

## بهره گیری از طبقه‌بندی‌کننده‌ی عصبی - فازی برای سیستم‌های تشخیص نفوذ

محمد حسن نتاج صلحدار<sup>1</sup>

تاریخ دریافت: ۱۳۹۹/۱۲/۰۸

تاریخ پذیرش: ۱۴۰۰/۰۶/۲۲

### چکیده

یکی از ابزارهای هوش مصنوعی، سیستم استنتاج تطبیق‌پذیر عصبی - فازی است، که در این مقاله برای ساخت سیستم تشخیص نفوذ مورد استفاده قرار گرفته است و ما آن را دسته‌بندی‌کننده‌ی عصبی - فازی می‌نامیم. سیستم تشخیص نفوذ مبتنی بر طبقه‌بندی‌کننده شبکه عصبی-فازی یک سیستم تشخیص نفوذ مبتنی بر ناهنجاری است که از منطق فازی و شبکه عصبی برای تشخیص اینکه فعالیت مخرب در یک شبکه انجام می‌شود، استفاده می‌کند. در این مقاله به تشریح معماری دسته‌بندی‌کننده‌ی عصبی - فازی و مؤلفه‌های آن می‌پردازیم. قوانین فازی نمونه برای برخی از حملات ایجاد شده و نتایج آزمایش با داده‌های واقعی شبکه شرح داده شده است. آزمایشات و ارزیابی‌های ما با مجموعه NSLKDD مجموعه داده‌های تشخیص نفوذ انجام شده است که نسخه‌ای از مجموعه داده‌های ارزیابی نفوذ KDD Cup99 است که توسط آزمایشگاه‌های MIT Lincoln تهیه و مدیریت شده است. در نهایت، این مقاله سعی بر آن دارد تا با بررسی عملکرد مدل "سیستم استنتاج تطبیق‌پذیر عصبی - فازی" بر روی یک مجموعه استاندارد و جامع میزان کارایی مدل طراحی شده را نشان دهد.

واژگان کلیدی: سیستم تشخیص نفوذ، شبکه عصبی، طبقه‌بندی‌کننده‌ی عصبی - فازی، انفیس، NSLKDD

<sup>1</sup> دانشگاه شهید چمران اهواز - پردیس صنعتی شهدای هوپزه، اهواز، ایران. مری. [n.solhdar@scu.ac.ir](mailto:n.solhdar@scu.ac.ir)  
نویسنده مسئول: محمد حسن نتاج صلحدار

## مقدمه

سیستم‌های تشخیص نفوذ<sup>1</sup> (IDS) در حال حاضر جزء اصلی‌ترین و کامل‌ترین قسمت‌های یک سیستم پایش شبکه می‌باشند. سیستم‌های تشخیص نفوذ فناوری‌های تقریباً جدیدی هستند و این نوید را به ما می‌دهند که به ما در جهت شناسایی نفوذهایی که به شبکه انجام می‌شود کمک خواهند کرد [1]. تشخیص نفوذ در واقع فرآیندی است که در آن رویدادها و رخدادها را یک سیستم یا شبکه پایش شده و براساس این پایش‌ها وقوع نفوذ به آن شبکه یا سیستم تشخیص داده می‌شود. یک نفوذ در واقع فعالیت یا عملی است که توسط آن محرمانگی، صحت و تمامیت و یا دسترس پذیری به منابع دچار اختلال و یا تعرض می‌شود [2].

IDSها نیاز به جمع آوری داده‌های کافی برای ساختن یک مدل ریاضی پیچیده دارند، که در مورد ترافیک شبکه پیچیده غیر عملی است [3]. و همچنین تشخیص نفوذهای جدید برای IDSها کار دشواری است، و پایگاه داده مبتنی بر امضا، باید به صورت دستی و مکرر به روز شود [4]. برای حل محدودیت‌های روش‌های فوق، تعدادی از تکنیک‌های داده‌کاوی معرفی شده‌اند [5]. در بین این تکنیک‌ها، شبکه عصبی یکی از تکنیک‌های است که بسیار مورد استفاده قرار می‌گیرد و با موفقیت در تشخیص نفوذ به کار گرفته شده است [6]. با توجه به انواع مختلف شبکه عصبی، این تکنیک‌ها را می‌توان در سه دسته زیر طبقه‌بندی کرد: تشخیص نفوذ مبتنی بر شبکه عصبی نظارت شده، تشخیص نفوذ مبتنی بر شبکه عصبی بدون نظارت و تشخیص نفوذی مبتنی بر شبکه عصبی ترکیبی.

در این پژوهش، از سیستم استنتاج تطبیق پذیر عصبی - فازی، برای ساخت سیستم تشخیص نفوذ استفاده شده است. این سیستم برای بررسی مسائلی با معادلات غیرخطی بسیار کاربردی است [7]. دسته‌بندی‌کننده‌ی عصبی - فازی، ترکیبی از شبکه‌ی عصبی و سیستم استنتاج فازی است. یکی از مشکلات سیستم استنتاج فازی، عدم قابلیت یادگیری است. از طرف دیگر، شبکه‌ی عصبی قابلیت یادگیری دارد و می‌تواند بارها و بارها آموزش دیده و خود را مطابق آموزش‌های جدید برنامه‌ریزی کند. با ترکیب شبکه‌ی عصبی با قابلیت یادگیری و سیستم استنتاج فازی، به یک شبکه‌ی یادگیرنده می‌رسیم که هم قابلیت یادگیری دارد و هم قابلیت کارکردن با قوانینی دارد که مطابق قوانین سیستم استنتاج فازی هستند. همچنین در این تحقیق سعی شده است از شبکه عصبی - فازی به عنوان یک طبقه‌بندی‌کننده استفاده شود. در این مدل ابتدا قوانین فازی لازم بدون نیاز به یک فرد خبره و با استفاده از روش خوشه‌بندی کاهشی بر اساس مجموعه داده‌های آموزشی<sup>2</sup> انتخاب شده از مجموعه داده‌های NSLKDD<sup>3</sup> تولید می‌شود، سپس قوانین فازی بدست آمده برای ایجاد یک شبکه عصبی - فازی تطبیق پذیر به نام انفیس<sup>4</sup> به کار گرفته می‌شوند. اندازه‌ی قدرت هر قانون فازی در طی فرآیند آموزش شبکه‌ی عصبی - فازی محاسبه و در طول پروسه‌ی آموزش، با تنظیم مجدد توابع عضویت<sup>5</sup> که سازنده‌ی قوانین فازی هستند، شبکه به سمت خطای کمتر گام برمی‌دارد. در نهایت مدل بدست آمده به عنوان طبقه‌بندی‌کننده، برای کلاس‌بندی داده‌های آزمایشی<sup>6</sup> که همانند داده‌های آموزشی از مجموعه داده‌های NSLKDD استخراج شده‌اند به کار گرفته می‌شود. عدم کفایت یا مناسب بودن این

<sup>1</sup>Intrusion Detection Systems (IDS)

<sup>2</sup>Learning data

<sup>3</sup>International Conference on Knowledge Discovery and Data Mining

<sup>4</sup>Adaptive Network-Based Fuzzy Inference System (ANFIS)

<sup>5</sup>Membership Function .

<sup>6</sup>Testing data

مدل برای انجام عملیات طبقه‌بندی مورد بررسی قرار می‌گیرد. طبقه‌بندی‌کننده‌ی ارائه شده، به صورت یک طبقه‌بندی‌کننده‌ی دوگانه که رکوردهای اتصال را به دو کلاس نفوذ و عادی تقسیم می‌کند، استفاده خواهد شد و داده‌های آموزش و آزمایش مورد طبقه‌بندی قرار خواهند گرفت. در ادامه این مقاله کارهای که در این حوزه انجام شده، آورده شده است و در روش پیشنهادی الگوریتم دسته‌بندی‌کننده فازی، توضیح داده شده است و در ادامه پیاده‌سازی و نتایج آماری آورده شده است و بعد از آن در مورد معیارهای ارزیابی که در این تحقیق از آن استفاده شد توضیحاتی داده شده است و در انتها مقایسه و نتیجه‌گیری آمده است.

### کارهای انجام شده

اخیراً، روش‌های محاسبات نرم برای سیستم‌های تشخیص نفوذ استفاده می‌شود. برخی از این روشها در مجموعه داده KDD مورد ارزیابی قرار گرفته است. طبقه‌بندی‌کننده‌های مبتنی بر قانون فازی<sup>1</sup>، درخت تصمیم‌گیری<sup>2</sup>، ماشین‌های بردار پشتیبانی<sup>3</sup>، برنامه‌نویسی ژنتیکی<sup>4</sup> خطی در [8] مورد استفاده قرار گرفته‌شده‌اند تا اهمیت نمونه‌های محاسبات نرم<sup>5</sup> را برای مدل‌سازی سیستم‌های تشخیص نفوذ نشان دهند.

IDS مبتنی بر آماری، از روشهای آماری متنوعی از جمله تجزیه و تحلیل مؤلفه‌های اصلی<sup>6</sup>، تجزیه و تحلیل خوشه‌ای و چند متغیره، آنالیز بیزی استفاده می‌کند. فارون و بوکلیف<sup>7</sup> and در تحقیق خود با استفاده از خوشه‌بندی K-means سعی در بهبود آموزش شبکه عصبی کردن به نرخ تشخیص خطا 92 درصدی و نرخ هشدار غلطی 6,21 درصدی دست یافته‌اند [9].

دش<sup>8</sup> در مقاله خود [10] نشان داد که نتایج آزمون شبکه عصبی مصنوعی پیشنهادی قادر به تشخیص 94,9% حملات هستند. جونالاگادا و رد<sup>9</sup> در تحقیقات خود با به کارگیری دسته‌بندی‌کننده‌های عصبی - فازی در یک معماری دو لایه موفق شدند به نرخ تشخیص خطا 95,3% و نرخ هشدار غلطی 1,9% دست یابند [11]. در [12] از روش TLM<sup>10</sup> برای سیستم تشخیص نفوذ استفاده شده است و سعی کرد داده‌های پارازیت دار<sup>11</sup> را کاهش دهد. امبوسایدی<sup>12</sup> و همکارانش [13] نیز از روشهای ماشین بردار پشتیبان خاص با نام LSSTM<sup>13</sup> برای تشخیص نفوذ استفاده کردند. بقداد<sup>14</sup> معتقد است، عملکرد شبکه عصبی بدون نظارت پایین است. به خصوص برای حملات با تعداد کم، شبکه عصبی بدون نظارت همچنین از دقت کمتری برخوردار است. ایشان با به کارگیری شبکه‌های عصبی و ساخت شبکه با پیکربندی‌های مختلف و آموزش - های متعدد شبکه به نرخ تشخیص خطا 84,25 درصد و نرخ هشدار غلطی 15,75 درصد دست یافت [14].

شبکه عصبی ترکیبی، که ترکیبی از شبکه عصبی تحت نظارت<sup>15</sup> و شبکه عصبی بدون نظارت<sup>16</sup> و یا ترکیب شبکه عصبی با سایر تکنیک‌های داده‌کاوی برای تشخیص نفوذ است، یکی دیگر از نمونه‌های محاسبات نرم در سیستم‌های تشخیص نفوذ است [15]. انگیزه استفاده از شبکه عصبی ترکیبی غلبه بر محدودیتهای شبکه‌های عصبی تنها است. جیراپومین<sup>17</sup> و همکاران [16] استفاده از شبکه عصبی ترکیبی را هم برای رویت هرگونه حمله با استفاده از SOM کوهن<sup>18</sup> و هم برای طبقه‌بندی حمله‌ها با استفاده از شبکه‌های عصبی انتشار انعطاف پذیر پیشنهاد کردند. هوریس<sup>19</sup>

Noise  
 1Ambusaidi  
 2Least Square Support Vector Machine  
 3Baghdad  
 4Supervised  
 5Unsupervised  
 6Jirapummin  
 7Kohenen  
 8Foreis

1Fuzzy rule  
 2Decision tree  
 3Support vector machine  
 4genetic programing  
 5Soft computing  
 6Principal Component Analysis - PCA  
 7Faraoun and Boukelif  
 8Dash  
 9onnalagadda and Redd  
 10Two Layers Multi-class Detection

از ترکیبی SOM<sup>1</sup> و شبکه های تابعی (RBF<sup>3</sup>) استفاده کرد. این سیستم به طور کلی نتایج بهتری را از IDS مبتنی بر شبکه های RBF ارائه می دهد [17]. هان و چو<sup>3</sup> برای تعیین ساختار و وزن توالی های تماس، یک روش تشخیص نفوذ را بر اساس شبکه های عصبی تکاملی پیشنهاد کردند [18]. چن<sup>4</sup> و همکارانش IDS مبتنی بر درخت عصبی انعطاف پذیر ترکیبی را بر اساس درخت عصبی انعطاف پذیر، الگوریتم تکاملی و بهینه سازی ذرات (PSO) پیشنهاد دادند. نتایج تجربی نشان داد که روش پیشنهادی کارآمد است [19]. طوسی و کاهانی<sup>6</sup> [20] و قوش<sup>7</sup> و همکارانش [21] در تحقیقات خود برای سیستم تشخیص نفوذ پیشنهادی با استفاده از شبکه فازی به نرخ های تشخیص بهتر از 95,3% و 91,1% دست نیافتند. در ادامه روش پیشنهادی بیان می شود و در قسمت ارزیابی نتایج با چند روش دیگر مقایسه خواهد گردید.

## روش پیشنهادی

شبکه ی عصبی مصنوعی و منطق فازی، هر دو ابزارهای هوش مصنوعی هستند که می توانند برای ساخت یک سیستم هوشمند دیگر، مکمل یکدیگر باشند. شبکه ی عصبی مصنوعی ساختار محاسباتی سطح پایینی است که به خوبی با داده های خام اولیه کار می کند. در مقابل منطق فازی با استدلال های سطح بالایی که با استفاده از دانش یک متخصص در یک حوزه ی خاص بدست آمده اند، سروکار دارد.

یکی از معایب سیستم فازی، عدم توانایی یادگیری است که سبب می شود این سیستم ها قادر به تطبیق و تنظیم خود با محیط جدید نباشند. از طرف دیگر، اگرچه شبکه ی عصبی مصنوعی توانایی یادگیری دارد، اما برای انسان غیرشفاف و مانند یک جعبه سیاه<sup>8</sup> می باشد.

ادغام شبکه ی عصبی مصنوعی با سیستم فازی، نوید یک سیستم کامل تر نسبت به هر دو آن ها را می دهد. سیستم ترکیبی عصبی - فازی، می تواند توانایی های شبکه ی عصبی از قبیل پردازش موازی و قابلیت یادگیری را با ویژگی های سیستم فازی مانند توضیح پذیری و انسان پسندانه تر بودن، ادغام نماید. در نتیجه، شبکه ی عصبی شفاف تر می شود و سیستم فازی، قابلیت یادگیری پیدا می کند [22].

یک از انواع شبکه ی عصبی - فازی، از ترکیب شبکه ی عصبی مصنوعی با فازی سوجینو<sup>9</sup> شکل می گیرد. در سال 1993 ژانگ<sup>10</sup> برای اولین بار با مدنظر قرار دادن توانایی های تئوری فازی که مبتنی بر قواعد منطقی بوده و همچنین روش شبکه عصبی مصنوعی که توانایی استخراج دانش از اطلاعات عددی را دارند، سیستم استنتاج تطبیقی عصبی - فازی را ارائه نمود [23]. سیستم ارائه شده توسط ژانگ، انفیس خوانده می شود. اصطلاح انفیس مخفف عبارت، Adaptive Nero Fuzzy Inference System است که به اختصار، روش استنتاج فازی عصبی تطبیقی نیز نامیده می شود. در این تحقیق ما از انفیس برای ساخت سیستم تشخیص نفوذ استفاده و از عنوان دسته بندی کننده ی عصبی - فازی برای آن استفاده خواهیم کرد. دسته بندی کننده ی عصبی - فازی، به طور گسترده برای بررسی پدیده های با معادلات غیرخطی به کار گرفته شده است. بنابراین، ترکیب سیستم های فازی که بر قواعد منطقی استوار هستند و روش شبکه های عصبی مصنوعی که توان استخراج دانش از اطلاعات عددی را دارند، ما را قادر می سازد تا بتوانیم در کنار استفاده از دانش بشری از اطلاعات موجود نیز، در ساخت مدل استفاده کنیم. یکی از مزایای استفاده از سیستم انفیس این است که پس آموزش شبکه با استفاده نمونه های

<sup>7</sup>Ghosh

<sup>8</sup>Black Box

<sup>9</sup>Sugeno

<sup>10</sup>Jang

<sup>1</sup>Self-organization map

<sup>2</sup>Radial base function

<sup>3</sup>Han and Cho

<sup>4</sup>Chen

<sup>5</sup>Particle Swarm Optimization

<sup>6</sup>Toosi and Kahani

لایه ی 3: این لایه، لایه ی قوانین است. هر گره در این لایه، یک قانون فازی را نشان می دهد. هر گره در این لایه، ورودی خود را از خروجی های متناظر در لایه ی قبل می گیرد و خروجی آن قدرت آتش هر قانون می باشد که به صورت زیر محاسبه می شود:

$$y_i^{(3)} = \prod_{j=1}^k x_{ji}^{(3)} \quad (3)$$

که در آن  $x_{ji}^{(3)}$  ورودی و  $y_i^{(3)}$  خروجی گره  $i$ ام در لایه 3ام می باشد و  $k$  تعداد خروجی لایه دوم به لایه سوم است. لایه ی 4: لایه ی 4 لایه ی نرمالیزه سازی می باشد. در این لایه هر گره ورودی خود را از تمام گره های لایه ی قبل می گیرد و عدد بدست آمده برای هر گره در لایه ی قبل، در این لایه نرمالیزه می شود. خروجی هر گره در این لایه، قدرت قانون نرمالیزه شده ی گره متناظر لایه ی قبل می باشد. قدرت آتش نرمالیزه شده ی هر قانون، از تقسیم قدرت آتش هر قانون به جمع کل قدرت آتش قانون ها به دست می آید. قدرت قانون نرمالیزه شده میزان نفوذ هر قانون در خروجی شبکه را نشان می دهد.

$$y_i^{(4)} = \frac{x_{ii}^{(4)}}{\sum_{j=1}^n x_{ji}^{(4)}} = \frac{\mu_i}{\sum_{j=1}^n \mu_j} \quad (4)$$

در معادله ی بالا،  $x_{ji}^{(4)}$  خارج شده از گره  $j$  در لایه سوم به ورودی گروه  $i$  در لایه چهارم است،  $y_i^{(4)}$  خروجی گره  $i$ ام در لایه چهارم است و  $n$  تعداد کل قانونها می باشد. و  $\mu$  همان قدرت آتش هر گره است که در لایه ی قبل محاسبه شده است و ورودی لایه ی چهارم به حساب می آید.

لایه ی 5: این لایه، لایه ی غیرفازی سازی نام دارد. هر گره در این لایه، ورودی خود را از گره متناظر در لایه ی قبل می گیرد. مقدار هر گره در این لایه با فرض مرتبه صفر بودن عبارتست از:

آموزشی، سیستم نتایج حاصله را در قالب قوانین فازی که به صورت محاوره ای، قابل فهم و تحلیل می باشند، ارائه می دهد. این ویژگی می تواند در ارزیابی قوانین و مقایسه آنها با مفاهیم و قوانین افراد خبره حائز اهمیت باشد. در سیستم فازی سوجینو قوانین به صورت زیر استنتاج می شوند:

IF  $x_1$  is  $A_1$   
AND  $x_2$  is  $A_2$   
.....  
AND  $x_m$  is  $A_m$   
THEN  $y = f(x_1, x_2, \dots, x_m)$

### لایه های دسته بندی کننده ی عصبی - فازی

در این بخش به تشریح لایه های انفیس خواهیم پرداخت. ورودی و خروجی هر لایه مشخص شده و رابطه ی مرتبط با آن نیز بیان می شود [22].

لایه ی 1: لایه ی 1 لایه ی ورودی است. گره ها در این لایه داده ها را برای لایه ی 2 آماده می کنند. در این لایه هیچ تغییری روی داده ها صورت نمی گیرد بطوری که ورودی با خروجی برابر است.

$$y_i^{(1)} = x_i^{(1)} \quad (1)$$

لایه ی 2: در این لایه عمل فازی سازی روی داده ها انجام می شود. در مدل ژانگ، تابع فعالیت برای عمل فازی سازی داده ها، تابع زنگوله است که معادله ی آن به صورت زیر است:

$$y_i^{(2)} = \frac{1}{1 + \left(\frac{x_i^{(2)} - a_i}{c_i}\right)^{2b_i}} \quad (2)$$

در این معادله،  $x_i^{(2)}$  ورودی و  $y_i^{(2)}$  خروجی گره  $i$ ام در لایه دوم است و پارامترهای  $a$  و  $b$  و  $c$  به ترتیب شیب، عرض و مرکز تابع فعالیت می باشند.

روبه جلو پارامترهای پایه ثابت هستند و ورودی آنها از محاسبات هر پنج لایه عبور می کنند و پارامترهای متعاقب با روش حداقل مربع تشخیص داده می شوند در آن نقطه پارامترهای بعدی در علائم خطای کانال معکوس تعیین می شوند و برای به روز رسانی پارامترها به لایه 2 منتقل می شوند.

فرآیند گام به گام اجرای الگوریتم جستجوی بهینه به این صورت است:

مرحله 1: راه اندازی مشکل و پارامتر قابل تطبیق

مسئله بهینه سازی، متغیرهای تصمیم گیری و محدودیت ها شرح داده شده است.

مرحله 2: مکان و حافظه عامل را راه اندازی می کند.

مرحله 3: محاسبه تابع کیفیت

مرحله 4: مکان جدید ایجاد می کند

مرحله 5: امکان مکان جدید را بررسی می کند

اگر مکان جدید امکان پذیر بود، عامل مکان خود را ارتقا می دهد.

در غیر این صورت، عامل در مکان فعلی می ماند و به مکان جدید ایجاد شده نمی رود.

مرحله 6: محاسبه تابع هدف مکان جدید

مقدار تابع هدف برای مکان جدید هر عامل ارزیابی شد.

مرحله 7: حافظه را به روز می کند

مرحله 8: معیارهای خاتمه را تأیید می کند

مراحل 4 تا 7 تا زمانی که حداکثر تکرار مشخص شده، انجام شود، ادامه می یابد.

$$y_i^{(5)} = x_i^{(5)} [k_{i0}] = \bar{\mu}_i [k_{i0}] \quad (5)$$

و با فرض مرتبه 1 بودن انفیس با در نظر گرفتن دو ورودی  $x_1$  و  $x_2$  از رابطه زیر به دست می آید:

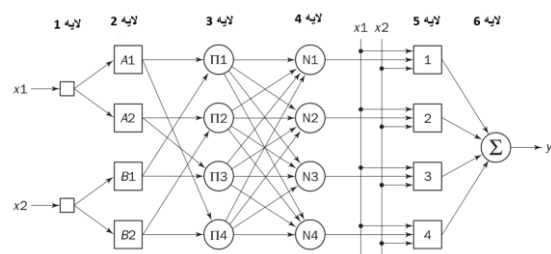
$$y_i^{(5)} = x_i^{(5)} [k_{i0} + k_{i1}x_1 + k_{i2}x_2] = \bar{\mu}_i [k_{i0} + k_{i1}x_1 + k_{i2}x_2] \quad (6)$$

که در این رابطه،  $x$  مقدار ورودی به هر گره،  $y$  خروجی گره و  $k_{ij}$  ها پارامترهای مربوط به قانون  $\mu_i$  می باشد. چنانچه انفیس از مرتبه 1 صفر باشد به جای  $x_1$  و  $x_2$  صفر قرار داده می شود.

لایه 6: لایه 6 تنها دارای یک گره می باشد که جمع گره های غیرفازی شده ی لایه 5 قبل را محاسبه می کند و رابطه ی آن به صورت زیر است:

$$y = \sum_{i=1}^n x_i^{(6)} \quad (7)$$

انفیس نشان داده شده در شکل 1 در واقع از لحاظ عملکردی معادل یک مدل فازی مرتبه اول سوچینو است.

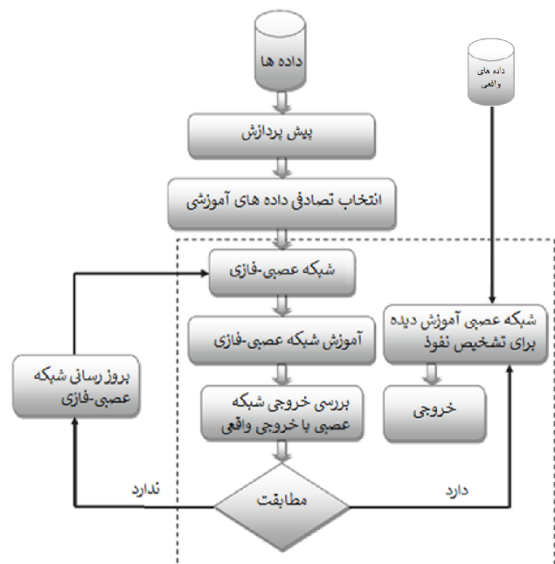


شکل 1: انفیس

## بهینه سازی پارامترهای دسته بندی کننده ی عصبی - فازی

برای افزایش دقت پیش بینی ANFIS و پرهیز از قرار گرفتن در حد بهینه محلی، یادگیری پارامتر توسط الگوریتم جستجوی بهینه انجام می شود [24]. روش یادگیری پارامتر ANFIS را می توان به عنوان تنظیم پارامترهای پایه و پارامترهای متعاقب آن تشریح کرد، الگوریتم جستجوی بهینه برای آموزش پارامترهایی که شامل یک کانال رو به جلو و یک کانال معکوس است، استفاده می شود. در کانال

ایده‌ی اصلی برای استفاده از داده‌های بررسی به منظور معتبرسازی مدل، این است که بعد از مرحله‌ی خاصی از آموزش امکان بوجود آمدن یادگیری بیش از اندازه وجود دارد. اگر این آموزش بیش از اندازه اتفاق بیافتد، نمی‌توان انتظار داشت که سیستم، نسبت به داده‌های آزمایشی که جدا از داده‌های آموزش هستند به خوبی پاسخ دهد. بنابراین برای جلوگیری از آموزش بیش از اندازه‌ی داده‌ها، از مجموعه‌داده‌های بررسی استفاده می‌کنیم. پس از اتمام فرایند آموزش، سیستم عصبی - فازی تطبیق‌پذیر با مینیمم خطا برای داده‌های بررسی به عنوان سیستم نهایی انتخاب و شروع به طبقه‌بندی می‌کنیم. شمای کلی روش پیشنهادی در شکل 2 آمده است.



شکل 2. شمای کلی روش پیشنهادی

برای تولید سیستم فازی اولیه، از روش خوشه‌بندی کاهش‌ی با شعاع همسایگی 0.5 استفاده شده است. برای این کار، از تابع  $genfis2$  که تابعی از برنامه‌ی متلب برای تولید قوانین فازی به روش خوشه‌بندی کاهش‌ی است، استفاده می‌شود. پارامترهای این تابع، مجموعه‌ی داده‌های آموزشی ورودی، خروجی و شعاع همسایگی است. همانطور که ذکر شد از 20000 داده‌ی آموزشی برای آموزش دسته‌بندی‌کننده

هنگامی که سطح پایان به دست آمد، بهترین مکان حافظه نسبت به مقدار تابع هدف به عنوان راه حل مسئله در نظر گرفته شد.

## پیاده سازی و نتایج آماری

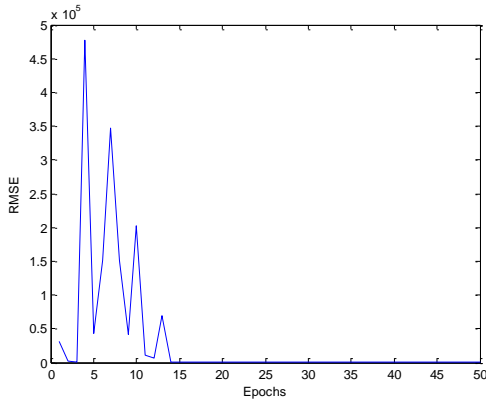
همانطور که در بخش‌های قبلی اشاره کردیم 41 ویژگی برای هر اتصال در مجموعه‌داده‌های NSLKDD وجود دارد. این ویژگی‌ها دارای سه فرم پیوسته، گسسته و نمادین هستند. فهرست کاملی از نام و دامنه‌ی مقادیر هر یک از ویژگی‌ها در ضمیمه 1 آمده است. طبقه‌بندی‌کننده‌ی مورد استفاده‌ی ما نمی‌تواند داده‌ها را به این صورت پردازش کند. چرا که شبکه‌ی عصبی - فازی مورد استفاده در این تحقیق توانایی استفاده از داده‌های نمادین را ندارد. بنابراین قبل از ساختن مدل نیاز به پیش پردازشی روی داده‌ها داریم که این پیش پردازش در حقیقت تبدیل مقادیر نمادین به مقادیر عددی است. در این پیش پردازش ویژگی‌های نمادین که شامل انواع پروتکل‌ها، خدمات و پرچم‌ها است به مقادیر عددی در بازه‌ی 0 تا  $N-1$  نگاشت می‌شوند که در آن  $N$  تعداد نمادهای هر ویژگی نمادین است. به عنوان مثال ویژگی Protocol\_type با سه نماد UDP, TCP و ICMP به سه عدد 0, 1 و 2 نگاشت می‌شود. بقیه ویژگی‌ها به همان صورت اولیه استفاده می‌شوند. علاوه بر این، هر اتصال بسته به نوع خود که عادی یا نفوذ می‌باشد، با دو کلاس 0 و 1 برچسب می‌خورد. سپس 20000 داده‌ی تصادفی از مجموعه‌ی داده‌های آموزشی NSLKDD به عنوان داده‌های آموزشی و 3188 داده‌ی تصادفی از همین مجموعه داده‌ها به عنوان داده‌های بررسی انتخاب شدند که هیچ اشتراکی با داده‌های قسمت اول ندارند.

استفاده خواهد شد. ابتدا توسط این 20000 مجموعه داده برای تولید قوانین فازی استفاده خواهد شد. این مجموعه‌ی 20000 تایی، شامل 20000 رکورد داده است. هر رکورد نشان‌دهنده‌ی یک اتصال است و با 41 ویژگی مشخص می‌گردد. یعنی 20000 ردیف داده که هر ردیف شامل 41 ویژگی است. همچنین برای هر اتصال یک عدد وجود دارد که نشان‌دهنده‌ی نوع اتصال است. نوع اتصال می‌تواند عادی و یا نفوذ باشد که اتصال عادی با عدد 0 و اتصال نفوذ با عدد 1 نشان‌دهنده می‌شود. در ضمیمه 2 انواع حملات و تعداد آنها در مجموعه داده NSLKDD بصورت کامل آمده است. بنابراین در مجموع هر ردیف شامل 42 ستون است که 41 ستون اول ویژگی‌های اتصال را نشان می‌دهند و ستون 42 نوع اتصال را مشخص می‌کند. تابع  $genfis2$ ، 41 ستون اول از 20000 اتصال را به عنوان پارامتر اول، ستون 42 داده‌ها یعنی نوع اتصال را به عنوان پارامتر دوم (20000 ردیف شامل ستون 42 داده‌ها) و شعاع همسایگی که عددی بین 0 و 1 است را به عنوان پارامتر سوم دریافت می‌کند. خروجی این تابع، سیستم فازی است که قوانین آن بر اساس خوشه‌بندی کاهشی تولید شده‌اند. تعداد قوانین فازی و تعداد توابع عضویت تولید شده به ازای هر ورودی، 7 است. همینطور تعداد خوشه‌ها نیز برابر 7 هستند.

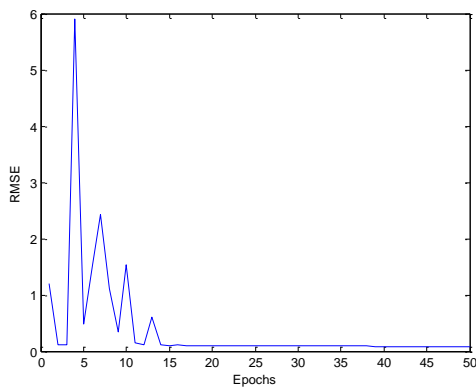
برای تنظیم بیشتر و تطبیق دادن توابع عضویت با داده‌های آموزشی از مجموعه داده‌های آموزشی همراه با داده‌های بررسی مورد نظر، شبکه انفیس را در 50 دوره آموزش دادیم. ساختار انفیزی که برای آموزش استفاده شد 632 گره داشت که مجموعاً 868 پارامتر داشت که 294 پارامتر خطی و 574 پارامتر غیرخطی بودند.

مجدور میانگین مربع خطا ( $RMSE^2$ )، بعد از 50 دوره آموزش به ترتیب 0/0822715 برای داده‌های آموزشی و

0/0771951 برای داده‌های بررسی به دست آمد. شکل‌های 3 و 4 مقدار  $RMSE$  را به صورت تابعی از دوره برای داده‌های آموزشی و داده‌های آزمایشی به ترتیب نشان می‌دهند.



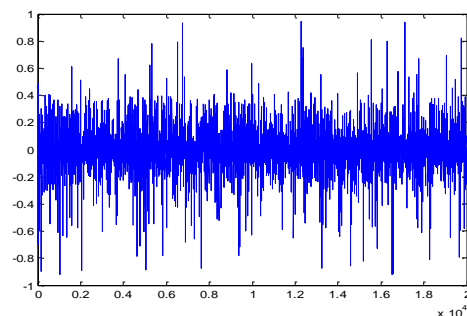
شکل 3: میزان خطا به ازای دوره‌های آموزش برای داده‌های آموزشی



شکل 4: میزان خطا به ازای دوره‌های آموزش برای داده‌های آزمایش

اغلب سعی بر آن است تا با آموزش یک شبکه بتوان مقادیر مناسبی از خروجی را برای داده‌های ورودی بدست آورد و این عمل همیشه به صورت 100 درصد امکان پذیر نیست و هر شبکه‌ی آموزش دیده، به طور معمول مقداری خطا در تخمین خروجی مورد انتظار دارا می‌باشد. شکل 4 تفاوت مقدار واقعی و مقدار بدست آمده از خروجی انفیس را برای داده‌های آموزشی بعد از 50 دوره آموزش نشان می‌دهد.

انفیس، عددی بین صفر و  $0/5$  خواهد بود. همچنین چنانچه خروجی حاصل از انفیس بین  $0$  تا  $0/5$  باشد حاصل تفاوت مقدار اصلی و مقدار بدست آمده توسط انفیس، عددی بین  $0$  تا  $0/5$  است. این تفاوت را به صفر گرد می کنیم. همچنین اگر حاصل تفاوت مقدار واقعی و مقدار بدست آمده توسط انفیس، عددی بیشتر از  $0/5$  یا کوچکتر از  $0/5$  باشد آن را به عدد صحیح غیر صفر گرد می کنیم و بعنوان پیش بینی اشتباه ثبت می کنیم. بر این اساس پس از عملیات گرد کردن، اعداد صفر نشان دهنده ی پیش بینی درست و اعداد غیر صفر نشان دهنده ی پیش بینی غلط هستند، برای پیدا کردن حد آستانه از آزمایش و خطا استفاده شد تا مقدار مناسب برای این مقدار مشخص شود. که استفاده از روشهای دیگر و مقادیر دیگر باعث میشد که نرخ تشخیص خطا و نرخ هشدار غلط به اندازه مطلوب نباشد. در شکل 6 نمودار تفاوت خروجی واقعی و خروجی انفیس را به صورت گرد شده نشان داده شده است. بر اساس نتایج بدست آمده از آموزش انفیس و آزمایش دسته بندی کننده ی عصبی - فازی با داده های آموزشی که در شکل 6 ارائه شده است، از کل 20000 داده ی آموزشی، تعداد تشخیص های درست برابر 19910 عدد بوده است. یعنی تعداد صفرهای بدست آمده، حاصل از اختلاف مقدار واقعی اتصال و مقدار بدست آمده توسط انفیس، برابر 19910 تا از 20000 اتصال است. این یعنی 99/55 درصد از اتصالات به درستی تشخیص داده شده اند. این تشخیص شامل تشخیص درست داده های عادی و نفوذ است. یعنی از 20000 اتصال که شامل اتصالات عادی و نفوذ است، 99/55 درصد، به درستی تشخیص داده شده اند. از بین این 20000 اتصال، 10693 اتصال عادی و 9307 اتصال نفوذ بوده اند. در الگوریتم 1، مراحل انجام کار و نحوه شمارش تعداد تشخیص درست و نادرست ارائه شده است.



شکل 5: تفاوت مقدار واقعی و مقدار بدست آمده از خروجی انفیس برای داده های آموزشی

همانطور که از شکل 5 می توان استنتاج نمود خروجی شبکه ی انفیس که در اصل شماره کلاس داده ی ورودی می باشد (0 برای اتصال های عادی و 1 برای اتصال های نفوذ) یک مقدار عددی صحیح نمی باشد. بنابراین ما نیاز داریم مقدار خروجی بدست آمده را برای بدست آوردن شماره کلاس گرد کنیم [20]. در این شکل محور افقی تعداد تکرار و محور عمودی میزان اختلاف مقدار واقعی و مقدار بدست آمده از خروجی انفیس را نشان میدهد.

ایده ی اصلی ما برای محاسبه ی تفاوت مقدار واقعی و مقادیر خروجی انفیس این است که اگر خروجی درست پیش بینی شده باشد مقدار حاصل از تفاوت باید صفر باشد. یعنی اگر فرض کنیم یک اتصال عادی باشد و مقدار 0 داشته باشد، در صورت تشخیص درست نیز مقدار صفر خواهد داشت و حاصل اختلاف این دو صفر است. همینطور در صورتی که اتصال از نوع نفوذ باشد مقدار 1 خواهد گرفت و در صورت تشخیص درست نیز مقدار 1 می گیرد که باز هم حاصل اختلاف صفر می شود.

پس اگر پیش بینی درست انجام شده باشد حاصل تفاوت مقدار واقعی اتصال و مقدار بدست آمده توسط انفیس، باید صفر باشد. برای مقادیر غیر صحیح از تکنیک گرد کردن استفاده می کنیم. بدین صورت که فرضاً یک اتصال عادی باشد، بنابراین مقدار خروجی واقعی برابر 0 است. حال اگر مقدار بدست آمده از انفیس عددی بین 0 و  $0/5$  باشد، حاصل تفاوت مقدار اصلی و مقدار بدست آمده توسط

$$FPR = \frac{FP}{(TN+FP)} \quad (10)$$

$$F\_measure = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \quad (11)$$

که:

True Positive (TP): تعداد نمونه های نفوذ که به صورت نفوذ طبقه بندی شده اند.

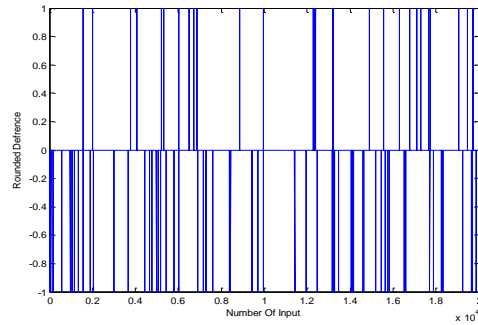
True Negative (TN): تعداد نمونه های نرمال که به صورت نرمال طبقه بندی شده اند.

False Positive (FP): تعداد نمونه های نرمال که به صورت نفوذ طبقه بندی شده اند.

False Negative (FN): تعداد نمونه های نفوذ که به صورت نرمال طبقه بندی شده اند.

در جدول 1 مدل پیشنهادی را با چهار معیار ذکر شده در رابطه های 8 تا 11 با داده های آموزشی و آزمایشی مورد بررسی قرار داده ایم و در ادامه مدل پیشنهادی را با استفاده از معیارهای نظیر، "نرخ تشخیص خطا" و "نرخ هشدار غلط"، که کارایی سیستم های تشخیص نفوذ طراحی شده را بهتر نشان می دهند [25]، با مدل های دیگر مقایسه کردیم. برای نرخ تشخیص خطا، هرچه این نرخ به 1 نزدیکتر باشد کارایی بهتر را نشان میدهد و هرچه نرخ هشدار غلط کمتر باشد (به صفر نزدیکتر باشد) بهتر است.

برای آزمایش دسته بندی کننده با داده های آموزشی، داده های آموزشی مورد استفاده برای آموزش، به 4 دسته تقسیم و نتایج ارائه شده در جدول 1 میانگین نتایج بدست آمده از آزمایش با 4 دسته داده ی آموزشی است. نتایج این جدول نشان می دهد که طبقه بندی کننده مورد نظر در کلاس بندی داده های آموزشی که بر اساس آنها آموزش دیده است، بسیار خوب عمل می کند. برای درک بهتر موضوع، 9307



شکل 6: مقدار گرد شده ی تفاوت خروجی واقعی و خروجی انفیس

الگوریتم 1. سیستم تشخیص نفوذ پیشنهادی مبتنی بر انفیس  
{ تعیین تشخیص صحیح ترافیک توسط انفیس }

```

Input Dataset
Preprocess Dataset //function
Select random Dataset
Construct ANFIS Classifier with train and test dataset
//function
Lx: Output of the classification
Fx: Real Output in Dataset (feature of 42)
begin
  Sx: Lx - Fx
  if -0.5 <= Sx <= 0.5 then
    True Detection
  else
    False Detection
end
end
    
```

## معیارهای ارزیابی

معیارهای استاندارد گوناگونی برای ارزیابی سیستم های تشخیص نفوذ ارائه شده است. از جمله می توان به نرخ تشخیص خطا<sup>1</sup> و نرخ هشدارهای غلط<sup>2</sup> اشاره نمود. نرخ تشخیص خطا از تقسیم تعداد حملاتی که به درستی تشخیص داده شده اند بر کل تعداد حملات بدست می آید و نرخ هشدارهای غلط در حقیقت نسبت تعداد اتصالات عادی که به اشتباه به عنوان نفوذ تشخیص داده شده اند به کل تعداد اتصالات عادی است. در ادامه معیارهای ارزیابی آورده شده است:

$$Accuracy = \frac{(TP+TN)}{(TP+FP+FN+TN)} \quad (8)$$

$$Recall (TPR) = \frac{(TP)}{(TP+FN)} \quad (9)$$

بوده است که داده‌های انتخاب شده در فاز آموزش و آزمایش کاملاً تصادفی باشند. اما در عین حال نتایج به دست آمده، می‌تواند تا حد زیادی توانایی دسته‌بندی‌کننده عصبی - فازی تطبیق پذیر را در این مساله‌ی طبقه‌بندی نشان دهد و این اطمینان را به ما بدهد که طبقه‌بندی‌کننده ارائه شده در این بخش، به اندازه‌ی کافی توانمند است و کارایی آن خارج از حد انتظار نمی‌باشد چرا که رویکرد ما در استفاده از دسته‌بندی‌کننده‌ی عصبی - فازی این بوده است که انفیس به علت قدرت یادگیری بدون دانش اولیه و استفاده از منطق فازی و شبکه‌عصبی، باید طبقه‌بندی‌کننده‌ی قدرتمندی باشد.

### تحلیل و مقایسه نتایج

محققان در زمینه‌ی بهینه‌سازی سیستم‌های تشخیص نفوذ تلاش‌های زیادی کرده‌اند. روش‌های متعددی برای ساخت سیستم تشخیص نفوذ توسط محققان به کار گرفته شده است. در این قسمت ما به مقایسه‌ی نتایج حاصل از سیستم تشخیص نفوذ ساخته شده توسط دسته‌بندی‌کننده‌ی عصبی - فازی با سایر روش‌هایی که برای ساخت این سیستم مورد استفاده قرار گرفته‌اند خواهیم پرداخت.

در جدول 2 نیز به مقایسه‌ی نتایج حاصل از این تحقیق با نتایج سایر تحقیقات خواهیم پرداخت. نکته‌ای که باید به آن توجه داشت، این است که به دلیل استفاده از داده‌های تصادفی در هنگام آموزش، ممکن است که این مقایسه تا حدی عادلانه نباشد، ولی نتایج جداول برای پذیرفتن طبقه‌بندی‌کننده عصبی - فازی، به عنوان یک طبقه‌بندی‌کننده مناسب، کفایت می‌کند و نشان‌دهنده‌ی این مطلب است که این طبقه‌بندی‌کننده از توانایی لازم در حد روش‌های مشابه و حتی بیشتر، برخوردار است. همانطور که در جدول 2 مشخص است، نرخ تشخیص خطا و نرخ هشدار غلط

اتصال از 20000 اتصالی که برای آموزش استفاده شده بودند از نوع نفوذ بوده‌اند. در هنگام آزمایش دسته‌بندی‌کننده با داده‌های آموزشی، 9281 اتصال از 9307 اتصال نفوذ، توسط دسته‌بندی‌کننده به درستی از نوع نفوذ تشخیص داده شدند. برای نرخ هشدار غلط نیز بر اساس تعریف، مقدار آن محاسبه و در جدول 1 ارائه شده‌است.

به منظور بررسی بیشتر کارایی طبقه‌بندی‌کننده‌ی ارائه شده و بررسی عملکرد طبقه‌بندی‌کننده‌ی عصبی - فازی در قبال داده‌هایی که در فرآیند آموزش مورد استفاده قرار نگرفته‌اند (داده‌های آزمایش)، 4075 داده‌ی تصادفی دیگر از مجموعه داده‌های آزمایش NSLKDD به منظور ارزیابی سیستم انتخاب شده‌اند که با داده‌های آموزشی و آزمایشی هیچ اشتراکی ندارند. نتیجه‌ی آزمایش دسته‌بندی‌کننده با این داده‌ها در جدول 1 ارائه شده است. در این جدول کارایی طبقه‌بندی‌کننده‌ی ارائه شده را بر اساس آزمایش با 4075 داده که در فاز آموزش استفاده نشده‌اند و جزء داده‌های آزمایش مجموعه داده NSLKDD هستند، نشان می‌دهد.

جدول 1: ارزیابی مدل پیشنهادی با داده‌های آموزشی و آزمایشی

نوع ارزیابی	داده‌های آموزش	داده‌های آزمایش
Accuracy	99/81	95/79
TPR	99/72	95/65
FPR	0/06	0/219
F_measure	99/76	95/73

همچنین برای بررسی بیشتر طبقه‌بندی‌کننده‌ی عصبی - فازی ارائه شده برای سیستم تشخیص نفوذ، نتایج حاصل از این طبقه‌بندی‌کننده، با نتایج تحقیقات دیگر که با روش‌های مختلف اقدام به ساخت سیستم تشخیص نفوذ کرده‌اند مقایسه شده است. هرچند که این مقایسه به علت وجود عدم دسترسی به داده‌های سایر تحقیقات در هنگام آموزش و آزمایش، نمی‌تواند عادلانه باشد. چرا که در سایر روش‌ها مشخص نیست که دقیقاً از کدام اتصالات NSLKDD برای آموزش و آزمایش استفاده شده است. سعی ما بر این

بدست آمده توسط این مقاله، نسبت به سایر روش‌ها، برتری قابل قبولی دارد.

برای مقایسه‌ی نتایج بدست آمده در بخش داده‌های آموزش نیز، نتایج حاصل از آموزش دسته‌بندی کننده‌ی عصبی - فازی با داده‌های آموزشی را با یک تحقیق دیگر که نتایج آن بر اساس آزمایش با داده‌های آموزشی ارائه شده است، مقایسه نموده‌ایم. در بخش آموزش این تحقیق ما تعداد 20000 داده‌ی تصادفی از مجموعه داده‌ی NSLKDD را انتخاب کرده بودیم و آموزش نیز بوسیله دسته‌بندی کننده‌ی عصبی - فازی انجام شد.

جدول 2: مقایسه‌ی نتایج آزمایش سیستم تشخیص نفوذ با سایر نتایج بر اساس معیارهای نرخ تشخیص خطا و نرخ هشدار غلط

روش ساخت سیستم تشخیص نفوذ	نرخ تشخیص خطا	نرخ هشدار غلط	مجموعه داده
CSOM+RBF[26]	95/27	1/64	NSL_KDD
Deep Neural Network[27]	94/62	0/97	NSL_KDD
TLMD [12]	93/32	0/06	KDD 99
LSSVM-IDS [13]	91/12	0/38	KDD 99, NSL_KDD, Kyoto 2006+
Fuzzy classification [۱۰]	94/4	0/35	DARPA
ESC-IDS [۲۰]	95/3	1/9	KDD 99
PNrule [۲۱]	91/1	0/4	NSL_KDD
PCA [۲۸]	93/83	0/616	KDD 99
<b>Proposed IDS</b>	95/56	0/219	<b>NSL_KDD</b>

پاندا<sup>۱</sup> و همکارانش، در تحقیق خود از تعداد 25192 اتصال مجموعه داده‌ی NSLKDD برای آموزش مدل خود استفاده کردند [29]. آنها در تحقیق خود از یک روش ترکیبی استفاده نمودند. این روش ترکیبی شامل درخت تصمیم<sup>۲</sup>

تجزیه و تحلیل مؤلفه‌های اصلی<sup>۳</sup>، ماشین بردار پشتیبان<sup>۴</sup>، Grading، و جنگل تصادفی<sup>۵</sup> می‌باشد.

این محققین، از مجموعه داده‌های آموزشی، برای آزمایش سیستم تشخیص نفوذ خود استفاده نموده‌اند. نتایجی که این محققان در تحقیق خود برای آزمایش با داده‌های آموزشی مورد استفاده‌ی خود بدست آورده‌اند، به همراه نتایج حاصل از این تحقیق، در جدول 3 ارائه شده است.

جدول 3: مقایسه‌ی نتایج این تحقیق برای آزمایش سیستم تشخیص نفوذ با داده‌های آموزشی باروش ترکیبی Panda

روش	نرخ تشخیص خطا	نرخ هشدار غلط
Hybrid Intelligent Approach	99/5	0/5
<b>Proposed IDS</b>	99/72	0/06

آنچه در جدول 3 مشخص است، این است که روش دسته‌بندی کننده‌ی عصبی - فازی به تنهایی با روش ترکیبی که در بالا ذکر شد رقابت می‌کند. وقتی ما دسته‌بندی کننده را با 41 ویژگی آموزش می‌دهیم (در تحقیق پاندا و همکارانش نیز از 41 ویژگی استفاده شده است)، نرخ تشخیص خطا و نرخ هشدار غلط هر دو نسبت به روش دیگر بهینه‌تر هستند و این خود گواهی بر قدرت این دسته‌بندی کننده عصبی - فازی می‌باشد.

## نتیجه گیری

در این مقاله، به مقایسه‌ی نتایج حاصل شده از دسته‌بندی کننده‌ی عصبی - فازی با چند تحقیق دیگر پرداختیم. ابتدا نتایج مقادیر بدست آمده برای نرخ تشخیص خطا و نرخ هشدار غلط حاصل از دسته‌بندی کننده که آموزش دیده بود و با داده‌های آزمایشی پایگاه داده‌ی NSLKDD آزمایش شده بود، با نتایج معیارهای سایر محققین مقایسه شد. نتایج

<sup>4</sup>Support Vector Machine

<sup>5</sup>Random Forest

<sup>1</sup>Panda

<sup>2</sup>Decision Tree

<sup>3</sup>Principal Component Analysis

- [4] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," *Peer-to-Peer Networking and Applications*, vol. 12, no. 2, pp. 493-501, 2019.
- [5] S.-Y. Wu and E. J. E. S. w. A. Yen, "Data mining-based intrusion detectors," vol. 36, no. 3, pp. 5605-5612, 2009.
- [6] N. Pandeeswari and G. Kumar, "Anomaly detection system in cloud environment using fuzzy clustering based ANN," *Mobile Networks and Applications*, vol. 21, no. 3, pp. 494-505, 2016.
- [7] I. Škrjanc, J. A. Iglesias, A. Sanchis, D. Leite, E. Lughofer, and F. Gomide, "Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A survey," *Information Sciences*, vol. 4, 90pp. 344-368, 2019.
- [8] A. Abraham and R. Jain, "Soft computing models for network intrusion detection systems," in *Classification and clustering for knowledge discovery*: Springer, 2005, pp. 191-207.
- [9] K. Faraoun and A. Boukelif, "Neural networks learning improvement using the K-means clustering algorithm to detect network intrusions," *INFOCOMP Journal of Computer Science*, vol. 5, no. 3, pp. 28-36, 2006.
- [10] T. Dash, "A study on intrusion detection using neural networks trained with evolutionary algorithms," *Soft Computing*, vol. 21, no. 10, pp. 2687-2700, 2017.
- [11] R. S. K. Jonnalagadda and R. P. Reddy, "A literature survey and comprehensive study of intrusion detection," *International Journal of Computer Applications*, vol. 81, no. 16, pp. 40-47, 2013.
- [12] Y. Yuan, L. Huo, and D. Hogrefe, "Two layers multi-class detection method for network intrusion detection system," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, 2017: IEEE, pp. 767-772.
- [13] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE transactions*
- جدول 2 حکایت از کیفیت و قدرت دسته‌بندی‌کننده‌ی عصبی - فازی در مقایسه با سایر نتایج دارد. همانطور که در بخش‌های قبلی نیز اشاره شد پایگاه‌داده‌ی NSLKDD دارای دو دسته مجموعه‌داده‌ی تفکیک‌شده برای آموزش و آزمایش سیستم تشخیص نفوذ می‌باشد. این دو مجموعه-داده هیچ اشتراکی ندارند و بر این اساس می‌توان سیستم تشخیص نفوذ ساخته شده را به طور کامل محک زد. این محک توسط "مجموعه‌داده‌ی بررسی" انجام می‌شود. علاوه بر این، توسط "مجموعه‌داده‌ی آزمایش"، به آزمایش سیستم تشخیص نفوذ ساخته شده پرداخته‌ایم. در نهایت، به این نتیجه می‌رسیم که دسته‌بندی‌کننده‌ی عصبی - فازی مبتنی بر فازی سوجینو، می‌تواند به عنوان یک دسته‌بندی‌کننده‌ی قابل قبول برای سیستم تشخیص نفوذ مورد استفاده قرار گیرد.
- در این تحقیق باور ما بر این بوده است که، طبقه‌بندی - کننده‌ی عصبی - فازی، به دلیل نوع ساختارش می‌تواند یک دسته‌بندی‌کننده‌ی قدرتمند باشد و همچنین نتایجی که بررسی شد قابلیت بهبود دارند. این قابلیت بهبود بستگی به عوامل زیادی دارد که از آن جمله می‌توان به استفاده از پایگاه‌داده‌ی بهتر نسبت به NSLKDD اشاره کرد.

## مراجع

- [1] E. Corchado and Á. Herrero, "Neural visualization of network traffic data for intrusion detection," *Applied Soft Computing*, vol. 11, no. 2, pp. 2042-2056, 2011.
- [2] S. A. Hofmeyr, S. Forrest, and A. Somayaji, "Intrusion detection using sequences of system calls," *Journal of computer security*, vol. 6, no. 3, pp. 151-180, 1998.
- [3] G. Folino and P. Sabatino, "Ensemble based collaborative and distributed intrusion detection systems: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 1-16, 2016.

- transactions on systems, man, and cybernetics*, vol. 23, no. 3, pp. 665-685, 1993.
- [24] A. G. Hussien *et al.*, "Crow Search Algorithm: Theory, Recent Advances, and Applications," *IEEE Access*, vol. 8, pp. 173548-173565, 2020, doi: 10.1109/ACCESS.2020.3024108.
- [25] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, p. 107247, 2020.
- [26] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, and A. Razaque, "Cascaded hybrid intrusion detection model based on SOM and RBF neural networks," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 21, p. e5233, 2020.
- [27] M. M. U. Chowdhury, F. Hammond, G. Konowicz, C. Xin, H. Wu, and J. Li, "A few-shot deep learning approach for improved intrusion detection," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, 2017: IEEE, pp. 456-462.
- [28] R. Beghdad, "Critical study of neural networks in detecting intrusions," *Computers & security*, vol. 27, no. 5-6, pp. 168-175, 2008.
- [29] M. Panda, A. Abraham, and M. R. Patra, "Hybrid intelligent systems for detecting network intrusions," *Security and Communication Networks*, vol. 8, no. 16, pp. 2741-2749, 2015.
- [14] R. Beghdad, "Critical study of supervised learning techniques in predicting attacks," *Information Security Journal: A Global Perspective*, vol. 19, no. 1, pp. 22-35, 2010.
- [15] J. Gomez and D. Dasgupta, "Evolving fuzzy classifiers for intrusion detection," in *Proceedings of the 2002 IEEE Workshop on Information Assurance*, 2002, vol. 6, no. 3: New York: IEEE Computer Press, pp. 321-323.
- [16] C. Jirapummin, N. Wattanapongsakorn, and P. Kanthamanon, "Hybrid neural networks for intrusion detection system," in *Proc. of ITC-CSCC*, 2002, vol. 7, pp. 928-931.
- [17] T. Horeis, "Intrusion detection with neural networks—combination of self-organizing maps and radial basis function networks for human expert integration," *Computational Intelligence Society Student Research Grants*, 2003.
- [18] S.-J. Han and S.-B. Cho, "Evolutionary neural networks for anomaly detection based on the behavior of a program," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 3, pp. 559-570, 2005.
- [19] Y. Chen, A. Abraham, and B. Yang, "Hybrid flexible neural-tree-based intrusion detection systems," *International journal of intelligent systems*, vol. 22, no. 4, pp. 337-352, 2007.
- [20] A. N. Toosi and M. Kahani, "A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers," *Computer communications*, vol. 30, no. 10, pp. 2201-2212, 2007.
- [21] S. Ghosh, A. Pal, A. Nag, S. Sadhu, and R. Pati, "Network anomaly detection using a fuzzy rule-based classifier," *Computer, Communication and Electrical Technology*, pp. 61-65, 2017.
- [22] M. Negnevitsky and A. Intelligence, "A guide to intelligent systems," *Artificial Intelligence, 2nd edition*, Pearson Education, 2005.
- [23] J.-S. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE*

ضمیمه ۱

جدول لیست کامل ویژگی های تعریف شده برای هر داده ی اتصال

برجسب	ویژگی های شبکه	توضیحات	بازه ی مقادیر
A	Duration	طول اتصال بر حسب ثانیه	21-0
B	protocol_type	نوع پروتکل به عنوان مثال tcp, udp و ...	Tcp - udp - icmp
C	Service	سرویس شبکه در مقصد، به عنوان مثال http, telnet, ftp, ...	http, talent, ftp_data, private, remote_job, mtp, eco_i_suodup
D	Flag	حالت اتصال به شبکه	SF, S0, S1, S2, S3, REJ, SH, PSTR, ...
E	src_bytes	تعداد بایت های یک داده ارسالی از مبدا به مقصد	1379963888-0
F	dst_bytes	تعداد بایت های ارسالی از مقصد به مبدا	1309937401-0
G	Land	1 اگر اتصال از له host/port یکسان، در غیر این صورت 0	1-0
H	wrong_fragment	تعداد قطعات اشتباه	3-0
I	Urgent	تعداد بسته های ضروری	3-0
J	Hot	تعداد شاخص های hot	77-0
K	num_failed_logins	تعداد تلاش های انجام شده برای ورود	5-0
L	logged_in	1 اگر ورود موفق داشته باشد؛ در غیر این صورت 0	1-0
M	num_compromised	تعداد اتصالات در معرض خطر	7479-0
N	root_shell	1 اگر پوشش ریشه بدست آمده باشد؛ در غیر این صورت 0	1-0
O	su_attempted	1 اگر تلاش برای اجرای دستور su root انجام شده باشد در غیر این صورت 0	1-0
P	num_root	تعداد دسترسی به ریشه	7468-0
Q	num_file_creations	تعداد عملیات ایجاد فایل	43-0
R	num_shells	تعداد ایجاد دستورات shell	2-0
S	num_access_files	تعداد دستورات دسترسی به فایل	9-0
T	num_outbound_cmds	تعداد دستورات خارج از محدوده برای دسترسی به نشست ftp	0
U	is_host_login	1 اگر ورود متعلق به hot لیست باشد، در غیر این صورت 0	1-0
V	is_guest_login	1 اگر کاربر بعنوان مهمان وارد شود؛ در غیر این صورت 0	1-0
W	Count	تعداد اتصال به یک میزبان در اتصالات اخیر در 2 ثانیه گذشته	511-0
X	srv_count	تعداد اتصالاتی که سرویس یکسان در اتصال اخیر در بیشتر از 2 ثانیه دارند	511-0
Y	serror_rate	درصد اتصالاتی که خطای SYN دارد	1-0
Z	srv_serror_rate	درصد اتصالاتی که خطای SYN دارد	1-0
AA	rerror_rate	در صد اتصالاتی که خطای REJ دارد	1-0
AB	srv_rerror_rate	در صد اتصالاتی که خطای REJ دارد	1-0
AC	same_srv_rate	درصد اتصالاتی که سرویس یکسان دارند	1-0
AD	diff_srv_rate	درصد اتصالاتی که سرویس مختلف دارند	1-0
AE	srv_diff_host_rate	درصد اتصالاتی که میزبان مختلف دارند	1-0
AF	dst_host_count	تعداد اتصالات از یک میزبان به مقصد در طی یک زمان مشخص	255-0
AG	dst_host_srv_count	تعداد اتصالات از یک میزبان به مقصد برای دستیابی به سرویس	255-0
AH	dst_host_same_srv_rate	درصد اتصالات از یک میزبان به مقصد برای دستیابی به یک سرویس یکسان	1-0
AI	dst_host_diff_srv_rate	درصد اتصالات از یک میزبان به مقصد برای دستیابی به سرویس های مختلف	1-0
AJ	dst_host_same_src_port_rate	درصد اتصالات از یک میزبان به مقصد از یک پورت یکسان	1-0
AK	dst_host_srv_diff_host_rate	درصد اتصالات از یک میزبان به مقصد های مختلف برای دستیابی به سرویس	1-0
AL	dst_host_serror_rate	درصد اتصالاتی از یک میزبان به یک مقصد مشخص که خطای SYN دارند	1-0
AM	dst_host_srv_serror_rate	درصد اتصالات از یک میزبان با یک سرویس مشخص به مقصد که خطای SYN دارند	1-0
AN	dst_host_rerror_rate	درصد اتصالاتی از یک میزبان به یک مقصد مشخص که خطای REJ دارند	1-0
AO	dst_host_srv_rerror_rate	درصد اتصالات از یک میزبان با یک سرویس مشخص به مقصد که خطای REJ دارند	1-0

جدول توزیع حملات

Class	Training			Testing		
	Attack names	Samples	Total	Attack names	Samples	Total
DOS	teardrop	979	391,458	Apache 2	794	229,853
	smurf	280,790		Back	1098	
	neptune	107,201		land	9	
	Pod	264		mailbomb	5000	
	Back	2203		neptune	58,001	
	Land	21		pod	87	
				processtable	759	
				smurf	164,091	
				teardrop	12	
				udpstorm	2	
Probe	satan	1589	4107	ipsweep	306	4166
	nmap	231		mscan	1053	
	ipsweep	1247		nmap	84	
	portsweep	1040		portsweep	354	
				saint	736	
				satan	1633	
U2R	perl	3	52	buffer overflow	22	70
	buffer overflow	30		loadmodule	2	
	rootkit	10		perl	2	
	loadmodule	9		ps	16	
				rootkit	13	
				sqlattack	2	
R2L	ftp write	8	1126	ftp write	3	16347
	Warezclient	1020		guess passwd	4367	
	Warezmaster	20		imap	1	
	Spy	2		multihop	18	
	guess passwd	53		named	17	
	Imap	12		phf	2	
	multihop	7		sendmail	17	
	phf	4		snmpgetattack	7741	
				Snmpguess	2406	
				warezmaster	1602	
				worm	2	
				xlock	9	
				xsnoop	4	
				httptunnel	158	

حملاتی که به صورت خاکستری مشخص شده است، حملاتی هستند که در داده های آزمایشی وجود دارد و در داده های آموزشی وجود نداشته است.