دیگر این پارامترهای در طول فرآیند آموزش تغییر می کنند. این روش یادگیری دقیقاً مشابه با روش شبکه های عصبی است. ANFIS از روش پس انتشار خطا[۵] یا ترکیت

---

۱ Feed-Forward
۲ Supervised Learning
۳ Premise
۴ Firing Strength
۵ Back Propagation

تخمین حداقل مربع[۱] خطا و پس انتشار خطا برای تنظیم پارامترها استفاده می کند. برای جزئیات بیشتر در مورد ساختار ANFIS و روش های آموزش آن می توانید به [۱۹] مراجعه کنید.

بیان این نکته حائز اهمیت است که ساختار اولیه ANFIS باید تعیین شود و سیستم استنتاج فازی- عصبی تطبیق پذیر مورد نظر ما فقط می تواند پارامتر های توابع عضویت را تنظیم نماید و فعالیت خاصی در قبال ایجاد ساختار کلی سیستم و قوانین فازی انجام نمی دهد. بنابراین سیستم استنتاج فازی باید قبل از آنکه فرآیند آموزش ANFIS آغاز شود به طریقی ایجاد شود. ایجاد این سیستم فازی علاوه بر اینکه می تواند به صورت دستی و با استفاده از دانش یک فرد خبره انجام شود، به دو روش خودکار دیگر امکان پذیر است روش بخش بندی شبکه ای[۲] و خوشه بندی کاهشی[۳]. همانطور که قبلا اشاره شد ما در این سیستم فرض کرده ایم که یک فرد خبره نمی تواند قوانین فازی لازمه را به راحتی و بر اساس مشاهده داده ها ارائه کند بنابراین در این مطالعه از روشهای خودکار استفاده شده است. در بخش بعدی این دو روش خودکار را با تاکید بر خوشه بندی کاهشی که در این پایاننامه استفاده شده است، بررسی خواهیم کرد.

۸ - ۳ خوشه بندی کاهشی

هدف از خوشه بندی شناسایی گروه های طبیعی داده ها از یک مجموعه داده ای بزرگ است که رفتار های مشابه دارند. این گروه ها به اجمال رفتار سیستم را نشان می دهند. این نکته قابل توجه است که می توان از اطلاعات خوشه ها برای تولید سیستم استنتاج فازی Sugeno استفاده کنیم و در حقیقت مدلی که به این ترتیب ساخته می شود رفتار داده ها را با حداقل تعداد قوانین نشان می دهد.

یک مجموعه داده آموزشی دوبعدی شامل ورودی X و خروجی مطلوب Y برای آن ورودی را همراه با یک خوشه خاص از این مجموعه داده، که مرکز آن (xi,yi) می باشد در نظر بگیرید. قانون i ام را می توان به صورت زیر نمایش داد:

if X *is* close to x_i, *then* Y *is* close to y_i.

به این ترتیب می توان با تعیین خوشه ها قوانین فازی لازم را ایجاد کرد. پس از آنکه ساختار قوانین تعیین شد می توان از روش پس انتشار یا گرادیان نزولی یا هر روش بهینه سازی دیگری برای شناسایی پارامترها و تنظیم آنها استفاده نمود.

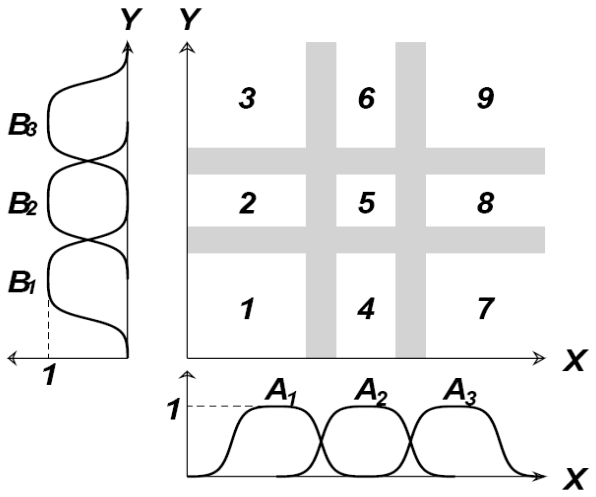
---

[۱] Least Square Estimation

[۲] Gird Partitioning

[۳] Subtractive Clustering

همانطور که قبلا اشاره کردیم با استفاده از دو روش بخش بندی شبکه ای یا خوشه بندی کاهشی می توان بدون نیاز به یک فرد خبره قوانین فازی را ایجاد کرد. در روش بخش بندی شبکه ای تمامی قوانین فازی ممکن بر اساس تعداد توابع عضویت برای هر ورودی که به صورت دستی وارد می شود، تولید می شوند. به عنوان مثال در یک فضای ورودی دو بعدی، با سه تابع عضویت مختلف برای هر مجموعه فازی ورودی، تعداد قوانین فازی که به این طریق ایجاد می شود ۹ قانون می باشد، که به این ترتیب تمامی ترکیبات مختلف برای ایجاد قوانین در نظر گرفته شده است. شکل ۱-۳ شمای کلی از این روش را نشان می دهد. این روش هنگامی که تعداد توابع عضویت و ورودی های سیستم کم است به خوبی عمل می کند. هنگامیکه تعداد ورودی ها و توابع عضویت زیاد می شود به طور قطع این روش پاسخگو نمی باشد زیرا تعداد قوانینی که تولید می شوند بسیار زیاد خواهد بود. همانطور که می دانید ما قصد داریم از مجموعه داده های KDD برای انجام آزمایشات و ارزیابی سیستم استفاده کنیم و از آنجاییکه تمامی ۴۱ خصیصه موجود در این مجموعه داده به عنوان ورودی طبقه بندی کننده های این سیستم استفاده می شود، استفاده از روش بخشبندی شبکه ای امکان پذیر نمی باشد. بنابراین ما از روش خوشه بندی کاهشی برای تعیین تعداد قوانین لازم و توابع عضویت و موقعیت اولیه آنها در این سیستم استفاده می کنیم وسپس از ANFIS برای تنظیم بهتر توابع عضویت استفاده خواهیم نمود.



شکل ۱-۳ بخشبندی شبکه ای در فضای دو بعدی با سه تابع عضویت برای هر ورودی [۱۹]

فرض کنید هیچ ایده واضح و شفافی نسبت به تعداد خوشه های موجود در یک مجموعه داده خاص وجود ندارد. خوشه بندی کاهشی [۶] یک روش سریع و یک گذره است که تعداد خوشه های لازم و مرکز هر خوشه را تخمین می زند. این روش در حقیقت توسعه ای از روش خوشه بندی تپه[۱] [۳۷] که توسط Yager و همکارانش ارائه شده است، می باشد.

[۱] Mountain Clustering

یک مجموعه داده با m نمونه $\{x_1, \ldots, x_m\}$ را در یک فضای N بعدی در نظر بگیرید(هر داده یک بردار N بعدی است). خوشه بندی کاهشی فرض می کند هر داده یک مرکز خوشه بالقوه است، سپس مقیاسی از پتانسیل هر داده بر اساس داده های اطرافش محاسبه می کند. که به آن مقایس انبوهی [۱]می گوییم. مقایس انبوهی برای داده x_j به صورت زیر محاسبه می شود:

$$D_j = \sum_{i=1}^{m} \exp\left(-\frac{|x_j - x_i|^2}{(r_a/2)^2}\right) \qquad (۳-۱)$$

که در آن r_a یک ثابت مثبت است که شعاع همسایگی[۲] را مشخص می کند. الگوریتم داده ایی را که بیشترین مقیاس انبوهی را دارد به عنوان اولین مرکز خوشه انتخاب می کند سپس پتانسیل داده های نزدیک به این مرکز خوشه را از بین می برد، آنگاه داده بعدی که بیشترین پتانسیل را برای مرکز خوشه شدن دارا می باشد(بزرگترین مقایس انبوهی باقیمانده) را به عنوان مرکز خوشه بعدی انتخاب می کند و پتانسیل داده های نزدیک به مرکز خوشه جدید برای مرکز خوشه شدن از بین می برد. فرآیند بدست آوردن یک خوشه جدید و از بین بردن پتانسیل داده های اطراف تا زمانی که پتانسیل همه داده ها از یک حد آستانه پایین تر قرار بگیرد ادامه می یابد. محدوده ای که هر مرکز خوشه در هر بعد از داده ها تحت تاثیر قرار می دهد شعاع خوشه نامیده می شود. شعاع خوشه کوچک موجب می شود که تعداد خوشه های پیدا شده زیاد (تعداد قوانین بیشتر) شوند و برعکس. اطلاعات خوشه ها که به این ترتیب به دست می آیند برای تعیین تعداد قوانین و تعداد توابع عضویت و مکان اولیه آنها در جهت تعیین ساختار سیستم استنتاج فازی به کار می روند.

یکی از مزایای اصلی استفاده از یک روش خوشه بندی برای پیدا کردن قوانین فازی این است که قوانین فازی بدست آمده از این طریق بسیار مناسب تر از قوانین فازی هستند که بدون خوشه بندی بدست آمده اند. در این مطالعه ما از روش خوشه بندی کاهشی برای تعیین قوانین فازی استفاده کرده ایم، که به این ترتیب ساختار مجموعه قوانین فازی بدست آمده برای فضای خصیصه های داده شده بسیار مناسب است.

۹ - ۳ الگوریتم های ژنتیک

الگوریتم ژنتیک یک روش حل مسائل بهینه سازی است که بر اساس فلسفه انتخاب اصلح در طبیعت[۳] ایجاد شده است و این فرایند در حقیقت از سیر تکامل زیستی ناشی شده است[۱۲]. الگوریتم ژنتیک با استفاده از تکرار و عملگر های ژنتیکی ، یک جمعیت[۴] را شامل مجموعه ای از افراد[۵] تغییر می دهد. سپس افرادی از

---

[۱] Density Measure

[۲] Neighborhood Radius

[۳] Natural Selection

[۴] Population

[۵] Individual

جامعه که جواب های بهینه تری را نسبت به دیگر افراد تولید می کنند از نسل جاری انتخاب کرده و از آنها برای تولید فرزندان در نسلهای بعدی استفاده می کند. یک الگوریتم ژنتیک هنگامی متوقف می شود که شرط توقف اتفاق بیفتد. شرط توقف می تواند حداکثر تعداد نسلهای تولید شده یا رسیدن به یک حد آستانه یا هر شرط دیگری باشد.

در جستجوی ژنتیک هر فرد از جامعه حاوی یک کروموزم است که این کروموزم به نوبه خود حاوی تعداد ژن می باشد که خصوصیات آن فرد از جامعه را کد می کند. علاوه بر این هر الگوریتم ژنتیک دارای تابعی به نام تابع شایستگی[۱] است که وظیفه این تابع بدست آوردن مقدارشایستگی،که در حقیقت معیاری است که قصد بهینه کردن آنرا داریم، برای یک کروموزم ورودی ،که به ازاء یک فرد از جامعه به این تابع ارسال شده است، می باشد. به عبارت دیگر تابع شایستگی یک کروموزوم را به عنوان ورودی دریافت می کند سپس ژن های آنرا کد گشایی می کند و بر این اساس مقدار عددی را بدست می آورد که این مقدار بیانگر میزان مناسب یا بهینه بودن این فرد از جامعه است.

سه عملگر ژنتیک معروف در زمینه الگوریتم ژنتیک عملگرهای جهش[۲]، تقاطع[۳]، انتخاب[۴] می باشند. عملگر جهش به طور تصادفی یا براساس یک الگوریتم خاص تعدادی از ژنهای یک کروموزم یا یک فرد از جامعه که برای این عمل انتخاب شده است تغییر می دهد و فرد جدید را ایجاد می کند. در تقاطع دو کروموزوم از قسمت های مختلف شکسته می شوند و ژنهای آنها با یکدیگر تعویض می شود و نمونه جدیدی ایجاد می گردد. این عملگر در حقیقت همانند ازدواج دو فرد با یکدیگر و تولد یک فرزند در نتیجه این ازدواج می باشد. عملگر انتخاب، والدین را برای ایجاد نسل بعدی بر اساس مقادیر خروجی تابع شایستگی برای آنها، می یابد.

۱۰ - ۳ نتیجه گیری

بررسی متون و مطالعه روشهای ارائه شده در زمینه تشخیص نفوذ حاکی از اهمیت روشهای محاسبات نرم در این زمینه تحقیقاتی است. روشهای محاسبات نرم گوناگونی برای تشخیص نفوذ تا کنون ارائه شده است. سیستم های فازی در این میان نقش به سزایی را بر عهده دارند و مطالعات از تمایل هر چه بیشتر برای استفاده از این روش ها در این زمینه خبر می دهد. نکته ضعف اصلی در مورد سیستم های فازی استفاده از فرد خبره برای ارائه قوانین فازی است در حالیکه انجام این عمل در مورد همه سیستم ها امکان پذیر نمی باشد، و علاوه بر این

---

[۱] Fitness Function

[۲] Mutation

[۳] Crossover

[۴] Selection

سیستم های فازی تطبیق پذیر نیستند به این معنی که پس از ایجاد یک سیستم فازی قوانین آن قابل تغییر نیستند و با تغییرات محیط عملیاتی قوانین باید بازنگری شوند. روشهای مختلفی برای الحاق قدرت یادگیری و توانایی انطباق به سیستم های فازی تاکنون ارائه شده است که از آن جمله به روشهای فازی- عصبی می توان اشاره نمود. در این فصل روش فازی- عصبی ارائه شده توسط آقای Jang به عنوان یکی از این روشها مورد بررسی قرار گرفت این روش از جمله روشهای محبوب در این زمینه است که در زمینه های تحقیقاتی گوناگون مورد استفاده قرار گرفته است. انگیزه ما از معرفی این روش در اینجا این است که این متد در زمینه تشخیص نفوذ تاکنون به کار گرفته نشده است. فصلهای بعدی این روش فازی- عصبی را برای حل مساله تشخیص نفوذ معرفی می کند و از آن به عنوان بخشی از چهارچوبه ارائه شده در این پایاننامه استفاده می کند.

فصل ۳: تشخیص نفوذ به روش فازی- عصبی

۱ – ۴ مقدمه

در این فصل سعی شده است ابتدا شبکه فازی- عصبی تطبیق پذیر به عنوان یک طبقه کننده معرفی می شود. در این مدل ابتدا قوانین فازی لازم بدون نیاز به یک فرد خبره و با استفاده از روش خوشه بندی کاهشی بر اساس مجموعه داده های آموزشی انتخاب شده از مجموعه داده های KDD تولید می شود، سپس قوانین فازی بدست آمده برای ایجاد یک شبکه عصبی- فازی تطبیق پذیر به نام ANFIS به کار گرفته شده اند و توابع عضویت در این ساختار با استفاده از داده های آموزشی استخراج شده، تنظیم می شوند. در نهایت مدل بدست آمده به عنوان طبقه بندی کننده برای کلاس بندی داده های آزمایش که همانند داده های آموزش از مجموعه داده های KDD استخراج شده اند به کار گرفته می شود. عدم کفایت یا مناسب بودن ابن مدل برای انجام عملیات طبقه بندی در ادامه این مطالعه مورد بررسی قرار می گیرد. پس از آن در ادامه این فصل، طبقه بندی کننده ارائه شده به دو صورت، یکبار به صورت یک طبقه کننده دو گانه که رکوردهای اتصال را به دو کلاس حمله و غیر حمله تقسیم می کند و بار دیگر به صورت یک طبقه بندی کننده چندگانه که الگوهای ورودی را به پنج کلاس مختلف که شامل چهار کلاس حمله و کلاس داده های نرمال می باشد طبقه بندی میکند. در انتها این دو گونه مختلف از طبقه بندی کننده ارائه شده با هم مقایسه می شوند و نتجه گیری لازم انجام می گیرد.

۲ – ۴ شبکه عصبی- فازی تطبیق پذیر به عنوان طبقه بندی کننده

همانطور که در فصل ۲ اشاره کردیم ۴۱ خصیصه برای هر رکورد اتصال در مجموعه داده های KDD cup 99 وجود دارد. این خصیصه ها سه فرم مختلف پیوسته، گسسته، و سمبولیک هستند که تغییرات نسبتا زیادی در محدوده مقادیر و تفکیک پذیری دارند. روشهای طبقه بندی الگو داده ها داده ها را به این شکل نمی توانند پردازش کنند. بنابراین قبل از ساختن مدل طبقه بندی نیازبه انجام پیش پردازشی روی داده ها خواهیم داشت. پیش پردازش داده ها در این مطالعه در حقیقت شامل تبدیل مقادیر سمبولیک به مقادیر عددی است. خصیصه های سمبولیک مانند انواع پروتکل، سرویس ها و پرچم ها به مقادیر ۰ تا N-1 که در آن N تعداد سمبول ها برای هر خصیصه است نگاشت شده اند. به عنوان مثال خصیصه Protocol_Type با سه سمبول مختلف TCP، UDP، ICMP به سه عدد ۰، ۱، ۲، تبدیل می شود. بقیه خصیصه ها به همان صورت اولیه استفاده شده اند. علاوه بر این الگو ها همچنین به یکی از دو کلاس ۱ برای حمله و ۰ برای نرمال برچسب خورده اند.

سپس ۱۵۰۰۰۰ داده تصادفی از مجموعه ۱۰٪ داده به عنوان داده های آموزشی و ۴۰۰۰۰ رکورد تصادفی از همان مجموعه داده ها که اشتراکی با داده های اولیه نداشتند به عنوان داده های بررسی[1] انتخاب شدند.

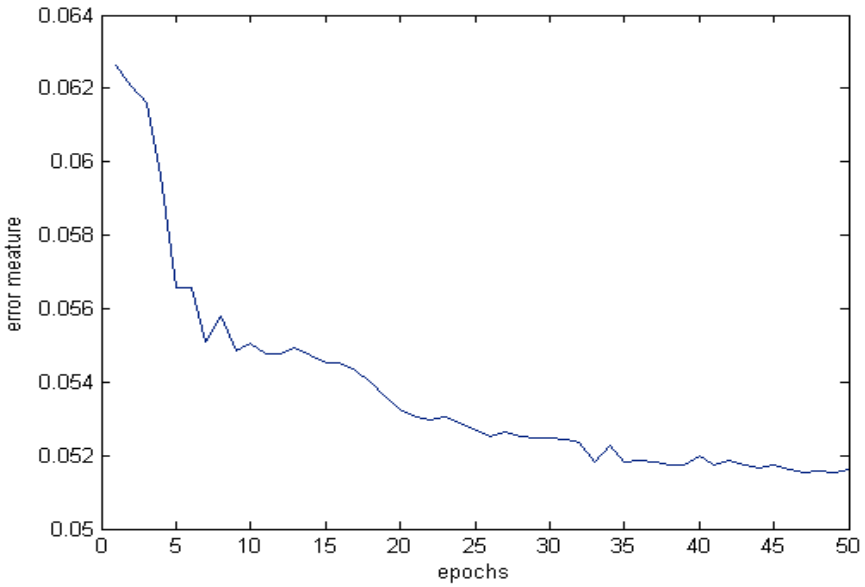
---

[1] Checking data

ایده اصلی برای استفاده از داده های بررسی به منظور معتبرسازی مدل این است که بعد از نقطه خاصی از آموزش مدل شروع به یادگیری بیش از اندازه [۱]می کند. اگر این آموزش بیش از اندازه اتفاق بیفتد، نمی توان انتظار داشت که سیستم فازی مورد نظر نسبت به مجموعه داده های غیر وابسته به داده های آموزشی به خوبی پاسخ دهد. هنگامی که داده های بررسی برای آموزش ANFIS مورد استفاده قرار می گیرند، پس از اتمام فرایند آموزش سیستم استنتاج فازی نهایی با مینیمم خطا برای داده های بررسی به عنوان سیستم نهایی انتخاب می شود.

برای تولید سیستم فازی اولیه از روش خوشه بندی کاهشی با $r_a=0.5$ (شعاع همسایگی ۰/۵) استفاده کردیم. دو قانون فازی و دو تابع عضویت برای هر ورودی به این ترتیب حاصل شد. سپس برای تنظیم بیشتر و تطبیق دادن توابع عضویت با داده های آموزشی از مجموعه داده های آموزشی همراه با داده های بررسی مورد نظر شبکه ANFIS را آموزش دادیم. ساختار ANFISی که برای آموزش استفاده شد ۲۱۲ نود داشت که مجموعا ۲۸۴ پارامتر تطبیق پذیر دارد که ۱۶۴ پارامتر در بخش مقدم و ۸۴ پارامتر در بخش تالی قرار گرفته اند.

مجذور میانگین مربع خطا[۲] (RMSE) برای داده های آموزشی و داده های بررسی بعد از ۵۰ دوره[۳] آموزش به ترتیب ۰/۰۵۱۶ برای داده های آموزشی و ۰/۲۸۳۶ برای داده های بررسی بدست آمد. شکل ۱-۴ مقدار RMSE را به صورت تابعی از دوره برای داده های آموزشی نشان می دهد.



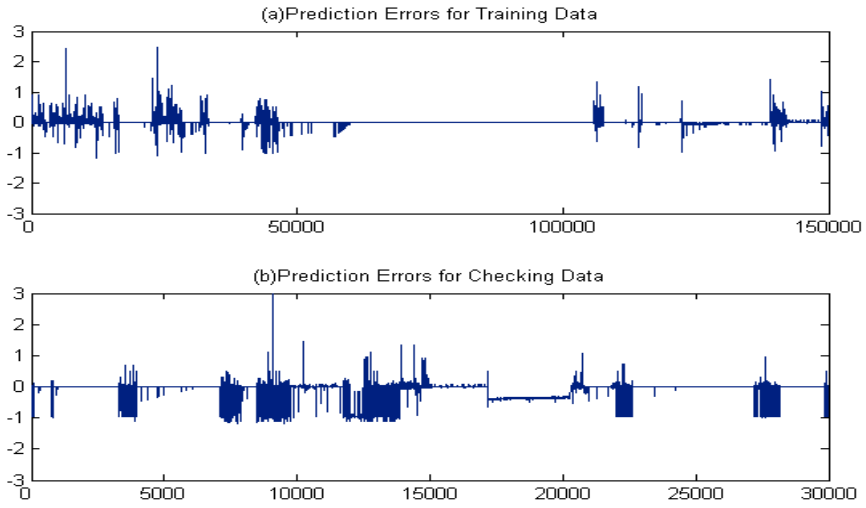
شکل ۱-۴ میزان خطا به ازا دوره های آموزش برای داده های آموزشی

[۱] Overfitting

[۲] Root Mean Squared Error

[۳] Epoch

اغلب سعی بر آن است تا با آموزش یک شبکه بتوان مقادیر مناسبی از خروجی را برای داده هایی ورودی بدست آورد و این عمل همیشه به صورت ۱۰۰٪ امکان پذیر نیست و هر شبکه آموزش دیده به طور معمول مقداری خطا در تخمین خروجی مورد انتظار دارا می باشد. شکل ۲-۴ تفاوت مقدار واقعی[۱] و مقدار بدست آمده[۲] از خروجی ANFIS را برای داده های آموزشی و داده های بررسی بعد از ۵۰ دوره آموزش نشان می دهد.



شکل ۲-۴ تفاوت مقدار واقعی و مقدار بدست آمده از خروجی ANFIS برای (a) داده های آموزشی (b) داده های بررسی

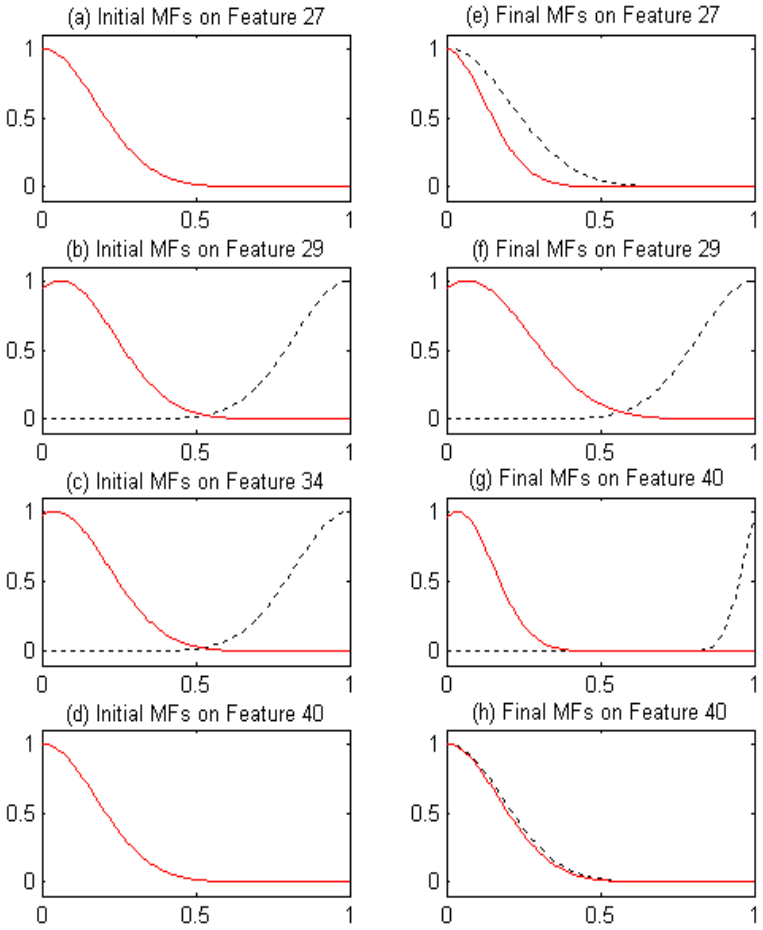
همانطور که از شکل ۲-۴ می توان استنتاج نمود خروجی شبکه ANFIS که در اصل شماره کلاس داده های ورودی می باشد (۰ برای الگوهای ورودی نرمال و ۱ برای الگوهای ورودی حمله) یک مقدار عددی صحیح نمی باشد. بنابراین ما نیاز داریم مقدار خروجی را برای بدست آوردن شماره کلاس گرد کنیم. Γ پارامتری است که ما عمل گرد کردن را بر اساس آن انجام می دهیم. Γ شعاع همسایگی را مشخص می کند و مقادیری که در فاصله کمتر از Γ با مقدار عددی صحیح مورد نظر (شماره کلاس) قرار می گیرند به این مقدار صحیح گرد می شوند. مقدار Γ در بازه ۰ تا ۰/۵ قابل تغییر است. در این مطالعه مقادیر خروجی که خارج از بازه گرد کردن قرار گیرند، باعث می شوند که طبقه بندی کننده، ورودی مربوطه برای آن خروجی را به عنوان یک الگو طبقه بندی نشده معرفی کند. در ادامه این فصل تاثیرات Γ بر کارایی سیستم ارائه شده بررسی خواهد شد.

---

[۱] Desierd Value

[۲] Perdicted Value

قبلا اشاره کردیم که آموزش ANFIS موجب تنظیم بهتر و تطبیق پذیری بیشتر توابع عضویت اولیه خواهد شد. توابع عضویت اولیه و نهایی برای چند نمونه از خصیصه های ورودی در شکل ۳-۴ نمایش داده شده است.



شکل ۳-۴ توابع عضویت برای چهار خصیصه ورودی نمونه (a)(b)(c)(d) قبل از آموزش (e)(f)(g)(h) بعد از آموزش

معیارهای استاندارد گوناگونی برای ارزیابی سیستم های تشخیص نفوذ ارائه شده است که از آن جمله می توان به *نرخ کشف*[۱] و *نرخ هشدارهای غلط*[۲] اشاره نمود. نرخ کشف از تقسیم تعداد حملاتی که بدرستی تشخیص داده شده اند بر کل تعداد حملات بدست می آید و نرخ هشدار های غلط در حقیقت نسبت تعداد اتصالات نرمال که به اشتباه به عنوان حمله تشخیص داده شده اند به کل تعداد اتصالات نرمال است. معیار ارزیابی دیگری که در این بخش مورد بررسی قرار گرفته است نرخ طبقه بندی است که از تقسیم تعداد الگو

---

۱ Detection Rate

۲ False Alarm Rate

هایی که به درستی طبقه بندی شده اند به کل تعداد الگو ها محاسبه می شود. جدول شماره ۱-٤ نرخ تشخیص و نرخ هشدار های غلط و نرخ طبقه بندی را برای داده های آموزشی با استفاده از روش طبقه بندی ارائه شده در این فصل نشان می دهد. مقدار Γ در محاسبه مقادیر ۰/۵ در نظر گرفته شده است. نتایج جدول نشان می دهد که طبقه بندی کننده مورد نظر در کلاس بندی داده های آموزشی و بررسی که بر اساس آنها آموزش دیده است بسیار خوب عمل می کند و نرخ طبقه بندی بسیار بالا است.

جدول ۱-۴: درصد نرخ هشدار غلط، نرخ تشخیص و نرخ طبقه بندی برای داده های آموزشی و داده های بررسی

داده	در صد نرخ هشدار غلط	در صد نرخ تشخیص	در صد نرخ طبقه بندی
آموزشی	۰/٦۱	۹۹/۷۵	۹۹/٦۸
بررسی	۱/٦	۹۱/۰۰	۹۲/٤٤

به منظور بررسی بیشتر کارایی طبقه بندی کننده ارائه شده و بررسی عملکرد طبقه بندی کننده فازی- عصبی در قبال داده هایی که آموزشی در قبال آن ها ندیده است، ۴۰۰۰۰ داده تصادفی دیگر از مجموعه داده های KDD به منظور ارزیابی سیستم انتخاب شدند. برای کاهش اثرات انتخاب تصادفی ۵ سری از این داده ها که با هم و با داده های آموزشی و بررسی هیچ اشتراکی ندارند انتخاب شدند و میانگین نتایج حاصله برای این ۵ سری محاسبه شد. در این قسمت از این مطالعه برای بررسی بیشتر توانایی طبقه کننده فازی- عصبی ارائه شده این طبقه بندی کننده با چند روش دیگر که آزمایشات مشابهی داشته اند مقایسه شده است. هر چند که این مقایسه به علت وجود عدم دسترسی به داده های آزمایشات انجام شده در کارهایی که مورد مقایسه قرار گرفته اند و همینطور از باب تعداد الگو های انتخابی نمی تواند عادلانه باشد، اما می تواند تا حد زیادی توانایی شبکه فازی- عصبی تطبیق پذیر را در این مساله طبقه بندی نشان دهد و این اطمینان را به ما بدهد که طبقه بندی کننده ارائه شده در این بخش که اساس ساختار کلی سیستم ارائه شده در این پایاننامه را تشکیل می دهد به اندازه کافی توانمند هست و کارایی آن خارج از حد انتظار نمی باشد. جدول ۲-٤ کارایی طبقه بندی کننده ارائه شده را بر اساس داده های آزمایش فوق همراه با دو روش فازی دیگر [۱, ۱۳] و روش استفاده از قوانین RIPPER که در [۲۲] آمده است، نشان می دهد.

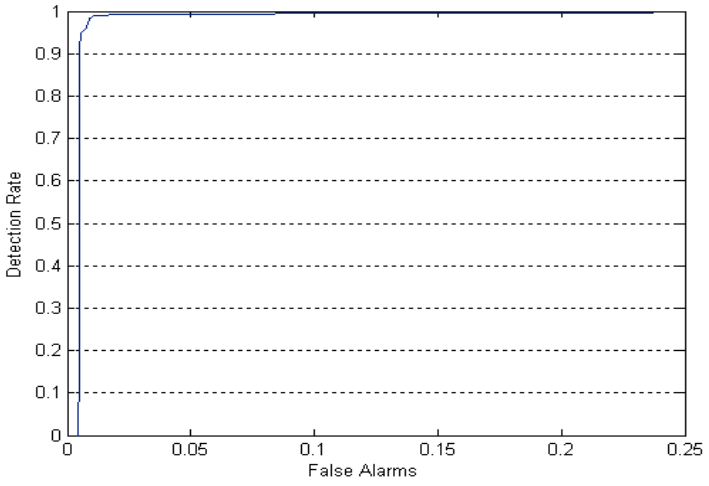
جدول ۲-۴: درصد نرخ هشدار غلط، در صد نرخ تشخیص و پیچیدگی زمانی الگوریتم های مختلف

روش	در صد نرخ هشدار غلط	در صد نرخ تشخیص	مرتبه زمانی
طبقه بندی کننده فازی- عصبی	۰/۵۹	۹۹/۵٤	O(n)
SRPP [۱]	۳/۵۸	۹۹/۰۸	O(n)

EFRID [۱۳]	۷	۹۸/۹۶	O(n)
RIPPER [۲۲]	۲/۰۲	۹٤/۲٦	O(n log n)

همانطور که قبلاً اشاره کردیم هر چند که این مقایسه تا حدی عادلانه نمی باشد، ولی نتایج جدول برای پذیرفتن طبقه بندی کننده فازی- عصبی به عنوان یک طبقه بندی کننده مناسب کفایت می کند و نشان دهنده این مطلب است که این طبقه بندی کننده از توانایی لازم در حد روشهای مشابه برخوردار است.

در ادامه این بخش ما تحلیل ویژگی عامل گیرنده[۱] (ROC) را برای ارزیابی کارایی طبقه بندی کننده ارائه شده به کار می گیریم. منحنی ROC در حقیقت نشان می دهد که تغییرات پارامتر های سیستم چگونه بر معیار های ارزیابی سیستم تاثیر می گذارد. ما برای تولید منحنی ROC، مقدار پارامتر Γ را در بازه ۰ تا ۰/۲ تغییر دادیم و سپس نقاط بدست آمده به صورت زوج $(FA,DR)_{\Gamma}$ که در آن FA در صد نرخ هشدار های غلط و DR درصد نرخ تشخیص می باشد را بر روی نمودار رسم کردیم. شکل ۴-۴ منحنی ROC را برای طبقه بندی کننده فازی- عصبی به ازاء مقادیر مختلف Γ نشان می دهد. منحنی ROC می تواند مشخص کند که طبقه بندی کننده چه موقع کارایی خوبی دارد و برای هر طبقه بندی کنند بالاترین نقطه سمت چپ نمودار، تشخیص بهینه را بر اساس نرخ تشخیص و نرخ هشدار نشان می دهد. اگر منحنی ROC برای طبقه بندی کننده A در تمامی قسمتهای نمودار بالای منحنی طبقه بندی کننده B قرار گیرد آنگاه می توان ادعا کرد که طبقه بندی کننده A از طبقه بندی کننده B بهتر است[۱۳] و هر چه منحنی ROC برای یک سیستم طبقه بندی به گوشه سمت چپ و بالای نمودار نزدیک باشد طبقه بندی کننده مناسبتری است. منحنی شکل ۴-۴ توانایی خوب طبقه بندی کننده ارائه شده را نشان می دهد.



شکل ۴-۴ منحنی ROC برای طبقه بندی کننده فازی- عصبی؛ ۰≤Γ≤۰/۲

[۱] Receiver Operating Characteristic

۳ – ۴ شبکه عصبی فازی تطبیق پذیر به شکل طبقه بندی کننده دوگانه و چندگانه

همانطور که می دانید هدف اصلی یک سیستم تشخیص نفوذ یافتن فعالیتهای نفوذی و غیر قانونی و گزارش کردن آنها به مدیر شبکه می باشد. در عمل می توان تشخیص نفوذ را به عنوان یک فرایند طبقه بندی دوگانه شامل تسهیم داده ها به دو کلاس حمله و نرمال معرفی نمود، معمولا سیستم های تشخیص رفتار غیر عادی به این طبقه بندی بسنده می کنند و هیچ گزارشی در مورد نوع وکلاس حمله مورد نظر ارائه نمی کنند. حال آنکه سیستم های تشخیص سوء استفاده اغلب نوع کلاسی را که حمله مورد نظر به آن وابسته است نیز معرفی می کنند. با توجه به مطالبی که در فصل دوم در مورد داده های KDD ارائه شده است، این داده ها شامل چهار کلاس کلی حملات هستند. در این بخش سعی شده است طبقه بندی کننده فازی- عصبی ارائه شده در فصل قبل به دو صورت طبقه بندی دوگانه و طبقه بندی کننده چندگانه مقایسه شود. در طبقه بندی دوگانه، مدل طبقه بندی کننده با داده های آموزشی که به دو کلاس داده های نرمال و داده های حمله برچسب خورده اند آموزش داده شده است. این مدل دقیقا مشابه با مدل ارائه شده در بخش قبل می باشد و بر خلاف این مدل طبقه بندی کننده دوگانه، مدل طبقه بندی کننده چندگانه بر اساس داده های آموزشی که به پنج کلاس مختلف، شامل کلاس ۰ برای داده های نرمال، کلاس ۱ برای داده های Probe، کلاس برای داده های Dos و کلاس ۳ و ۴ به ترتیب برای داده های کلاس هاس U2R و R2L نگاشت شده اند، آموزش داده شده است.

۴۸۸۴۰ داده تصادفی از مجموعه ۱۰٪ داده های که KDD به عنوان داده های آموزشی و ۴۸۸۴ داده دیگر که هیچ اشتراکی با داده فوق ندارند به عنوان داده های بررسی برای آموزش دو طبقه بندی کننده مذکور انتخاب شدند. جدول ۳-٤ توزیع الگو ها در مجموعه های آموزشی و بررسی برای آموزش مدل های طبقه بندی کننده دوگانه و چندگانه نشان می دهد.

جدول ۳-۴: توزیع الگو های آموزشی برای داده های آموزشی و بررسی

کلاس	آموزش	بررسی
Normal	۲۵۰۰۰	۲۵۰۰
Dos	۲۰۰۰۰	۲۰۰۰
U2R	٤۰	٤
R2L	۸۰۰	۸۰
Probe	۳۰۰۰	۳۰۰
	٤۸۸٤۰	٤۸۸٤

همانند روش ارائه شده در بخش قبل، در این قسمت نیز روش خوشه بندی کاهشی با شعاع همسایگی ۰/۵ برای بخش بندی داده های آموزشی و تولید ساختار اولیه سیستم استنتاج فازی به کار گرفته شد. تمامی توابع عضویت ورودی از نوع توابع گوسی با چهار پارامتر هستند. در مرحله بعد برای تنظیم بیشتر توابع عضویت داده های آموزشی این بار به منظور آموزش ساختار ANFIS به کار گرفته می شود. ANFIS ارائه شده برای طبقه بندی دوگانه شامل ۳۸۰ نود است و در مجموع ۴۹۶ پارامتر تنظیم شونده دارد که ۳۲۸ مورد از آنها پارامتر های بخش مقدم و ۱۶۸ مورد پارامتر های بخش تالی می باشند. میانگین RMSE (مجذور میانگین مربع خطا) برای این مدل پس از ۵۰ دوره آموزش ۲۳۴۹/۷۴ و ۰/۲۰۷۴۴۸ به ترتیب برای داده های آموزش و داده های بررسی می باشد. ANFIS استفاده شده برای طبقه بندی کننده چندگانه شامل ۳۲۸ پارامتر تطبیق پذیر در بخش مقدم و ۱۶۸ پارامتر تطبیقی در بخش تالی می باشد و مقدار میانگین RMSE برای داده های آموزش ۱۰۱۶۴/۷ و ۰/۲۶۴۴۱۹ برای داده های بررسی می باشد. بزرگتر بودن مقدار RMSE برای داده های آموزشی نسبت به داده های بررسی همانطور که در این قسمت اتفاق افتاده است، یک اتفاق غیر معمول است. به نظر می رسد این تفاوت به علت تفاوت زیاد در تعداد داده های آموزش و بررسی، همین طور کم بودن دوره های آموزشی است. در هر حال آنچه براساس آزمایشات دیگری که انجام گرفت مسلم است بیانگر این مطلب است که این تفاوت تاثیر بسزایی بر نتایج مقایسه ندارد.

لازم به یادآوری مجدد است که ساختار شبکه فازی- عصبی تطبیق پذیر (ANFIS) یک خروجی دارد و ما برای مشخص کردن شماره کلاس مورد نظر خروجی ساختار مورد نظر را با شعاعی که آن را Γ نامیدیم، گرد می کنیم. مجدداً در این بخش تاثیرات این پارامتر بر پارامتر های ارزیابی سیستم بررسی می شود.

در ادامه این بخش به منظور خلاصه سازی و جلوگیری از تکرار به دو طبقه بندی کننده مورد بررسی در این بخش با نام های مستعار ارائه شده در جدول ٤-٤ رجوع می کنیم.

جدول ۴-۴: نام های مستعار برای دو روش طبقه بندی استفاده شده

نام مستعار	روش
B-NFC	طبقه بندی کننده فازی- عصبی دوگانه
M-NFC	طبقه بندی کننده فازی- عصبی چندگانه

جدول ٥-٤ درصد نرخ تشخیص و درصد هشدار های غلط را برای داده های آموزش و داده های بررسی به ازاء دو طبقه بندی کننده ارائه شده در این بخش، پس از ۵۰ دوره آموزش و پارامتر گرد کردن Γ برابر با ۰/۵ نشان می دهد.

جدول ۵-۴: در صد نرخ هشدار های غلط و نرخ تشخیص برای داده های آموزشی و بررسی به ازاء طبقه بندی کننده B-NFC و M-NFC

طبقه بندی کننده	داده	درصد هشدار غلط	در صد نرخ تشخیص
B-NFC	آموزش	۰/۱۷	۹۶/۵۰
	بررسی	۰/۰۸	۹۶/٦٤
M-NFC	آموزش	۵/٦۳	۹۸/٤۳
	بررسی	۳/٦٤	۹۹/٤۱

همانطور که از نتایج جدول قابل مشاهده است طبقه بندی کننده چندگانه در میزان تشخیص بهتر از طبقه بندی کننده دوگانه است، ولی در عوض نرخ هشدار های غلط این روش نیز بیشتر از روش طبقه بندی دوگانه است. بنابراین هیچ قضاوت درستی در مورد اینکه کدام یک از این دو طبقه بندی کننده بهتر عمل می کنند نمی توان ارائه داد. در ادامه این بخش در مورد این مطلب بحث بیشتری خواهیم کرد.

برای ارزیابی بهتر این دو مدل مورد بررسی در این بخش دو آزمایش مختلف طراحی شد تا توانایی های این دو روش با هم مقایسه شوند. در آزمایش اول کل داده های تست مجموعه داده های KDD به عنوان مجموعه تست انتخاب شدند و نتایج طبقه بندی هر یک از دو روش بر روی این داده ها مورد بررسی قرارگرفت. نتایج بدست آمده در جدول ٦-٤ نشان داده شده است. با آنکه حدود ۵۰۰۰۰ داده از کل داده های آموزشی برای آموزش طبقه بندی کننده ها مورد استفاده قرار گرفته است جدول نتایج نسبتا خوبی را نشان می دهد. لازم به یاد آوری است که در مجموعه داده های تست KDD حملات جدیدی وجود دارند که هیچ الگو مشابهی در مجموعه آموزش برای آنها وجود ندارد. علاوه براین مجدداً نتایج جدول به خوبی مشخص نمی کند که کدام طبقه بندی کننده بهتر عمل می کند هر چند که انتظار داریم طبقه بندی کننده دوگانه نتایج بهتری را ارائه کند. راه حل مشکل استفاده از منحنی ROC برای درک هر چه بهتر عملکرد این دو روش طبقه بندی است که در ادامه به آن خواهیم پرداخت.

جدول ۶-۴: نرخ هشدارهای غلط و نرخ تشخیص برای کل الگو های مجموعه تست KDD حاصل از طبقه بندی دوگانه و چندگانه

طبقه بندی کننده	درصد نرخ هشدار های غلط	درصد نرخ تشخیص
B-NFC	۰/۳	۸۹/٤۳
M-NFC	۳/۴	۹۱/۱٤

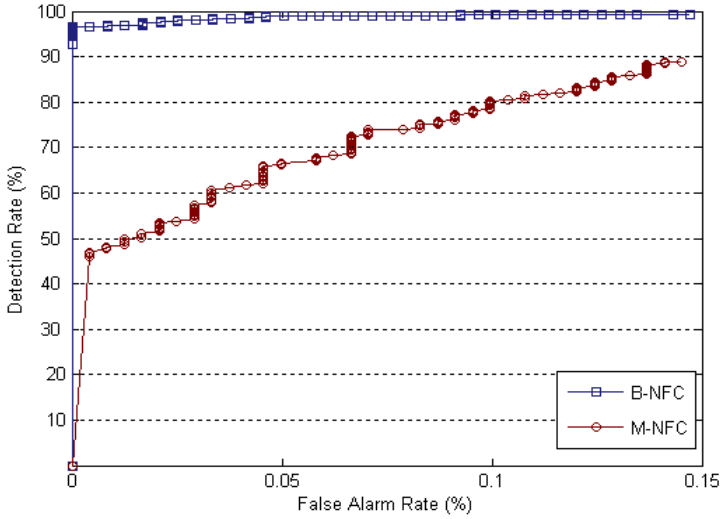
در آزمایش دوم، ۴۰۰۰۰ رکورد اتصال از مجموعه داده های آموزشی KDD که همان منبع انتخاب داده های آموزش برای طبقه بندی کننده های ارائه شده است و هیچ اشتراکی با آنها ندارند به صورت تصادفی انتخاب کردیم. برای کاهش اثرات انتخاب تصادفی الگو ها پنج سری از داده های آزمایشی با همان تعداد الگو که هیچ اشتراکی با هم ندارد انتخاب شدند و نتایج آزمایشات برای آنها میانگین گیری شد. جدول ۷-٤ نتایج

آزمایش را برای طبقه بندی کننده دوگانه و چندگانه نشان می دهد که نتایج بدست آمده کارایی مناسبی را نشان می دهند و نتایج بدست آمده عملکرد نسبتا بهتری را برای طبقه بندی کننده دو گانه نسبت به طبقه بندی کننده چندگانه ارائه می کند. برای اطمینان هر چه بیشتر این مدعا منحنی ROC برای این دو طبقه بندی کننده بدست آمد.

جدول ۷-۴: نرخ هشدارهای غلط و نرخ تشخیص برای ۴۰۰۰۰ الگو تصادفی از مجموعه آموزش KDD حاصل از طبقه بندی دوگانه و چندگانه

طبقه بندی کننده	درصد نرخ هشدار های غلط	درصد نرخ تشخیص
B-NFC	۰/۱۳	۹۹/۶۰
M-NFC	۴/۶۱	۹۹/۹۰

همانطور که قبلا اشاره کردیم منحنی ROC نشان می دهد که تغییرات پارامتر های سیستم چگونه بر معیار های ارزیابی سیستم تشخیص نفوذ تاثیر می گذارد. منحنی ROC می تواند مشخص کند که طبقه بندی کننده چه موقع کارایی خوبی دارد و برای هر طبقه بندی کنند بالاترین نقطه سمت چپ نمودار تشخیص بهینه را بر اساس نرخ تشخیص و نرخ هشدار نشان می دهد. لازم به یاد آوری است اگر منحنی ROC برای طبقه بندی کننده A در تمامی قسمتهای نمودار بالای منحنی طبقه بندی کننده B قرار گیرد آنگاه می توان ادعا کرد که طبقه بندی کننده A از طبقه بندی کننده B بهتر است[۱۳]. بنابر مطالب فوق ما منحنی ROC را برای دو طبقه بندی کننده ارائه شده در این بخش را با تغییر پارامتر Γ بدست آوردیم. شکل ۵-۴ منحنی بدست آمده برای این دو مدل را نشان می دهد. منحنی نشان می دهد که طبقه بندی کننده فازی عصبی دوگانه به طور محسوسی بهتر از مدل چندگانه عمل می کند. در حقیقت منحنی ROC نشان می دهد که مدل چقدر می تواند در تشخیص الگو ها موفق باشد و در عین حال نرخ هشدار های غلط کمی داشته باشد. منحنی نشان می دهد که طبقه بندی کننده چندگانه در مقادیر کمتر از ٪۰/۱ از هشدار های غلط نرخ تشخیص بسیار کمی دارد.



شکل ۵-۴ منحنی ROC برای طبقه بندی کننده های فازی- عصبی دو گانه و چندگانه؛ ۰/۵≤Γ≤۰

۴ - ۴ نتیجه گیری

در این فصل مدل طبقه بندی کننده ساده ای با استفاده از شبکه عصبی- فازی تطبیق پذیر ارائه شد. مدل ارائه شده قادر است قوانین فازی لازم را بدون نیاز به یک فرد خبره با استفاده از روش خوشه بندی کاهشی تولید کند. سپس قوانین فازی بدست آمده برای ایجاد یک شبکه عصبی- فازی تطبیق پذیر به نام ANFIS به کار گرفته شده اند و توابع عضویت در این ساختار با استفاده از داده های آموزشی استخراج شده به خوبی تنظیم می شوند و در نهایت برای طبقه بندی الگو های آزمایشی به کار گرفته می شوند. ساختار ANFIS به طور ذاتی برای انجام طبقه بندی طراحی نشده است و در اصل ANFIS یک سیستم استنتاج فازی است که می تواند همانند یک شبکه عصبی یک مدل پیچیده را تخمین بزند. ما در این فصل ANFIS را با استفاده از روش ارائه شده به عنوان یک طبقه بندی کننده به کار گرفتیم. نتایج بدست آمده تایید کننده این مدعا است که ANFIS به عنوان یک سیستم عصبی- فازی تطبیق پذیر می تواند در ایجاد یک سیستم تشخیص نفوذ بر اساس مدل طبقه بندی مناسب باشد. در ادامه سعی شد مدل طبقه بندی ارائه شده به دو شکل طبقه بندی کننده دو گانه و چندگانه مورد ارزیابی قرار گیرد. مدل طبقه بندی کننده دوگانه با داده های آموزشی که به دو کلاس داده های نرمال و داده های حمله برچسب خورده اند آموزش داده شده است، در حالیکه مدل طبقه بندی کننده چندگانه بر اساس داده های آموزشی که به پنج کلاس مختلف، شامل کلاس ۰ برای داده های نرمال، کلاس ۱ برای داده های Probe، کلاس برای داده های Dos و کلاس ۳ و ۴ به ترتیب برای داده های کلاس های U2R و R2L نگاشت شده اند، آموزش داده شده است. نتایج بدست آمده برای آزمایشات انجام شده بیانگر این حقیقت است که طبقه بندی کننده دوگانه به طور محسوسی از طبقه بندی کننده چندگانه بهتر عمل می کند و این در حقیقت انگیزه ای است که طبقه بندی کننده دوگانه ارائه شده به شکل فوق در چهار چوبه ارائه شده در این پایاننامه به کار گرفته شود.

فصل چهارم: تشخیص نفوذ به روش محاسبات نرم تکاملی

۱ – ۵ مقدمه

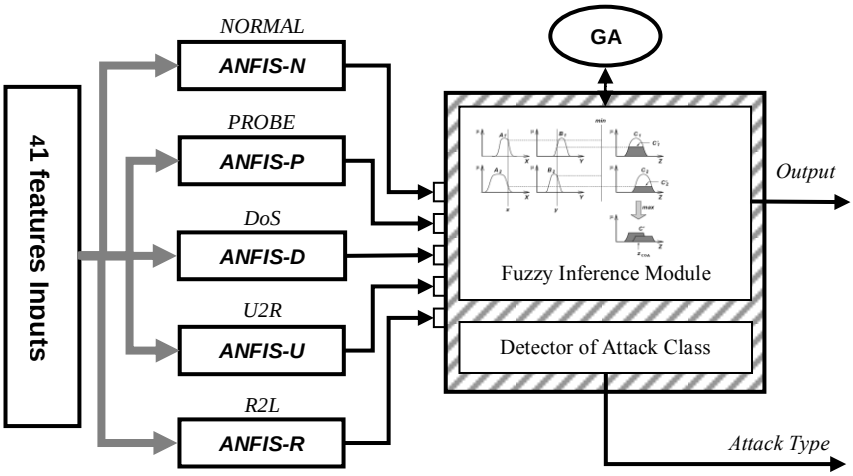
انگیزه اصلی برای انجام این مطالعه ارائه چهارچوبه ای شامل چندین روش محاسبات نرم با هدف ایجاد یک طبقه بندی کننده الگو است که این طبقه بندی کننده بتواند بهتر از الگوریتم هایی عمل کند، که تنها از یک روش استفاده می کنند. ما در این فصل ساختار سیستم ارائه شده را بررسی می کنیم. ابتدا به طور مختصر ساختار سیستم و چگونگی عملکرد آنرا شرح می دهیم، سپس مجموعه داده های انتخابی از مجموعه داده های KDD را برای آموزش سیستم شرح خواهیم داد. پس از آن گام به گام به تشریح لایه های چهارچوبه ارائه شده می پردازیم. فصل بعد به نتایج ارزیابی کارایی سیستم و آزمایشات اختصاص دارد.

۲ – ۵ معماری سیستم

معماری ارائه شده برای سیستم تشخیص نفوذ بر اساس محسابات نرم تکاملی شامل دو لایه است. لایه اول شامل پنج ماژول شبکه عصبی- فازی تطبیق پذیر است که برای کشف فعالیتهای تفوذی از داده های ورودی آموزش داده شده اند. هر ماژول به یکی از کلاسهای موجود در مجموعه داده متعلق است و میزان وابستگی داده ورودی را به آن کلاس مشخص بیان می کند. مقدار ۱ وابستگی کامل و مقدار ۱- برای بیان عدم وابستگی استفاده می شود. به عبارت بهتر هر ANFIS، در لایه اول با مجموعه ای از داده ها شامل داده هایی در کلاس مربوط به آنANFIS (با برچسب ۱) و داده های کلاس های دیگر که با ۱- برچسب خورده اند آموزش داده می شود. لازم به یاد آوری است که ساختار ANFIS شامل یک خروجی است که می تواند مقادیر پیوسته عددی را در فضای اعداد حقیقی را تولید کند. اصلیترین انگیزه برای استفاده از پنج ماژول ANFIS به شکل فوفق، نتایجی بدست آمد در فصل قبل است، که بیانگر این مطلب می باشد که شبکه عصبی- فازی تطبیق پذیر به شکل یک طبقه بندی کننده دو گانه بهتر عمل می کند. بنابراین سیستم ارائه شده در این فصل از پنج ANFIS که هر یک به صورت یک طبقه بندی کننده دو گانه که داده های ورودی را به دو کلاس مختلف یعنی کلاس داده های متعلق و کلاس داده های غیر متعلق به کلاس کلی تری که آن طبقه بندی کننده به برای آن در نظر گرفته شده است، تقسیم بندی می کند.

لایه دوم در این چهارچوبه یک ماژول استنتاج فازی است که بر اساس نتایج تجربی به دست آمده طراحی شده است. وظیفه اصلی این ماژول تصمیم گیری نهایی در مورد نفوذی بودن الگوی مورد بررسی است. ماژول تصمیم گیری فازی خروجی طبقه بندی کننده های عصبی- فازی در لایه اول را به عنوان ورودی دریافت می کند سپس یک خروجی تولید می کند که این خروجی باینگر نفوذی یا نرمال بودن رکورد اتصال ورودی است. پس از آن اگر الگو ورودی جاری را به عنوان یک الگو غیر قانونی یا نفوذی تشخیص داده شد، طبقه بندی کننده لایه اول که خروجی آن نزدیکترین مقدار به ۱ (یعنی تعلق به کلاس مربوطه) باشد، کلاس الگو تهاجمی را مشخص می کند.

برای رسیدن به بهترین نتایج، الگوریتم ژنتیک برای بهینه سازی موتور تصمیم گیری فازی لایه دوم به کار گرفته شده است. ساختار الگوریتم ژنتیک به کار گرفته شده در بخشهای بعدی به تفصیل مطرح خواهد شد. شکل ۱-۵ بلاک دیاگرام شماتیکی را برای معماری سیستم ارائه شده نشان می دهد.



شکل ۱-۵: بلاک دیاگرام ساختار سیستم

۳ – ۵ منابع داده

همانطور که قبلا اشاره کردیم ۴۱ خصیصه برای هر رکورد اتصال در مجموعه داده های KDD وجود دارد که به فرم های پیوسته، گسسته و سمبولیک هستند[۲۰] و روشهای طبقه بندی الگو نمی توانند این داده ها را به فرمت اصلی خود به کار گیرند. ما در این بخش پیش پردازشی همانند آنچه در فصل قبل به آن اشاره کردیم را استفاده می کنیم به این شکل که داده های سمبولیک را به داده های عددی تبدیل می کنیم. به عبارت دیگر خصیصه های سمبولیک مانند انواع پروتکل، سرویس ها و پرچم ها به مقادیر ۰ تا N-1 که در آن N تعداد سمبول ها برای هر خصیصه است نگاشت شده اند. به عنوان مثال خصیصه Protocol_Type با سه سمبول مختلف TCP، UDP، ICMP به سه عدد ۰، ۱، ۲، نگشت شده است. بقیه خصیصه ها به همان صورت اولیه استفاده شده اند.

تمامی این ۴۱ خصیصه به عنوان ورودی سیستم مورد استفاده قرار گرفته اند و در این سیستم تلاشی برای استفاده از روشهای کاهش خصیصه ها انجام نگرفته است [۱۵, ۲۹] . از دیدگاه طبقه بندی هر فرایند طبقه بندی شامل دو فاز است، که شامل آموزش پارامتر های طبقه بندی کننده با استفاده از داده های آموزشی و استفاده از طبقه بندی کننده برای کلاس بندی داده های آزمایش می باشد. همانند فصل قبل زیر مجموعه ۱۰٪ داده های آموزشی KDD به عنوان منبع داده های آموزشی استفاده شده است. از آنجائیکه مجموعه ۱۰٪ داده های آموزشی برای اهداف ما بسیار بزرگ است، زیر مجموعه های مختلفی شامل داده های آموزش و بررسی از این مجموعه داده به صورت تصادفی برای آموزش سیستم ارائه شده انتخاب شده اند. در مورد لزوم استفاده

از داده های بررسی قبلا صحبت کرده ایم و به بیان این مطلب کـه اگـر داده هـای بررسـی اسـتفاده نشـوند ایـن امکان وجود دارد که سیستم بیش از حد آموزش ببیند و برای داده هـای غیـر وابسـته بـه داده هـای آمـوزش بـه خوبی عمل نکند.

بررسی نتایج الگوریتم های مختلف آموزش ماشین نشان می دهد که سیستم های تشـخیص نفـوذ مبتـی بـر تشخیص رفتار غیر عادی از مدل های مبتی بر تشخیص سوء استفاده برای مجموعه داده های KDD بهتر عمـل می کنند [۲۱, ۳۳]. این ممکن است به این علت باشد که داده های تست در مجموعه KDD شـامل حملاتـی جدیدی هستند که هیچ امضاء مشابهی مناسبی در مجموعه داده های آموزش ندارند. به عبـارت دیگـر، بـه نظـر میرسد، تعداد نمونه های نفوذی در مجموعه آموزش برای آموزش یـک تشـخیص دهنـده سـوء اسـتفاده کـافی نیست، هر چند که نمونه های نفوذی در مجموعه تست انحراف لازم از داده های نرمال برای آنکه یک سیسـتم تشخیص رفتار غیر عادی بتواند آنها را بیابد، دارا هستند. نتایج بدست آمده در [۳۴] و [۳۸] بیـانگر ایـن مـدعا است. از آنجائیکه هر طبقه بندی کننده در سیستم ارائه شده به صورت یک مدل تشـخیص سـوء اسـتفاده عمـل می کند و هدف اصلی انتخاب مجموعه های آموزشی و بررسی مناسب برای فـاز آمـوزش سیسـتم اسـت، زیـر مجموعه های آموزشی و بررسی به صورتیکه در جداول آمده اند انتخاب شدند، که در آنها تعـداد الگـو هـای کلاس نرمال تقریبا برابر با مجموع الگوهای موجود در کلاس های دیگر است. با این سیاست با توجه به اینکـه هر طبقه بندی کننده به شکل یک طبقه بندی کننده دوگانه عمل می کند، سـعی شـده اسـت تـا حـدی توانـایی تشخیص رفتار غیرعادی را به هر طبقه بندی کننده اضافه کنیم.

دو گروه داده برای آموزش سیستم استفاده شده است که توزیع الگو ها در این دو زیر مجموعه آموزشـی در جداول ۱-۵ و ۲-۵ آمده است. زیـر مجموعـه هـای انتخـابی شـامل تعـداد الگـو هـای متفـاوتی هسـتند. زیـر مجموعه کوچکتر شامل تعداد کمی از الگو های آموزشی است، تا به این وسیله نشان دهیم سیستم ارائه شده بـا وجود تعداد کم داده های آموزشی از قابلیتهای خوبی برخوردار است. زیـر مجموعـه بزرگتـر شـامل داده هـای آموزشی بیشتر است و به این منظور انتخاب شده است که کارایی سیستم را هر چه بیشتر نمایش دهد.

برای کاهش اثرات انتخاب تصادفی ده زیر مجموعه با همان توزیع جداول ۱-۵ و ۲-۵ انتخاب شده است و نتایج بدست آمده در فصل بعد بر اساس میانگین گیری از نتایج حاصل بدست آمده است. لازم به ذکر است به جهت اینکه نتایج حاصله از هر جهت منصفانه باشد، در طول فرایند آموزش و بهینه سازی هیچ دسترسی به داده های تست وجود نداشته است و تمامی داده از مجموعه آموزش انتخاب شده اند.

جدول ۱-۵: توزیع الگو ها در مجموعه آموزشی اول که از زیر مجموعه ۱۰٪ KDD cup انتخاب شده اند.

طبقه بندی کننده	داده	Normal	Probe	Dos	U2R	R2L	مجموع
ANFIS-N	آموزش	۲۰۰۰۰	٤۰۰۰	۱۵۰۰۰	٤۰	۱۰۰۰	۴۰۰۴۰
	بررسی	۲۵۰۰	۱۰۷	۲۰۰۰	۱۲	۱۲٦	٤۷٤۵

ANFIS-P	آموزش	۱۰۰۰۰	٤۰۰۰	۵۰۰۰	٤۰	۱۰۰۰	۲۰۰٤۰
	بررسی	۱۰۰۰	۱۰۷	۵۰۰	۱۲	۱۲٦	۱۷٤۵
ANFIS-D	آموزش	۲۵۰۰۰	٤۰۰۰	۲۰۰۰۰	٤۰	۱۰۰۰	۵۰۰٤۰
	بررسی	٦۰۰۰	۱۰۷	۵۰۰۰	۱۲	۱۲٦	۱۱۲٤۵
ANFIS-U	آموزش	۲۰۰	۵۰	۵۰	٤٦	۵۰	۳۹٦
	بررسی	۱۰۰	۲۵	۲۵	٦	۲۵	۱۸۱
ANFIS-R	آموزش	٤۰۰۰	۱۰۰۰	۲۰۰۰	٤۰	۱۰۰۰	۸۰٤۰
	بررسی	۲۰۰۰	۵۰۰	۱۰۰۰	۱۲	۱۲٦	۳٦۳۸

جدول ۲-۵: توزیع الگو ها در مجموعه آموزشی دوم که از زیر مجموعه ۱۰٪ KDD cup انتخاب شده اند.

طبقه بندی کننده	داده	Normal	Probe	Dos	U2R	R2L	مجموع
ANFIS-N	آموزش	۱۵۰۰	۵۰۰	۵۰۰	۵۲	۵۰۰	۳۰۵۲
	بررسی	۱۵۰۰	۵۰۰	۵۰۰	۰	۵۰۰	۳۰۰۰
ANFIS-P	آموزش	۱۵۰۰	۵۰۰	۵۰۰	۵۲	۵۰۰	۳۰۵۲
	بررسی	۱۵۰۰	۵۰۰	۵۰۰	۰	۵۰۰	۳۰۰۰
ANFIS-D	آموزش	۱۵۰۰	۵۰۰	۵۰۰	۵۲	۵۰۰	۳۰۵۲
	بررسی	۱۵۰۰	۵۰۰	۵۰۰	۰	۵۰۰	۳۰۰۰
ANFIS-U	آموزش	۱۵۰۰	۵۰۰	۵۰۰	٤٦	۵۰۰	۳۰٤٦
	بررسی	۱۵۰۰	۵۰۰	۵۰۰	٦	۵۰۰	۳۰۰٦
ANFIS-R	آموزش	۱۵۰۰	۵۰۰	۵۰۰	۵۲	۵۰۰	۳۰۵۲
	بررسی	۱۵۰۰	۵۰۰	۵۰۰	۰	۵۰۰	۳۰۰۰

۴ – ۵ طبقه بندی کننده های عصبی – فازی

همانند روشهای ارائه شده در فصل قبل مجددا روش خوشه بندی کاهشی با شعاع همسایگی ۰/۵ برای بخش بندی مجموعه های آموزشی و تولید ساختار سیستم استنتاج فازی برای هر ANFIS به کار گرفته شد. برای تنظیم بهتر و تطبیق هر چه بیشتر توابع عضویت، مجموعه های آموزشی ذکر شده برای فاز آموزش استفاده شدند. هر ANFIS ۱۰۰ دوره آموزش داده شد و ساختار حاصله با مینیمم خطا برای داده های بررسی انتخاب شد. ساختار ANFIS های حاصل برای یک سری از ۱۰ سری داده های آموزشی سری ۱ و ۲ به عنوان

مثال در جدول ارائه شده است. میانگین RMSE در پایان ۱۰۰ دوره برای همه دوره های آموزش محاسبه شده است.

جدول ۳-۵: ساختار پنج ANFIS برای یک سری نمونه از ۱۰ سری داده های آموزشی سری اول

کلاس	پارامترهای مقدم	پارامترهی تالی	تعداد قوانین	تعداد توابع عضویت	RMSE میانگین
Normal	۴۱۰	۲۱۰	۵	۵	۰/۲۸۶۶۷
Probe	۵۷٤	۲۹٤	۷	۷	٤/۷۰۸۲
DoS	۲٤٦	۱۲٦	۳	۳	۰/۱۸۴۷
U2R	۶۵۶	۳۳۶	۸	۸	۳۱/۱۱۸۳
R2L	۳۲۸	۱٦۸	٤	٤	۰/۵۹۹٦

جدول ۴-۵: ساختار پنج ANFIS برای یک سری نمونه از ۱۰ سری داده های آموزشی سری دوم

کلاس	پارامترهای مقدم	پارامترهی تالی	تعداد قوانین	تعداد توابع عضویت	RMSE میانگین
Normal	۶۵۶	۳۳٦	۸	۸	۲/۷e+۶
Probe	٦۵٦	۳۳٦	۷	۷	۱٦۰۹۱/۲
DoS	۳۷۸	۷۳۸	۹	۹	۳۳/۸۳۶۵
U2R	۳۷۸	۷۳۸	۹	۹	۴/۶e+۷
R2L	۳۷۸	۷۳۸	۹	۹	۲۴۸۰/۵۴

۵ – ۵ ماژول تصمیم گیری فازی

ماژول تصمیم گیری فازی دارای پنج ورودی است، که هر ورودی در اصل خروجی حاصل از ANFIS لایه قبل است. ماژول استنتاج فازی بر اساس این ورودی ها مشخص می کند که آیا اتصال جاری یک حمله است یا خیر. یک سیستم استنتاج فازی پنج- ورودی، یک- خروجی ممدانی با غیر فازی ساز مرکز ثقل و--- ---------- برای این منظور استفاده شده است. هر مجموعه فازی ورودی شامل دو تابع عضویت است که تمامی این توابع ازنوع توابع گوسی با ۴ پارامتر انتخاب شده اند. موتور تصمیم گیری فازی ارائه شده از قوانین فازی ارائه شده در حافظه انجمنی فازی[1] در جدول ۵-۵ نمایش داده شده اند.

جدول ۵-۵: حافظه انجمنی فازی برای قوانین فازی ماژول تصمیم گیری فازی

Normal	PROBE	DoS	U2R	R2L	Output

[1] Fuzzy associative map

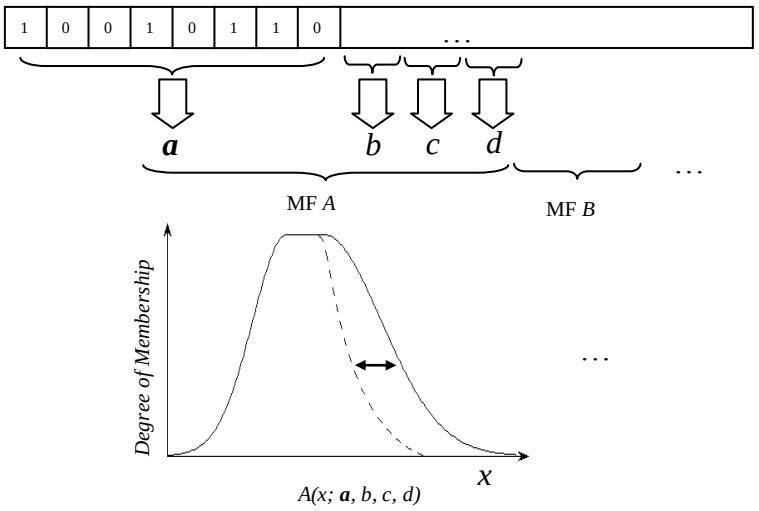
High	-	-	-	-	Normal
-	¬High	¬High	¬High	¬High	Normal
-	High	-	-	-	Attack
-	-	High	-	-	Attack
-	-	-	High	-	Attack
-	-	-	-	High	Attack
Low	-	-	-	-	Attack
-	Low	Low	Low	Low	Normal

خروجی سیستم فازی بین ۱ و ۱- تغییر می کند و مشخص می سازد که رکورد جاری به چه میزان غیر قانونی است. مقدار ۱ نشان می دهد که رکورد اتصال جاری به طور کامل نفوذی است و مقدار ۱- بیانگر نرمال بودن الگوی ورودی است. رکورد های اتصال با میزان نفوذی بودن بالای ۰ به عنوان الگو های نفوذی تشخیص داده می شوند. هنگامی که یک رکورد اتصال توسط ماژول تصمیم گیری فازی به عنوان اتصال غیر قانونی تشخیص داده شد، کلاس حمله تشخیص داده شده بر اساس ماژول ANFIS ی که بیشترین مقدار نفوذی بودن را برای این رکورد صادر کرده است مشخص می شود.

۶ – ۵ ماژول الگوریتم ژنتیک

وظیفه این بخش از چهار چوبه ارائه شده که بر اساس یک الگوریتم ژنتیک عمل می کند، بهنیه سازی توابع عضویت موتور تصمیم گیری فازی ارائه شده در بخش قبل است. انگیزه اصلی برای استفاده از یک روش بهینه سازی در این قسمت این مطلب است که هر کدام از طبقه بندی کننده های لایه قبل بر اساس میزان آموزش ، تعداد داده ها و توزیع الگوها دارای توان خاصی در طبقه بندی الگو ها هستند. به عنوان مثال طبقه بندی کننده U2R با تعداد الگوهای بسیار کمی آموزش دیده است، بنابراین نتایج حاصل از طبقه بندی این ماژول به اندازه دیگر ماژولهای این لایه مستدل نمی باشد. اگر خواننده گرامی به خاطر بیاورد در فصل قبلی ما خروجی طبقه بندی کننده را با پارامتر خاصی گرد می کردیم، یعنی با شعاعی خاص مقادیر اطراف یک شماره کلاس را به مقدار مطلق شماره کلاس مورد نظر نگاشت می کردیم. در معماری ارائه شده در این فصل در قسمت موتور استنتاج فازی در حقیقت یک تابع گوسی برای هر کلاس خروجی از هر طبقه بندی کننده لایه اول در نظر گرفته ایم که این تابع گوسی در حقیقت مشابه با پارامتر گرد کردن اشاره شده عمل می کند با این تفاوت که ما در این بخش با خروجی هر طبقه بندی کننده به عنوان یک متغیر زبانی برخورد کرده ایم که نه تنها داده های اطراف یک کلاس را به آن کلاس نگاشت می کند، بلکه میزان وابستگی به آن کلاس را نیز مشخص می کند و در نهایت موتور استنتاج فازی برای اساس این مقادیر تصمیم نهایی را اتخاذ می کند. بنابراین اگر بتوان این توابع عضویت گوسی را به طریقی بر اساس میزان قدرت طبقه بندی کننده ای که ورودی مربوط به این مجموعه فازی شامل تابع عضویت فوق را تولید می کند تنظیم کنیم به طور قطع نتایج بهتری حاصل خواهد شد.

در جستجوی ژنتیک، هر فرد (کروموزوم) پارامتر های مربوط به توابع عضویت هر یک از مجموعه های فازی سیستم استنتاج فازی را کد می کند. هر کروموزوم شامل ۳۲۰ژن که در اصل هر ژن یک بیت اطلاعاتی می باشد، است. هر ۸ بیت از این اطلاعات یک پارامتر از ۴ پارامتر تابع عضویت را مشخص می کند. شکل ۲-۵ فرایند کدگشایی هر فرد از جمعیت را نشان می دهد.



شکل ۲-۵: فرآیند کد گشایی شماتیک برای هر فرد از جمعیت الگوریتم ژنتیک به کار رفته

الگوریتم ژنتیکی که در این بخش برای بهینه سازی توابع عضویت ورودی ماژول تصمیم گیری فازی استفاده شده است از زیر مجموعه ای که به صورت تصادفی با توزیع ارائه شده در جدول ٦-٥ از زیر مجموعه ۱۰٪ KDD استخراج شده است. با توجه به این حقیقت که فرآیند بهینه سازی الگوریتم ژنتیک همیشه نتایج مطلقی را ایجاد نمی کند، ما فاز بهنیه سازی را ۳ بار انجام داده و نتایج حاصل برای آزمایشات را به ازا هر یک از سه ساختار بدست آمده میانگین گیری کردیم. علاوه بر این برای کاهش اثرات انتخاب تصادفی ۵ سری از این داده ها را انتخاب کردیم و هر بار بهینه سازی را بر اساس یکی از این پنج سری از داده ها انجام دادیم.

همانطور که قبلا اشاره کردیم مهمترین بخش یک الگوریتم ژنتیک توابع شایستگی هستند که میزان صلاحیت هر فرد از جامعه را ارزیابی می کند. در اصل تابع شایستگی، تابعی است که ما می خواهیم آنرا بهینه کنیم. در این مطالعه دو تابع شایستگی مختلف در نظر گرفته شده است.

جدول ۶-۵ : درصد نرخ هشدار غلط، نرخ تشخیص و نرخ طبقه بندی برای داده های آموزشی و داده های بررسی

داده	Normal	Probe	Dos	U2R	R2L
تعداد الگو ها	۲۰۰	۱۰٤	۲۰۰	۵۲	۱۰٤

قبل از اینکه به تشریح جزئیات توابع شایستگی استفاده شده در این مطالعه به پردازیم، لازم است به معیار های استانداردی که برای ارزیابی سیستم های تشخیص نفوذ ارائه شده است بپردازیم که در توابع شایستگی ارائه شده نقش قابل توجهی را ارائه می کنند. نرخ تشخیص و نرخ هشدار های غلط دو مورد از معروفترین این معیار ها هستند که در فصل قبل تعریف شدند و در اینجا برای یادآوری مجددا آنها را تعریف می کنیم. نرخ تشخیص نسبت تعداد الگو های حمله که بدرستی توسط طبقه بندی کننده تشخیص داده شده است به کل تعداد الگو های حمله موجود می باشد، در حالیکه نرخ هشدار های غلط در حقیقت نسبت تعداد اتصالات نرمالی که به عنوان حمله تشخیص داده شده اند به کل تعداد اتصالات نرمال می باشد. لازم به ذکر است که نرخ تشخیص برای یک کلاس خاص از حملات که از آن با عنوان نرخ طبقه بندی آن کلاس یاد می کنیم، برابر نسبت تعداد حملات آن کلاس که بدرستی تشخیص داده شده اند و تعداد کل الگو های موجود در آن کلاس است. معیار دیگری که برای ارزیابی الگوریتم های طبقه بندی بنا شده است، معیار هزینه برای نمونه[۱] است و این همان معیاری است که در مسابقه KDD به عنوان معیار گزینش روش برتر استفاده شده است[۳۲]. ما به این معیار با نام اختصاری CPE اشاره می کنیم. CPE به صورت زیر محاسبه می شود:

$$CPE = \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m} CM(i,j) * C(i,j) \qquad (۵-۱)$$

که در آن CM و C ماتریسهای برهم ریختگی[۲] و هزینه[۳] هستند و N تعداد کل نمونه های آزمایش را نشان می دهد و m تعداد کلاس های موجود در عملیات طبقه بندی است. یک ماتریس بر هم ریختگی یک ماتریس مربعی است که هر ستون آن به یک کلاس پیش بینی شده[۴] و هر سطر آن به یک کلاس واقعی[۵] اختصاص دارد. هر درایه در سطر i و ستون j، CM(i,j)، تعداد نمونه های که بدرستی طبقه بندی نشده اند را نشان می دهد که در اصل به کلاس i متعلق بوده است و در کلاس j طبقه بندی شده است. درایه های روی قطر اصلی ماتریس تعداد نمونه هایی که بدرستی طبقه بندی شده اند را مشخص می کند. ماتریس هزینه از نظر ساختار شبیه به ماتریس بر هم ریختگی است با این تفاوت که درایه C(i,j) در این ماتریس هزینه جریمه برای طبقه بندی نادرست یک الگو از کلاس i در کلاس j را مشخص می کند. بنابراین درایه های قطر اصلی ماتریس C همواره دارای مقدار صفر دارند، زیرا قطر اصلی بیانگر طبقه بندی درست نمونه هاست.

---

[۱] Cost per example

[۲] Confusion Matrix

[۳] Cost Matrix

[۴] Predicted Class

[۵] Actual Class

ماتریس هزینه ای که در مسابقه یادگیری طبقه بندی کننده ها KDD'99 به کار گرفته شد در جدول ۵-۷(a) نشان داده شده است[۲۰]. هر چه مقدار CPE برای یک طبقه بندی کننده پایین تر باشد بیانگر این مطلب است که آن طبقه بندی کننده بهتر عمل کرده است.

حال که معیار های لازم برای ارزیابی یک سیستم تشخیص نفوذ بر اساس طبقه بندی را معرفی کردیم، قادر هستیم در مورد توابع شایستگی که در الگوریتم ژنتیک سیستم ارائه شده از آن استفاده کرده ایم صحبت کنیم. دو تابع شایستگی مختلف در این مطالعه در نظر گرفته شده است. تابع شایستگی اول، تابعی است که سعی می کند مقدار CPE را بر اساس ماتریس هزینه ارائه شده در جدول ۵-۷(b) به حداقل ممکن برساند. استفاده از یک تابع شایستگی با این ساختار سعی می کند میزان نرخ تشخیص را تا حد ممکن بالا ببرد در حالیکه میزان هشدار های غلط را در حداقل ممکن نگهداری کند و به این ترتیب نرخ طبقه بندی کلاس های مختلف و نرخ تشخیص کلی ماکزیمم می شوند در حالیکه میزان هشدار های غلط مینیمم می گردد. در حقیقت این تابع شایستگی حد تعادلی بهینه ای را بین نرخ تشخیص و نرخ هشدار های غلط ارائه می دهد.

جدول ۵-۷: ماتریس های هزینه؛ ستون ها به کلاس های پیش بینی شده و سطر ها به کلاس واقعی تعلق دارند. (a) ماتریس هزینه برای مسابقه طبقه بندی KDD'99. (b) ماتریس هزینه با هزینه طبقه بندی نادرست یکسان برای همه کلاس ها.

		Predicted				
		Normal	PROBE	DoS	U2R	R2L
Actual	Normal	0	1	2	2	2
	PROBE	1	0	2	2	2
	DoS	2	1	0	2	2
	U2R	3	2	2	0	2
	R2L	4	2	2	2	0

(a)

		Predicted				
		Normal	PROBE	DoS	U2R	R2L
Actual	Normal	0	1	1	1	1
	PROBE	1	0	1	1	1
	DoS	1	1	0	1	1
	U2R	1	1	1	0	1
	R2L	1	1	1	1	0

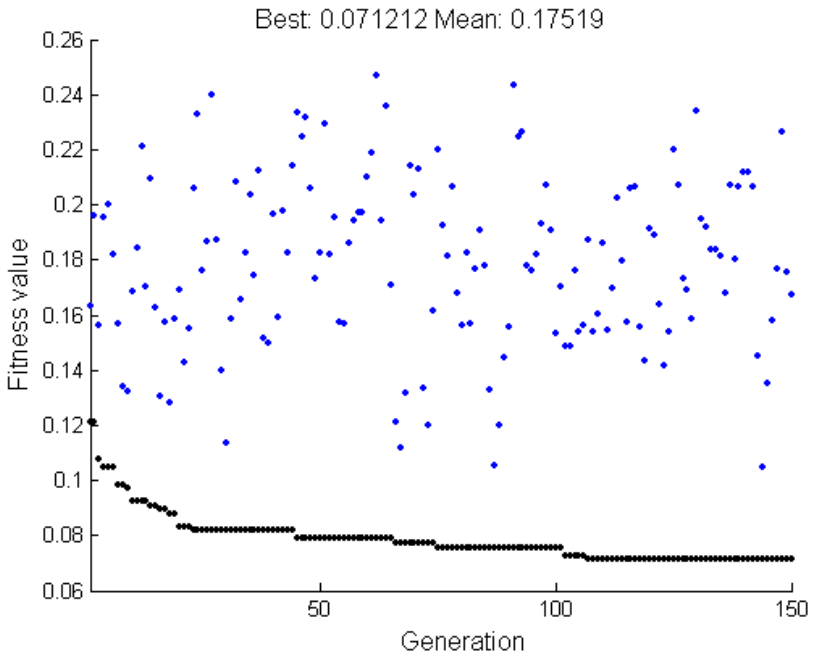
(b)

تابع شایستگی دیگری که مورد استفاده قرار گرفته است از ماتریس هزینه ای که برای ارزیابی نتایج مسابقه KDD مورد استفاده قرار گرفته است را به کار می گیرد[۲۰]. این تابع شایستگی میزان CPE را برای یک فرایند طبقه بندی با ماتریس هزینه فوق را در جهت کاهش این مقدار بهینه می کند. ما این ماتریس هزینه در اینجا استفاده کرده ایم تا به بهترین نرخ طبقه بندی که بر اساس ماتریس هزینه ای که برای ارزیابی شرکت کنندگان در مسابقه KDD استفاده شده است، برسیم. شبه کدی مورد استفاده برای توابع شایستگی در ادامه آمده است.

```
Function Y=fitness(X)
```

```
Begin
     Transform input vector(X)to output cube(M[5,2,4])
     //5 Input fuzzy set each has 2 Membership functions (MF) and each MF contains 4
     parameters.//
     Read FIS //Fuzzy Inference System//
     For each input i=1 to 5
          For Membership Function (MF) j=1 to 2
               For each parameter k=1 to 4
                    FIS.Input(i).MF(j).params(k)=M(i,j,k)
     Write FIS
     Read dataset
     Do classification based on FIS
     Calculate CPE
     Return Y=CPE
End
```

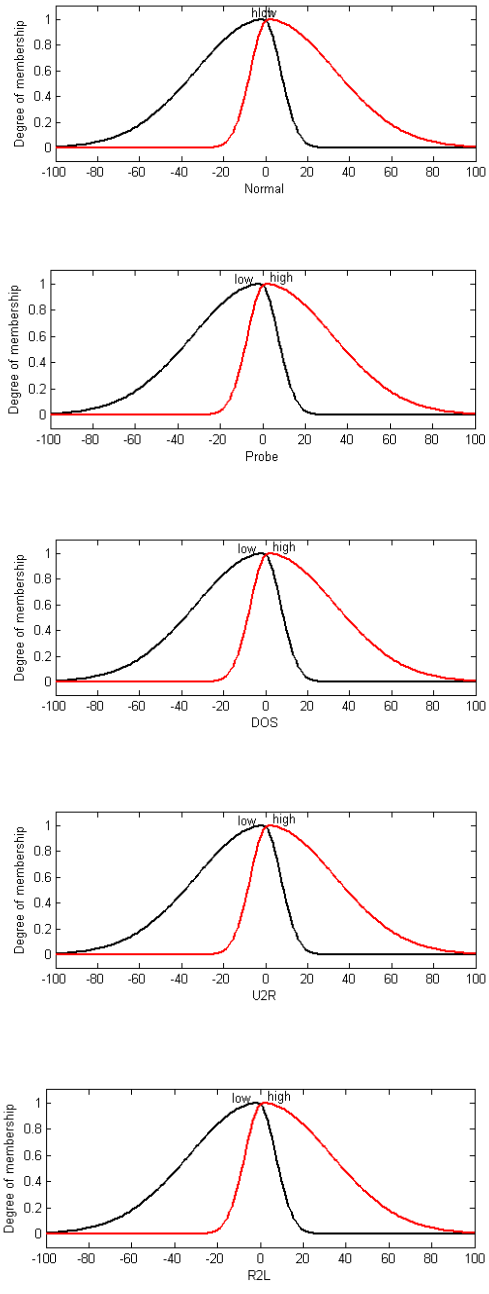
شبه کد شامل تبدیل بردار ورودی که همان کروموزوم توضیح داده شده در این بخش شامل ۳۲۰ بیت می باشد که بر اساس ساختار شکل ۲-۵ به پارامترهای مربوط به هر تابع عضویت می باشد. سپس در ادامه این داده های انتخابی برای عملیات بهنیه سازی که در جدول ٦-٥ آمده است خوانده می شوند و طبقه بندی بر اساس سیستم ارائه شده انجام می گیرد. در نهایت مقدار CPE بر اساس یکی از ماتریس های هزینه ارائه شده در جداول ۵-۷(a) و ۵-۷(b) محاسبه می شود. بدین ترتیب میزان مناسب بودن یک فرد از جامعه در الگوریتم ژنتیک محاسبه می گردد و بر اساس استفاده از عملگر ژنتیک توضیح داده شده نسل های جدید ایجاد می شوند. شکل ۳-۵ مقدار میانگین و کمینه CPE به عنوان خروجی تابع شایستگی را برای یک ساختار نمونه از سیستم ارائه شده در طی فرآیند بهینه سازی توسط الگوریتم ژنتیک به ازاء ۱۵۰ نسل تولید شده نشان می دهد.



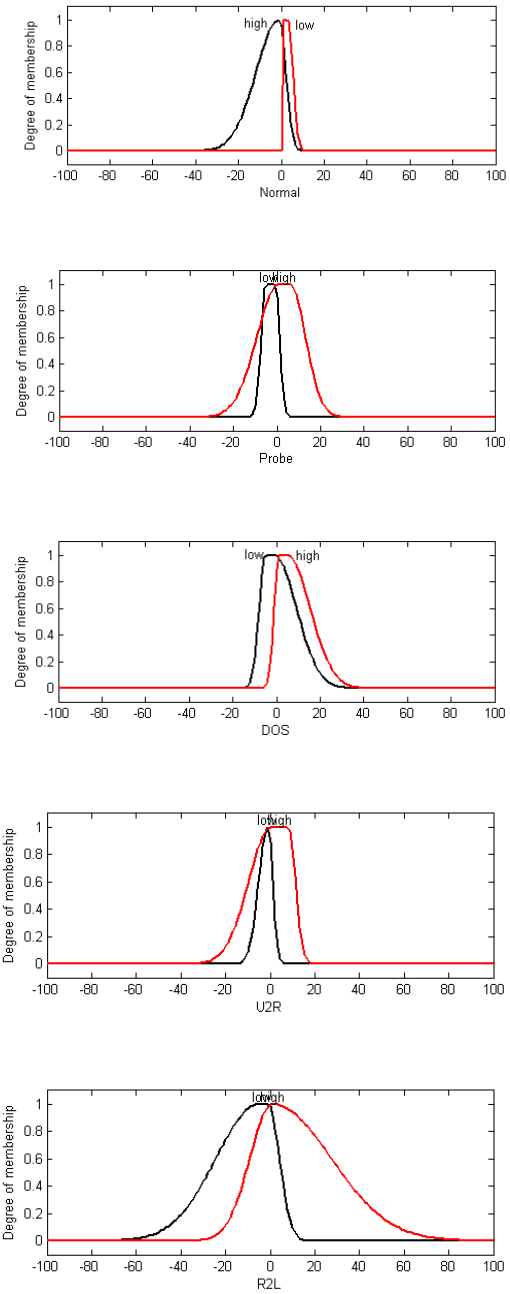
شکل ۳-۵: مقدار مینیمم و میانگین خروجی تابع شایستگی برای هر نسل در فرآیند بهینه سازی الگوریتم ژنتیک برای یک ساختار نمونه از سیستم تشخیص نفوذ ارائه شده

شکل توابع عضویت ماژول تصمیم گیری فازی قبل و بعد از فرآیند بهینه سازی توسط الگوریتم ژنتیک نشان می دهد.

قبل از بهینه سازی ژنتیکی



بعد از بهینه سازی ژنتیکی



شکل ۴-۵: توابع عضویت مجموعه های فازی ورودی برای موتور تصمیم گیری فازی قبل و بعد از بهینه سازی ژنتیک

از آنجائیکه ما به منظور کاهش اثرات انتخاب تصادفی ۱۰ انتخاب مختلف را برای سری اول و دوم داده های آموزشی ارائه شده در جداول ۱-۵ و ۲-۵ استفاده می کنیم و دو تابع شایستگی مختلف که هر کدام ۳ بار به ازاء هر یک سری از ۵ سری زیر مجموعه های انتخابی برای بهینه سازی اجرا می گردند در نهایت به ازاء هر یک از دو سری داده های آموزشی۱۵۰ ساختار مختلف حاصل می شود که نتایج بر اساس میانگین گیری از خروجی این ۱۵۰ ساختار مختلف انجام می گیرد.

۷ – ۵ نتیجه گیری

ما در این فصل ساختار و معماری سیستم تشخیص نفوذی را بر اساس مدل طبقه بندی ارائه کردیم که این ساختار از چند روش محاسبات نرم بهره می گیرد. این روشها شامل روش فازی- عصبی، سیستم استنتاج فازی و در نهایت الگوریتم ژنتیک می باشد. معماری ارائه شده شامل ۲ لایه اصلی است. لایه اول از پنج شبکه فازی- عصبی تطبیق پذیر که هر کدام به صورت یک طبقه بندی کننده دوگانه که رکورد های اتصال ورودی را به دو کلاس داده های متعلق و داده های غیر متعلق تقسیم بندی می کنند، تشکیل شده است. خروجی هر یک از این شبکه ها به عنوان ورودی یک سیستم استنتاج فازی پنج ورودی عمل می کنند، که در هر ورودی یک مجموعه فازی شامل دو تابع عضویت وجود دارد، تابع عضویت اول برای حد بالای تعلق به یک کلاس و خروجی دوم برای حد پایین تعلق به آن کلاس در نظر گرفته شده است. سیستم استنتاج فازی بر اساس یک سری قوانین تجربی تصمیم می گیرد که اتصال ورودی فعلی یک اتصال غیر قانونی است یا خیر. در صورتیکه یک رکورد اتصال به عنوان نفوذ یا حمله تشخیص داده شود بر اساس مکانیزم خاصی کلاس آن حمله مشخص می شود. طبقه کنندی فازی- عصبی لایه اول که بیشترین میزان نفوذی بودن را نشان می دهد کلاس نمونه جاری را مشخص می کند. به منظور بهینه سازی عملکرد موتور تصمیم گیری فازی الگوریتم ژنتیک با دو تابع شایستگی مختلف که بر اساس معیار هزینه برای هر نمونه عمل می کنند، توابع عضویت سیستم استنتاج فازی را بهینه سازی می کند. ما چهارچوبه ارائه شده در این فصل را که یک مدل طبقه بندی الگو است، سیستم تشخیص نفوذ بر اساس روش محاسبات نرم تکاملی یا به اختصار ESC-IDS می نامیم.

فصل پنجم: آزمایشات و بررسی نتایج

۱ – ۶ مقدمه

در این فصل سیستم تشخیص نفوذ بر اساس روش محاسبات نرم تکاملی مورد ارزیابی و نتایج حاصل با چند روش دیگر مورد مقایسه قرار می گیرد. ارزیابی سیستم بر اساس معیار های مطرح در زمینه تشخیص نفوذ انجام گرفته است. هدف اصلی از انجام این پایاننامه ارائه یک سیستم تشخیص نفوذ در سطح شبکه کامپیوتری بوده است، ولی لازم به ذکر است که سیستم تشخیص نفوذ بر اساس روش محاسبات نرم تکاملی به طور اساسی یک سیستم طبقه بندی الگو است که می تواند برای طبقه مجموعه داده هایی مختلف مورد استفاده قرار گیرد، با این فرض که داده هایی که قصد طبقه بندی آنها را داریم باید دارای این خصوصیت باشد که داده در سطح کلان به دو کلاس مختلف متعلق هستند و سپس داده ها در هر یک از دو کلاس اشاره شده به گروه های کوچکتری تقسیم بندی می شوند. نتایج ارزیابی سیستم که در ادامه این فصل آمده است نشان می دهد که سیستم تشخیص نفوذ بر اساس روش محاسبات نرم تکاملی می تواند در طراحی سیستم های تشخیص نفوذ بسیار موثر باشد و ما معتقدیم که این سیستم می تواند در زمینه تشخیص نفوذ بسیار بهتر از نتایج بدست آمده در این فصل عمل کند هر چند که دلیل قانع کننده ای برای این مدعا نداریم. آنچه مانع از نمایش حد نهایی توانایی های این روش می شود مشکلاتی است که بر داده های KDD که به عنوان مجموعه داده هایی که ارزیابی و مقایسه سیستم بر اساس آن انجام گرفته است وارد است [۱۸, ۳۳, ۳۸] و ما برای انجام مقایسه نا گزیر به استفاده از این مجموعه داده ها هستیم.

۲ – ۶ بررسی نتایج و ارزیابی سیستم

همانطور که در فصل ۲ به آن اشاره کردیم، داده های KDD cup 99 شامل زیر مجموعه است که به آن زیر مجموعه تست می گوییم این زیر مجموعه شامل ۳۱۱۰۲۹ رکورد اتصال می باشد که هر کدام به یک کلاس از چهار کلاس حمله Dos,U2R, Probe, R2l و کلاس داده های نرمال برچسب خورده اند. در این مجموعه داده حمله هایی وجود دارند که هیچ الگو مشابهی برای آنها در مجموعه داده های آموزش وجود ندارد و به آنها حمله های جدید می گوییم. توزیع الگو های اتصال و حمله های جدید در این مجموعه در جداول ۲-۳ و ۳-۳ فصل ۲ آورده شده است. در ادامه این فصل ما به ارزیابی سیستم تشخیص نفوذ ارائه شده در این پایان نامه بر اساس این مجموعه داده می پردازیم. همانطور که قبلاً اشاره کردیم ما از دو تابع شایستگی مختلف و دو سری داده آموزشی مختلف برای ایجاد سیستم استفاده کردیم بنابراین ۴ نسخه مختلف از این سیستم به وجود آمد که ما در ادامه این فصل از نام های اختصاری که در جدول ۱-٦ آمده است برای رجوع به آنها استفاده می کنیم.

جدول ۱-۶: نام های مستعار برای دو روش طبقه بندی استفاده شده

نام مستعار	روش
ESC-KDD-1	نسخه حاصل از آموزش با داده های سری اول و تابع شایستگی با ماتریس هزینه مورد استفاده در مسابقه KDD'99
ESC-EQU-1	نسخه حاصل از آموزش با داده های سری اول و تابع شایستگی با ماتریس هزینه با هزینه یکسان برای طبقه بندی نادرست
ESC-KDD-2	نسخه حاصل از آموزش با داده های سری دوم و تابع شایستگی با ماتریس هزینه مورد استفاده در مسابقه KDD'99
ESC-EQU-2	نسخه حاصل از آموزش با داده های سری دوم و تابع شایستگی با ماتریس هزینه با هزینه یکسان برای طبقه بندی نادرست

نتایج طبقه بندی مجموعه داده های تست براساس معیار های تعریف شده در فصل قبل و هر یک از نسخه های سیستم ESC-IDS در جدول ۲-۶ آمده است. جدول ۲-۶ نتایج قابل توجهی را برای نسخه های حاصل از آموزش با داده های سری دوم، با تعداد نمونه های آموزشی در حدود ۳۰۰۰۰ الگو که در آنها تکرار نیز وجود دارد و بسیار کمتر از کل نمونه های آموزشی می باشد، ارائه می کند. به عبارت دیگر با توجه به جدول می توان بیان نمود که با کم شدن الگو های آموزشی سیستم همچنان از سطح کارایی قابل قبولی برخوردار است.
از آنجا که ما نتایج جدول ۲-۶ از نتایج حاصل از ۱۵۰ ساختار مختلف اشاره شده میانگین گیری شده است مقدار واریانس هر یک از این مقادیر در جدول ۳-۶ ارائه شده است. علاوه بر این ما به عنوان نمونه ماتریس برهم ریختگی را که در درک بهتر عملکرد طبقه بندی کننده، موثر است برای یکی از ساختارهای حاصل از آزمایشات در جدول ارائه کرده ایم. جدول نشان می دهد که سیستم در کلاس U2R دارای میزان هشدار های غلط بسیار زیادی است و که می تواند عملکرد سیستم را تحت تاثیر قرار دهد. البته این اتفاق دور از انتظار ما نیست و علت آنرا می توان در پایین بودن تعداد نمونه های آموزشی در این کلاس حمله جستجو نمود.

جدول ۲-۶: در صد نرخ طبقه بندی، نرخ تشخیص(DTR)، نرخ هشدار های غلط(FA) و هزینه برای هر نمونه(CPE) در روشهای مختلف سیستم تشخیص نفوذ ESC-IDS برای داده های تست مجموعه داده های KDD'99

نسخه	Normal	Probe	DoS	U2R	R2L	DTR	FA	CPE
ESC-KDD-1	۹۸/۲	۸۴/۱	۹۹/۵	۱۴/۱	۳۱/۵	۹۵/۳	۱/۹	۰/۱۵۷۹
ESC-EQU-1	۹۸/۴	۸۹/۲	۹۹/۵	۱۲/۸	۲۷/۳	۹۵/۳	۱/۶	۰/۱۶۸۷
ESC-KDD-2	۹۶/۵	۷۹/۲	۹۶/۸	۸/۳	۱۳/۴	۹۱/۶	۳/۴	۰/۲۴۲۳
ESC-EQU-2	۹۶/۹	۷۹/۱	۹۶/۳	۸/۲	۱۳/۱	۸۸/۱	۳/۲	۰/۲۴۹۳

جدول ۳-۶: مقدار واریانس برای مقادیر محاسبه شده در جدول ۲-۶

نسخه	Normal	Probe	DOS	U2R	R2L	DTR	FA	CPE
ESC-KDD-1*	۱/۲۳	۱۱/۷٤	۰/۰۸	۵/۶۱	۱۱/۲۹	۰/۱۰	۱/۲۳	۱/۲۹
ESC-EQU-1*	۱/۰۴	۲۶/۱۵	۰/۰۹	۸/۱۹	۳۱/۷۴	۰/۱۶	۱/۰۴	۲/۸۴
ESC-KDD-2	۱/۱۹	۱۵/۷۲	۰/۰۶	۲/۶۴	۰/۱۳	۰/۷۱	۱/۱۹	۲/۰۱
ESC-EQU-2	۲/۱۷	۲۵/۹۵	۴/۶۴	۲/۸۳	۱/۰۴	۴/۴۳	۲/۱۷	۱/۰۴

* مقادیر سطر اول و دوم جدول همگی در E-4 ضرب شوند.

جدول ۴-۶: ماتریس برهم ریختگی برای داده های تست مجموعه داده های KDD'99 به ازاء نتایج حاصل از یک ساختار نمونه از سیستم تشخیص نفوذ ESC-IDS

		Predicted					دقت
		Normal	PROBE	DoS	U2R	R2L	
Actual	Normal	۵۸۸۰۹	۴۷۸	۲۵۱	۷۷۴	۲۸۱	٪۹۸/۴۷
	PROBE	۱۹۶	۳۵۴۱	۲۷۶	۴۹	۱۰۴	٪۶۴/۹۷
	DoS	۵۳۴	۴۹	۲۲۸۵۲۴	۶۴۱	۱۰۵	٪۹۹/۷۶
	U2R	۸۵	۶۴	۲۴	۲۹	۲۶	٪۱۶/۶۷
	R2L	۱۰۶۹۸	۲۲	۱۷	۵۶	۵۳۹۶	٪۳۱/۶۸
هشدار غلط		٪۱۶/۳۷	٪۱۴/۷۶	٪۰/۲۵	٪۹۸/۱۳	٪۸/۷۳	CPE=۰/۱۵۴۹

نتایج جداول فوق کارایی مناسب و در خور توجه سیستم ESC-IDS را نشان می دهد و مقدار CPE بدست آمده حاکی از عملکرد بسیار خوب این سیستم می باشد. در ادامه برای ارزیابی هر چه بیشتر رویکرد فوق ما سیستم ESC-IDS را با چندین روش یادیگیری ماشین که نتایج آزمایشات خود را بر روی داده های KDD ارائه کرده اند و همچنین دو شرکت کننده برتر در مسابقه KDD [۲۳, ۳۱] مقایسه کرده ایم. جدول ٥-٦ کارایی هر یک از این روشها را بر اساس معیار های مقایسه تعریف شده در فصل قبل مقایسه می کند. روش ارائه شده کارایی بهتری را در بعضی از کلاس های حمله نسبت به دیگر رقیبان ارائه می کند و مقدار بی سابقه CPE برابر با ۰/۱۵۷۹ بیانگر توانایی این سیستم در تشخیص نفوذ است. به راحتی از نتایج بدست آمده می توان استنباط نمود که سیستم ESC-IDS کارایی خوبی در تشخیص نفوذ در شبکه دارد و نرخ تشخیص در این سیستم نسبت به دیگر روشهای ارائه شده بیشتر است در حالیکه نرخ هشدار های غلط قابل قبولی را ارائه می دهد. ما ادعا می کنیم که نتایج جدول در بعضی از سطر ها می تواند غیر عادلانه باشد برای مثال دو روش Parzen-window [۳۸] و RSS-DSS [۳۴] روش های تشخیص آنامولی هستند و فقط قادر به تشخیص نفوذی یا عدم نفوذی بودن رکورد های اتصال هستند و هیچ اطلاعاتی را در مورد نوع حمله ارائه نمی کنند. بیان حقیقت فوق دلیلی برای این مدعا نیست که چون سیستم ما یک طبقه بندی چهار کلاسه را برای حملات ارائه می دهد روشهای مناسب تری می باشد، بلکه ما با بیان این مطلب قصد داریم توجه خواننده را به این نکته جلب کنیم که در روشهای فوق وقتی رکوردی به عنوان حمله شناخته می شود کلاس آن هر چه که باشد به عنوان یک الگوی طبقه بندی شده درست در آن کلاس قرار می گیرد ولی در روش ارائه شده در این

مطالعه بعضی از الگو ها علیرغم تشخیص درست به عنوان حمله ممکن است در کلاس حمله ای نادرست طبقه بندی شوند. به عبارت دیگر در روشهای فوق تشخیص یک حمله به معنی طبقه بندی درست است در حالیکه در روشهای دیگر که همانند روش ما عمل می کنند تشخیص حمله به معنای طبقه بندی درست نیست و امکان طبقه بندی اشتباه وجود دارد. اثبات این مدعا نرخ تشخیص حملات بالاتر در روش ارائه شده است که نشان می دهد روش ما در تشخیص حملات بسیار خوب عمل می نماید در حالیکه ممکن است در طبقه بندی کلاس دچار اشتباه شود.

در پایان نتایج جدول دلیلی بر این مدعا هستند که روش ESC-IDS یک روش جدید مناسب برای سیستم های تشخیص نفوذ می باشد که بر اساس ترکیب چند روش محاسبات نرم عمل می کند.

جدول ۵-۶: در صد نرخ طبقه بندی، نرخ تشخیص(DTR)، نرخ هشدار های غلط(FA) و هزینه برای هر نمونه(CPE) در الگوریتم های مختلف تشخیص نفوذ برای داده های تست مجموعه داده های KDD'99

CPE	FA	DTR	R2L	U2R	DoS	Probe	Normal	نسخه
0.1579	1.9	95.3	31.5	14.1	99.5	84.1	98.2	ESC-IDS
n/r	3.5	94.4	12.4	76.3	99.7	86.8	96.5	RSS-DSS [۳٤]
0.2024	n/r	n/r	31.2	93.6	96.7	99.2	97.4	Parzen-Window [۳۸]
0.2285	n/r	n/r	9.6	29.8	97.3	88.7	n/r	Multi-Classifier [۳۲]
0.2331	0.6	91.8	8.4	13.2	97.1	83.3	99.5	Winner of KDD [۳۱]
0.2356	0.6	91.5	7.3	11.8	97.5	84.5	99.4	Runner Up of KDD [۲۳]
0.2371	0.4	91.1	10.7	6.6	96.9	73.2	99.5	PNrule [۳]

۳ - ۶ نتیجه گیری

در این فصل ما به مقایسه نتایج حاصل از سیستم تشخیص نفوذ به روش محاسبات نرم تکاملی با استفاده از طبقه بندی کننده الگو فازی- عصبی پرداختیم. نتایج مناسب بودن این روش را در تشخیص نفوذ نشان می دهد و سیستم تشخیص نفوذ مقدار CPE را معیار برای ارزیابی سیستم های تشخیص نفوذ است به طور چشمگیری کاهش داده است. آنچه توجه بیش از پیش ما را به این سیستم جلب می کند توانایی این سیستم در تطبیق پذیری برای شرایط مورد نظر طراح سیستم است که در آن می توان با تعریف ماتریس هزینه های مختلف سیستم را برای شرایط مختلف بهینه نمود.

فصل ششم: نتیجه گیری و کار های آتی

۱ – ۷ اهداف پایان نامه

در این فصل اهداف پایان نامه به طور مختصر بیان خواهد شد و نتایج بدست آمده را مرور می کنیم و در انتها موضوعاتی که می‌تواند در ادامه کار این پایان نامه مطرح و مورد بررسی بیشتر قرار می گیرد مطرح خواهیم نمود.

به طور کلی در این پایاننامه یک روش تشخیص نفوذ در شبکه که بر اساس ترکیب چندین روش یادگیری ماشین و محاسبات نرم شکل گرفته است ارائه شد. ما از شبکه فازی- عصبی تطبیق پذیر در این سیستم استفاده نمودیم زیرا این این متد می تواند بدون نیاز به یک فرد خبره و بر اساس داده های نمونه سیستم را مدل کند. ما از سیستم استنتاج فازی که بر اساس خروجی شبکه های فازی-عصبی تصمیم گیری نهایی را انجام می دهد استفاده کردیم زیرا هر شبکه فازی- عصبی تطبیق پذیر یک خروجی بیشتر ندارد و نتایج بدست آمده نشان می دهد هر چه این خروجی به مقدار مورد نظر ما نزدیک تر باشد جواب های بهتری بدست خواهیم آورد و این در حقیقت انگیزه ای بود برای اینکه هر یک از این خروجی ها را به عنوان یک متغیر زبانی در نظر بگیریم. ما از چند طبقه بندی کننده فازی- عصبی که هر یک برای یک کلاس خاص از داده ها عمل می کنند استفاده کردیم چون آزمایشات نشان می دهد شبکه عصبی – فازی تطبیق پذیر در شکل یک طبقه بندی کننده دوگانه بهتر از یک طبقه بندی کننده چندگانه عمل می کند. و در نهایت ما از الگوریتم ژنتیک استفاده کردیم چون نمونه های آموزشی ما در کلاس های مختلف داده ها از توزیع متفاوتی برخوردار بودند و در نتیجه هر طبقه بندی کننده فازی- عصبی قدرت متفاوت داشت. الگوریتم ژنتیک این قابلیت را به سیستم می دهد که طبقه بندی کننده های ضعیف تر نقش کمرنگ تری را در تصمیم گیری نهایی ارائه کنند. نتایج مناسب بودن این سیستم ارائه شده را در تشخیص نفوذ نشان می دهد و سیستم تشخیص نفوذ مقدار CPE را که معیاری برای ارزیابی سیستم های تشخیص نفوذ است به طور چشمگیری کاهش داده است. علاوه بر این نرخ تشخیص را تا حد قابل قبولی بالا برده است در حالیکه نرخ هشدار های غلط در سطح مناسبی می باشد. آنچه توجه بیش از پیش ما را به این سیستم جلب می کند توانایی این سیستم در تطبیق پذیری برای شرایط مورد نظر طراح سیستم است که در آن می توان با تعریف ماتریس هزینه های مختلف سیستم را برای شرایط مختلف بهینه نمود.

۲ – ۷ تحقیقات بیشتر

ما از تمامی ۴۱ مشخصه موجود در داده های KDD برای آموزش استفاده کردیم در حالیکه روش های مختلفی برای انتخاب مجموعه کوچکتری از مشخصه ها برای آموزش سیستم وجود دارد که به آنها با نام روشهای انتخاب مشخصه می گوییم. یکی از مهمترین فعالیتهای آتی که در ادامه این پروژه قابل انجام است استفاده از روشهای مختلف در این زمینه به منظور کاهش مشخصه ها است.

در سیستم مورد نظر ما الگوریتم ژنتیک فقط در تنظیم پارامترهای توابع عضویت ورودی شرکت دارد در حالیکه این بهینه سازی می تواند تا حد دستکاری قوانین و پارامتر های خروجی یا نوع توابع عضویت پیش

روی کند و از اهداف آتی این پروژه می توان به دستکاری پارامتر های بیشتری از ماژول تصمیم گیری فازی اشاره نمود.

از فعالیتهای دیگری که در ادامه این پروژه می توان به آنها اشاره نمود استفاده از طبقه بندی کننده های دیگری مانند شبکه های عصبی یا ماشینهای پشتبانی بردار[۱] می باشد.

[۱] Support Vector Machine

مراجع

١. Abadeh M. S, Habibi J., Lucas C., "**Intrusion detection using a fuzzy genetics-based learning algorithm**", Journal of Network and Computer Applications, August 2005.

٢. Abraham A., Jain R., "**Soft Computing Models for Network Intrusion Detection Systems**", Journal of Soft Computing in Knowledge Discovery: Methods and Applications, Saman Halgamuge and Lipo Wang (Eds.), Studies in Fuzziness and Soft Computing, Springer Verlag Germany, Chapter 16, 20 pages, 2004.

٣. Agarwal R., Joshi M. V., PNrule: "**A New Framework for Learning classifier Models in Data Mining**", Technical Report TR 00-015, Depratment of Computer Science, University of Minnesota, 2000.

٤. Axelsson S., Intrusion detection systems: "**A survey and taxonomy, Department of Computer Engineering**", Chalmers University, Tech. Rep. 99-15, 2000.

٥. Chavan S., Shah K., Dave N., Mukherjee S., Abraham A. and Sanyal S., "**Adaptive Neuro-Fuzzy Intrusion Detection System**", IEEE International Conference on Information Technology: Coding and Computing (ITCC' 04), USA, IEEE Computer Society, Vol. 1, pp. 70-74, 2004.

٦. Chiu, S., "**Fuzzy Model Identification Based on Cluster Estimation**", Journal of Intelligent & Fuzzy Systems, Vol. 2, No. 3, September. 1994.

٧. DARPA Intrusion Detection Evaluation: http://www.ll.mit.edu/SSt/ideval/result/result_index.html.

٨. Debar H., Dacier M., Wespi A., "**Towards a taxonomy of intrusion detection systems**", Computer Networks, 31(8):805–822, April 1999.

٩. Denning D. E., "**An Intrusion Detection Model**", IEEE Transaction on Software Engineering, Vol. SE-13, No. 2,pp. 222-232,February 1987.

١٠. Dickerson J. E., Juslin J., Koukousoula O., and Dickerson J. A., "**Fuzzy Intrusion Detection**", In IFSA World Congress and 20th North American Fuzzy Information Processing Society (NAFIPS) International Conf., Volume 3, Vancouver, Canada, pp. 1506–1510, North American Fuzzy Information Processing Society (NAFIPS), July 2001.

١١. Gao, M. and Zhou M. C., **"Fuzzy intrusion detection based on fuzzy reasoning Petri nets**", Proceeding of the IEEE International Conference on Systems, Man and Cybernetics, Washington D. C., Oct. 5-8, pp. 1272 – 1277, 2003.

١٢. Goldberg, David E., "**Genetic Algorithms in Search"**, Optimization & Machine Learning, Addison-Wesley, 1989.

١٣. Gomez J., Dasgupta D., **"Evolving Fuzzy Classifiers for Intrusion Detection**", Proceeding Of 2002 IEEE Workshop on Information Assurance, United States Military Academy, West Point NY, June 2001.

١٤. Guan Y., Ghorbani A. and Belacel N., "**Y-means: A Clustering Method for Intrusion Detection**", Proceedings of Canadian Conference on Electrical and Computer Engineering. Montreal, Quebec, Canada. May 4-7, 2003.

١٥. Hofmann A., Horeis T., Sick B. "**Feature Selection for Intrusion Detection: An Evolutionary Wrapper Approach**", pp. 1563- 1568 vol.2, Proceedings of IEEE International Joint Conference on Neural Networks, 2004.

١٦. Ilgun K., Kemmerer R.A., and Porras P.A., "**State Transition Analysis: A Rule-Based Intrusion Detection Approach,**" IEEE Transaction on Software Engineering, Vol 2, No 3, 21(3), March 1995.

١٧. Ishibuchi H., Nakashima T., Murata T., "**A fuzzy classifier system that generates fuzzy if-then rules for pattern classification problems**", Proceedings of second IEEE international conference on evolutionary computation, Perth, Australia, November, pp. 759–64, 1995.

١٨. J. McHugh, Testing Intrusion Detection Systems: **"A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory**", Proceeding of ACM TISSEC 3(4) pp. 262-294, 2000.

١٩. Jang J.-S. R., **"ANFIS: Adaptive-Network-based Fuzzy Inference Systems",** IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 3, pp. 665-685, May 1993.

٢٠. KDD Cup 1999 Intrusion detection dataset: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

٢١. Laskov P., Dussel P., Schafer C., "**Learning intrusion detection supervised or unsupervised?",** Proceedings of ICIAP, pp. 50-57,2005.

٢٢. Lee W., Stolfo S.J., Mok K., "**A data mining framework for building intrusion detection models**", Proceedings of IEEE Symposium on Security and Privacy, pp 120 –132, 1999.

٢٣. Levin I., KDD-99 Classifier Learning Contest LLSoft's Results Overview, SIGKDD Explorations, ACM SIGKDD, 1(2) 67-75, 2000.

٢٤. Lippmann R., Haines J.W., Fried D. J.,Korba J., Das K., "**Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation**", . Recent Advances in Intrusion Detection 2000: 162-182, 2000.

٢٥. Liu J., Kwok J., **"An extended genetic rule induction algorithm**", Proceedings of the Congress on Evolutionary Computation Conference, 2000.

٢٦. M. Mahoney, P. K. Chan, "**An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection**", RAID 2003 pp. 220-237.

٢٧. Mamdani E. H., Assilian S., "**An experiment in linguistic synthesis with a fuzzy logic controller"**, International Journal of Man-Machine Studies,7(1):1-13, 1975.

٢٨. Mohajerani M.,Morini A.,Kianie M. "**NFIDS: A Neuro-Fuzzy Intrusion Detection System**", IEEE 2003.

٢٩. Mukkamala S., Sung A. H., "**Feature Ranking and Selection for Intrusion Detection Systems**", Proceedings of International Conference on Information and Knowledge Engineering, pp.503-509, 2002.

٣٠. Nauck D., Kruse r., **"NEFCLASS - A Neuro-Fuzzy approach for the classification of data"**, presented at the Symposium on applied Computing, Nashville, USA, 1995.

٣١. Pfahringer B., Winning the KDD99 Classification Cup: Bagged Boosting, SIGKDD explorations, 1(2), 65-66, 2000.

٣٢. Sabhnani M. R., Serpen G., **"Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context**", Proceedings of International Conference on Machine Learning: Models, Technologies, and Applications, Las Vegas, Nevada, 209-215, 2003.

٣٣. Sabhnani M. R., Serpen G., **"Why machine learning algorithms fail in misuse detection on KDD intrusion detection data set"**, Intelligent Data Analysis. Vol. 8, no. 4, pp. 403-415, 2004.

٣٤. Song D., Heywood M.I., Zincir-Heywood A.N., **"Training Genetic Programming on Half a Million Patterns: An Example from Anomaly Detection**", IEEE Transactions on Evolutionary Computation, 2005.

٣٥. Takagi T., Sugeno M., "**Fuzzy identification of systems and its applications to modeling and control"**, IEEE Transaction on Systems, Man, and Cybernetics, 15:116-132, 1985.

٣٦. Vladimir M., Alexei V., Ivan S., **"The MP13 Approach to the KDD'99 Classifier Learning Contest**", SIGKDD Explorations, ACM SIGKDD, 1(2) 76-77, 2000.

٣٧. Yager, R., D. Filev, "**Generation of Fuzzy Rules by Mountain Clustering, Journal of Intelligent & Fuzzy Systems**", Vol. 2, No. 3, pp. 209-219, 1994.

٣٨. Yeung D. Y. ,Chow C., **"Parzen-window Network Intrusion Detectors, Sixteenth International Conference on Pattern Recognition**", Quebec City, Canada, pp. 11-15, August 2002.

٣٩. Zadeh L. A., **"Role of Soft Computing and Fuzzy Logic in the Conception, Design and Development of Information/Intelligent Systems"**, Computational Intelligence: soft Computing and Fuzzy-Neuro Integration with Application, O. Kaynak, L.A. Zadeh, B. Turksen, I.J. Rudas(Eds.), pp 1-9, 1998.

٤٠. Zhang Z., Li J., Manikopoulos C., Jorgenson J. and Ucles J., **"HIDE: a Hierarchical Network Intrusion Detection System Using Statistical Preprocessing and Neural Network Classification"**, Proceedings of the 2nd Annual IEEE Systems, Mans, Cybernetics Information Assurance Workshop, West Point, NY, 2001.

ضمیمه

ضمیمه ۱: لیست مشخصه های موجود در رکوردهای اتصال در مجموعه *KDD*

Feature name	Type	Description
1. duration	continuous	length (number of seconds) of the connection
2. protocol_type	discrete	type of the protocol, e.g. tcp, udp, etc.
3. service	discrete	network service on the destination, e.g., http, telnet, etc.
4. src_bytes	continuous	number of data bytes from source to destination
5. dst_bytes	continuous	number of data bytes from destination to source
6. flag	discrete	normal or error status of the connection
7. land	discrete	1 if connection is from/to the same host/port; 0 otherwise
8. wrong_fragment	continuous	number of ``wrong'' fragments
9. urgent	continuous	number of urgent packets
10. hot	continuous	number of ``hot'' indicators
11. num_failed_logins	continuous	number of failed login attempts
12. logged_in	discrete	1 if successfully logged in; 0 otherwise
13. num_compromised	continuous	number of ``compromised'' conditions
14. root_shell	discrete	1 if root shell is obtained; 0 otherwise
15. su_attempted	discrete	1 if ``su root'' command attempted; 0 otherwise
16. num_root	continuous	number of ``root'' accesses
17. num_file_creations	continuous	number of file creation operations
18. num_shells	continuous	number of shell prompts
19. num_access_files	continuous	number of operations on access control files
20. num_outbound_cmds	continuous	number of outbound commands in an ftp session
21. is_hot_login	discrete	1 if the login belongs to the ``hot'' list; 0 otherwise
22. is_guest_login		1 if the login is a ``guest''login; 0 otherwise
23. count	continuous	number of connections to the same host as the current connection in the past two seconds
24. serror_rate	continuous	% of connections that have ``SYN'' errors
25. rerror_rate	continuous	% of connections that have ``REJ'' errors
26. same_srv_rate	continuous	% of connections to the same service
27. diff_srv_rate	continuous	% of connections to different services
28. srv_count	continuous	number of connections to the same service as the current connection in the past two seconds
29. srv_serror_rate	continuous	% of connections that have ``SYN'' errors
30. srv_rerror_rate	continuous	% of connections that have ``REJ'' errors
31. srv_diff_host_rate	continuous	% of connections to different hosts
32. dst_host_count	continuous	count for destination host
33. dst_host_srv_count	continuous	srv_count for destination host
34. dst_host_same_srv_rate	continuous	same_srv_rate for destination host
35. dst_host_diff_srv_rate	continuous	diff_srv_rate for destination host
36. dst_host_same_src_port_rate	continuous	same_src_port_rate for destination host
37. dst_host_diff_host_rate	continuous	diff_host_rate for destination host
38. dst_host_serror_rate	continuous	serror_rate for destination host
39. dst_host_srv_serror_rate	continuous	srv_serror_rate for destination host

40. dst_host_rerror_rate	continuous	rerror_rate for destination host
41. dst_host_srv_rerror_rate	continuous	srv_serror_rate for destination host

Abstract

An intrusion detection system's main goal is to classify activities of a system into two major categories: normal and suspicious (intrusive) activities. Intrusion detection systems usually specify the type of attack or classify activities in some specific groups. The objective of this thesis is to incorporate several soft computing techniques into the classifying system to detect and classify intrusions from normal behaviors based on the attack type in a computer network. Among the several soft computing paradigms, neuro-fuzzy networks, fuzzy inference approach and genetic algorithms are investigated in this work. A set of parallel neuro-fuzzy classifiers are used to do an initial classification. The fuzzy inference system would then be based on the outputs of neuro fuzzy classifiers, making final decision of whether the current activity is normal or intrusive. Finally, in order to attain the best result, genetic algorithm optimizes the structure of our fuzzy decision engine. The experiments and evaluations of the proposed method were performed with the KDD Cup 99 intrusion detection dataset. This thesis shows that our proposed method can be effective in intrusion detection compared with similar models.