رویکردی برای تشخیص حملات DDoS در محیط رایانش ابری با استفاده از آنتروپی و بهینه‌سازی ازدحام ذرات

مهدی آسایش‌جو[۱]، مهدی صادق‌زاده[۲*]، مازیار گنجو[۳]

۱: گروه کامپیوتر، واحد بوشهر، دانشگاه آزاد اسلامی، بوشهر، ایران، mehdi.asayeshjo@gmail.com
۲*: گروه کامپیوتر، واحد ماهشهر، دانشگاه آزاد اسلامی، ماهشهر، ایران، sadegh_1999@yahoo.com
۳: گروه فناوری اطلاعات، واحد بوشهر، دانشگاه آزاد اسلامی، بوشهر، ایران، ganjoo@gmail.com

تاریخ دریافت: ۱۳۹۹/۱۲/۲۰ تاریخ پذیرش: ۱۴۰۰/۰۴/۰۹

چکیده

در دسترس بودن سرویس‌های ابری یکی از مهمترین نگرانی‌های ارائه‌دهندگان خدمات ابری است. در حالی که سرویس‌های ابری عمدتا از طریق اینترنت منتقل می‌شوند، مستعد حملات مختلفی هستند که ممکن است منجر به درز اطلاعات حساس شود. حمله منع سرویس توزیع شده (*DDoS*) به عنوان یکی از مهمترین تهدیدات امنیتی برای محیط رایانش ابری شناخته می‌شود. این حمله تلاشی صریح توسط یک مهاجم برای جلوگیری و عدم دسترسی به خدمات یا منابع مشترک در یک محیط ابری است. روش‌های زیادی برای پیش‌بینی سریع و دقیق این حملات پیشنهاد شده است. با این حال، با توجه به گستردگی ویژگی‌های موجود تلاش‌ها با توجه به اهمیت مسئله همچنان ادامه دارد. رویکردی آنتروپی و بهینه‌سازی ازدحام ذرات از الگوریتم‌های رایج در یادگیری ماشین هستند، که برای مسائل پیش‌بینی و دسته‌بندی بسیار محبوب می‌باشند. در این مقاله رویکردی ترکیبی برای مقابله با حمله *DDoS* در محیط رایانش ابری مورد بحث قرار گرفته است. این روش اهمیت روش‌های مبتنی بر انتخاب ویژگی‌های موثر و مدل‌های دسته-بندی را برجسته می‌کند. در اینجا، رویکردی بر مبنای آنتروپی و بهینه‌سازی ازدحام ذرات برای مقابله با این حملات در محیط رایانش ابری ارائه می‌شود. دسته‌بندی داده‌های با ابعاد بالا معمولاً به انتخاب ویژگی به عنوان یک مرحله قبل از پردازش برای کاهش ابعاد نیاز دارد. با این حال، انتخاب ویژگی‌های موثر یک کار چالش برانگیز است که در این مقاله از بهینه‌سازی ازدحام ذرات برای اینکار استفاده می‌شود. در اینجا، مدل دسته‌بندی پیشنهادی بر مبنای استفاده از ساختمان داده درخت جستجوی دودویی متوازن و دیکشنری توسعه یافته است. شبیه‌سازی براساس مجموعه داده‌های *NSL-KDD* و *CICDDoS2019* انجام شده که نتایج برتری روش پیشنهادی را با میانگین دقت تشخیص ۹۹,۸۴٪ نسبت به الگوریتم‌های *AGA* و *E-SVM* اثبات می‌کند.

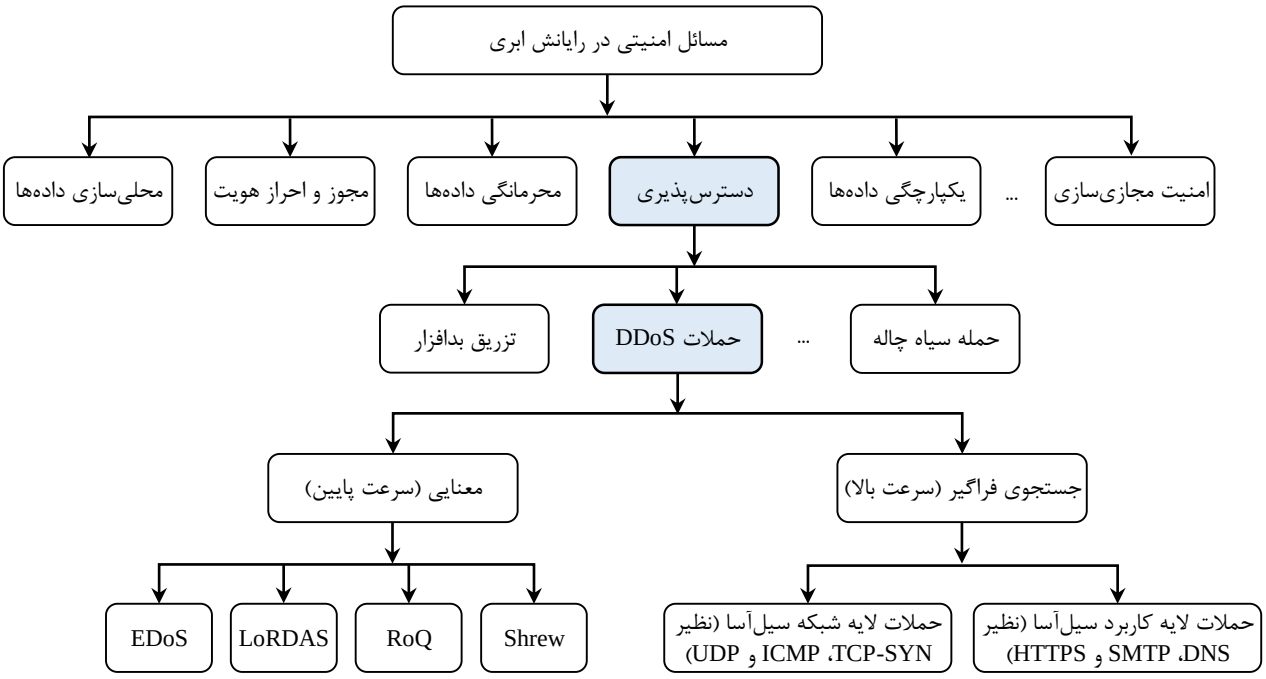
واژه های کلیدی: رایانش ابری، تشخیص حملات، آنتروپی، بهینه‌سازی ازدحام ذرات، حمله *DDoS*.

۱- مقدمه

محاسبات ابری اصطلاحی است که برای ارائه خدمات میزبانی تحت اینترنت به کار رفته که پتانسیل بسیار خوبی را برای بهبود بهره-وری و کاهش هزینه‌ها ارائه می‌دهد [۱]. با وجود مزایای بیشمار رایانش ابری، این فناوری با چندین مسئله امنیتی روبرو است [۲-۴]. در میان مسائل امنیتی، در دسترس بودن به عنوان مهمترین نگرانی مهم ذکر شده است [۵]، زیرا عملکرد اصلی رایانش ابری ارائه خدمات درخواستی است. یکی از عمده‌ترین تهدیدها برای مسئله دسترس‌پذیری، حمله منع از سرویس توزیع شده (*DDoS*) می‌باشد [۶]. حمله *DDoS* نوع خاصی از حمله منع از سرویس (*DoS*) است که در آن چندین ماشین توزیع شده در معرض حمله، سرور ابر قربانی را هدف قرار می دهند [۷]. بر اساس گزارش آربور (*Arbor*)، درصد حملات *DDoS* با هدف قرار دادن خدمات و منابع رایانش ابری هر ساله در حال افزایش است [۸]. این حملات باعث اختلال در خدمات، زیان اقتصادی و بسیاری از تأثیرات کوتاه مدت و بلند مدت بر ارائه دهندگان خدمات ابری شده است بنابراین، شناسایی این نوع حملات در مقایسه با حملات جستجوی فراگیر دشوار است [۹].

با پیشرفت‌های فنی، پلتفرم‌های ابری از نظر منابع روز به روز قدرتمندتر می‌شوند [۱۰]. به دلیل وجود این منابع بزرگ در سمت ابر،

تهدید کردن آنها حتی با حملات *DDoS* دشوار است. نمونه‌ای از چنین حمله ناموفق *DDoS* در [۱۱] گزارش شده است که توسط یک گروه ناشناس علیه ابر آمازون راه‌اندازی شده بود. در نتیجه، مهاجمان در حال تلاش برای کاهش کیفیت خدمات ابر بدون شناسایی زود هنگام با استفاده از حملات پیچیده *DDoS* هستند. در سال ۲۰۱۵ یک حمله شدید *DDoS* علیه آمازون آغاز شد که باعث خسارت مالی روزانه ۳۰ هزار دلاری شد [۱۲]. شکل ۱ سناریوی کاملی از حمله *DDoS* و انواع آن را در یک محیط رایانش ابری ارائه می‌دهد.



شکل ۱: حمله *DDoS* و انواع آن در یک محیط رایانش ابری [۸]

با افزایش حجم روزافزون داده‌ها در سال‌های اخیر بخصوص در داده‌های شبکه، مشکل داده‌های با ابعاد بالا بوجود آمده است که باعث کاهش کارایی و دقت الگوریتم‌های یادگیری ماشین و شناسایی الگو شده است [۱۳]. دو طبقه عمده از روش‌های کاهش ابعاد وجود دارند. استخراج ویژگی و انتخاب ویژگی [۱۴]. در استخراج ویژگی هدف یافتن یک نگاشت از فضای ویژگی‌های کنونی به یک فضای با ابعاد کمتر است که در آن کمترین اتلاف اطلاعات (با در نظر گرفتن معیارهای جداپذیری دسته‌ها) ایجاد می‌گردد. مسئله انتخاب ویژگی نیز در حقیقت همان استخراج ویژگی است. با این تفاوت که نگاشت مذکور به انتخاب زیر مجموعه‌ای از مجموعه ویژگی‌های فضای اصلی محدود می‌شود. با توجه به خصوصیات داده‌های شبکه در این مقاله استفاده از الگوریتم‌های انتخاب ویژگی مورد توجه می‌باشند. برای مسئله انتخاب ویژگی، راه‌حل‌ها و الگوریتم‌های زیادی پیشنهاد شده است [۱۵]. مشکل بعضی از الگوریتم‌ها، پیچیدگی و بار محاسباتی زیاد آنها است که البته با ظهور کامپیوترهای سریع و منابع ذخیره‌سازی بزرگ، امروزه این مشکل کمتر به چشم می‌آید. ولی از طرف دیگر، مجموعه‌های داده‌ای بسیار بزرگ در مسائل جدید باعث شده است که همچنان پیدا کردن یک الگوریتم سریع برای این کار مهم باشد. علاوه بر این، زمانی که طیف گسترده‌ای از داده‌ها با خصوصیات ناهمگون و پراکنده وجود دارد، نیاز به الگوریتم‌های کاهش ابعاد دقیق‌تر ضروری به نظر می‌رسد. بنابراین، از اهداف این تحقیق می‌توان به کاهش داده‌های موجود در شبکه برای تشخیص موثرتر حملات *DDoS* اشاره کرد.

روش‌های تشخیص حملات *DDoS* زیادی در منابع گزارش شده است [۱۲-۱۴]، اما تنها تعداد کمی از آنها در یک محیط شبکه واقعی استفاده شده‌اند و عملکرد مؤثری دارند [۱۶]. طراحی و پیاده‌سازی یک سیستم دفاع ایده‌آل و عملی واقعاً دشوار است [۱۷]. لذا باید قبل از ارائه روش دفاعی به ویژگی‌هایی که در تشخیص این نوع حمله موثر هستند آگاهی لازم را داشت [۱۷]. با توجه به فراوانی استفاده از ویژگی‌های مختلف، در این مقاله از یک الگوریتم بهینه‌سازی ازدحام ذرات برای انتخاب زیرمجموعه‌ای از ویژگی‌های موثر در تشخیص حملات *DDoS* استفاده می‌شود. علاوه بر این، برای مدل‌سازی تشخیص حملات از یک رویکرد دسته‌بندی مبتنی بر محاسبه

آنتروپی اطلاعات استفاده می‌شود. مدل دسته‌بندی پیشنهادی بر مبنای استفاده از ساختمان داده درخت جستجوی دودویی متوازن و دیکشنری توسعه یافته است. اعتقاد بر این است که این مقاله میتواند محققان حوزه امنیت را تحریک کند تا راه‌حل‌های دفاعی موثر برای جلوگیری از حملات *DDoS* در فضای ابر ایجاد کنند.

در ادامه این مقاله؛ در بخش دوم به بررسی برخی از جدیدترین کارهای انجام شده در زمینه تشخیص حملات در محیط رایانش ابری می‌پردازیم. روش پیشنهادی در بخش سوم مطرح شده و نتایج حاصل از آن در بخش چهارم ارائه می‌گردد. در نهایت، بخش پنجم نتیجه‌گیری و پیشنهادات اختصاص داده شده است.

۲- مروری بر کارهای انجام شده

اگرچه حملات *DDoS* انواع مختلفی دارند، اما هدف همه آنها این است که با ارسال بسته‌های درخواست با نرخ بالا، سرورها، دیوارهای آتش یا سایر دستگاه‌های تعریف شده محیط را تحت فشار قرار دهند [۱۸]. در این راستا، شبکه‌های ابری با وب‌سایت‌های غیرقابل دسترس بیش از سایر پلتفرم‌ها تحت فشار قرار می‌گیرند [۱۷]. در ادامه این بخش، در مورد تکنیک‌های تشخیص نفوذ موجود برای محیط رایانش ابری بحث می‌شود.

در [۱۹]، یک سیستم تشخیص نفوذ مبتنی بر الگوریتم ژنتیک و قوانین فازی ارائه شده است. مدل دسته‌بندی روی مجموعه داده *NSL-KDD* با استفاده مجموعه قوانین اگر-آنگاه روی ویژگی‌های انتخاب شده توسط الگوریتم ژنتیک ایجاد شده است. علاوه بر این، دامنه‌های شرط در قوانین فازی با الگوریتم ژنتیک و با هدف حداقل‌سازی تعداد نمونه‌های دسته‌بندی کاذب نشده انتخاب می‌شود. در این مقاله، انتخاب ویژگی مبتنی بر همبستگی (*CFS*) و ارزیابی زیرمجموعه سازگاری (*CSE*) نیز تجزیه و تحلیل شده است. در [۲۰]، روشی مبتنی بر روش جستجوی تطبیقی تصادفی حریصانه با یک دسته‌بندی تصادفی تبرید (*GAR-forest*) را برای تشخیص حملات در ابر پیشنهاد شده است. این روش برای دو دسته‌بندی باینری و دسته‌بندی چند برچسب روی مجموعه داده *NSL-KDD* ارائه شده است. این روش مجموعه‌ای از درختان تصمیم سازگار تصادفی را از طریق بهره اطلاعاتی تولید می‌کند. نتایج این روش نشان می‌دهد که دقت تشخیص حملات نسبت به مدل‌های *RF*، *C4.5*، *NB* و *MLP* بهبود یافته است.

در [۲۱]، یک مدل دسته‌بندی با استفاده از شبکه‌های عصبی مصنوعی برای مجموعه داده *NSL-KDD* پیشنهاد شد. این روش بر اساس هر دو مدل دو کلاسه و پنج کلاسه توسعه یافته است. نتایج بر اساس معیارها مختلف مورد تجزیه و تحلیل قرار گرفت و نتایج دقت بهتری را برای این روش نشان می‌دهد. با این حال، هنوز هم باید نسبت دسته‌بندی بهبود یابد. میزان تشخیص این روش ۸۱٫۲٪ برای نفوذ و ۷۹٫۹٪ برای دسته‌بندی زیرحملات می‌باشد. در [۲۲]، یک چارچوب تشخیص نفوذ موثر با استفاده از یک روش بهینه-سازی دقیق و تطبیقی ارائه شده است. این روش از بهینه‌سازی ازدحام ذرات آشفته با تغییر زمان (*TVCPSO*) برای تنظیم همزمان پارامترها و انتخاب ویژگی‌ها استفاده می‌کند. در اینجا، مدل‌سازی بر اساس چندین معیار برنامه نویسی خطی (*MCLP*) و ماشین بردار پشتیبان (*SVM*) انجام شده است. علاوه بر این، یک تابع هدف وزن‌دار ارائه شده که بین حداکثر میزان تشخیص، حداقل‌سازی میزان هشدار کاذب و همچنین تعداد ویژگی‌های انتخابی توازن ایجاد می‌کند.

در [۲۳]، الگوریتم *AGA* به عنوان یک سیستم تشخیص نفوذ معرفی شده است. *AGA* از الگوریتم ژنتیک و تکنیک‌های داده‌کاوی بر اساس گاهش ویژگی‌ها برای تشخیص نفوذ استفاده می‌کند. در اینجا، یک الگوریتم ژنتیک بهبودیافته برای کار انتخاب ویژگی‌های موثر پیشنهاد شده که پارامترهای آن به صورت تطبیقی کنترل می‌شوند. برای مثال، پارامتر نرخ جهش در ابتدای کار دارای مقدار نسبتاً بالایی است و در روند اجرای الگوریتم به ترتیب کاهش می‌یابد. *AGA* به راحتی قابل اجرا است و از پیچیدگی محاسباتی پایینی برخوردار می‌باشد. نتایج تجربی این الگوریتم روی مجموعه داده *NSL-KDD* دقت بالاتری در تشخیص نفوذ با هشدار کاذب و تعداد ویژگی‌ها کمتر فراهم می‌کند. در [۲۴]، الگوریتم *E-SVM* برای مقابله با حملات *DDoS* در محیط رایانش ابری ارائه شده است. این الگوریتم بر اساس اندازه‌گیری آنتروپی اطلاعات و مدل *SVM*، ناهنجاری ترافیک شبکه را تشخیص می‌دهد. *E-SVM* از شش ویژگی شامل آدرس *IP* منبع، پورت منبع، آدرس *IP* مقصد، درگاه مقصد، نوع بسته و تعداد بسته‌های شبکه برای مدل‌سازی استفاده می‌کند. نتایج تجربی نشان می‌دهد که این الگوریتم میتواند در مجموعه داده‌های مقیاس بزرگ ترافیک ناهنجاری شبکه را با دقت بالاتری تشخیص دهد.

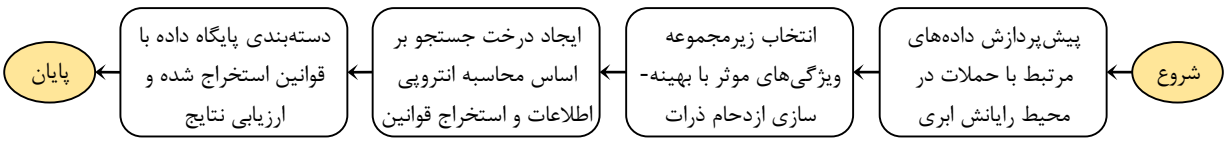
در [۲۵]، یک مکانیسم دفاعی با استفاده از شبکه‌های نرم‌افزار محور (*SDN*) برای تشخیص حملات *DDoS* در یک محیط رایانش

ابری معرفی شده است. این کار مکانیسم جدیدی را مورد بحث قرار می‌دهد که نه تنها حملات *DDoS* را شناسایی و کاهش می‌دهد بلکه موقعیت منابع حملات را نیز ردیابی می‌کند. در اینجا، حملات با استفاده از تغییرات آنتروپی اطلاعات شناسایی می‌شود و منابع حمله با استفاده از طرح علامت‌گذاری بسته ردیابی می‌شوند. این روش منابع حمله را بین ۱۴,۴۵ میلی‌ثانیه تا ۱۰,۰۲ ثانیه شناسایی و دقت ۹۷,۶٪ را فراهم می‌کند. در [۲۶]، رویکردی برای تشخیص حملات زمان واقعی *DDoS* بر اساس آنتروپی و با استفاده از چارچوب هادوپ پیشنهاد شده است. در این مقاله، از آنتروپی آدرس‌های منبع به عنوان معیار سنجش *DDoS* و تجزیه و تحلیل حجم زیادی از ترافیک در زمان واقعی استفاده شده است. نتایج این روش نشان می‌دهد که حملات *DDoS* در زمان واقعی با دقت بهتری قابل تشخیص هستند.

در [۲۷]، سیستم شناسایی حمله *DDoS* برای ابر خصوصی مبتنی بر *OpenStack* معرفی شده است. در اینجا سیستمی با فایروال یکپارچه *OpenStack* و برنامه نویسی سوکت برای نظارت بر ترافیک شبکه پیشنهاد شده است. در [۲۸]، تشخیص حملات *DDoS* از طریق یادگیری ماشین و روش‌های آماری در شبکه نرم‌افزار محور (*SDN*) بررسی شده است. این روش از سه بخش کلکتور، آنتروپی و دسته‌بندی تشکیل شده است. نتایج تجربی به دست آمده با استفاده از مجموعه داده‌های *UNB-ISCX*، *CTU-13* و *ISOT* نشان می‌دهد که این روش از نظر دقت در تشخیص حملات *DDoS* در *SDN* عملکرد خوبی دارد. در [۲۹]، کاهش ویژگی برای شناسایی حملات *DDoS* برجسته شده است. این مقاله یک روش کاهش ویژگی را با ترکیب تکنیک‌های انتخاب ویژگی بهره اطلاعاتی (*IG*) و همبستگی (*CR*) پیشنهاد می‌کند. این چارچوب در مجموعه داده *CICDDoS2019* با دسته‌بندی *J48* آزمایش شده است. این روش حداقل و حداکثر کاهش ویژگی‌های اصلی را به ترتیب ۵۶٪ و ۸۲٪ گزارش داده است.

۳- روش پیشنهادی

هدف این مقاله شناسایی و کاهش حملات *DDoS* در محیط رایانش ابری است. در اینجا، یک الگوریتم ترکیبی برای شناسایی این حملات براساس ویژگی‌های موثر پیشنهاد می‌شود که که ترافیک حمله *DDoS* را از ترافیک واقعی جدا می‌کند. به کارگیری و دسترسی به داده‌های اولیه مناسب در داده کاوی به منظور کشف دانش، به عنوان آماده‌سازی یا پیش‌پردازش داده‌ها یاد می‌شود. اهمیت پیش‌پردازش داده‌ها به دلیل این واقعیت است که؛ فقدان داده باکیفیت برابر با فقدان کیفیت در نتایج کاوش است و ورودی بد خروجی بدی را به دنبال دارد. بنابراین، در گام اول از روش پیشنهادی داده‌های ورودی با هدف بهبود کیفیت پیش‌پردازش می‌شوند. یکی از مشکلات پیاده‌سازی سیستم‌های پیش‌بینی و تشخیص زیاد بودن اطلاعات و بالا بودن تعداد ویژگی‌های ورودی است. وجود ویژگی‌های نامرتبط و زائد در مجموعه داده بر عملکرد مدل‌های یادگیری تأثیر منفی گذاشته و همچنین پیچیدگی محاسباتی را افزایش می‌دهد. در این مقاله برای کاهش پیچیدگی محاسباتی از الگوریتم بهینه‌سازی ازدحام ذرات استفاده می‌شود. این الگوریتم با هدف کاهش ابعاد داده‌ها، زیرمجموعه‌ای از ویژگی‌های موثر را برای استفاده در مدل دسته‌بندی انتخاب می‌کند. علاوه بر این، برای ایجاد مدل دسته‌بندی از تکنیک محاسبه آنتروپی اطلاعات و ایجاد درخت جستجوی دودویی متوازن استفاده می‌شود. این درخت به یک گرامر تبدیل می‌شود، بطوریکه در این نگاشت گره ریشه به عنوان نماد شروع و هر گره به عنوان یک متغیر در گرامر در نظر گرفته می‌شود. به طور کلی، مراحل انتخاب ویژگی و ایجاد مدل دسته‌بندی در یک پروسه انجام می‌شود، جائیکه در بخش انتخاب ویژگی برای محاسبه برازندگی راه‌حل‌ها از دقت مدل دسته‌بندی استفاده می‌شود. شکل ۲ فلوچارت روش پیشنهادی را نشان می‌دهد.



شکل ۲: فلوچارت روش پیشنهادی

۳-۱- پیش‌پردازش داده‌ها

در این تحقیق اطلاعات پایگاه داده مورد استفاده شامل اطلاعات رکوردهای ورودی شبکه است، بطوریکه برچسب رکوردها (حمله و

غیرحمله) نیز دردسترس می‌باشد. به طور کلی، برای ایجاد هر مدل دسته‌بندی، اعمال پیش‌پردازش روی داده‌ها لازم است. در اینجا، پیش‌پردازش به منظور بهبود کیفیت داده‌های واقعی اعمال می‌شود. این مرحله از روش تحقیق شامل دو بخش آماده‌سازی و نرمال-سازی است. در مرحله آماده‌سازی، مقادیر ویژگی‌هایی که ارزش‌های کیفی دارند به ارزش کمی تبدیل می‌شوند. برای اینکار به هر مقدار متمایز از هر ویژگی یک عدد تخصیص داده شده و این اعداد بجای مقادیر کیفی در پایگاه داده جایگزین می‌شوند. علاوه بر این، رکوردهایی با داده‌های از دست رفته از پایگاه داده حذف می‌شوند.

در مرحله بعد، داده‌ها نرمال‌سازی می‌شوند. نرمال‌سازی برای نگاشت داده‌ها از بازه فعلی به یک بازه مشخص است. اینکار بدلیل تنوع ویژگی‌ها، کاهش تاثیر ویژگی‌هایی با مقادیر بالا و همچنین نزدیک کردن پیش‌بینی‌های مدل‌های دسته‌بندی به یکدیگر انجام می‌شود. در این مقاله از تکنیک $ZScore$ برای نرمال‌سازی استفاده می‌شود، همانطور که در رابطه (۱) نشان داده شده است. این روش میانگین و انحراف معیار برای هر ویژگی را به ترتیب برابر ۰ و ۱ قرار می‌دهد.

$$x_{i,j}^{ZScore} = \frac{x_{i,j} - \mu_j}{\sigma_j} \quad (۱)$$

در این رابطه، $\boldsymbol{x_{i,j}}$ و $\boldsymbol{x_{i,j}^{ZScore}}$ به ترتیب مقدار واقعی و مقدار نرمال شده ویژگی $\boldsymbol{j}$-ام در نمونه $\boldsymbol{i}$-ام است. $\boldsymbol{\mu_j}$ و $\boldsymbol{\sigma_j}$ به ترتیب میانگین و انحراف معیار مقادیر تمام نمونه‌ها برای ویژگی $\boldsymbol{j}$-ام می‌باشد.

۳-۲- انتخاب ویژگی‌ها با الگوریتم بهینه‌سازی ازدحام ذرات

با توجه به حجم گسترده ویژگی‌های پایگاه‌های داده تشخیص حملات در محیط‌های رایانش ابری، در این مقاله با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات، زیرمجموعه ویژگی‌های بهینه جستجو و انتخاب می‌شوند. مراحل الگوریتم به صورت زیر است:

گام اول: ایجاد جمعیت اولیه از ذرات به صورت تصادفی، به طوریکه هر ذره یک زیرمجموعه از ویژگی‌ها را بیان می‌کند. طول هر ذره برابر با اندازه ویژگی‌های اصلی، $\boldsymbol{m}$، است. هر عنصر از ذره دارای مقدار ۰ یا ۱، به ترتیب به معنی انتخاب و یا عدم انتخاب ویژگی است. شکل ۳ ساختار ذرات را نشان می‌دهد.

f_1	f_2	...	f_k	...	f_m

شکل ۳: ساختار ذرات در مسئله انتخاب ویژگی

در اینجا، $\boldsymbol{f_j}$ به مفهوم وضعیت ویژگی $\boldsymbol{j}$-ام در ذره است و به صورت یک عدد باینری (۰ عدم انتخاب ویژگی و ۱ انتخاب ویژگی) بیان می‌شود، جائیکه $\boldsymbol{j = 1, 2, \dots, m}$ است.

گام دوم: تابع برازندگی برای هر ذره محاسبه می‌شود. در اینجا از ترکیب دقت مدل دسته‌بندی پیشنهاد شده، مجموع شباهت بین ویژگی‌های انتخاب‌شده و همچنین تعداد ویژگی‌های انتخاب شده برای محاسبه تابع برازندگی استفاده می‌شود. دلیل استفاده از این سه فاکتور اهمیت آنها در عملکرد نهایی مدل دسته‌بندی است. هدف اصلی انتخاب ویژگی حصول دقت بهتر در مدل دسته‌بندی است، بنابراین فاکتور دقت در تابع هدف استفاده شده است. فاکتور تعداد ویژگی‌های انتخاب شده باعث کاهش پیچیدگی مدل می‌شود، چون تعداد کمتر ویژگی‌های استفاده شده به پیش‌بینی سریع و مدل‌سازی بهتر منجر می‌شود. فاکتور شباهت بین ویژگی‌ها نیز در کاهش پیچیدگی مدل کمک می‌کند، چون انتخاب ویژگی‌هایی با شباهت بالا عملا باعث انتخاب ویژگی‌هایی با بار اطلاعاتی مشابهی می‌شود. بنابراین، در نظر گرفتن حداقل شباهت بین ویژگی‌ها در نهایت باعث ایجاد مدل بهتری برای دسته‌بندی می‌شود. رابطه (۲) تابع برازندگی برای ذره $\boldsymbol{k}$-ام را تعریف می‌کند.

$$Fitness(P_k) = \frac{Acc(p_k)}{|p_k| \times \sum_{f_i, f_j \in p_k} Sim(f_i, f_j)} \quad (۲)$$

در این رابطه، $\boldsymbol{Acc(p_k)}$ دقت دسته‌بندی با توجه به ویژگی‌های انتخاب شده در ذره $\boldsymbol{k}$-ام، $\boldsymbol{|p_k|}$ تعداد ویژگی‌های انتخاب شده

توسط ذره $\boldsymbol{k}$-ام و $\boldsymbol{Sim(f_i,f_j)}$ شباهت بین ویژگی‌های $\boldsymbol{f_i}$ و $\boldsymbol{f_j}$ از ذره $\boldsymbol{k}$-ام را نشان می‌دهد. در اینجا، مطابق رابطه (۳) از ضریب همبستگی پیرسون برای محاسبه شباهت بین ویژگی‌ها استفاده می‌شود.

$$sim(f_i, f_j) = \frac{\sum_{k=1}^{n}(f_{i,k} - \overline{f_i})(f_{j,k} - \overline{f_j})}{\sum_{i=1}^{n}(f_{i,k} - \overline{f_i})^2 \sum_{i=1}^{n}(f_{j,k} - \overline{f_j})^2} \quad (۳)$$

جائیکه، $\boldsymbol{f_{i,k}}$ و $\boldsymbol{f_{j,k}}$ نمونه $\boldsymbol{k}$-ام را به ترتیب برای ویژگی‌های $\boldsymbol{f_i}$ و $\boldsymbol{f_j}$ نشان می‌دهد، $\overline{\boldsymbol{f_i}}$ و $\overline{\boldsymbol{f_j}}$ به ترتیب میانگین همه نمونه‌ها برای ویژگی‌های $\boldsymbol{f_i}$ و $\boldsymbol{f_j}$ است و $\boldsymbol{n}$ به تعداد کل نمونه‌ها اشاره دارد.

گام سوم: موقعیت ذرات با استفاده از رابطه (۴) برورسانی می‌شوند.

$$p_k = \langle p_k + v_k \rangle \quad (۴)$$

در این رابطه، $\boldsymbol{v_k}$ و $\boldsymbol{p_k}$ به ترتیب سرعت و موقعیت فعلی ذره $\boldsymbol{k}$-ام است. ترم $\langle \boldsymbol{p_k + v_k} \rangle$ به طور کلاسیک برای محیط پیوسته استفاده می‌شود، با این حال فضای مسئله فوق گسسته است. از این‌رو، این ترم مطابق رابطه (۵) به صورت گسسته توسعه داده می‌شود.

$$p_{k,j} = \begin{cases} 0 & \langle p_{k,j} + v_{k,j} \rangle < 0.5 \\ 1 & otherwise \end{cases} \quad (۵)$$

در این رابطه، $\boldsymbol{p_{k,j}}$ مقدار ویژگی $\boldsymbol{j}$-ام (موقعیت) از ذره $\boldsymbol{k}$ می‌باشد. در اینجا، موقعیت ویژگی بر اساس سرعت حرکت ذره بروزرسانی می‌شود.

گام چهارم: سرعت ذرات با استفاده از رابطه (۶) بروزرسانی می‌شوند.

$$v_k = v_k \times \omega + c_1 \times r_1 \times (pbest_k - p_k) + c_2 \times r_2 \times (gbest - p_k), \quad v_k \in [v_{min}, v_{max}] \quad (۶)$$

در این رابطه، $\boldsymbol{c_1}$ و $\boldsymbol{c_2}$ پارامترهای ثابت یادگیری، $\boldsymbol{r_1}$ و $\boldsymbol{r_2}$ اعداد تصادفی در بازه ۰ و ۱، $\boldsymbol{pbest_k}$ بهترین موقعیت ذره $\boldsymbol{k}$-ام تاکنون و $gbest$ بهترین موقعیت سراسری همه ذرات می‌باشند. سرعت ذرات در هر ویژگی به یک مقدار $\boldsymbol{v_{max}}$ محدود می‌شوند. اگر مجموع شتاب‌ها باعث شوند که سرعت در یک ویژگی از $\boldsymbol{v_{max}}$ بیشتر شود، مقدار سرعت در آن ویژگی برابر با $\boldsymbol{v_{max}}$ قرار می‌گیرد. به طور مشابه سرعت ذرات در هر ویژگی به یک مقدار $\boldsymbol{v_{min}}$ نیز محدود می‌شوند.

گام پنجم: موقعیت‌های بهینه $pbest$ و $gbest$ با توجه به مقادیر برازندگی ذرات محاسبه می‌شود.

گام ششم: مراحل قبل تا برقراری شرط خاتمه الگوریتم تکرار می‌شوند، در این مقاله از تعداد ثابت تکرار برای شرط خاتمه الگوریتم استفاده شده است.

۳-۳- ایجاد مدل دسته‌بندی بر اساس محاسبه آنتروپی اطلاعات

در این بخش مدل دسته‌بندی بر مبنای استفاده از ساختمان داده درخت جستجوی دودویی متوازن و آرایه انجمنی (دیکشنری) ارائه می‌شود. در این الگوریتم از پارامترهای «نرخ ورودی به هر میزبان»، «نرخ خروجی از هر میزبان»، «نرخ افزایش نسبت ورودی به خروجی در هر میزبان» و «آنتروپی» برای شناسایی حملات استفاده شده است. نرخ ورودی، میزان بسته ورودی به یک میزبان و نرخ خروجی، میزان بسته خروجی از هر میزبان است که به صورت نسبی مطابق رابطه (۷) در نظر گرفته می‌شود.

$$R(p) = \frac{input - rate(p)}{output - rate(p)} \quad (۷)$$

در این رابطه، $\boldsymbol{p}$ یک پیوند آدرس خاص برای یک میزبان است.

نرخ افزایش نسبت ورودی به خروجی در هر میزبان به صورت $\boldsymbol{R_t(p)}$ در نظر گرفته می‌شود، جائیکه نشان‌دهنده نسبت نرخ ورودی به نرخ خروجی در زمان $\boldsymbol{t}$ برای میزبان $\boldsymbol{p}$ است. برای لحاظ کردن نرخ افزایش نسبت $\boldsymbol{R(p)}$، پنجره‌های زمانی به طول $\boldsymbol{w}$ در نظر گرفته می‌شود. در اینجا، برای اعمال نرخ تغییرات $\boldsymbol{R(p)}$ نسبت به زمان، از رابطه (۸) استفاده می‌شود

$$Slope(p) = \frac{R_{t_{end}}(p) - Slope_{t_{start}}(p)}{t_{start} - t_{end}}, \quad w = [t_{start}, t_{end}] \tag{۸}$$

در این رابطه، با لحاظ شدن میزان $Slope$ مرحله قبل، میزان و چگونگی تغییرات $R(p)$ برای تشخیص حملات در نظر گرفته شده است. بنابراین، در مجموع برای تشخیص حملات $DDoS$ از دو پارامتر R و $Slope$ استفاده می‌شود.

در این مقاله، برای کار دسته‌بندی، از یک ساختمان داده درخت برای شمارش و نگهداری نرخ ورودی و خروجی میزبان و نرخ افزایشی آن استفاده می‌شود. ساختمان داده در نظر گرفته شده یک درخت جستجوی دودویی متوازن ۲۵۶ تایی چهار سطحی می‌باشد که برای پوشش کامل یک پیوند آدرس نسخه چهار می‌باشد. با محاسبه مقادیر $R(p)$، $Slope(p)$و با در نظر گرفتن دو مقدار $\boldsymbol{R_{min}}$ و $\boldsymbol{R_{max}}$ به عنوان دو آستانه حملات $DDoS$ مطابق رابطه (۹) شناسایی می‌شوند.

$$\begin{cases} If\ (R_{min} \le R \le R_{max})\ and\ (slope \le slope_{max}), & valid\ \ action \\ If\ (R < R_{min})\ or\ (R > R_{max})\ and\ (slope > slope_{max}), & invalid\ \ action \end{cases} \tag{۹}$$

با توجه به ذخیره کردن تمامی پیوندهای آدرس در درخت و تکرار عمل بروزرسانی با دریافت هر بسته باید درخت یک بار پیمایش شود. از این‌رو، نیاز به مکانیزمی برای کاهش تعداد عملیات پیمایش درخت وجود دارد. بدین منظور از رویکرد فشرده‌سازی درخت جستجوی دودویی استفاده شده، جائیکه گره‌های درخت ۲۵۶ تایی به یک درخت جستجوی دودویی متوازن نگاشت می‌شود. در اینجا برای فشرده‌سازی درخت از الگوی گرامر استفاده شده، جائیکه مسیر از درخت به یک گرامر تبدیل می‌شود. در این نگاشت گره ریشه به عنوان نماد شروع و هر گره به عنوان یک متغیر در گرامر در نظر گرفته می‌شود. همچنین، هر یال به عنوان انتقالی از متغیر مبدأ به متغیر مقصد است. بعلاوه، در پیمایش در درخت جستجوی دودویی، پیمایش به هر گره که وارد می‌شود، درختی که تاکنون پیمایش شده است با استفاده از روال فشرده‌سازی، فشرده شده و در یک آرایه انجمنی (دیکشنری) ذخیره می‌شود. دلیل ذخیره این است که در بسته‌های دیگر اگر قسمتی از مسیر پیمایش یکسان باشد، دیگر درخت پیمایش نشود بلکه این مسیر از آرایه انجمنی مربوطه در زمان ثابت استخراج و به مسیر فعلی اضافه شود.

در ادامه روش پیشنهادی با محاسبه آنتروپی اطلاعات توسعه می‌یابد. آنتروپی یک مفهوم مهم تئوری اطلاعات است که تصادفی بودن (بی‌نظمی) داده‌های وارد شده به شبکه را اندازه‌گیری می‌کند. هرچه داده‌ها تصادفی باشند، آنتروپی بیشتری خواهند داشت. بر این اساس، برای شناسایی انحراف (تغییرات) در رفتار بسته‌ها، یک مقدار آستانه آنتروپی تنظیم می‌شود. با مقایسه مقدار حقیقی آنتروپی بسته و مقدار آستانه، اگر مقدار آنتروپی بسته‌ها با مقدار آستانه یکسان نباشد، یک حمله صورت گرفته یا تغییراتی در بی‌نظمی داده‌ها حاصل شده است. تغییر یا انحراف در آنتروپی برای شناسایی ترافیک حمله روی پیوندهای آدرس است. در اینجا، مقدار آنتروپی با توجه به پنجره زمانی $\boldsymbol{w}$ محاسبه می‌شود. با فرض اینکه تعداد بسته رد و بدل شده در پنجره زمانی $\boldsymbol{w}$ برابر $\boldsymbol{N}$ باشد؛ آنگاه آنتروپی مطابق رابطه (۱۰) محاسبه می‌شود.

$$H_w = -\frac{1}{N}\sum_{c} n_i \log_2(\frac{n_i}{N}) \tag{۱۰}$$

در این رابطه، $\boldsymbol{H_w}$ مقدار آنتروپی برای پنجره زمانی $\boldsymbol{w}$ و $\boldsymbol{n_i}$ تعداد تکرار پیوند آدرس $\boldsymbol{i}$-ام در طول پنجره است. برای در نظر گرفتن پارامتر آنتروپی از اختلاف دو مقدار آنتروپی در دو پنجره زمانی متوالی استفاده می‌شود. از این‌رو، اگر میزان تغییرات اختلاف آنتروپی در دو پنجره زمانی متوالی از مقدار ثابت گاما بیشتر باشد، ترافیک به عنوان یک حمله $DDoS$ شناسایی می‌شود. بنابراین؛

$$\begin{matrix} \Delta H_1 = H_{K+1} - H_K \\ \Delta H_2 = H_{K+2} - H_{K+1} \end{matrix} \quad \begin{cases} If\ \left(\gamma_{min} < \dfrac{\Delta H_1}{\Delta H_2} < \gamma_{max}\right) & Normal\ \ Transfer \\ If\ \left(\dfrac{\Delta H_1}{\Delta H_2} \ge \gamma_{max}\right)\ or\ \left(\dfrac{\Delta H_1}{\Delta H_2} \le \gamma_{min}\right) & DDoS\ \ \ Attack \end{cases} \tag{۱۱}$$

۴- نتایج شبیه سازی

در این بخش، آزمایش‌های گسترده‌ای برای ارزیابی و مقایسه روش پیشنهادی ارائه می‌شود. شبیه‌سازی با نرم‌افزار متلب ورژن ۲۰۱۹ و آزمایش‌ها توسط یک *PC* با پردازنده اینتل ۷ هسته، فرکانس ۳,۲ گیگاهرتز و حافظه ۱۶ گیگابایت انجام شده است. در این بخش مجموعه داده‌ها و معیارهای ارزیابی توضیح داده شده و سپس نتایج و مقایسه‌ها ارائه می‌گردد.

۴-۱- مجموعه داده‌ها

در این مقاله از رکوردهای مجموعه داده‌های *NSL-KDD* [۳۰] و *CICDDoS2019* [۳۱] برای شبیه‌سازی روش پیشنهادی و انجام آزمایش‌ها استفاده می‌شود. *NSL-KDD* نسخه توسعه یافته از مجموعه داده *KDDCUP* است و شامل یک کلاس نرمال و چهار کلاس نفوذ (*Dos*، *R2L*، *U2R* و *Probe*) می‌باشد. این مجموعه داده از ۴۱ ویژگی، ۲۲۵۴۴ نمونه در بخش آزمایش و ۱۲۵۹۷۳ نمونه در بخش آموزش تشکیل شده است. علاوه بر این، هر نوع رکورد نفوذ شامل چندین نوع زیرحمله است جائیکه جزئیات آن در [۳۰] قابل دسترس می‌باشد. مجموعه داده *CICDDoS2019* به طور اختصاصی حملات *DDOS* را ارائه می‌دهد. این مجموعه داده شامل ۸۰ ویژگی و ۲۳۰۰۰ رکورد است که ۱۲ نوع حمله *DDoS* در بخش آموزش و ۷ حمله در بخش آزمایش را ارائه می‌دهد.

۴-۲- معیارهای ارزیابی

یکی از مهمترین بخش‌های هر مدل مبتنی بر دسته‌بندی، تحلیل میزان کارایی آن است، جائیکه اینکار توسط معیارهای ارزیابی انجام می‌شود. معیارهای ارزیابی برای تحلیل کارایی یک مدل طبقه‌بندی از برچسب‌های کلاس واقعی و برچسب‌های کلاس پیش‌بینی استفاده می‌کنند. وقوع حالات مختلف این دو نوع برچسب با ترم‌های *FN*، *TN*، *FP* و *TP* نشان داده می‌شود. وقوع این حالات در یک مدل با دو کلاس میتواند به صورت یک ماتریس درهمریختگی مطابق جدول ۱ نشان داده شود.

جدول ۱: ماتریس درهمریختگی برای یک مدل دسته‌بندی با دو کلاس

کلاس واقعی	کلاس پیش‌بینی نفوذ	کلاس پیش‌بینی نرمال
رکورد نفوذ	*TP* (مثبت حقیقی)	*FN* (منفی کاذب)
رکورد نرمال	*FP* (منفی حقیقی)	*TN* (مثبت کاذب)

در اینجا، مطابق جدول ۲ از تعداد ویژگی‌های انتخاب شده، دقت، درستی، فراخوان و اندازه‌گیری F به عنوان معیارهای ارزیابی استفاده می‌شود.

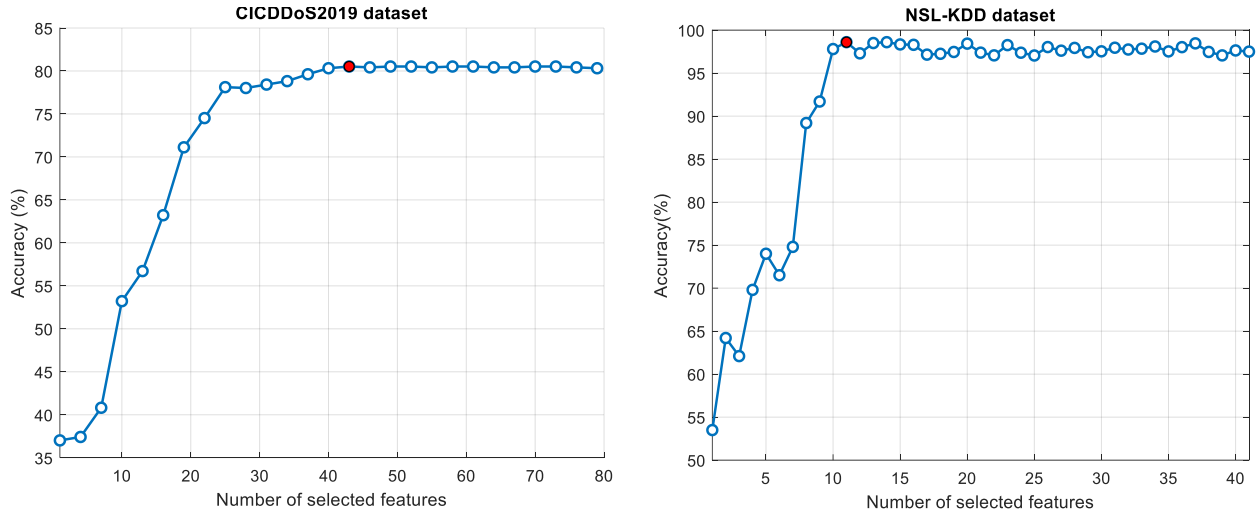
جدول ۲: معیارهای ارزیابی

معیار	توصیف	رابطه
دقت	نسبت نمونه‌های مثبت حقیقی و منفی حقیقی به کل نمونه‌های مجموعه آزمایش	$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$
درستی	نسبت نمونه‌های مثبت حقیقی به کل نمونه‌های پیش‌بینی شده به صورت مثبت	$Precision = \frac{TP}{TP + FP}$
فراخوان	نسبت نمونه‌های مثبت حقیقی به کل نمونه‌های واقعی موجود	$Recall = \frac{TP}{TP + FN}$
اندازه‌گیری F	میانگین هارمونیک دو معیار درستی و فراخوان را در نظر می‌گیرد.	$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$

۴-۳- نتایج و مقایسه‌ها

در این بخش، همه نتایج بر اساس تکنیک اعتبارسنجی *10-Fold* روی مجموعه داده‌های *NSL-KDD* و *CICDDoS2019* گزارش می‌شوند، از اینرو مقایسه‌ها در شرایط یکسانی انجام شده است.

استفاده از تکنیک انتخاب ویژگی‌ها با طول متغیر در الگوریتم *PSO* علاوه بر انتخاب ویژگی‌های موثر، تعداد بهینه این ویژگی‌ها را نیز ارائه می‌دهد. شکل ۴ دقت روش پیشنهادی را با تعداد مختلف ویژگی‌ها نشان می‌دهد. در اینجا بهترین معیار دقت برای هر تعداد ویژگی مختلف گزارش شده است. نتایج نشان دهنده بهترین دقت طبقه‌بندی ۹۸٫۶۵٪ با ۱۱ ویژگی برای مجموعه داده *NSL-KDD* و ۸۰٫۵۲٪ با ۴۳ ویژگی برای مجموعه داده *CICDDoS2019* است.



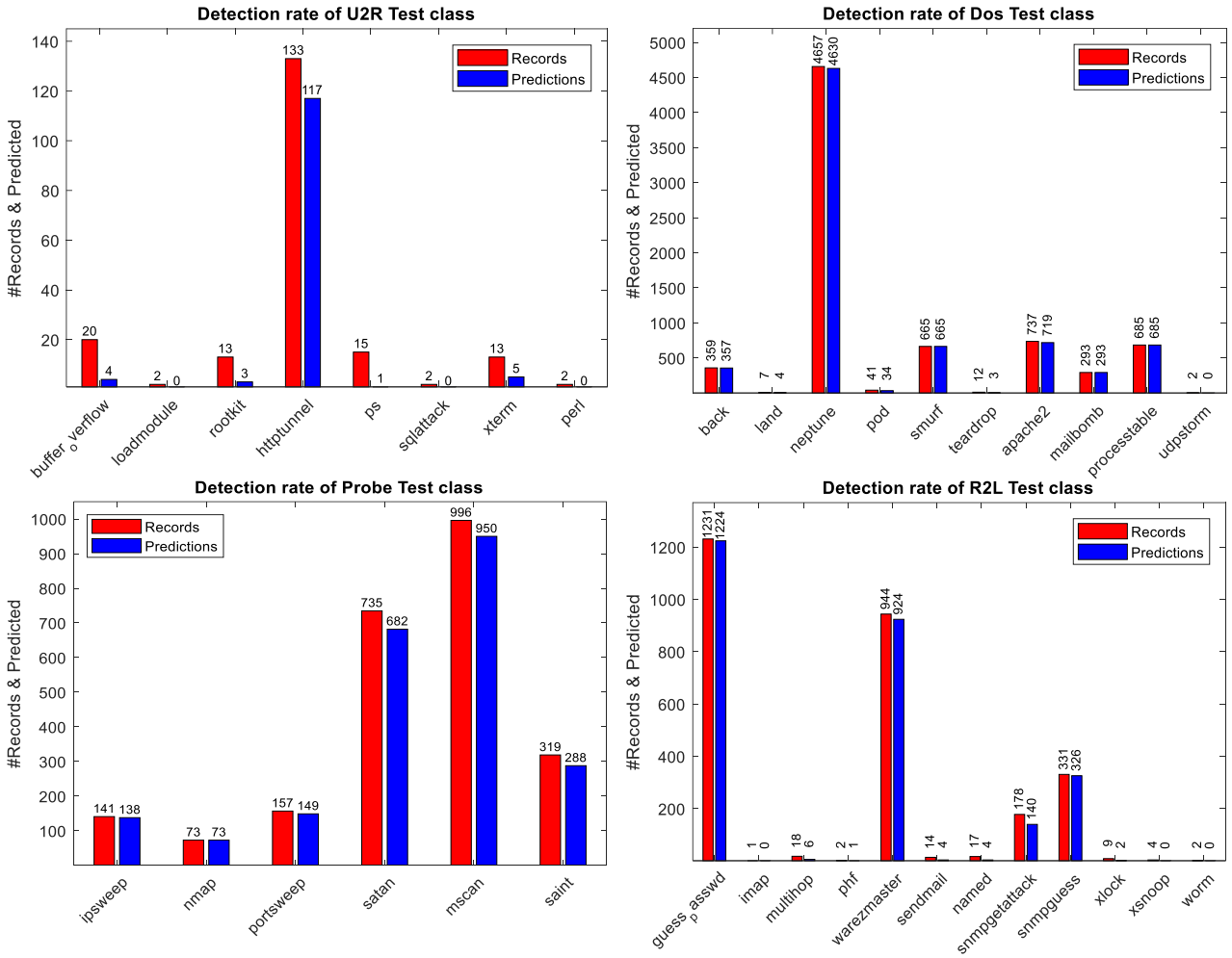
شکل ۴: دقت روش پیشنهادی با تعداد ویژگی‌های مختلف

به طور کلی، مجموعه داده *NSL-KDD* شامل ۴۱ ویژگی در دو کلاس نرمال و نفوذ است. جزئیات ویژگی‌های برای این مجموعه داده در جدول ۳ ارائه شده است. علاوه بر این، در این جدول زیرمجموعه ویژگی‌های انتخاب شده توسط روش پیشنهادی برجسته شده است. این ویژگی‌ها توسط الگوریتم ازدحام ذرات جستجو شده و در حالت میانگین گزارش شده است.

جدول ۳: ویژگی‌های مجموعه داده *NSL-KDD* و ویژگی‌های بهینه انتخاب شده

شماره	نام ویژگی	شماره	نام ویژگی	شماره	نام ویژگی	شماره	نام ویژگی
۱	duration	۱۱	num_failed_logins	۲۱	is_host_login	۳۱	srv_diff_host_rate
۲	protocol_type	۱۲	logged_in	۲۲	is_guest_login	۳۲	dst_host_count
۳	service	۱۳	num_compromised	۲۳	Count	۳۳	dst_host_srv_count
۴	flag	۱۴	root_shell	۲۴	srv_count	۳۴	dst_host_same_srv_rate
۵	src_bytes	۱۵	su_attempted	۲۵	serror_rate	۳۵	dst_host_diff_srv_rate
۶	dst_bytes	۱۶	num_root	۲۶	srv_serror_rate	۳۶	dst_host_same_src_port_rate
۷	Land	۱۷	num_file_creations	۲۷	rerror_rate	۳۷	dst_host_srv_diff_host_rate
۸	wrong_fragment	۱۸	num_shells	۲۸	srv_rerror_rate	۳۸	dst_host_serror_rate
۹	Urgent	۱۹	num_access_files	۲۹	same_srv_rate	۳۹	dst_host_srv_serror_rate
۱۰	Hot	۲۰	num_outbound_cmds	۳۰	diff_srv_rate	۴۰	dst_host_rerror_rate
						۴۱	dst_host_srv_rerror_rate

در مجموعه داده *NSL-KDD* چندین نوع نفوذ وجود دارد و هر نوع نیز شامل چندین نوع حمله می‌باشد. در اینجا عملکرد روش پیشنهادی برای هر نوع نفوذ گزارش شده است. شکل ۵ به ترتیب نتایج برای نوع نفوذهای *Dos*، *R2L*، *U2R* و *Probe* نشان می‌دهد. در اینجا، نتایج بر اساس تعداد کل رکوردها و تعداد رکوردهای صحیح تشخیص داده شده در بخش آزمایش برای هر زیرکلاس ارائه شده است.



شکل ۵: نتایج پیش‌بینی برای زیرکلاس‌های نفوذ از مجموعه داده *NSL-KDD*

نتایج مقایسه روش‌ها مختلف بر اساس معیار زمان اجرا در جدول ۴ گزارش شده است. تمام نتایج براساس میانگین اعتبارسنجی *10-fold* است. روش پیشنهادی و روش *AGA* با توجه به استفاده از الگوریتم تکاملی در بدنه خود تقریبا زمان اجرای مشابهی دارند، با این حال روش پیشنهادی به دلیل استفاده از ویژگی‌های کمتر در مدل‌سازی زمان اجرای کمتری دارد. علاوه بر این، روش *E-SVM* از رویکردهای تکاملی استفاده نمی‌کند و به همین دلیل زمان اجرای کمتری نسبت به سایر روش‌ها دارد. این آزمایش برای هر دو مجموعه داده استفاده شده نتایج مشابهی ارائه می‌دهد.

جدول ۴: مقایسه روش‌های مختلف از لحاظ زمان اجرا

روش‌ها	مجموعه داده *NSL-KDD*	مجموعه داده *CICDDoS2019*
AGA [۲۳]	۹۴۸	۷۷۰
E-SVM [۲۴]	۳۵۱	۲۱۸
روش پیشنهادی	۷۹۴	۵۳۵

در ادامه نتایج روش پیشنهادی روی مجموعه داده *NSL-KDD* با دو الگوریتم *AGA* [۲۳] *E-SVM* [۲۴] مقایسه می‌شود. این مقایسه در جدول ۵ بر اساس معیارهای دقت، درستی، فراخوان و اندازه‌گیری *F* انجام شده است. نتایج برتری روش پیشنهادی را با دقت تشخیص ۹۹,۸۴٪ نسبت به الگوریتم‌های *AGA* و *E-SVM* در اغلب معیارها نشان می‌دهد. به طور کلی روش پیشنهادی در معیار دقت نسبت به *AGA* حدود ۰,۵٪ برتری دارد. این برتری در معیار اندازه‌گیری *F* نسبت به *E-SVM* حدود ۷٪ است.

جدول ۵: مقایسه روش پیشنهادی با الگوریتم‌های مشابه بر اساس مجموعه داده *NSL-KDD*

معیار ارزیابی	روش‌ها	*Normal*	*Dos*	*R2L*	*U2R*	*Probe*	میانگین
دقت	*AGA* [۲۳]	۹۹,۸۲	۹۹,۸۵	۹۹,۶۴	۹۸,۶۸	۹۹,۷۷	۹۹,۸۱
	E-SVM [۲۴]	–	–	–	–	–	–
	روش پیشنهادی	۹۹,۷۹	۹۹,۹۳	۹۹,۵۰	۹۸,۲۴	۹۹,۸۰	۹۹,۸۴
درستی	*AGA* [۲۳]	۹۹,۸۷	۹۹,۹۷	۹۹,۴۲	۹۷,۵۸	۹۹,۷۱	–
	E-SVM [۲۴]	۹۹,۶۳	۹۹,۷۶	۹۰,۵۵	۷۶,۸۵	۹۸,۳۲	۹۲,۳۴
	روش پیشنهادی	۹۹,۸۷	۹۹,۸۱	۹۹,۳۹	۹۶,۸۷	۹۹,۷۸	۹۹,۸۰
فراخوان	*AGA* [۲۳]	۹۹,۷۷	۹۹,۷۳	۹۹,۸۶	۹۹,۸۳	۹۹,۸۲	–
	E-SVM [۲۴]	۹۹,۳۴	۱۰۰,۰	۹۱,۳۰	۷۹,۳۸	۹۹,۱۲	۹۴,۰۵
	روش پیشنهادی	۹۹,۸۲	۱۰۰,۰	۹۹,۹۶	۹۸,۰۰	۹۹,۹۳	۹۹,۸۷
اندازه‌گیری *F*	*AGA* [۲۳]	۹۹,۸۲	۹۹,۹۳	۹۹,۶۴	۹۸,۶۹	۹۹,۷۷	–
	E-SVM [۲۴]	۹۹,۴۸	۹۹,۸۸	۹۰,۹۲	۷۸,۱۰	۹۸,۷۲	۹۳,۱۹
	روش پیشنهادی	۹۹,۴۸	۹۹,۹۰	۹۹,۶۷	۹۷,۴۳	۹۹,۸۵	۹۹,۸۳

در ادامه نتایج شبیه‌سای روش پیشنهادی روی مجموعه داده *CICDDoS2019* ارائه شده است. جدول ۶ نتایج معیارهای ارزیابی مختلف را برای روش پیشنهادی و چهار الگوریتم معمول یادگیری ماشین (درخت تصمیم *ID3*، جنگل تصادفی، نایو بیز و رگرسیون لجستیک) نشان می‌دهد [۳۱].

جدول ۶: مقایسه روش پیشنهادی با الگوریتم‌های یادگیری ماشین روی مجموعه داده *CICDDoS2019*

الگوریتم‌ها	دقت	درستی	فراخوان	اندازه‌گیری *F*
درخت تصمیم *ID3*	۷۶,۹۱	۷۸,۴۱	۶۵,۰۰	۷۱,۰۸
جنگل تصادفی	۷۶,۲۵	۷۷,۳۲	۵۶,۲۵	۶۵,۱۲
نایو بیز	۴۰,۰۵	۴۱,۸۷	۳۱,۵۴	۳۵,۹۸
رگرسیون لجستیک	۲۵,۴۱	۲۵,۴۶	۲۰,۹۸	۲۳,۰۱
روش پیشنهادی	۸۰,۵۲	۸۴,۰۵	۷۳,۱۲	۷۸,۲۰

با توجه به سه معیار ارزیابی (درستی، فراخوان و اندازه‌گیری *F*)، بیشترین دقت مربوط به الگوریتم‌های جنگل تصادفی و درخت تصمیم *ID3* است. همچنین، از نظر معیار فراخوان *ID3* بهترین عملکرد را ارائه داده است. رگرسیون لجستیک در کل بدترین نتیجه را گزارش داده است. بر اساس آزمایشات برای آموزش درخت تصمیم *ID3* تنها چند دقیقه زمان نیاز است. پس از *ID3* جنگل تصادفی، نایو بیز و رگرسیون لجستیک از نظر زمانی به ترتیب در رتبه‌های بعدی هستند. با در نظر گرفتن زمان اجرا و معیارهای ارزیابی، *ID3* بهترین الگوریتم یادگیری ماشین با کمترین زمان اجرا و بالاترین دقت است. با این حال، روش پیشنهادی در مقایسه با چهار الگوریتم معمول یادگیری ماشین عملکرد بهتری با دقت ۸۰,۵۲٪ را ارائه داده است.

۵- نتیجه‌گیری

از ویژگی‌های بارز رایانش ابری (مانند سرویس درخواستی، جمع‌آوری منابع، دسترسی به شبکه گسترده، ذخیره‌سازی داده‌ها و غیره) توسط مهاجمان برای شروع حمله منع سرویس توزیع شده استفاده می‌شود. به طور کلی، حملات *DDoS* در چنین محیطی با ایجاد حجم عظیمی از ترافیک مخرب برای اختلال منابع سرورهای قربانی استفاده می‌کند. با استفاده از ویژگی‌های برجسته رایانش ابری، حمله سریع و پیچیده *DDoS* برای یک مهاجم آسان می‌شود. بنابراین، این مقاله بر روی شناسایی و تجزیه و تحلیل حملات *DDoS* در محیط رایانش ابری متمرکز است. به طور کلی، به دلیل استفاده از اینترنت برای سرویس‌دهی، محیط رایانش ابری در برابر حملات مخرب آسیب‌پذیر است. بنابراین، این محیط نیاز به یک سیستم تشخیص نفوذ برای مقابله با چنین حملاتی دارد. این مقاله یک سیستم ترکیبی برای تشخیص این نوع حملات پیشنهاد می‌کند، بطوری که از آنتروپی و بهینه‌سازی ازدحام ذرات تشکیل شده تا دقت تشخیص را در محیط رایانش ابری بهبود بخشد. روش پیشنهادی با الگوریتم‌های موجود بر اساس مجموعه داده‌های *NSL-KDD* و *CICDDoS2019* مقایسه شده است. نتایج بدست آمده نشان می‌دهد که روش پیشنهادی میتواند حملات را با دقت تشخیص بالاتری نسبت به تکنیک‌های موجود تشخیص دهد. این مقاله محققان امنیتی را به توسعه راه‌حل‌های موثر در زمینه پیشگیری، شناسایی و ایمنی در برابر حملات *DDoS* در محیط ابر تحریک می‌کند. کارهای آینده چگونگی بهره‌برداری از ویژگی‌های برجسته رایانش ابری توسط مهاجمان برای شروع حملات مختلف *DDoS* را مورد بحث قرار می‌دهد.

مراجع

[1] S. Q. A. Shah , F. Z. Khan and M. Ahmad “The impact and mitigation of ICMP based economic denial of sustainability attack in cloud computing environment using software defined network.” Computer Networks, vol.187, pp.107825, 2021.

[2] M. Haddadi and R. Beghdad “A Confidence Interval Based Filtering Against DDoS Attack in Cloud Environment: A Confidence Interval Against DDoS Attack in the Cloud.” International Journal of Information Security and Privacy (IJISP), vol.14,no.4, pp.42-56, 2020.

[3] R. SaiSindhuTheja and G. K. Shyam “An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment.” Applied Soft Computing, vol.100, pp.106997, 2021.

[4] O. Osanaiye, K. K. R. Choo and M. Dlodlo “Distributed denial of service (DDoS) resilience in cloud: Review and conceptual cloud DDoS mitigation framework.” Journal of Network and Computer Applications, vol.67, pp.147-165, 2016.

[5] P. S. Bawa, S. U. Rehman and S. Manickam “Enhanced mechanism to detect and mitigate economic denial of sustainability (EDoS) attack in cloud computing environments.” Int. J. Adv. Comput. Sci. Appl, vol.8, no.9, pp.51-58, 2017.

[6] H. A. Kholidy, “Detecting impersonation attacks in cloud computing environments using a centric user profiling approach.” Future Generation Computer Systems, vol.117, pp.299-320, 2021.

[7] S. Subashini and V. Kavitha “A survey on security issues in service delivery models of cloud computing.” Journal of network and computer applications, vol.34, no.1, pp.1-11, 2011.

[8] N. Agrawal and S. Tapaswi “Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges.” IEEE Communications Surveys & Tutorials, vol.21, no.4, pp.3769-3795, 2019.

[9] G. Somani., M. S. Gaur, D. Sanghi, M. Conti and R. Buyya “DDoS attacks in cloud computing: Issues, taxonomy, and future directions.” Computer Communications, vol.107, pp.30-48, 2017

[10] A. Praseed and P. S. Thilagam “DDoS attacks at the application layer: Challenges and research perspectives for safeguarding Web applications.” IEEE Communications Surveys & Tutorials, vol.21, no.1, pp. 661-685, 2018.

[11] G. Somani, M. S. Gaur, D. Sanghi, , M. Conti, M. Rajarajan and R. Buyya “Combating DDoS attacks in the cloud: requirements, trends, and future directions.” IEEE Cloud Computing, vol.4,no.1, pp.22-32, 2017.

[12] A. Shameli-Sendi, M. Pourzandi, M. Fekih-Ahmed and M. Cheriet “Taxonomy of distributed denial of service mitigation approaches for cloud computing.” Journal of Network and Computer Applications, vol.58, pp.165-179, 2015.

[13] S. T. Zargar, J. Joshi and D. Tipper “A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks.” IEEE communications surveys & tutorials, vol.15,no.4, pp.2046-2069, 2013.

[14] N. Agrawal and S. Tapaswi “A lightweight approach to detect the low/high rate IP spoofed cloud DDoS attacks.” in IEEE 7th International Symposium on Cloud and Service Computing (SC2),2017, pp. 118-123.

[15] G. I. Shidaganti, A. S. Inamdar, S. V. Rai and A. M. Rajeev “SCEF: A model for prevention of DDoS attacks from the cloud.” International Journal of Cloud Applications and Computing (IJCAC), vol.10, no.3,pp. 67-80, 2020.

[16] H. F. El-Sofany "A New Cybersecurity Approach for Protecting Cloud Services against DDoS Attacks." International Journal of Intelligent Engineering and Systems, vol.13,no.2,pp. 205-215, 2020.
[17] A. Bhardwaj, V. Mangat, R. Vig, , S. Halder, and, M. Conti "Distributed denial of service attacks in cloud: State-of-the-art of scientific and commercial solutions." Computer Science Review, vol.39, pp.100332, 2021.
[18] A. Saied, R. E. Overill and T. Radzik "Detection of known and unknown DDoS attacks using Artificial Neural Networks." Neurocomputing, vol.172, pp.385-393, 2016.
[19] S. Rastegari, P. Hingston and C. P. Lam "Evolving statistical rulesets for network intrusion detection." Applied soft computing, vol.33, pp.348-359, 2015.
[20] N. K. Kanakarajan and, K. Muniasamy"Improving the accuracy of intrusion detection using gar-forest with feature selection." In Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA), 2015, pp. 539-547.
[21] B. Ingre and A. Yadav "Performance analysis of NSL-KDD dataset using ANN." in international conference on signal processing and communication engineering systems,2015, pp. 92-96.
[22] S. M. H. Bamakan, H. Wang, T. Yingjie and Y. Shi "An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization." Neurocomputing, vol.199, pp.90-102, 2016.
[23] M. Ghalehgolabi and A. Rezaeipanah "Intrusion Detection System Using Genetic Algorithm and Data Mining Techniques Based on the Reduction." International Journal of Computer Applications Technology and Research, vol.6,no.11, pp.461-466, 2017.
[24] C. Yang "Anomaly network traffic detection algorithm based on information entropy measurement under the cloud computing environment." Cluster Computing, vol.22,no.4, pp.8309-8317, 2019.
[25], N. Agrawal, and, S. Tapaswi "An SDN-Assisted Defense Mechduanism for the Shrew DDoS Attack in a Cloud Computing Environment." Journal of Network and Systems Management, vol.29,no.2, pp.1-28, 2021.
[26], A. Sharma, , C. Agrawal, , A. Singh, and, K. Kumar "Real-time DDoS detection bsed on entropy using Hadoop framework." In Computing in Engineering and Technology,2020, pp. 297-305.
[27] K. B. Virupakshar, M. Asundi, K. Channal, P. Shettar, S. Patil, and D. G. Narayan, "Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud", Procedia Computer Science, vol.167, pp.2297-2307, 2020.
[28] A. B. Dehkordi, M. Soltanaghaei, and F. Z. Boroujeni, "The DDoS attacks detection through machine learning and statistical methods in SDN". The Journal of Supercomputing, vol.77, no.3,pp. 2383-2415, 2021.
[29] D. Kshirsagar, and S. Kumar , "A feature reduction based reflected and exploited DDoS attacks detection system", Journal of Ambient Intelligence and Humanized Computing,vol.13,no.3, pp.1-13, 2021.
[30] Nsl-kdd data set for network based intrusion detection systems." Available on: http://nsl.cs.unb.ca/KDD/NSL-KDD.html, November 2020.
[31] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy." In International Carnahan Conference on Security Technology (ICCST), 2019, pp. 1-8.

An Approach to DDoS Attacks Detection in Cloud Computing Environment Using Entropy and Particle Swarm Optimization

Mehdi Asayesh Joo[1], Mehdi Sadeghzadeh[2*], Maziyar Ganjoo[3]

1: Department of Computer, Bushehr Branch, Islamic Azad University, Bushehr, Iran, mehdi.asayeshjo@gmail.com
2*: Department of Computer, Mahshahr Branch, Islamic Azad University, Mahshahr, Iran, sadegh_1999@yahoo.com
3: Department of Information Technology, Bushehr Branch, Islamic Azad University, Bushehr, Iran, ganjoo@gmail.com

ABSTRACT:

Availability of cloud services is one of the most important concerns of cloud service providers. While cloud services are mainly transmitted over the Internet, they are prone to various attacks that may lead to the leakage of sensitive information. Distributed Denial-of-Service (DDoS) attack is known as one of the most important security threats to the cloud computing environment. This attack is an explicit attempt by an attacker to block or deny access to shared services or resources in a cloud environment. Many methods have been proposed to quickly and accurately predict these attacks. However, given the breadth of features available, efforts are still ongoing given the importance of the issue. Entropy approach and particle swarm optimization are common algorithms in machine learning, which are very popular for prediction and categorization problems. This paper discusses a hybrid approach to dealing with DDoS attack in the cloud computing environment. This method highlights the importance of effective feature-based selection methods and classification models. Here, an entropy-based approach and particle swarm optimization to counter these attacks in a cloud computing environment is presented. Classification on high-dimensional data typically requires feature selection as a pre-processing step to reduce the dimensionality. However, effective features selecting is a challenging task, which in this paper uses particle swarm optimization. Here, the proposed classification model is developed based on the use of a balanced binary search tree and dictionary data structure. The simulation is based on the NSL-KDD and CICDDoS2019 datasets, which prove the superiority of the proposed method with an average detection accuracy of 99.84% over the AGA and E-SVM algorithms.

KEYWORDS: Cloud computing, attack detection, entropy, particle swarm optimization, DDoS attack.