
بسم الله الرحمن الرحيم

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

رشته مهندسی کامپیوترگرایش هوش مصنوعی و رباتیک

پایان‌نامه کارشناسی ارشد

رده‌بندی و تحلیل خودکار ترافیک شبکه بر اساس کاربرد با استفاده از روش‌های یادگیری ترکیبی

نگارنده: مهسا ناظمی‌گلیان

استاد راهنما

دکتر هدی مشایخی

شهریور ۱۳۹۵

شماره: ۳۰۷ ر ف ک ـ
تاریخ: ۹۵/۷/۲۹
ویرایش:

باسمه تعالی

مدیریت تحصیلات تکمیلی

فرم شماره ۷: صورتجلسه نهایی دفاع از پایان نامه دوره کارشناسی ارشد

با تأییدات خداوند متعال و با استعانت از حضرت ولی عصر (عج) ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد خانم / آقای مهسا ناظمی گلیان به شماره دانشجویی ۹۳۱۷۲۵۴ رشته مهندسی کامپیوتر گرایش هوش مصنوعی و رباتیک تحت عنوان ردهبندی و تحلیل خودکار ترافیک شبکه بر اساس کاربرد با استفاده از روشهای یادگیری ترکیبی

که در تاریخ ۹۵/۶/۱۸ با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می‌گردد:

قبول (با درجه عالی امتیاز ۱۸) ☑ دفاع مجدد ☐ مردود ☐

نوع تحقیق: نظری ☐ عملی ☐

۱- عالی (۲۰ - ۱۹)
۲- بسیار خوب (۱۸/۹۹ - ۱۸)
۳- خوب (۱۷/۹۹ - ۱۶)
۴- قابل قبول (۱۵/۹۹ - ۱۴)
۵- نمره کمتر از ۱۴ غیر قابل قبول

عضو هیأت داوران	نام ونام خانوادگی	مرتبهٔ علمی	امضاء
۱- استادراهنمای اول	دکتر هدی مشایخی	استادیار	
۲- استادراهنمای دوم			
۳- استاد مشاور			
۴- نماینده شورای تحصیلات تکمیلی			
۵- استاد ممتحن اول	دکتر حمید حسن‌پور	استاد	
۶- استاد ممتحن دوم	دکتر امیدرضا معروضی	استادیار	

نام و نام خانوادگی رئیس دانشکده: دکتر حمید حسن‌پور

تاریخ و امضاء و مهر دانشکده:

۶

تقدیم به پدر و مادر عزیزم

که مهر آسمانی‌شان آرام‌بخش آلام زمینی‌ام است.

به استوارترین تکیه‌گاهم، دستان پرمهر پدرم و به زیباترین نگاه زندگیم، چشمان پرمهر مادرم

که هرچه آموخته‌ام در مکتب عشق شما آموختم، باشد که حاصل تلاشم نسیم‌گونه غبار خستگی‌تان را بزداید.

از استاد گرامی سرکار خانم دکتر هدی مشایخی بسیار سپاسگزارم چرا که بدون راهنمایی‌های ایشان تأمین این پایان نامه بسیار مشکل می‌نمود.

تعهد نامه

اینجانب **مهسا ناظمی گلیان** دانشجوی دوره کارشناسی ارشد رشته **مهندسی کامپیوتر** دانشکده**مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی شاهرود** نویسنده پایان‌نامه **رده‌بندی و تحلیل خودکار ترافیک شبکه بر اساس کاربرد با استفاده از روش‌های یادگیری ترکیبی** تحت راهنمائی **دکتر هدی مشایخی** متعهد می شوم.

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « **Shahrood University of Technology** » به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافتهای آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است

.

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رایانه ای، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده

گسترش روزافزون استفاده از شبکه اینترنت انگیزه‌ی بیشتری برای مهاجمان در جهت ایجاد حملات اینترنتی گسترده‌تر ایجاد نموده است. در دهه اخیر، با افزایش تهدیدات ناشی از این نوع حملات، حفظ امنیت شبکه‌ها و نظارت بر ترافیک شبکه‌ها از اهمیت فراوانی برخوردار شده است. به طور کلی ترافیک موجود در هر شبکه می‌توانند با دو نوع هدف ایجاد شده باشند؛ اهداف مخرب و حمله و یا اهداف سالم و کاربردهای عادی. رده‌بندی جریان‌های ترافیک بسته به نوع کاربرد و هدف جریان‌های تولیدی، صورت می‌گیرد. در سال‌های اخیر بات‌نت‌ها به عنوان گسترده‌ترین و خطرناک‌ترین تهدیدها در بستر اینترنت شناخته شده‌اند. تاکنون در جهت شناسایی این نوع حملات رویکردهای متفاوتی معرفی شده است که رایج‌ترین و موثرترین آن‌ها رویکردهای مبتنی بر یادگیری ماشین می‌باشند. یکی از مهم‌ترین دلایل گرایش محققان به سمت رویکردهای مبتنی بر یادگیری ماشین، قدرت تعمیم‌پذیری بیشتر این روش‌ها برای شناسایی حملات بات‌نت‌های جدید می‌باشد.

به دلیل اهمیت ویژه‌ی بات‌نت‌ها در دهه اخیر، در این پژوهش، یک سیستم تشخیص بات‌نت بر اساس یادگیری افزایشی و بر مبنای رده‌بندی ترافیک ارائه شده است. در این سیستم، جریان‌های ترافیک مورد بررسی قرار گرفته و بر حسب این که این جریان‌ها اهداف مخرب داشته و ویژگی‌هایی مشابه با ویژگی‌های بات‌نت‌ها داشته‌اند و یا این که سالم هستند، به دو دسته سالم یا بات‌نت دسته‌بندی می‌گردند. رده‌بند مورد استفاده بر پایه الگوریتم K– نزدیک‌ترین همسایه عمل می‌کند. آموزش در این روش پیشنهادی به صورت افزایشی انجام شده و سیستم در حین اجرا دائماً رده‌بند خود را با توجه به انواع نمونه‌های جدیدی که مشاهده می‌کند به روزرسانی می‌نماید؛ بنابراین در تشخیص بات‌نت‌های جدید به سطح بالاتری از تعمیم‌پذیری دست می‌یابد. این سیستم علاوه بر اینکه همانند سایر روش‌های برخط، همواره روند یادگیری

را ادامه می‌دهد، قادر است بدون داشتن برچسب واقعی نمونه‌های جدید، برچسب آن‌ها را پیش‌بینی نموده و از آن‌ها در یک رده‌بندی با ناظر استفاده نماید. علاوه بر این، به منظور دست یافتن به یک ارزیابی معتبر از عملکرد واقعی سیستم، که این نوع ارزیابی در میان پژوهش‌های انجام شده بسیار کم دیده می‌شود، سیستم به وسیله‌ی یک مجموعه داده‌ی جامع و معتبر مورد ارزیابی قرار گرفته است که از درجه‌ی بالایی از تنوع بات‌نت‌ها برخوردار می‌باشد. نتایج آزمایش‌ها و مقایسه‌های انجام شده، نشان می‌دهد که این سیستم قادر است در محیط پویا با انواع مختلف بات‌نت‌ها، به خوبی عمل کند. بیشترین بهبود حاصل در نرخ تشخیص در این سیستم نسبت به سیستم‌های مشابه ۱۳٪ می‌باشد.

کلمات کلیدی: رده‌بندی ترافیک، یادگیری ماشین، تشخیص بات‌نت، یادگیری افزایشی

مقاله مستخرج از پایان‌نامه

۱. ناظمی گلیان، م. و مشایخی، ه. (۱۳۹۵)، " تشخیص باتنت براساس رده‌بندی ترافیک و یادگیری افزایشی"، کنفرانس بین المللی مهندسی کامپیوتر و فناوری اطلاعات، تهران، ایران.

فهرست مطالب

فهرست اشکال

فهرست جداول

فصل اول

مقدمه

امروزه گسترش روزافزون استفاده از شبکه اینترنت انگیزه‌ی بیشتری برای مهاجمان برای ایجاد حملات اینترنتی گسترده‌تر به کمک ابزارها و روش‌های پیچیده‌تر ایجاد نموده است. با افزایش تهدیدات ناشی از این نوع حملات، حفظ امنیت شبکه‌ها و در واقع نظارت بر ترافیک شبکه‌ها از اهمیت فراوانی برخوردار شده است. یکی از مهم‌ترین وظایف نظارت بر ترافیک، رده‌بندی ترافیک شبکه جهت شناسایی وقوع حمله می‌باشد که نقش بسیار مهمی در مدیریت و امنیت شبکه دارد.

روش‌های کلاسیک رده‌بندی جریان ترافیک، شامل پیش‌بینی مبتنی بر پورت[1] و بررسی مبتنی بر محموله[2] [۱] می‌باشد که در محیط شبکه‌های کنونی به دلیل مسائلی چون پورت‌های پویا[3] و برنامه‌های کاربردی رمزگذاری شده[4] دارای محدودیت بوده و کارایی خود را از دست داده‌اند. همچنین تا پیش از چند سال اخیر، تقریباً تمامی برنامه‌های کاربردی اینترنت از پورت‌های با پروتکل لایه انتقال شناخته شده استفاده می‌کردند که به راحتی قابل شناسایی بودند. اما اخیراً تعداد برنامه‌های کاربردی که از پورت‌های تصادفی و یا غیراستاندارد استفاده می‌کنند به شدت افزایش یافته است[۲].

بنا به دلایل مطرح شده و نیز اهمیت رده‌بندی درست جریان‌های ترافیک، رویکردهای جدیدی بر مبنای نظارت بر بسته‌ها، تکنیک‌های آماری، روش‌های مبتنی بر رفتار و نیز اعمال تکنیک‌های یادگیری ماشین جهت رده‌بندی مبتنی بر ویژگی‌های آماری جریان، مورد بررسی قرار گرفته‌اند[۲].

از مهم‌ترین کاربردهای رده‌بندی می‌توان به سیستم‌های تشخیص نفوذ و انواع حملات اشاره نمود. ترافیک‌های موجود در هر شبکه می‌توانند با دو نوع هدف ایجاد شده باشند؛ اهداف مخرب و یا اهداف سالم و کاربردهای عادی. رده‌بندی جریان‌های ترافیک با توجه به این اهداف و نوع کاربرد

[1] Port-based prediction
[2] Payload-based Inspection
[3] Dynamic ports
[4] Encrypted applications

جریان‌های تولیدی، صورت می‌گیرد. در چند سال گذشته، اهداف و انگیزه‌های تهدیدات بدافزارها به طور قابل ملاحظه‌ای تغییر یافته وتلاش‌های انجام شده در جهت سازماندهی بهتر و سودمحوری بیشتر صورت گرفته است؛ این مساله سبب به وجود آمدن گونه‌ی جدیدی از تهدیدات شده است که زیرساخت هزاران شبکه در سراسر دنیا را در معرض خطر قرار می‌دهد. در مرکز این تهدیدات شبکه‌ای از میزبان‌ها قرار دارد که توسط یک مهاجم یا همان بات مرکزی[۵] کنترل می‌شود. این میزبان‌ها یک بات‌نت را تشکیل می‌دهند [۳].

بات، رایانه‌ی آلوده شده به یک بدافزار است که بدون آگاهی و اراده‌ی کاربر و از راه دور توسط یک یا چند عامل انسانی کنترل می‌شود. به این عامل کنترل کننده، بات مرکزی یا مدیر بات گفته می‌شود [۴]. کنترل بات‌ها توسط بات مرکزی و ایجاد ارتباط با بات‌ها از طریق کانال‌های فرمان و کنترل[۶]، با هدف انجام فعالیت‌های مخرب، شکل می‌گیرد که این کانال‌ها بر روی پروتکل‌های ارتباطی مختلف نظیر IRC،HTTP و P2P عمل می‌کنند. در حقیقت، بات مرکزی با در اختیار گرفتن کنترل رایانه‌های آلوده، به صورت توزیع شده از توان پردازشی آن‌ها در جهت اهداف تهاجمی خود استفاده می‌نماید.

در چند سال اخیر، تعداد بات‌نت‌ها به طور قابل توجهی افزایش یافته و تبدیل به یکی از بزرگترین تهدیدات بدافزاری شده است[۳]؛ از جمله حملاتی که توسط بات‌نت‌ها صورت می‌گیرد، حمله‌های جلوگیری از سرویس‌دهی توزیعی[۷]، ارسال هرزنامه‌ها و توسعه بدافزارها[۸]، سرقت اطلاعات محرمانه[۹]، دزدی هویت[۱۰] و کلاه‌برداری با کلیک کردن[۱۱] می‌باشد [۵].

[5] Botmaster
[6] Command and control channels (C&C channels)
[7] Distributed denial of service
[8] Spamming and Spreading Malware
[9] Information Leakage
[10] Identity fraud
[11] Click fraud

بنابراین به منظور افزایش امنیت شبکه‌ها لازم است اقدامات جدی در جهت شناسایی بات‌نت‌ها صورت گیرد و روش‌ها و تکنیک‌هایی با کارایی و دقت مناسب در این زمینه مطرح شده و مورد استفاده قرار گیرد تا از نفوذ و پیشروی هرچه بیشتر بات‌نت‌ها در سطح شبکه‌های اینترنتی جلوگیری گردد. از آنجایی که هر بات مرکزی می‌تواند در هر زمان صدها و حتی ده‌ها هزار رایانه متصل به اینترنت را برای اقدامات تهاجمی خود به کار گیرد، پهنای باند تجمعی وسیع و تعداد زیاد منابع حمله، حملات ناشی از بات‌نت‌ها را بسیار خطرناک، و مقابله و دفاع در برابر این حملات را بسیار مشکل نموده است [۶].

بات‌نت‌ها دارای ترکیبی از خواص سایر انواع حملات می‌باشند. آن‌ها همچون ویروس[۱۲]ها قابلیت پنهان کردن خود را دارند، از طرفی همانند کرم[۱۳]ها از قابلیت انتشار برخوردارند. به علاوه، کاربرانی که رایانه‌ی آن‌ها عضو یک بات‌نت می‌شود از این موضوع بی اطلاع می‌مانند و بات مرکزی بدون اطلاع صاحبان رایانه‌ها، آن‌ها را در اقدامات سودجویانه و تهاجمی خود شرکت می‌دهند. بنابراین توسعه‌ی روش‌های شناسایی بات‌نت‌ها از حساسیت بالاتری برخودار می‌گردد.

تاکنون رویکردهای مختلفی برای تشخیص بات‌نت‌ها پیشنهاد شده است[۷]. در سال‌های اخیر رویکردهای مبتنی بر یادگیری ماشین از لحاظ کارایی به عنوان موثرترین روش‌ها شناخته شده است و تلاش‌های اغلب محققان به این راستا سوق داده شده است. در این پژوهش نیز روشی مبتنی بر یادگیری ماشین جهت شناسایی بات‌نت‌ها، پیشنهاد و پیاده‌سازی شده است. در واقع در این پایان‌نامه، جریان‌های ترافیک مورد بررسی قرار گرفته و بر حسب این که این جریان‌ها اهداف مخرب

[12] Virus
[13] Worm

داشته و ویژگی‌هایی مشابه با ویژگی‌های بات‌نت‌ها داشته‌اند و یا این که سالم هستند، به دو دسته سالم یا بات‌نت دسته‌بندی می‌گردند.

با وجود اینکه بیش از یک دهه از به کارگیری روش‌های مبتنی بر یادگیری ماشین در این زمینه می‌گذرد، همچنان تلاش برای رسیدن به کارایی و دقت بالاتر و در عین حال هزینه کمتر ادامه دارد. مساله رده‌بندی دقیق ترافیک شبکه به ویژه برای سیستم‌های تشخیص نفوذ و حمله، بر اساس ویژگی‌های آماری جریان، هنوز به عنوان مساله‌ای حل نشده مطرح می‌گردد. رده‌بندی بر این اساس می‌تواند با استفاده از الگوریتم‌های دسته‌بندی با ناظر و یا بدون ناظر و یا ترکیبی از آن‌ها انجام گیرد. در این پایان‌نامه تلاش در جهت افزایش نرخ تشخیص، تعمیم‌پذیری، اعتبار ارزیابی سیستم می‌باشد.

۱-۱- شرح مسأله

جریان‌های ترافیک درون و میان شبکه‌ها با اهداف متفاوتی تولید می‌شوند. این اهداف می‌توانند در راستای کاربردهای معمولی و سالم باشندو یا مخرب بوده و جهت حمله به شبکه‌ها ایجاد گردند. یکی از مهم‌ترین اهداف تحلیل ترافیک، رده‌بندی جریان‌ها بر اساس نوع این اهداف و کاربردها می‌باشد تا بتوان نفوذ و حملات مختلف اینترنتی به شبکه را شناسایی نمود.

در میان انواع بدافزارهای[۱۴] موجود در مقیاس اینترنت، بات‌نت‌ها به عنوان یکی از جدیدترین نوع بدافزارها، بیش‌ترین تهدیدات را برای سامانه‌های اینترنتی به وجود آورده‌اند [۴]. تا به امروز روش‌ها و تکنیک‌های متعددی جهت طراحی سیستم‌های تشخیص نفوذ بات‌نت‌ها پیشنهاد شده است که در فصل دوم به برخی از آن‌ها اشاره شده است. این روش‌ها مبتنی بر رده‌بندی جریان‌هاست و بر اساس اینکه کاربرد جریان‌ها سالم و یا انجام فعالیت در بات‌نت بوده است، عمل می‌کنند.

[14] Malware

به طور کلی می‌توان چالش‌های اساسی در طراحی سیستم‌های تشخیص بات‌نت را در سه مورد خلاصه کرد. اولین مساله، نحوه‌ی ارزیابی سیستم است که باید با مجموعه داده‌ی جامعی انجام شود که هم شامل ترافیک عادی باشد و هم ترافیک‌های ناسالم تولید شده توسط انواع مختلفی از بات‌نت‌ها را پوشش دهد[۸]. به علاوه، داده‌های آموزش و داده‌هایی که جهت ارزیابی مورد استفاده قرار می‌گیرند تا حد امکان نباید با یکدیگر هم‌پوشانی داشته باشند. بدون وجود چنین مجموعه داده‌ای، میزان نرخ تشخیص اندازه گیری شده نمی‌تواند به کارایی سیستم در دنیای واقعی تعمیم داده شود [۹].

چالش دوم، تعمیم‌پذیری برای شناخت حملات جدید می‌باشد. نظر به رشد روزافزون انواع جدید بات‌نت‌ها، از سیستم‌های تشخیص انتظار می‌رود که تا حد زیادی ظرفیت شناسایی بات‌نت‌های جدید و دیده نشده را داشته باشند. البته رویکردهای مبتنی بر یادگیری ماشین، در این زمینه نسبت به روش‌های دیگر مانند روش‌های مبتنی بر قانون ، بهتر عمل می‌کنند؛ اما با این وجود هنوز نتوانسته‌اند انتظار محققین را به خوبی برآورده کنند. یکی از مهم‌ترین عواملی که در میزان تعمیم‌پذیری سیستم تاثیر به سزایی دارد، ویژگی‌هایی است که در این سیستم‌ها به عنوان مشخصه‌های تعیین کننده‌ی وجود یک بات‌نت مورد استفاده قرار می‌گیرند [۹] و [۱۰].

مساله‌ی سوم، امکان آموزش مجدد سیستم و به روزرسانی آن و درواقع استفاده از داده‌های آزمون به منظور بالا بردن کارایی و قدرت تشخیص سیستم می‌باشد، که این امر منجر به تطابق سیستم با ماهیت پویای اینترنت در دنیای واقعی می‌گردد[۱۱]. یک سیستم تشخیص هوشمند نباید به یک مجموعه‌ی آموزشی ایستا و ثابت محدود باشد، بلکه لازم است مجموعه‌ی آموزشی خود را به صورت مداوم به روزرسانی کند و هر بار مدل رده‌بند یا خوشه‌بند خود را مجدداً تشکیل دهد تا بتواند در دنیای واقعی به سطح بالایی از دقت تشخیص و کارایی برسد.

اگر یک سیستم برای شناسایی باتنت‌هایی که قبلا مشابه آن‌ها در مجموعه‌ی آموزشی دیده شده، نرخ تشخیص بسیار بالایی داشته باشد اما قادر نباشد از نمونه‌های جدیدی که برای ارزیابی وارد سیستم شده‌اند، در مراحل بعد در جهت بالا بردن قدرت تشخیص بهره ببرد، در دنیای واقعی دارای ارزش عملیاتی نمی‌باشد. با وجود اینکه این مساله از اهمیت بسیار بالایی برخوردار است و تا حدی دو چالش دیگر را نیز تحت تاثیر قرار می‌دهد، کمتر مورد توجه محققان قرار گرفته است.

۱-۲- اهمیت انجام پژوهش

در سال‌های اخیر، گسترش روزافزون باتنت‌ها و نیز افزایش توانایی آنان در مقابله با سیستم‌های شناسایی کننده، محققان را به تلاش برای طراحی و به کارگیری روش‌های هوشمند با قابلیت تشخیص بیشتر واداشته است.

حملات مختلفی می‌تواند توسط باتنت‌ها صورت گیرد. احتمال استفاده از باتنت‌ها برای انگیزه‌های جنایی و یا اهداف مخرب، همانطور که در فصل ۱ به آن اشاره شد، می‌توانند در پنج گروه شامل حمله‌های جلوگیری از سرویس‌دهی توزیعی، ارسال هرزنامه‌ها و توسعه بدافزارها، سرقت اطلاعات محرمانه، دزدی هویت و کلاه‌برداری با کلیک کردن دسته‌بندی شوند[۵]. خسارات فراوان ناشی از این نوع حملات، اهمیت پرداختن به مساله‌ی تولید سیستم‌های تشخیص باتنت با کارایی و دقت بالا را نشان می‌دهند.

امروزه باتنت‌ها به سرعت در حال رشد و تغییر در ساختار خود می‌باشند[۱۲] با توجه به تهدیدات گسترده‌ی باتنت‌ها، نیاز به تولید سیستم‌های تشخیص باتنت با تعمیم‌پذیری بالا، که علاوه بر نمونه‌های موجود باتنت قادر به تشخیص انواع جدید باتنت‌ها نیز باشد، به شدت احساس می‌شود.

از آنجایی که تا قبل از چند سال اخیر، مجموعه داده‌ی جامع و بزرگی از بات‌نت‌ها در اختیار پژوهشگران نبود تا از آن برای آموزش و ارزیابی سیستم‌های تشخیص خود استفاده نمایند، اغلب پژوهش‌های انجام شده سیستم خود را توسط مجموعه داده‌های کوچک که شامل انواع بسیار محدودی از بات‌نت‌ها بودند ارزیابی می‌کردند. این امر باعث شده است که علی رغم اینکه نتایج ارزیابی اعلام شده توسط این پژوهش‌ها دقت و نرخ تشخیص بسیار بالایی را نشان می‌دهند، این سیستم‌ها در دنیای واقعی دقت و کارایی بسیار کمتری داشته باشند.

حتی در سال‌های اخیر نیز کمتر پژوهشگرانی در پژوهش خود از یک مجموعه داده‌ی جامع استفاده نموده است. بنابراین لازم است سیستم‌های تشخیص بات‌نتی تولید شوندکه دقت گزارش شده توسط طراحان آن تا حد امکان مشابه به دقت عملکرد آن سیستم در دنیای واقعی باشد؛ که این امر با بکارگیری یک مجموعه داده‌ی جامع از بات‌نت‌ها در آموزش و ارزیابی سیستم، تا حد زیادی میسر خواهد شد.

۱-۳- هدف پژوهش

هدف از این پایان‌نامه تحلیل جریان‌های ترافیک و رده‌بندی آن‌ها بر اساس کاربرد می‌باشد؛ به این معنی که مشخصه‌های جریان‌ها بررسی شده و چنانچه این مشخصه‌ها بیانگر این موضوع باشد که جریانی با هدف مخرب ایجاد شده است، وجود یک حمله شناسایی و اعلام می‌گردد. در غیر این صورت کاربرد جریان‌ها معمولی و سالم تلقی می‌گردد.

با توجه به گستردگی نوع حملات، سیستم پیشنهادی، در جهت تشخیص یک نوع مشخص از حمله‌ها (بات‌نت‌ها) طراحی شده است. در این سیستم، به هر سه چالش بیان شده در بخش ۱-۲ رسیدگی شده و با در نظر گرفتن این مسائل، یک سیستم کارا جهت تشخیص وجود بات‌نت طراحی

گردیده است. به این معنی که علاوه بر استفاده از یک مجموعه داده‌ی جامع و بکارگیری ویژگی‌های موثر جهت شناسایی بات‌نت‌ها که تا حد زیادی مستقل از نوع بات‌نت می‌باشند و افزایش تعمیم‌پذیری سیستم را به دنبال دارد، آموزش سیستم نیز به صورت افزایشی انجام می‌گیرد.

این سیستم که بر اساس یک رده‌بند ترکیبی مبتنی بر k- نزدیک‌ترین همسایه [15]عمل می‌کند، زمانی که در محیط واقعی قرار می‌گیرد نمونه‌های آزمون را رده‌بندی کرده و بدون نیاز به اطلاع داشتن از برچسب واقعی این نمونه‌ها، از آن‌ها برای به روزرسانی مجموعه‌ی آموزشی خود بهره می‌برد. بدین ترتیب عملیات یادگیری به صورت افزایشی ادامه می‌یابد.

۱-۴- مروری بر فصل‌ها

در ادامه در فصل دوم، ابتدا تعاریف نظری و عملیاتی و به طور کلی مفاهیم مرتبط با موضوع پژوهش و سپس مروری بر برخی از پژوهش‌های مرتبطی که تاکنون در این زمینه انجام گرفته، ارائه می‌شود. در فصل سوم، روش پیشنهادی به طور کامل شرح داده شده و در فصل چهارم نتایج حاصل از آزمایش‌های انجام شده جهت ارزیابی این روش، گزارش می‌شود. در نهایت نیز در فصل پنجم، نتیجه‌گیری به عمل می‌آید.

[15] K-Nearest Neighbor (KNN)

فصل دوم

ادبیات پژوهش

۲-۱- رده بندی ترافیک و کاربرد

رده‌بندی جریان‌های ترافیک در شبکه‌های کامپیوتری کاربردهای امنیتی، کنترلی و مدیریتی مختلفی دارد و به طور کلی بر طبق ویژگی‌های جریان‌ها، محتویات بسته‌ها و الگوریتم‌های یادگیری ماشین انجام می‌شود.

رده‌بندی ترافیک به هنگام، برای سیستم‌های تشخیص نفوذ به عنوان ورودی محسوب می‌شود. هم‌چنین، برای نظارت بر شبکه اطلاعات آماری لازم را فراهم می‌کند. انواع متفاوتی از کاربردهای شبکه در اینترنت وجود دارد که دارای مشخصه‌های آماری مختلفی می‌باشند. بنابراین، به دلیل وجود این سطح از تنوع در مشخصه‌های آماری، معمولا به منظور رده‌بندی ترافیک، از رده‌بندی‌های آماری بهره برده می‌شود [۱۳].

از کاربردهای امنیتی رده‌بندی، رده‌بندی بر اساس کاربرد جریان‌ها جهت تشخیص حملات اینترنتی می‌باشد. بسته به اهداف میزبان‌ها، جریان‌های ورودی و خروجی آن‌ها دارای مشخصه‌هایی می‌باشند. در رده‌بندی بر اساس کاربرد، طبق همین مشخصه‌ها، جریان‌ها به دو دسته سالم و حمله تقسیم می‌گردند.

۲-۲- بات‌نت

بات‌نت‌ها شبکه‌هایی از رایانه‌های هماهنگی هستند که تحت کنترل مهاجمان (بات‌های مرکزی) قرار گرفته‌اند . عملکرد بات‌های موجود در یک بات‌نت به صورت مستقل از هم نیست بلکه

هر باتنت برای دریافت و پاسخ‌دهی به فرمان‌های بات مرکزی یک زیرساخت ارتباطی به نام کانال فرمان و کنترل [16] دارد [۱۴]. باتنت که دارای خواص مشترکی با ویروس‌ها، کرم‌ها و تروجان[17]ها می‌باشد، ابزار مخرب پیچیده‌ای است و در دنیای اینترنتی امروز، یکی از مهم‌ترین مشکلات به شمار می‌رود [۱۵].

اندازه یک باتنت را پیچیدگی و نیز تعداد رایانه‌هایی مشخص می‌کنند که بدون اطلاع صاحبان آن‌ها، تحت تصرف و کنترل باتنت قرار گرفته‌اند. علت محبوبیت باتنت‌ها میان مجرمان اینترنتی، قابلیت برنامه‌ریزی و تنظیم مجدد آن‌ها به منظور مقابله باروش‌های مختلف امنیتی و یا به دلیل ایجاد حملات جدید می‌باشد [۱۴].

۲-۲-۱- انواع باتنت

باتنت‌ها را می‌توان بر اساس دو معیار مرتبط با کانال‌های فرمان و کنترل، شامل ساختار و پروتکل طبقه‌بندی نمود. بر اساس ساختار کانال کنترل و فرمان، به سه دسته‌ی متمرکز و غیرمتمرکز و ترکیبی و بر اساس پروتکل مورد استفاده در این کانال‌ها، به سه نوع مبتنی بر IRC[18]، مبتنی بر HTTP و نظیر به نظیر[19] دسته‌بندی می‌شوند[۱۴و۱۶]. در ادامه مختصری در مورد برخی ازاین باتنت‌ها شرح داده می‌شود.

۱) باتنت متمرکز

[16] Command and control channel (C&C channel)
[17] Trojan Horse
[18] Internet relay chat
[19] Peer-to-Peer

ساختار متمرکز مبتنی بر مدل مشتری- سرویس‌دهنده[20] است، طوری که همه‌ی بات‌ها به طور مستقیم به یک یا تعداد کمی از سرویس‌دهنده‌های کنترل و فرمان متصل هستند. این سرویس‌دهنده‌ها بات‌ها را با یکدیگر هماهنگ کرده و هم‌چنین به بات‌ها برای انجام عملیات فرمان می‌دهند[۱۶].

اگرچه طراحی این نوع ساختار نسبت به ساختارهای غیرمتمرکز ساده‌تر است و هماهنگی و انتشار سریع فرمان‌ها در آن‌ها به صورت بهینه‌تر صورت می‌گیرد، اما شناسایی این نوع بات‌نت‌ها سریع‌تر صورت می‌گیرد و چنانچه یکی از سرویس‌دهنده‌ها شناسایی شده و از کار انداخته شود، مجموعه‌ی بات‌نت به سادگی از بین خواهد رفت [۱۶]. در کانال‌های ارتباطی این نوع بات‌نت‌ها معمولا از پروتکل‌های IRC یا HTTP استفاده می‌شود [۱۷].

۲) بات‌نت مبتنی بر IRC

اولین بات‌نت‌ها در سال ۱۹۹۹ مبتنی بر پروتکل IRC ظاهر شدند[۱۸]. هر سرویس‌دهنده‌ی IRC کانال‌های متنوعی را میزبانی می‌کند. در این نوع از بات‌نت‌ها، کانال کنترل و فرمان در سرویس دهنده‌ی IRC ایجاد می‌شود و بات‌ها عضو این کانال می‌گردند[۱۴].

۳) بات‌نت غیرمتمرکز

در این نوع معماری، یک سرویس دهنده‌ی کنترل و فرمان متمرکز وجود ندارد، بلکه در آن بات‌نت‌هابی مختلف از طریق پروتکل‌های نظیر به نظیر با یکدیگر در ارتباط هستند. به عبارت دیگر، بات‌ها هم‌زمان هم به عنوان مشتری و هم سرویس‌دهنده کنترل و فرمان عمل می‌کنند[۱۶]. بنابراین

[20] Client-server

چنانچه یکی از بات‌ها شناسایی و غیرفعال گردد، لزوماً منجر به شناسایی و از بین رفتن سایر بات‌ها نخواهد شد. این نوع ساختار، اگرچه در مقابله در برابر شناسایی از قابلیت بیشتری نسبت به ساختار متمرکز برخوردار است، اما طراحی و پیاده‌سازی آن نیز دارای پیچیدگی بیشتری است[۱۹].

۴) بات‌نت نظیر به نظیر

همانطور که پیش‌تر بیان شد، در بات‌نت‌های با ساختار غیرمتمرکز اغلب از پروتکل‌های ارتباطی نظیر به نظیر استفاده می‌گردد. همانند شبکه‌های نظیر به نظیر که نسبت به تغییرات پویا انعطاف‌پذیرند، ارتباطات بات‌نت نظیر به نظیر نیز با از دست دادن تعدادی از بات‌ها مختل نخواهد شد.

در یک بات‌نت نظیر به نظیر هیچ سرویس‌دهنده‌ی مرکزی وجود ندارد و همه‌ی بات‌ها به یکدیگر متصل هستند و به عنوان مشتری و سرویس‌دهنده‌ی کنترل و فرمان عمل می‌کنند. بات‌نت‌های نظیر به نظیر در مقایسه با بات‌نت‌های متمرکز مزایای بیشتری از خود نشان می‌دهند و دفاع در برابر آن‌ها برای جامعه‌ی امنیتی دشوارتر است [۱۹].

۵) بات‌نت ترکیبی

معماری ترکیبی بات‌نت‌ها از مزایای هر دو نوع معماری متمرکز و غیرمتمرکز بهره می‌برد. به این معنی که در این ساختار، بات‌ها عملکردهای متفاوتی را از خود نشان می‌دهند. برخی از آن‌ها به صورت موقت نقش سرویس‌دهنده‌ی کنترل و فرمان را به عهده می‌گیرند و عمل هماهنگ‌سازی بات‌نت و نیز انتشار فرمان‌ها را انجام می‌دهند، در حالی که سایر بات‌ها منتظر فرمان می‌مانند[۲۰].

۲-۲-۲- چرخه حیات بات‌نت

چرخه حیات باتنت‌ها را می‌توان به سه مرحله اصلی شکل‌گیری، فرمان و کنترل وحمله تقسیم کرد. در مرحله اول بات مرکزی بات‌های خود را از طرق مختلف پراکنده می‌سازد تا کد دودویی مخرب خود را بر روی رایانه‌های آسیب‌پذیر زیادی نصب کند و در نتیجه آن‌ها نیز عضو باتنت گردند. در مرحله فرمان و کنترل، کانال ارتباطی کنترل و فرمان بین بات مرکزی و سایر بات‌ها ایجاد می‌شود. و در مرحله سوم، بات‌ها فرمان‌هایی جهت انجام فعالیت‌های بدخواهانه دریافت و آن‌ها را اجرا می‌کنند[۱۴].

اما این مراحل، خود شامل مراحل و عملیات بیشتری هستند که در برخی مطالعات اخیر به صورت دقیق‌تر بیان شده‌اند [۲۱]. در مرجع [۲۲] چرخه‌ی حیات یک باتنت معمول را در پنج مرحله شرح می‌دهد. این مراحل شامل سرایت آلودگی اولیه، تزریق ثانویه، اتصال، کنترل و فرمان مخرب و به روزرسانی و نگه‌داری می‌باشد. در ادامه هریک از این مراحل شرح داده می‌شود.

۱) **سرایت آلودگی اولیه:** در این مرحله، مهاجم، یک زیرشبکه‌ی آسیب‌پذیر هدف را جستجو کرده، و این ماشین قربانی را از طرق مختلف آلوده می‌کند[۲۲].

۲) **تزریق ثانویه:** بعد از سرایت آلودگی اولیه، در مرحله تزریق ثانویه، میزبان‌های آلوده شده یک پوسته کد۲۱ را اجرا می‌کنند. پوسته کد، تصویر دودویی بات واقعی را از موقعیت مشخص به وسیله FTP، HTTP و یا P2P واکشی می‌کند. کد دودویی بات خود را روی ماشین هدف راه‌اندازی می‌کند. زمانی که برنامه‌ی بات راه‌اندازی شود رایانه‌ی قربانی به یک زامبی ۲۲ تبدیل شده و کد مخرب را اجرا می‌کند [۲۲و۲۳].

[21] Shell-code
[22] Zombie

۳) **اتصال:** در مرحله‌ی اتصال، برنامه‌ی بات یک کانال کنترل و فرمان ایجاد می‌کند و رایانه آلوده شده را به سرویس‌دهنده‌ی کنترل و فرمان[۲۳] متصل می‌کند. با ایجاد کانال کنترل و فرمان، رایانه آلوده شده نیز عضوی از ارتش بات‌نت مهاجم می‌گردد[۲۲].

۴) **کنترل و فرمان مخرب:** پس از مرحله‌ی اتصال، فعالیت‌های کنترل و فرمان واقعی بات‌نت آغاز خواهد شد. بات مرکزی از این کانال برای انتشار فرمان‌ها در میان ارتش بات‌های خود استفاده می‌کند. برنامه‌ی بات فرمان‌های ارسال شده از جانب بات مرکزی را دریافت و اجرا می‌کند. کانال کنترل و فرمان، بات مرکزی را قادر می‌کند تا عملکرد بات‌های خود را از راه دور کنترل کرده تا انجام فعالیت‌های مختلف غیرقانونی را هدایت کند [۲۲].

۵) **به‌روزرسانی و نگهداری:** آخرین مرحله، نگهداری بات‌ها به صورت زنده و به روز می‌باشد. در این مرحله به بات‌ها فرمان داده می‌شود که یک دودویی به روزرسانی شده را دانلود کنند. کنترل کننده‌های بات ممکن است به چندین دلیل به به‌روزرسانی کردن بات‌های خود نیاز داشته باشند. برای مثال، ممکن است برای فرار کردن از تکنیک‌های شناسایی و یا اینکه مایل باشند کارکردهای بیشتری را به بات‌های خود اضافه کنند. هم‌چنین گاهی دودویی به روز رسانی شده، بات‌ها را به یک سرویس‌دهنده‌ی کنترل و فرمان متفاوت هدایت می‌کند. این عمل مهاجرت سرویس‌دهنده نامیده می‌شود و برای زنده نگه داشتن بات‌نت توسط بات مرکزی بسیار مفید است [۲۲].

۲-۲-۳ روش‌های تشخیص بات‌نت

[23] Command and control server

روش‌های تشخیص باتنت را می‌توان بر اساس سه معیار گروه‌بندی کرد. معیار اول، موقعیت باتنت در چرخه حیات به هنگام تشخیص است. معیار دوم و سوم به ترتیب، رویکرد یادگیری و سطح تحلیل همبستگی می‌باشد [۱۴].

بر اساس معیار اول، تشخیص می‌تواند در مراحل آغازین، یعنی در زمان شکل‌گیری باتنت و ایجاد کانال کنترل و فرمان و یا در مرحله حمله‌ی باتنت، صورت گیرد. به طور کلی، دقت روش‌هایی که در مرحله حمله شناسایی را انجام می‌دهند بالاتر است [۱۴]. اما از طرفی، تشخیص باتنت‌ها در مراحل آغازین، از مشارکت آن‌ها در فعالیت‌ها و حمله‌های مخرب جلوگیری می‌کند.

بر اساس معیار دوم، رویکرد یادگیری می‌تواند نظارتی یا غیرنظارتی باشد [۲۴]. معمولا در یادگیری برخط، به دلیل محدودیت نیاز به داده‌های برچسب‌گذاری شده، از روش‌های نظارتی استفاده نمی‌شود. بلکه از روش‌های غیر نظارتی استفاده می‌شود، زیرا این روش‌ها نیاز به اطلاعات قبلی از باتنت‌ها و داده‌های برچسب‌گذاری شده ندارند. برای به کارگیری روش‌های نظارتی، لازم است به طریقی این محدودیت برطرف شود.

در نهایت، بر اساس معیار سوم روش‌های تشخیص باتنت می توانند براساس دو سطح مختلف از تحلیل همبستگی(سطح انفرادی و سطح گروهی)عمل کنند. در تحلیل سطح انفرادی، شناسایی بر اساس رفتارهای فردی هر سیستم صورت می‌گیرد و رفتار سایر سیستم‌های آلوده در نظر گرفته نمی‌شود. مزیت این روش‌ها در این است که اگر در شبکه مورد نظر، تنها یک بات وجود داشته باشد، آن را تشخیص دهند [۱۴].

روش‌هایی که تحلیل در سطح گروهی انجام می‌شود، به این صورت عمل می‌کنند که چنانچه الگوی رفتاری مشابهی میان دو یا چند سیستم مشاهده کنند، آن‌ها را به عنوان اعضای باتنت

تشخیص می‌دهند. دقت عمل این نوع روش‌ها بیشتر از روش‌هایی است که تنها رفتار فردی سیستم مورد توجه قرار می‌گیرد.اما از طرفی، این روش‌ها فقط بات‌هایی را در شبکه می‌توانند تشخیص دهند که عضو یک بات‌نت مشترک هستند[۱۴].

روش تشخیص بات‌نت ارائه شده در این پژوهش بر اساس سه معیار مطرح شده، از نوع نظارتی، تحلیل سطح انفرادی و تشخیص در مرحله آغازین می‌باشد.

۲-۳- بررسی پژوهش‌های انجام شده

تاکنون رویکردهای متعددی جهت شناسایی انواع مختلف بات‌نت‌ها ارائه شده است. تعداد زیادی از این رویکردها در مراجع [۷]، [۲۵] و [۲۶] به تفصیل بیان شده است. یکی از چالش‌های اساسی در سیستم‌های تشخیص بات‌نت‌ها، همچون سایر سیستم‌های تشخیص نفوذ[۲۴] توانایی تشخیص بات‌نت‌های نوع جدید و تاکنون دیده نشده، می‌باشد. از آنجایی که یادگیری ماشین قادر به توسعه‌ی الگوریتم‌ها و تکنیک‌های کاراتری برای رویارویی و تشخیص درست بات‌نت‌های جدید بوده است، اخیراً از میان رویکردهای مختلف شناسایی، روش‌های مبتنی بر یادگیری ماشین بیشتر مورد توجه محققین قرار گرفته است.

مطالعات ارائه شده در این زمینه، از الگوریتم‌های یادگیری بدون ناظر، نظیر پژوهش‌های [۲۷]، [۲۸] و [۲۹] و نیز الگوریتم‌های یادگیری با ناظر، هم‌چون پژوهش‌های [۳۰]، [۳۱] و [۳۲]، جهت رده‌بندی یا خوشه‌بندی ترافیک شبکه استفاده نموده‌اند؛ تحلیل ترافیک شبکه می‌تواند در سطح

[24] Intrusion detection systems (IDS)

جریان و یا بسته‌ها صورت پذیرد، که به دلیل ظرفیت بالاتر تشخیص با بررسی در سطح جریان، در مطالعات اخیر، اغلب در این سطح انجام شده‌اند.

با وجود این که نرخ تشخیص گزارش شده توسط اغلب این مطالعات در سطح بالایی (بالاتر از ۹۰٪) قرار دارند، اما با به‌کارگیری آن‌ها در دنیای واقعی، نیاز به بهبود بیشتر در این تکنیک‌ها هم‌چنان احساس می‌شود[۹]. از طرفی دقت اعلام شده‌ی مربوط به یک روش، باید تحت شرایط قابل قبولی ارزیابی شود.

از جمله‌ی این شرایط، استفاده از مجموعه داده‌های آزمونی است که دارای انواع مختلفی از بات‌نت‌ها باشد. به عبارتی، چنانچه از مجموعه داده‌ای جهت آموزش یک سیستم استفاده شود که دارای تعداد محدودی از انواع بات‌نت‌ها باشد و در میان داده‌های آزمون نیز نوع جدید و دیده نشده‌ای از بات‌نت وجود نداشته باشد، نمی‌توان از آن سیستم در دنیای واقعی انتظار چنین نرخ تشخیص بالایی را داشت. از آنجایی که فراهم نمودن چنین مجموعه داده‌ی جامعی، به خاطر مشکلاتی که در مراجع [۳۳] و [۳۴] به خوبی به آن‌ها اشاره شده است کار دشواری است، اغلب مطالعات انجام شده در این زمینه مجموعه داده‌هایی را به کار گرفته‌اند که دارای تنوع کمی بوده و برای ارزیابی قابل قبول عملکرد سیستم‌ها مناسب نیستند.

در رابطه با دو چالش اول مطرح شده در بخش ۱-۱، در مطالعات اخیر اقداماتی انجام شده است[۳۵]. به عنوان نمونه، در پژوهش [۳۶]، جهت تشخیص بات‌نت یک روش تحلیل ایستا به نام DeDroid ارائه شده و از مجموعه داده Drebin استفاده شده است که نسبته به سایر پایگاه‌های داده تا آن زمان، از جامعیت نسبتا بالاتری برخوردار بوده است [۹]و [۳۶]. یکی از جدیدترین مطالعات صورت گرفته در زمینه‌ی تشخیص بات‌نت از طریق تحلیل ترافیک، که نرخ تشخیص بالایی (۹۹٪) در آن گزارش شده است روش ارائه شده در مرجع[۳۷] می‌باشد، این روش، از ویژگی‌های مبتنی بر

جریان و روش یادگیری درخت تصمیم C4.5 بهره برده است. اما همانطور که کارایی این روش در مطالعه‌ی[۹] مورد بحث و بررسی قرار گرفته است، باید بیان نمود که در مرجع [۳۷] مجموعه داده‌ی جامعی جهت ارزیابی به کار گرفته نشده است و همچنین در بین داده‌های آموزش و آزمون آن هم‌پوشانی وجود دارد.

مطالعه‌ی [۹]، یک مجموعه داده‌ی جامع[۲۵] ، که نسبت به سایر مجموعه داده‌های موجود از تنوع بسیار بالاتری برخوردار است را تولید نمود و به سیستم معرفی شده در[۳۷] اعمال کرد. مشاهده شد که نرخ تشخیص در این حالت به شدت کاهش یافت(۶۸٪). بنابراین اهمیت و تاثیر مجموعه داده‌ی استفاده شده بر روی ارزیابی به خوبی آشکار می‌شود.

در پژوهش [۹]در جهت افزایش قدرت تشخیص سیستم اقداماتی صورت گرفت. برای این منظور ویژگی‌هایی جهت شناسایی انتخاب شدند که وابستگی زیادی به انواع محدود و مشخصی از بات‌نت‌ها نداشته باشند و تا حد زیادی قابل تعمیم به انواع بات‌نت‌ها باشند. با بکارگیری این ویژگی‌ها در سیستم، نرخ تشخیص سیستم مورد نظر تا حد قابل توجهی (۷۵٪) افزایش یافت. لازم به ذکر است که نرخ تشخیص در [۹] را نمی‌توان با اغلب سیستم‌های تشخیص بات‌نت دیگر مقایسه نمود؛ زیرا اگرچه ممکن است مطالعات دیگر نظیر[۳۷]و [۳۸]. به دقت بالایی رسیده باشند، اما نتایج آنها هنگامی می‌تواند با نتایج [۹] مقایسه شود که توسط یک مجموعه داده‌ی جامعی نظیر آنچه در [۹] استفاده شد، مورد ارزیابی قرار گرفته باشند.

مطالعه [۹] اگرچه در جهت رسیدگی به چالش اول و دوم اقدام کرده است و نسبت به سایر مطالعات موجود، به نتایج بسیار بهتر و معتبرتری رسیده است، اما به مساله‌ی سوم، یعنی آموزش در

[25] ISCX botnet data set

محیط پویا نپرداخته است. در زمینه‌ی رسیدگی به این مساله در مقالات دیگر تلاش‌هایی صورت گرفته است که اغلب این مطالعات از خوشه‌بندی بدون ناظر استفاده نموده‌اند [۱۱]، [۳۹] و [۴۰].

در این پژوهش تلاش شده است که با در نظر گرفتن تمامی چالش‌های مطرح شده، یک رده‌بند با ناظر برای تشخیص بات‌نت‌ها طراحی شود. از آنجایی که مطالعات اخیر[۹]، [۳۷]، [۳۲]و[۴۱] نشان داده است که ویژگی‌های مبتنی بر جریان در سیستم‌های تشخیص بات‌نت و نیز دسته‌بندی ترافیک کارایی بهتری را نسبت به سایر ویژگی‌ها به ارمغان می‌آورد، در این پژوهش نیز از این نوع ویژگی‌ها استفاده شده است.

یکی از مهم‌ترین مزایای این نوع ویژگی‌ها این است که توانایی رویارویی و تشخیص ترافیک‌های رمزگذاری شده را دارد. همچنین، به دلیل اینکه ویژگی‌های مبتنی بر جریان، محتوای درون بسته‌ها را مورد تحلیل قرار نمی‌دهند و فقط از قسمت سرآیند[۲۶] استفاده می‌نماید، به کارگیری این ویژگی‌ها، میزان نفوذ به حریم اطلاعات محرمانه‌ی بسته‌ها را کاهش می‌دهد، و همچنین هزینه‌ی محاسباتی کمتری به دنبال خواهند داشت.

[26] Header

فصل سوم

روش پیشنهادی برای تحلیل ترافیک جهت رده‌بندی کاربرد

۳-۱- طرح کلی سیستم تحلیل و رده‌بندی ترافیک شبکه

به منظور رسیدگی به مساله‌ی پویا بودن محیط اینترنت و رشد روزافزون تنوع بات‌نت‌ها و در نتیجه ضرورت وجود سیستم‌های تشخیص بات‌نتی که یادگیری و عملیات شناسایی آن‌ها به صورت پویا انجام می‌گیرد، در این پژوهش، یک سیستم تشخیص بات‌نت با یادگیری افزایشی طراحی شده است. در واقع این سیستم یک سیستم تشخیص خودفراگیر است که در آن از یادگیری ترکیبی استفاده می‌گردد. الگوریتم پایه در این سیستم، الگوریتم نظارتی K- نزدیک‌ترین همسایه[۲۷] میباشد. به دلیل ساختار سیستم، الگوریتم پایه‌ی مورد استفاده بهتر است غیرپارامتریک باشد.

همانطور که بیان شد، الگوریتم مورد استفاده در این سیستم، برخلاف اغلب روش‌های برخط موجود، از نوع نظارتی می‌باشد. اگرچه روش‌های غیرنظارتی نیاز به داده‌های برچسب‌گذاری ندارند و تشخیص برخط را تسهیل می‌کنند، اما دارای دشواری‌هایی می‌باشند؛ در این روش‌ها، پس از مرحله خوشه‌بندی، لازم است تعیین شود که هر خوشه متعلق به ترافیک‌های سالم است یا بات‌نت. با توجه به این که داده‌ای از قبل موجود نبوده، تشخیص این گروه‌ها نیاز به روش‌های پیچیده‌تری دارد. برای نمونه، در پژوهش [۳۹]، خوشه‌هایی که شباهت میان داده‌های آن از حد آستانه‌ای بیشتر باشند به عنوان خوشه بات‌نت‌ها تعیین می‌شوند و در غیر این صورت سالم هستند.

اما در روش‌های نظارتی این مرحله وجود ندارد و تنها با مقایسه داده‌های ورودی جدید با مجموعه داده برچسب‌گذاری شده‌ی موجود، می‌توان برچسب هر داده را تعیین کرد. اما از طرفی، روش‌های نظارتی نیاز به تعداد زیادی داده برچسب‌گذاری شده دارد و هم‌چنین از آن‌جایی که تشخیص، بر اساس مقایسه با تعداد محدودی صورت می‌گیرد، نسبت به روش‌های غیرنظارتی تعمیم‌پذیری کمتری دارد.

[27] K-Nearest Neighbor (KNN)

در این پایان‌نامه، با رفع محدودیت‌های ذکر شده در روش‌های نظارتی، یک سیستم برای تشخیص بات‌نت ارائه شده است که دارای قدرت تعمیم‌پذیری بوده و تنها با داشتن تعداد اندکی داده برچسب‌گذاری شده، می‌تواند بات‌نت‌ها را تشخیص دهد، زیرا سیستم پیشنهادی پس از هربار شناسایی جریان‌ها و اختصاص برچسب به آن‌ها، سعی می‌کند نمونه‌هایی را که به درستی برچسب اختصاص داده شده به آن‌ها اطمینان بیشتری دارد را در مجموعه آموزشی خود ذخیره کند. به این ترتیب دائما در حال به روزرسانی مجموعه آموزشی خود بر اساس نمونه‌های جدید می‌باشد. از سوی دیگر، میزان حافظه مصرفی و سربار ناشی از ذخیره نمونه‌های جدید را نیز در نظر گرفته و فقط نمونه‌های موثرتر را ذخیره می‌کند.

در ادامه ابتدا در یک نگاه اجمالی روند کلی سیستم پایه‌ی پیشنهادی بررسی می‌شود و سپس هریک از مراحل به صورت مجزا تشریح می‌گردد.

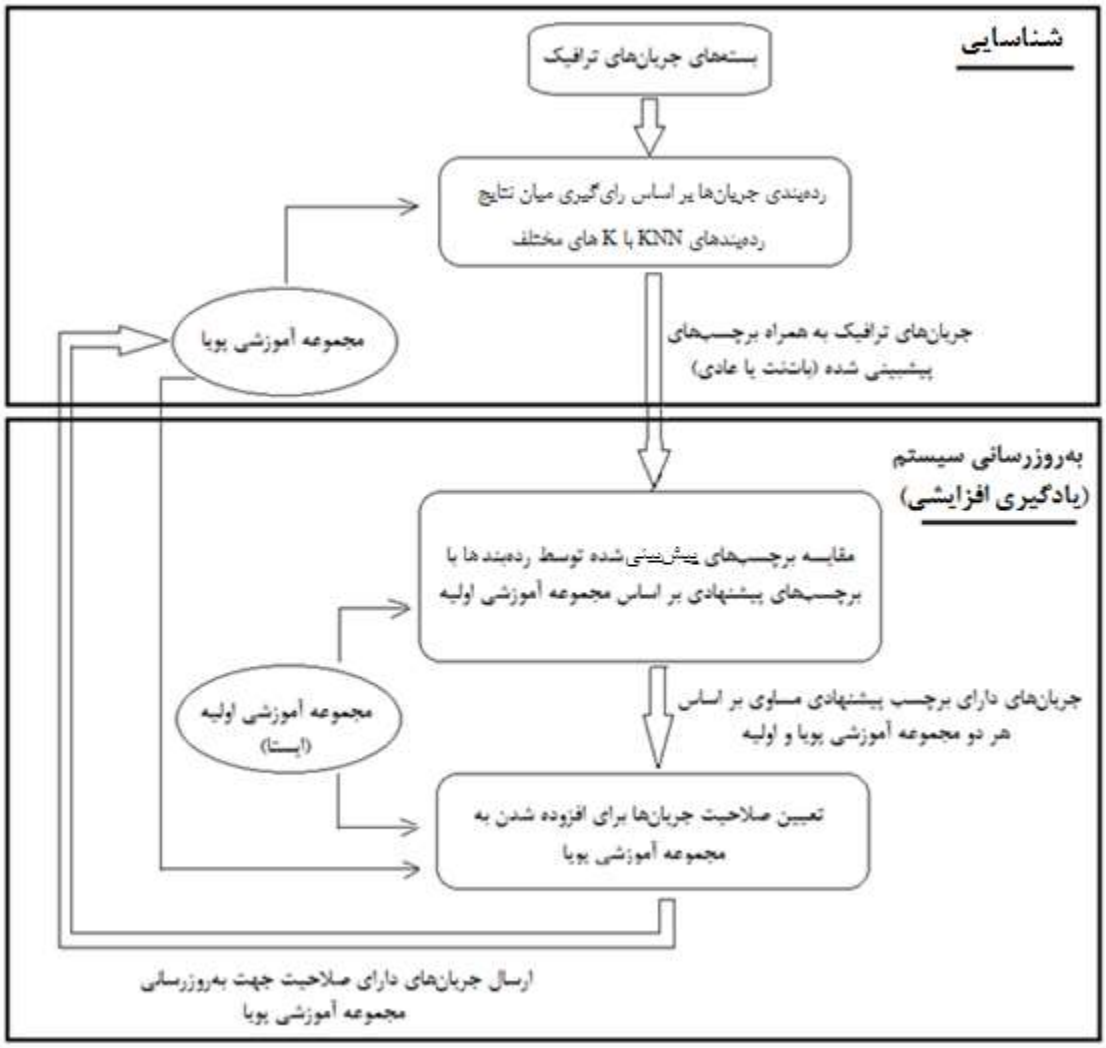
شکل ۳.۱ مراحل مختلف روش پیشنهادی را نشان می‌دهد. همانطور که مشاهده می‌شود این سیستم شامل دو بخش کلی می‌باشد؛ بخش شناسایی و تشخیص بات‌نت و بخش به روزرسانی سیستم که در آن یادگیری افزایشی به صورت مستمر صورت می‌گیرد.

در بخش اول هدف کلی سیستم، رده‌بندی ترافیک ورودی به دو دسته‌ی ترافیک عادی و معمول[۲۸] و ترافیک تولید شده توسط بات‌نت می‌باشد. قبل از راه‌اندازی و ورود به بخش اول، سیستم دارای یک مجموعه‌ی آموزشی اولیه ایستا می‌باشد که آموزش اولیه، بر مبنای الگوریتم پایه بر اساس این مجموعه صورت می‌گیرد. سپس بخش اول سیستم شروع به کار کرده و بر اساس مجموعه آموزشی پویا که در اولین اجرا، برابر با همان مجموعه اولیه ایستا است رده‌بندی جریان‌ها را انجام

[28] Normal traffic

می‌دهد. این رده‌بندی به کمک الگوریتم پایه، و بر اساس یادگیری ترکیبی است که شامل رای‌گیری میان خروجی‌های رده‌بندهای kNN با پارامترهای مختلف k می‌باشد.

در بخش دوم، مجموعه‌ی آموزشی پویا به تدریج به کمک ترافیک‌های ورودی جدید، با هدف ارتقای سطح تشخیص سیستم، به روزرسانی می‌گردد و پس از آن رده‌بندی سیستم بر اساس این مجموعه‌ی پویا انجام می‌گیرد.



شکل ۳-۱) مدل سیستم تشخیص باتنت پیشنهادی با قابلیت یادگیری افزایشی

به بیان دقیق‌تر، ابتدا تعدادی جریان ترافیک جهت شناسایی وارد رده‌بندها می‌شود. این سه رده‌بند همگی kNN هستند که تفاوت آن‌ها با یکدیگر در پارامتر k آن‌ها می‌باشد. این رده‌بندها براساس مجموعه آموزشی پویا (که در اولین تکرار برابر با همان مجموعه‌ی اولیه‌ی ایستا است)، برای هریک از این جریان‌ها یک برچسب پیش‌بینی می‌کنند که بیان کننده‌ی باتنت یا عادی بودن جریان می‌باشد. سپس از میان برچسب‌های پیشنهادی توسط هر یک از رده‌بندها، بر اساس رأی‌گیری،

برچسب محتمل‌تر به عنوان برچسب نهایی انتخاب می‌گردد. بنابراین در همین مرحله، شناسایی توسط سیستم صورت گرفته و قابل اعلام می‌باشد.

اما به منظور بهره‌برداری از ترافیک‌های جدید وارد شده برای ارتقا و به روزرسانی سیستم، بخش بعدی سیستم، یعنی به روزرسانی و یادگیری افزایشی آغاز به فعالیت می‌کند. در این بخش، ابتدا برچسب پیش‌بینی شده براساس مجموعه‌ی آموزشی پویا در بخش تشخیص باتنت با برچسبی که بر مبنای مجموعه‌ی آموزشی ایستای اولیه و بر اساس فاصله داده جدید از مراکز هریک از دسته‌ها (دسته‌ی باتنت‌ها وو دسته‌ی ترافیک عادی)، به این جریان‌ها نسبت داده می‌شود، مقایسه می‌گردد. جریان‌هایی که برچسب داده شده به آن‌ها در هر دو حالت یکسان باشد، انتخاب شده و به مرحله‌ی بعد منتقل می‌شوند.

در مرحله‌ی بعد که حساس‌ترین مرحله محسوب می‌شود، از میان جریان‌های منتقل شده، آن‌هایی که صلاحیت کافی برای پیوستن به مجموعه‌ی آموزشی پویا، جهت به‌روزرسانی و بالا بردن دقت رده‌بند را دارند انتخاب می‌گردند و به مجموعه آموزشی پویا افزوده می‌شوند. تعیین صلاحیت این جریان‌ها در طی دو زیرمرحله صورت می‌گیرد که نقشی اساسی در کارایی سیستم ایفا می‌کنند.

همان‌طور که مشاهده می‌شود سیستم ما می‌تواند از جریان‌های ترافیکی که به مرور جهت تشخیص وارد می‌شوند، جهت بالا بردن ظرفیت خود در تشخیص وجود باتنت‌ها استفاده نماید. اهمیت اضافه نمودن نمونه یا همان جریان جدید به مجموعه‌ی آموزشی در حین اجرای سیستم، زمانی آشکارتر می‌شود که باتنت‌های جدید وارد سیستم شوند. به این صورت که بعد از ساختن مجدد مدل‌های رده‌بندی بر اساس مجموعه‌ی آموزشی پویا که نمونه‌هایی از این نوع باتنت جدید به آن اضافه شده، میزان قدرت سیستم در شناسایی این نوع جدید از باتنت، نسبت به زمانی که برای

اولین بار این نمونه آزمایشی را مشاهده کرده بود، افزایش می‌یابد. بدین ترتیب یادگیری در این سیستم در زمان بهره‌برداری از سیستم نیز به صورت افزایشی ادامه می‌یابد.

البته ذخیره کردن نمونه‌های بیشتراز همان انواع بات‌نتی که در مجموعه داده موجود بوده نیز می‌تواند در افزایش دقت تشخیص بات‌نت‌های تکراری موثر باشد.

نکته‌ی حائز اهمیت در این سیستم که آن را از اغلب سیستم‌های مشابه متمایز می‌کند اینست که در این سیستم، بدون آگاهی داشتن از برچسب واقعی جریان‌های وارد شده، از آن‌ها در رده‌بندی با ناظر استفاده می‌نماید. به این معنی که سیستم ما برای ارتقای خود نیاز به داشتن جریان‌های با برچسب واقعی ندارد، و خود برچسب جریان‌ها را پیشبینی نموده و سپس از آن‌ها در رده‌بندی‌های بعدی استفاده می‌کند.

در سیستم‌های برخط موجود به دلیل محدودیت در اختیار داشتن تعداد زیاد داده‌های برچسب‌گذاری‌شده، اغلب از روش‌های نظارتی استفاده نشده است و روش‌های غیر نظارتی به کار گرفته شده‌اند که نیاز به داده برچسب‌گذاری شده ندارند. اما در سیستم ارائه شده در این پژوهش، با رفع این محدودیت، از یک روش تشخیص نظارتی استفاده شده است.

به طور کلی، از آنجایی که روش‌های غیرنظارتی نیاز به داده‌های برچسب‌گذاری نداشته و از طرفی به دلیل اینکه رفتارهای کلی بات‌نت‌ها را جهت تشخیص بات‌نت‌ها مورد توجه قرار می‌دهند و در نتیجه تعمیم‌پذیری بیشتری در جهت تشخیص بات‌نت‌های ناشناخته دارند، اغلب محققان در سیستم‌های تشخیص برخط از روش‌های غیرنظارتی بهره می‌گیرند [۱۴]. اما در این پژوهش، علاوه بر رفع محدودیت نیاز به داده‌های برچسب‌گذاری شده، با بهره‌گیری از ویژگی‌های کارا برای تشخیص و به ویژه با انجام یادگیری به طور افزایشی و استفاده از داده‌های جدید جهت ارتقای سیستم، قابلیت

تعمیم‌پذیری نیز تا حد قابل توجهی افزایش یافته است. الگوریتم مورد استفاده برای پیاده‌سازی این سیستم به ترتیب در شکل ۳-۲ مشاهده می‌گردد.

```
1: Init_TrainingSet ← Initialize training set with n data
2: Dynamic_TrainingSet ← Init_TrainingSet
3: Initialize 5 KNN_Models

4:  For each data packet do
5:      For each t ϵ incoming data do

        // detection
6:        Predicted label ←Vote among (kNNModel1.classify(t), kNNModel2.classify(t) ,
                    kNNModel3.classify(t), kNNModel4.classify(t), kNNModel5.classify(t))

        // update system (dynamic learning)
7:          if (predicted-label ==  suggested-label based on Init_TrainingSet) then
8:            if (qualification (t, predicted-label, Init_TrainingSet) == true) then
9:                      qualifiedSet← t
10:           end if
11:         end if
12:     end for
13:     System-Update (qualifiedSet)
14: end for
```

شکل ۳-۲) الگوریتم پیاده‌سازی سیستم تشخیص باتنت ارائه شده

ویژگی‌ها

ویژگی‌های استخراج شده در این پژوهش، ویژگی‌های مبتنی بر جریان هستند. از مهم‌ترین مزایای این نوع ویژگی‌ها توانایی تشخیص و رویارویی با ترافیک‌های رمزگذاری شده می‌باشد. همچنین، به دلیل عدم بررسی محتوای بسته‌ها و صرفاً استفاده از قسمت سرآیند، میزان نفوذ به حریم خصوصی و میزان هزینه‌ی محاسباتی کمتری وجود خواهد داشت.

هفت ویژگی نهایی که در این سیستم مورد استفاده قرار گرفته است، در جدول ۳-۱ مشاهده می‌شود. همه‌ی این ویژگی‌ها بجز آخرین ویژگی، مشخصه‌های آماری مستخرج از جریان‌ها می‌باشند. در این ویژگی‌ها، از اطلاعات موجود در سه لایه‌ی انتقال داده[۲۹]، شبکه[۳۰] و پیوند داده[۳۱] استفاده می‌شود. این ویژگی‌ها بر اساس محاسبه‌ی میزان همبستگی[۳۲]، و با انجام آزمایش‌های تجربی از میان ویژگی‌های پیشنهادی در پژوهش [۹]، انتخاب شده‌اند. برای این منظور ابتدا ۴ ویژگی‌ای که در مطالعه [۹] به عنوان بهترین ویژگی انتخاب گردیدند، جهت رده‌بندی بر اساس الگوریتم KNN انتخاب گردیدند. سپس هر بار یکی از ۱۴ ویژگی که در [۹]، به عنوان کاراترین ویژگی هایی که تاکنون در پژوهش‌ها از آن‌ها بهره برده شده است، معرفی شده‌اند، به وِیژگی‌ها افزوده می‌شدند و سپس دقت الگوریتم نسبت به حالت قبل سنجیده می‌شد. چنانچه افزودن آن ویژگی منجر به افزایش دقت می‌شد به عنوان یکی از ویژگی‌های منتخب تعیین می‌گردید، در غیر این صورت از انتخاب آن ویژگی صرف نظر می‌شد.

جدول ۳-۱) ویژگی‌های استخراج شده‌ی موجود در بردار ویژگی جریان‌ها

نام ویژگی	توضیح
All-inp	تعدادکل بسته‌های ورودی
Min_outp	طول کوچکترین بسته‌ی خروجی (byte)
BS	متوسط تعداد بیت بر ثانیه (bit/s)
Duration	طول مدت جریان(s)
APL	متوسط طول محتوای بسته(byte)
Min_toutps	حداقل زمان بین دو بسته ارسالی خروجی(ms)
Dest_Port	پورت مقصد

[29] Transport layer
[30] Network layer
[31] Data link layer
[32] correlation

۳-۲- بخش شناسایی

اولین گام جهت راه‌اندازی سیستم، اجرای آموزش اولیه و در واقع تولید مدل اولیه‌ی رده‌بندها می‌باشد. در این مرحله ابتدا تعداد محدودی (n) نمونه‌ی آموزشی برچسب‌گذاری شده از هر دو نوع ترافیک بات‌نت و ترافیک عادی، به عنوان مجموعه داده‌ی آموزشی اولیه، که در الگوریتم ارائه شده در شکل ۳-۲، با عنوان Init_TrainingSet نشان داده شده، ذخیره می‌شود. این نمونه‌ها به صورت تصادفی از میان نمونه‌های آموزشی موجود در مجموعه داده در اختیار، انتخاب شده‌اند. این مجموعه‌ی آموزشی اولیه تا پایان کار و در تمامی مراحل بدون تغییر باقی خواهد ماند و در مراحل موجود در بخش به‌روزرسانی و یادگیری افزایشی مورد استفاده قرار می‌گیرد. اولین مدل‌های رده‌بند k–نزدیک‌ترین همسایه با چهار k مختلف در این مرحله و براساس Init_TrainingSet تشکیل می‌شوند.

پس از انجام آموزش اولیه، سیستم آغاز به کار می‌نماید. در این بخش جریان‌های ترافیک ورودی به صورت بسته‌های هزارتایی وارد شده و توسط رده‌بندها مورد تحلیل قرار گرفته و هریک از رده‌بندها به جریان‌ها برچسب عادی و یا بات‌نت اختصاص داده می‌دهند. سپس با رای‌گیری میان برچسب‌های پیشنهادی، برچسبی که توسط تعداد بیشتری از رده‌بندها پیشنهاد شده باشد، به عنوان برچسب پیش‌بینی شده نهایی انتخاب می‌شود. به این ترتیب عملیات شناسایی و تشخیص بات‌نت در همین قسمت صورت می‌گیرد.

این رده‌بندها بر اساس مجموعه آموزشی پویا که به صورت مستمر در حال به‌روزرسانی شدن است عمل می‌کنند. همانطور که پیش‌تر بیان شد، در آغاز مجموعه‌ی آموزشی پویا برابر با همان مجموعه‌ی آموزشی اولیه می‌باشد.

۳-۳- بخش به‌روزرسانی (یادگیری افزایشی)

در این بخش با بهره بردن از داده‌های (جریان‌های ترافیک) ورودی بدون برچسب که جهت شناسایی به سیستم وارد شده‌اند و توسط رده‌بند ترکیبی، به هریک از آن‌ها برچسبی اختصاص داده شده است، آموزش افزایشی و به روزرسانی سیستم انجام می‌گردد. برای این منظور بر روی هر یک از بسته‌های داده‌ها، مراحل ۳-۳-۱ تا ۳-۳-۲ اجرا می‌شود. پس از بررسی تمام داده‌های موجود در هر بسته، عملیات به روزرسانی سیستم به کمک داده‌های منتخب انجام می‌شود.

۳-۳-۱- مقایسه‌ی برچسب‌های پیش‌بینی شده با برچسب‌های پیشنهادی براساس نمونه‌های اولیه

در این مرحله بر مبنای داده‌های موجود در مجموعه آموزشی اولیه، برای هر یک از داده‌های جدید یک برچسب پیشنهاد می‌شود. انتخاب این برچسب بر اساس فاصله‌ی داده از مراکز دو دسته باتنت‌ها و ترافیک‌های عادی صورت می‌گیرد. بنابراین فاصله‌ی اقلیدسی داده از مراکز هریک از دو دسته محاسبه می‌گردد. برچسب دسته‌ای که مرکز آن فاصله‌ی کمتری از داده داشته باشد، به عنوان برچسب پیشنهادی برای این داده معرفی می‌شود.

بدین ترتیب برای هریک از جریان‌ها برچسبی پیشنهاد می‌شود. چنانچه برچسب پیشنهادی مبتنی بر مجموعه آموزشی اولیه با برچسبی که در بخش شناسایی بر اساس رده‌بند ترکیبی مشخص شده برابر باشد، داده‌ی مورد نظر برای ورود به مرحله‌ی بعد انتخاب می‌شود. بنابراین در اینجا داده‌ها از فیلتر اولیه جهت پیوستن به مجموعه آموزشی پویا عبور می‌کنند.

علت این مقایسه این است که تنها داده‌های برچسب‌گذاری شده‌ای که برچسب‌های آن‌ها حقیقی است و سیستم از درستی آن‌ها اطمینان کامل دارد، داده‌های موجود در مجموعه اولیه می‌باشد. زیرا داده‌هایی که طی مراحل آینده به مجموعه پویا اضافه می‌گردد، داده‌های برچسب‌گذاری شده نیستند بلکه برچسب‌های آن‌ها توسط سیستم به آن‌ها اختصاص داده شده است. بنابراین تلاش بر اینست که از این حقایق اولیه‌ی موجود در سیستم غافل نشده و خیلی فاصله گرفته نشود؛ از طرفی این امر به این موضوع کمک می‌کند که اطلاعات مستخرج از داده‌های جدید با اطلاعات معتبری که داده‌های اولیه در اختیار قرار می‌دهند تناقضی نداشته باشند.

لازم به ذکر است که با توجه به اینکه داده‌های موجود در مجموعه اولیه ثابت و ایستا هستند، محاسبه‌ی مراکز دسته‌ها تنها یک بار انجام خواهد شد و نیاز به محاسبه‌ی مجدد در تکرارها و دفعات بعدی نخواهد بود. بنابراین بار محاسباتی زیادی را به الگوریتم اضافه نخواهد کرد.

۳-۳-۲- تعیین صلاحیت جریان‌ها برای افزوده شدن به مجموعه آموزشی پویا

این مرحله مهم‌ترین بخش سیستم می‌باشد و کارایی کل سیستم بستگی زیادی به عملکرد این قسمت دارد، زیرا داده‌هایی که صلاحیت اضافه شدن به مجموعه‌ی آموزشی پویا را دارند در این قسمت انتخاب می‌گردند. در انتخاب داده‌ها، باید چند مساله را درنظر داشت؛ چنانچه محدودیت‌های زیادی برای انتخاب داده‌ها اعمال شود و این عمل بسیار سخت‌گیرانه باشد، تعداد داده‌هایی که در نهایت برای به‌روزرسانی سیستم انتخاب می‌شوند بسیار کم بوده و در نتیجه علی رغم صرف زمان و انرژی زیاد، بهبود چندانی صورت نگرفته و مجموعه‌ی آموزشی در حد کمی توسعه خواهد یافت.

از طرف دیگر، اگر شرایط انتخاب داده‌ها بسیار ساده باشد و تعداد داده‌های منتخب زیاد شود، به سرعت اندازه‌ی مجموعه آموزشی و در نتیجه پیچیدگی مکانی و زمانی الگوریتم افزایش پیدا خواهد کرد. از طرفی امکان نامناسب بودن داده‌های افزوده شده به مجموعه آموزشی نیز افزایش می‌یابد که می‌تواند منجر به گمراهی سیستم و در نتیجه کاهش دقت سیستم شناسایی گردد.

داده‌های منتخبی که از مرحله ۳-۲-۱ وارد این مرحله شده‌اند، به منظور تعیین صلاحیت از دو فیلتر دیگر عبور خواهند کرد. در ادامه عملکرد هر یک از این فیلترها شرح داده می‌شود. در شکل ۳-۲، این مرحله با تابعی با عنوان Qualification نشان داده شده است.

نسبت قابل قبول بین فاصله‌ی داده از مراکز دو دسته

انتخاب دسته‌ای که داده‌ی مورد نظر احتمالا به آن تعلق دارد صرفاً بر اساس اینکه فاصله‌ی این داده از مرکز یک دسته نسبت به فاصله‌ی آن تا مرکز دسته‌ی دیگر بزرگتر است، می‌تواند از دقت عمل کاسته و سیستم را گمراه کند. در حقیقت لازم است میزان اختلاف این فاصله‌ها با یکدیگر نیز در نظر گرفته شود. برای مثال حالتی را فرض کنید که یک داده تقریبا در مرز بین دسته‌ی اول و دوم قرار گرفته و تنها با اختلاف کمی بر حسب فاصله از مراکز، به یکی از دسته‌ها نسبت داده می‌شود. در صورتی که احتمال تعلق چنین داده‌ای به دسته‌ی دیگر نیز بالاست. بنابراین این داده، نامزد خوبی برای مورد استفاده قرار گرفتن جهت به‌روزرسانی سیستم و پیوستن به مجموعه آموزشی پویا نمی‌باشد.

با توجه به این نکته، در این قسمت فیلتری قرار می‌گیرد که تنها داده‌هایی برای ارسال به مرحله‌ی بعد انتخاب شوند که فاصله‌ی اقلیدسی آن‌ها از مرکز دسته‌ی منتخب حداقل $1/\alpha$ برابر دسته‌ی دیگر باشد.

بسته به مقدار این آلفا، تعداد داده‌هایی که از این فیلتر عبور می‌کنند متغیر است. لازم است که ازمیان مقادیر مختلف α، با تکرار آزمایش، مقدار مناسب انتخاب گردد.

محدوده‌ی قابل قبول تغییر واریانس داده‌های موجود در هر دسته

جهت جلوگیری از تغییرات شدید در واریانس هر دسته در مجموعه‌ی آموزشی، شرط دیگری نیز به این بخش اضافه نمودیم. این به این معناست که ما تمایلی به تغییرات بسیار شدید در مجموعه‌ی آموزشی خود نداریم؛ زیرا احتمال کاهش تاثیر دانش اولیه را به دنبال دارد.

بنابراین داده‌ای که اضافه شدن آن منجر به افزایش واریانس دسته‌ی مربوطه، بیش از حد تعیین شده‌ای می‌گردد صلاحیت اضافه شدن به مجموعه آموزشی از آن سلب خواهد شد.

از طرفی تغییرات بسیار کم در مجموعه‌ی آموزشی پویا نیز مطلوب نیست. درنتیجه داده‌هایی که بیش از حد به داده‌های موجود در مجموعه شباهت دارند و بنابراین واریانس را در حد ناچیزی تغییر می‌دهند نیز، با هدف صرفه جویی در میزان حافظه‌ی مصرفی انتخاب نخواهند شد، زیرا این داده‌ها اطلاعات بیشتری به سیستم اضافه نمی‌کنند. تعیین حد بالا و پایین تغییرات قابل قبول واریانس دسته‌ها می‌تواند با تکرار آزمایش با مقادیر مختلف برای این حدود صورت گیرد.

همانطور که بیان شد عملکرد سیستم در این مرحله بر روی کارایی کل بسیار تاثیرگذار است. در حقیقت اگر تعداد زیادی از داده‌ها کاندید اضافه شدن به مجموعه‌ی آموزشی پویا شوند خطر کاهش دقت رده‌بند به وجود می‌آید؛ حتی اگر اغلب داده‌های اضافه شده هم به دسته‌ی حقیقی خود انتساب داده شوند، باز هم کارایی سیستم را از لحاظ پیچیدگی مکانی تحت تاثیر خود قرار می‌دهند. از سوی دیگر، چنانچه انتخاب این داده‌ها بسیار سخت‌گیرانه باشد و هربار تعداد بسیار کمی به

مجموعه اضافه شوند، آموزش افزایشی سیستم به شدت کند پیش می‌رود و آنگونه که انتظار می‌رود تاثیر چندانی در قدرت تشخیص باتنت‌های جدید نخواهد داشت. بنابراین نحوه‌ی تصمیم‌گیری برای انتخاب نمونه‌های مناسب دارای اهمیت زیادی خواهد بود.

۳-۳-۳- به‌روزرسانی سیستم

پس از عبور از فیلترهای تعیین صلاحیت، داده‌های برچسب‌گذاری شده‌ی منتخب (qualifiedSet) از هر بسته، جهت به‌روزرسانی سیستم به مجموعه‌ی آموزشی پویا اضافه می‌گردند. سپس رده‌بندها، براساس این مجموعه‌ی جدید مدل‌های خود را بازسازی نموده و از این پس در شناسایی از این مدل‌های جدید استفاده می‌شود. این مرحله در الگوریتم نشان داده شده در شکل ۳-۲ با تابعی با عنوان System-Update نمایش داده شده است.

نکته قابل توجه دیگر این است که اگرچه رده‌بند پایه‌ی مورد استفاده kNN است و به روزرسانی آن پیچیدگی محاسباتی بالایی ندارد و تنها لازم است تعدادی داده به مجموعه داده‌ی آن‌ها اضافه شود، اما انجام مکرر این عمل به ازای هربار یافتن داده‌ی مناسب به صرفه نبوده و بر کارایی سیستم تاثیر منفی خواهد گذاشت. به همین دلیل است که عملیات به روزرسانی سیستم به ازای هر داده انجام نمی‌شود، بلکه پس از بررسی تمام هزار عدد داده‌ی موجود در بسته، و در واقع به ازای هر بسته صورت می‌گیرد.

لازم است توجه شود که داده‌ی جدیدی که وارد سیستم می‌شود، پیش از آن که برای به روزرسانی سیستم مورد استفاده قرار گیرد، برچسب به آن اختصاص داده شده و عملیات تشخیص باتنت یا عادی بودن آن پایان یافته است. اما در بخش بعدی سیستم، این داده که در واقع به عنوان داده آزمون به سیستم اعمال شده بود، جهت ارتقای سیستم مورد بررسی و بهره‌برداری قرار می‌گیرد.

همانطور که ملاحظه نمودید، سیستم، بدون داشتن برچسب واقعی داده‌های جدید می‌تواند از آنها برای بهبود عملکرد خود بهره ببرد. نکته ارزشمند این است که این برچسب یک برچسب پیش‌بینی شده توسط سیستم است و برای بهره بردن از این داده جهت به روزرسانی، نیازی نیست این برچسب واقعی باشد؛ این موضوع سیستم ارائه شده را از سایر سیستم‌های تشخیص نظارتی مجزا می‌کند. زیرا یکی از مشکلات اصلی روش‌های نظارتی، ضرورت در اختیار داشتن تعداد زیاد داده‌های برچسب‌گذاری شده می‌باشد. اما در سیستم ارائه شده در این پژوهش، برای به کار گرفتن روش نظارتی نیازی به داشتن تعداد زیادی داده با برچسب واقعی نیست. بلکه با تعداد بسیار کمی داده برچسب‌گذاری شده سیستم راه‌اندازی شده و سپس برچسب‌های داده‌های جدیدی که در رده‌بندی خود بکار می‌گیرد را خود پیش‌بینی می‌کند. در واقع سیستم نیازی به دانستن برچسب حقیقی ندارد، بلکه آن‌ها را پیش‌بینی می‌کند. در نتیجه در محیط واقعی نیز قادر به بهره بردن از ترافیک‌های جدید ناشناخته‌ای که هر لحظه ممکن است وارد سیستم شوند می‌باشد.

فصل چهارم

پیاده‌سازی و ارزیابی روش پیشنهادی

به منظور ارزیابی روش ارائه شده، سیستم مورد نظر در جاوا پیاده‌سازی شده است و آزمایش‌های متعددی بر روی آن انجام گردیده است. در این فصل جزئیات و چگونگی این پیاده‌سازی شرح داده می‌شود. ابتدا در بخش ۴-۱ نحوه راه‌اندازی و تنظیم پارامترها بیان می‌شود. سپس آزمایش‌ها و ارزیابی نتایج بیان شده و در نهایت مقایسه‌ای بین سیستم پیشنهادی و تعدادی سیستم مشابه انجام می‌گردد.

۴-۱- تنظیمات پارامترها و راه‌اندازی

در این بخش، ابتدا مجموعه داده مورد استفاده معرفی می‌شود. سپس نحوه انتخاب مقادیر هریک از پارامترها بیان می‌گردد.

۴-۱ -۱- مجموعه داده‌ی باتنت ISCX

مجموعه داده‌ای که برای آموزش و ارزیابی سیستم مورد استفاده قرار می‌گیرد، به عنوان یکی از چالش‌های اصلی در میان سیستم‌های تشخیص باتنت و به طور کلی در سیستم‌های تشخیص نفوذ، نقش به سزایی در اعتبار دقت تشخیص گزارش شده از یک سیستم ایفا می‌کند. یک مجموعه داده باید دارای سطح بالایی از تنوع از نظر نوع باتنت‌ها باشد. هم‌چنین لازم است در بین داده‌های آزمون انواع جدیدی از باتنت‌ها وجود داشته باشد که در میان داده‌های آموزش دیده نشده باشند؛ تا بدین ترتیب میزان قابلیت سیستم در تشخیص انواع جدید باتنت سنجیده شود. به علاوه داده‌های آموزش و آزمون تا حد امکان نباید با یکدیگر هم‌پوشانی داشته باشند.

در این پژوهش، در جهت رفع این مشکل، ازمجموعه داده باتنت ISCX استفاده شد که تا کنون در بین مجموعه داده‌های باتنت موجود دارای جامعیت بیشتر و سطح تنوع بسیار وسیع‌تری از

باتنت‌ها می‌باشد[۹]. با توجه به آن که این مجموعه داده ترکیب غیریکنواخت و جامعی دارد، تا سطح بالایی قادر به شبیه‌سازی ترافیک واقعی می‌باشد؛ بنابراین ارزیابی کارایی سیستم ما می‌تواند تا حد زیادی قابل اعتماد باشد. این مجموعه داده در آزمایشگاه ISCX در دانشگاه UNB کانادا تهیه شده است[۳۳].

مجموعه داده باتنت ISCX، دارای ۷ نوع باتنت در مجموعه‌ی آموزشی خود به حجم ۵.۳ گیگابایت و ۱۶ نوع باتنت در مجموعه‌ی آزمون خود به حجم ۸.۵ گیگابایت می‌باشد. وجود تنوع بیشتر باتنت‌ها در مجموعه‌ی آزمون نسبت به مجموعه‌ی آموزشی این امکان را فراهم می‌کند که قابلیت سیستم در تشخیص باتنت‌های نوع جدید مورد ارزیابی قرار گیرد.

در جدول ۴-۱و۴-۲ لیست انواع باتنت‌های موجود به ترتیب در مجموعه آموزشی و مجموعه آزمون و نیز درصد فراوانی آن‌ها در این مجموعه داده نمایش داده شده است. جزئیات بیشتر در مورد این مجموعه داده و چگونگی تهیه‌ی آن در مرجع [۹] به تفصیل بیان شده است.

جدول۴-۱) توزیع انواع باتنت‌ها در مجموعه داده آموزشی ISCX [۹]

نام باتنت	نوع	سهم جریان‌ها در مجموعه داده

[33] University of New Brunswick

Neris	IRC	(٪ ۱۲) ۲۱۱۵۹
Rbot	IRC	(٪ ۲۲) ۳۹۳۱۶
Virut	HTTP	(٪ ۰.۹۴) ۱۶۳۸
NSIS	P2P	(٪ ۲.۴۸) ۴۳۳۶
SMTP Spam	P2P	(٪ ۶.۴۸) ۱۱۲۹۶
Zeus	P2P	(٪ ۰.۰۱) ۳۱
Zeus control (C&C)	P2P	(٪ ۰.۰۱) ۲۰

جدول ۴-۲) توزیع انواع بات‌نت‌ها در مجموعه داده آزمون ISCX [۹]

نام بات‌نت	نوع	سهم جریان‌ها در مجموعه داده
Neris	IRC	(٪ ۵.۶۷) ۲۵۹۶۷
Rbot	IRC	(٪ ۰.۰۱۸) ۸۳
Menti	IRC	(٪ ۰.۶۲) ۲۸۷۸
Sogou	HTTP	(٪ ۰.۰۱۹) ۸۹
Murlo	IRC	(٪ ۱.۰۶) ۴۸۸۱
Virut	HTTP	(٪ ۱۲.۸۰) ۵۸۵۷۶
NSIS	P2P	(٪ ۰.۱۶۵) ۷۵۷
Zeus	P2P	(٪ ۰.۱۰۹) ۵۰۲
SMTP Spam	P2P	(٪ ۴.۷۲) ۲۱۶۳۳
UDP Storm	P2P	(٪ ۹.۶۳) ۴۴۰۶۲
Tbot	IRC	(٪ ۰.۲۸۳) ۱۲۶۹
Zero Access	P2P	(٪۰.۲۲۱) ۱۰۱۱
Weasel	P2P	(٪ ۹.۲۵) ۴۲۳۱۳
Smoke Bot	P2P	(٪ ۰.۰۱۷) ۷۸
Zeus Control (C&C)	P2P	(٪ ۰.۰۰۶) ۳۱
ISCX IRC bot	P2P	(٪ ۰.۳۸۷) ۱۸۱۶

در این پژوهش، پس از استخراج ویژگی‌ها از داده‌های موجود در هر دو مجموعه آموزشی و آزمون، هریک از نمونه‌ها تبدیل به یک بردار ویژگی با مولفه های عددی پیوسته شدند. سپس ترتیب

قرار گرفتن آن‌ها در هر مجموعه به صورت تصادفی تعیین شد. برای انجام آزمایش‌ها و ارزیابی‌ها، از میان داده‌های مجموعه آزمون، دو دسته، هریک شامل ۳۰۰۰ داده انتخاب شد.

دسته‌ی اول شامل انواع داده‌های عادی و باتنت می‌باشد. در میان داده‌های باتنت، باتنت‌هایی از انواع جدید نیز که در میان داده‌های آموزشی دیده نمی‌شوند، وجود دارد. اما در دسته‌ی دوم، بخش عمده‌ی دسته را تنها انواع باتنت‌هایی تشکیل می‌دهند که در مجموعه آموزشی وجود نداشته‌اند. باقی دسته شامل داده‌های عادی می‌باشد. در ادامه جهت سهولت، به دسته اول، دسته آزمون تصادفی و به دسته دوم، دسته آزمون تعمیم‌پذیری گفته می‌شود.

به منظور انجام آزمایش سوم در بخش ۴-۲، به کمک داده‌های آزمون و آموزشی، یک مجموعه آموزشی دیگر نیز تشکیل می‌شود. در این مجموعه، ۲۰۰۰۰۰ داده‌ی اول به صورت تصادفی از مجموعه آموزشی انتخاب شده است. اما در ۲۵۰۰۰۰ داده بعدی، علاوه بر داده‌های مجموعه آموزشی، تعداد زیادی از همه‌ی انواع جدید باتنت که تنها در مجموعه آزمون وجود دارند، نیز به صورت پراکنده در میان داده‌های دیگر قرار داده شده‌اند.

۴-۱-۲- انتخاب رده‌بند پایه

با توجه به ساختار سیستم که در آن رده‌بند در طول اجرا دائماً به روزرسانی می‌شود، الگوریتم پایه مورد استفاده بهتر است غیر پارامتریک باشد. در نتیجه از میان الگوریتم‌های نظارتی غیر پارامتریک، دو الگوریتم KNN و بیز ساده[۳۴] که ساختار نسبتا ساده‌ای دارند، انتخاب شدند. به دلیل ساده‌تر بودن الگوریتم KNN، ابتدا این سیستم بر اساس این الگوریتم پیاده‌سازی شد؛ اما به دلیل اینکه الگوریتم بیزین ساده دارای رویکردی احتمالاتی است و با رویکرد KNN که بر اساس

[34] Naïve Bayes

فاصله اقلیدسی داده‌هاست متفاوت می‌باشد، یادگیری افزایشی بر پایه بیزین ساده نیز طراحی و پیاده‌سازی گردید.

جهت به کارگیری الگوریتم بیزین ساده، لازم است داده‌ها گسسته‌سازی شوند. برای این منظور به ازای هر ویژگی، محدوده تغییرات مقادیر ویژگی به ۱۰ بازه تقسیم‌بندی شده و بر اساس این که هر مقدار ویژگی در چه بازه‌ای قرار گرفته باشد، با مقدار مشخص شده به عنوان نماینده‌ی آن بازه جایگزین می‌گردد. تعیین این بازه‌ها به صورتی انتخاب گردیده است که تعداد مقادیرمختلف در همه بازه‌ها تقریبا به صورت یکنواخت پراکنده شده باشند. برای انتخاب این بازه‌ها از فیلتر گسسته‌سازی در نرم‌افزار وکا[۳۵] استفاده شده است.

پس از آماده‌سازی داده‌های گسسته‌سازی شده، از آن‌ها در پیاده‌سازی سیستم مورد نظر بر پایه بیزین ساده، استفاده شد. در این حالت نرخ تشخیص نهایی سیستم بر اساس دسته آزمون تصادفی، ۶۶٪ به دست آمد، که نسبت به حالتی که در آن از الگوریتم KNN استفاده شده بود، کمتر می‌باشد. یکی از علل کاهش دقت در این حالت می‌تواند عملیات گسسته‌سازی باشد، زیرا از دقت داده‌ها می‌کاهد. با توجه به کاهش دقت در این مورد و نیز ساده‌تر بودن الگوریتم KNN، سیستم پیشنهادی بر پایه KNN ارائه گردید.

۴-۱-۳- انتخاب مقادیر پارامترها

همانطور که در فصل سوم اشاره شده، انتخاب مقادیر مناسب برای پارامترهای متغیر سیستم نقش کلیدی در کارایی سیستم تشخیص دارد. در ادامه نحوه‌ی انتخاب این مقادیر برای پارامترها بیان می‌شود.

[35] Weka (v.3-8-0)

پارامتر k در الگوریتم ترکیبی مبتنی بر kNN

به منظور انتخاب بهترین مقادیر برای k، به ازای k =۱ تا k= ۱۰ الگوریتم kNN معمولی بر روی کل مجموعه آموزشی اجرا گردید و با دسته آزمون تصادفی، دقت عملکرد هر حالت مورد آزمون قرار گرفت. نتایج این آزمایش‌ها در جدول ۴-۳ قابل مشاهده است. همانطور که مشاهده می‌شود، دقت عملکرد (نرخ تشخیص) در k = ۳و۵و۹ نسبت به سایر حالات بیشتر است. بنابراین این مقادیر برای پارامترهای رده‌بندهای سیستم انتخاب می‌شود و تشخیص نهایی، بر اساس رای‌گیری بین رده‌بندهای kNN با این مقادیر k صورت می‌گیرد.

البته انتخاب مقادیر مناسب برای K بر اساس نتایج الگوریتم KNN معمولی دقیق نیست؛ زیرا الگوریتم مورد استفاده در سیستم به صورت افزایشی انجام می‌شود و در شرایط واقعی، در زمان راه‌اندازی سیستم همه‌ی داده‌ها در اختیار نیست تا بهترین K انتخاب گردد. اما در این پژوهش، از آنجایی که مجموعه داده از ابتدا در اختیار بود، به جای انتخاب مقدار تصادفی برای K، از مجموعه داده موجود بهره برده شد و مقادیر مناسب K بر اساس نتایج اعمال KNN روی این داده‌ها انتخاب گردیدند.

جدول ۴-۳) نرخ تشخیص kNN به ازای مقادیر متفاوت k

K	نرخ تشخیص
۱	٪ ۷۴/۸۵
۲	٪ ۶۹/۱۲
۳	٪ ۷۶/۹۲
۴	٪ ۷۴/۰۹
۵	٪ ۷۶/۹۵
۶	٪ ۷۴/۲۸
۷	٪ ۷۶/۸۵

۸	٪ ۷۵/۰۵
۹	٪ ۷۷/۱۲
۱۰	٪ ۷۳/۴۹

پارامتر آلفا (α)

برای انتخاب مقدار مناسب برای آلفا، لازم است که عملکرد سیستم را در یک آزمایش یکسان، به ازای مقادیر مختلف آلفا تکرار نمود. آزمایشی که برای این منظور انتخاب شد، اجرای سیستم تا زمانی که ۵۰ مرتبه عملیات به روزرسانی انجام شود، و ارزیابی سیستم در طول این مدت بر اساس دسته آزمون تصادفی بود. هر آزمایشی که بیشترین مقدار میانگین نرخ تشخیص در طول اجرا را داشت به عنوان بهترین مقدار برای آلفا تعیین شد. با اعمال مقادیر متعدد آلفا و تکرار آزمایش‌ها، مقدار نهایی ۱.۱۲ انتخاب گردید.

بازه‌ی تغییرات واریانس مجموعه آموزشی پویا

هدف از اعمال شرط محدودیت تغییرات واریانس، کاهش به صرفه‌ی تعداد داده‌های انتخاب شده برای افزودن به مجموعه پویا می‌باشد؛البته به طوری که اعمال این شرط دقت عملکرد سیستم را نسبت به حالتی که این شرط وجود نداشته باشد، کاهش ندهد. انتخاب بازه‌ی مناسب برای تغییرات نقش تعیین کننده در این موضوع دارد.

بنابراین برای انتخاب بازه مناسب، سیستم بدون وجود این شرط اجرا می‌شود. در هر مرحله تعداد داده‌هایی که برای افزوده شدن به مجموعه پویا انتخاب می‌شوند و نیز نرخ تشخیص سیستم در این وضعیت نیز بر اساس دسته آزمون تصادفی ثبت می‌شود. سپس این آزمایش با وجود شرط محدودیت بازه تغییرات واریانس بارها تکرار می‌شود. در هر تکرار، بازه‌های مختلفی اعمال می‌گردد.

نتایج هر تکرار با نتایج آزمایش بدون شرط مقایسه می‌گردد. به این ترتیب هر بازه‌ای که منجر به حصول نرخ تشخیص برابر با آزمایش بدون شرط شود و از طرفی تعداد داده‌های افزوده شده به طور قابل توجهی کمتر از این تعداد در آزمایش بدون شرط باشد، به عنوان بازه مناسب انتخاب می‌شود.

پس از انجام آزمایش‌های متعدد با بازه‌های مختلف، بازه [۰.۴۵و۰.۳۰] به عنوان بهترین بازه انتخاب گردید. همانطور که در بخش ۴-۴ نیز اشاره خواهد شد، میزان صرفه‌جویی حافظه‌ی نگهداری مجموعه پویا در این حالت نسبت به حالت بدون این شرط، حدود ۶۷٪ درصد می‌باشد.

۴-۲- آزمایش‌ها و ارزیابی نتایج

همانطور که پیش‌تر بیان شد سیستم ارائه شده برای بکارگیری در محیط پویا طراحی شده است؛ محیطی که هر لحظه ممکن است جریان جدیدی وارد سیستم شود که این جریان می‌تواند مشابه داده‌های پیشین و یا از نوع جدیدی باشد. سیستم موظف است در مورد جریان‌های ورودی به خوبی تصمیم‌گیری کند و در صورت لزوم آنها را در میان مجموعه آموزشی پویای خود ذخیره کند.

به منظور شبیه‌سازی چنین محیط پویایی، پس از راه‌اندازی اولیه سیستم و ایجاد مدل اولیه ردبندهای KNN براساس مجموعه آموزشی اولیه، تعدادی داده‌ی آزمون در طی چندین مرحله جهت فراهم نمودن شرایط آموزش افزایشی وارد سیستم می‌شود. این داده‌های آزمون در واقع نقش همان داده‌های آزمونی را ایفا می‌کنند که در یک محیط پویای واقعی وارد سیستم خواهند شد. در این بخش تمامی این مراحل با جزئیات بیشتر شرح داده می‌شود.

لازم به ذکر است که مجموعه آموزشی اولیه به صورت تصادفی از میان داده‌های مجموعه آموزشی مجموعه داده ISCX انتخاب می‌شود. تعداد داده‌های موجود در این مجموعه ۵۰۰ داده

می‌باشد. علت در نظر گرفتن تعداد کم برای مجموعه آموزشی اولیه نشان دادن کارایی سیستم ارائه شده در تشخیص، تنها با داشتن تعداد خیلی کمی داده برچسب‌گذاری شده می‌باشد؛ زیرا یکی از مهم‌ترین مزیت روش ارائه شده، رفع این محدودیت روش‌های نظارتی است که این روش‌ها برای دست یافتن به کارایی بالا نیاز به تعداد زیادی داده برچسب‌گذاری شده دارند. اما در این سیستم، به کمک یادگیری افزایشی تنها با داشتن یک مجموعه کوچک برچسب‌گذاری شده، کارایی بالایی به دست می‌اید.

آزمایش‌هایی که جهت ارزیابی سیستم اجرا می‌شوند، همگی یکسان هستند و تنها تفاوت آن‌ها در داده‌هایی است که در آموزش و ارزیابی مورد استفاده قرار می‌گیرند. بنابراین می‌توان عملکرد سیستم را در شرایط مختلف بررسی نمود.

آزمایش به این صورت است که در یک حلقه تکرار، هر بار بسته‌های داده که شامل ۱۰۰۰ داده می‌باشد، وارد سیستم می‌شود. با استفاده از الگوریتم ترکیبی، برچسب هر یک از این داده‌ها پیش‌بینی شده و بات‌نت یا عادی بودن آن‌ها تعیین می‌گردد. سپس این بسته داده که اکنون برچسب‌گذاری شده است، به بخش یادگیری افزایشی و به روزرسانی سیستم فرستاده می‌شود، در آن‌جا داده‌های مناسب به مجموعه پویا افزوده می‌گردند. جهت ارزیابی عملکرد سیستم و تاثیر یادگیری افزایشی در طول اجرا، به ازای هر ۵۰ بسته داده‌ای که به سیستم وارد می‌شود، یک بار سیستم تشخیص بر اساس یک مجموعه آزمون، ارزیابی می‌شود.

معیار ارزیابی در این پژوهش، همانند سایر پژوهش‌های مشابه، دو معیار متداول نرخ تشخیص و نرخ هشدار نادرست که به ترتیب در رابطه ۴-۱ و ۴-۲ نشان داده شده‌اند، می‌باشند. هم‌چنین در این پژوهش، نرخ تشخیص درست هر دو نوع بات‌نت و سالم نیز محاسبه شده است که در رابطه۴-۳ نمایش داده شده است.

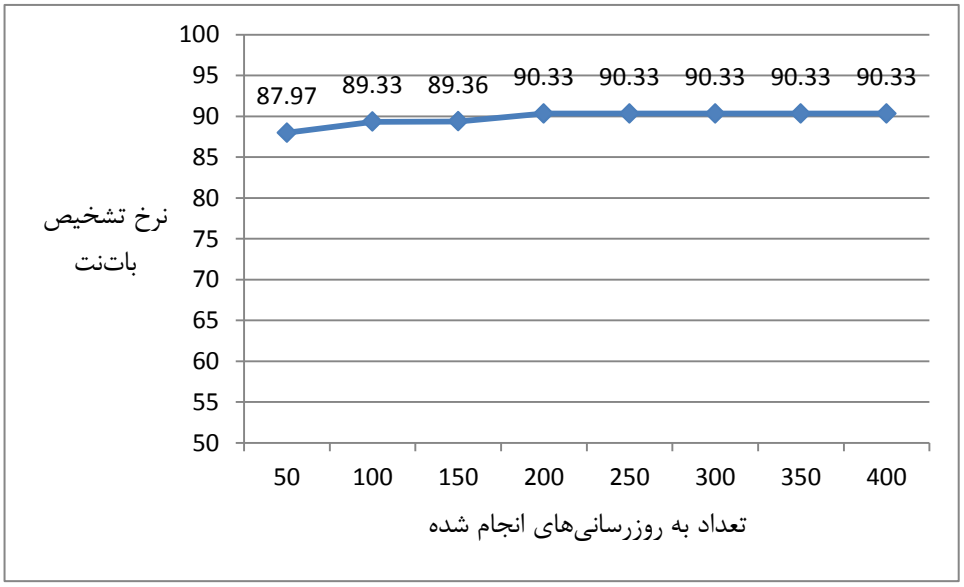
$$\text{نرخ تشخیص (باتنت‌ها)} = \frac{\text{تعداد داده‌های باتنتی که به درستی شناسایی شده‌اند}}{\text{تعداد باتنت‌های موجود}}$$ رابطه۴-۱)

$$\text{نرخ هشدار نادرست} = \frac{\text{تعداد داده‌های سالمی که باتنت تشخیص داده شده‌اند}}{\text{تعداد کل داده‌هایی که باتنت تشخیص داده شده‌اند}}$$ رابطه ۴-۲)

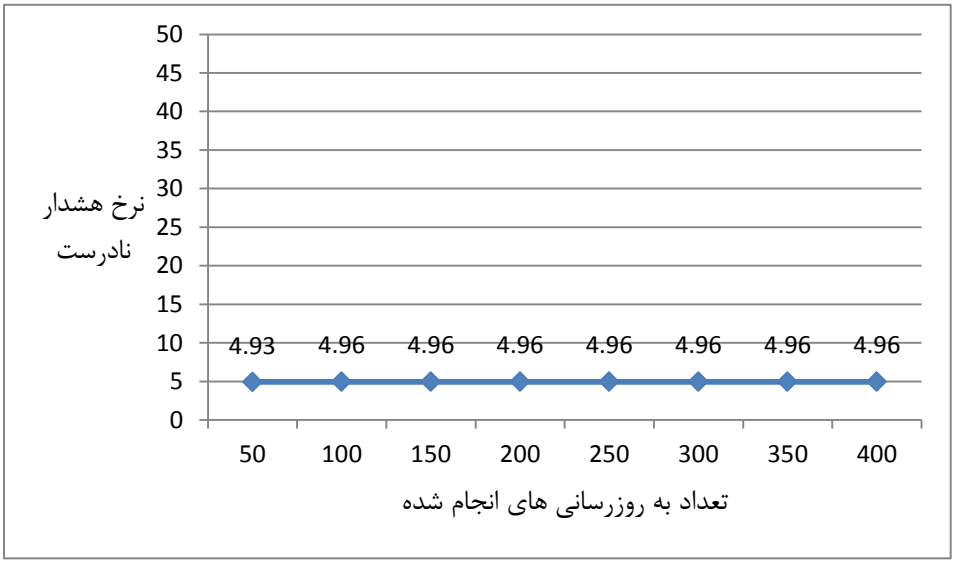
$$\text{نرخ تشخیص درست} = \frac{\text{تعداد داده‌های باتنت و سالمی که به درستی شناسایی شده}}{\text{تعداد داده‌های مورد شناسایی}}$$ رابطه ۴-۳)

همانطور که بیان شد، تنها تفاوت آزمایش‌های مختلف در مجموعه داده‌های مورد استفاده می‌باشد. در ادامه داده‌های به کار گرفته شده در هریک از سه آزمایش انجام شده و نیز هدف از انتخاب این داده‌ها بیان می‌گردد.

آزمایش اول: در آزمایش اول، بسته‌های داده‌ای که وارد سیستم می‌شوند، همان داده‌های مجموعه آموزشی ISCX می‌باشند که تبدیل به بردارهای ویژگی شده‌اند و با ترتیب تصادفی در این مجموعه قرار داده شده‌اند. جهت ارزیابی روش نیز، از بسته آزمون تصادفی استفاده می‌گردد. نتایج این ارزیابی‌ها در آزمایش اول، در نمودارهای شکل ۴-۱ و ۴-۲ و ۴-۳ نشان داده شده است.



شکل ۴-۱) نمودار نرخ تشخیص (باتنت) سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تصادفی



شکل ۴-۲) نمودار نرخ هشدار نادرست سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تصادفی

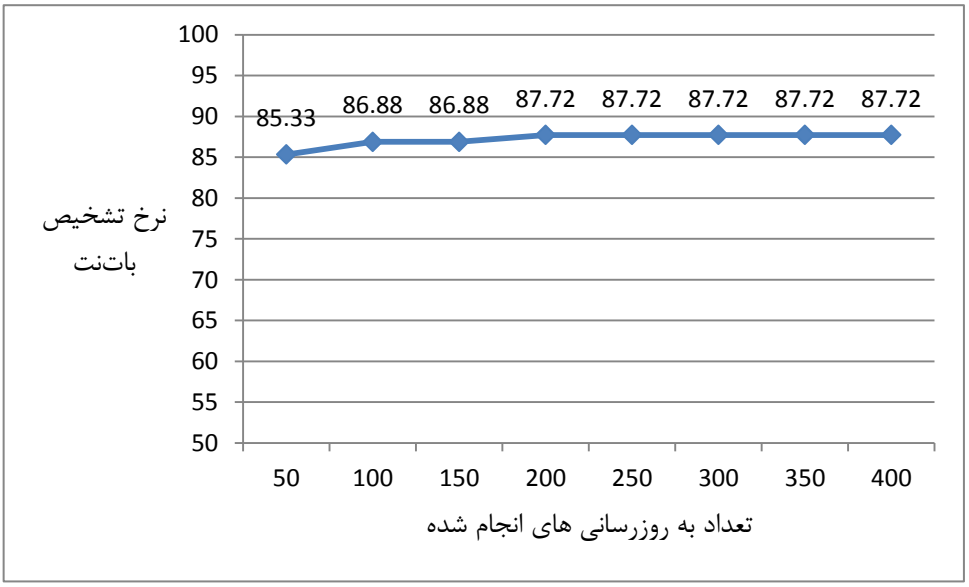
شکل ۴-۳) نمودار نرخ تشخیص درست سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تصادفی

در نمودار شکل ۴-۱، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۲ نیز، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم را نشان می‌دهد و محور عمودی نرخ هشدار نادرست سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۳، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص درست سیستم (بات‌نت‌ها و سالم‌ها) بعد از هر یک از این به روزرسانی‌ها می‌باشد.

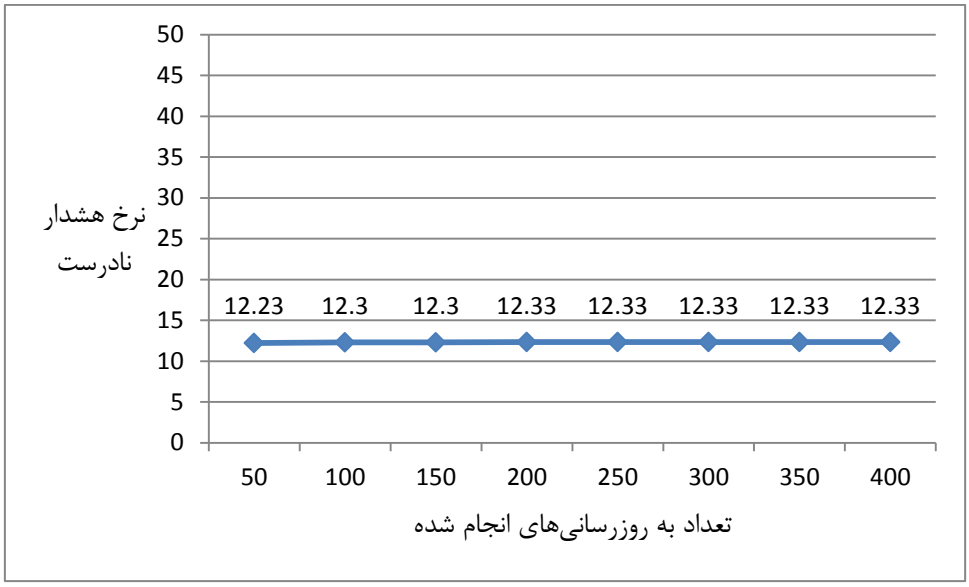
همان‌طور که در شکل ۴-۱ مشاهده می‌شود، نرخ تشخیص سیستم پس از ۵۰ مرتبه به‌روزرسانی حدود ۸۷.۹۷٪ بوده و رفته رفته در طی ورود داده‌های جدید به سیستم و به روزرسانی‌های بعدی، تا زمانی که ۲۰۰ مرتبه به روزرسانی انجام می‌شود این مقدار افزایش یافته و به نرخ تشخیص به ۹۰.۳۳٪ می‌رسد. از این لحظه به بعد، نرخ تشخیص ثابت مانده و روند کاهشی یا افزایشی ندارد، به این معنی که ذخیره داده‌های جدید در این بازه، تاثیری بر نرخ تشخیص نداشته است. روی هم رفته می‌توان نتیجه گرفت که آموزش افزایشی و به‌روزرسانی‌هایی سیستم، منجر به افزایش دقت و نرخ تشخیص سیستم شده‌اند. اما نرخ هشدار نادرست نیز همان‌گونه که در شکل ۴-۲ مشاهده می‌شود، روند افزایشی داشته است. نرخ تشخیص درست سیستم نیز با توجه به شکل ۴-۳، به تدریج افزایش یافته و از ۸۲.۲٪ به ۸۴.۴٪ می‌رسد.

جهت ارزیابی تاثیر یادگیری افزایشی بر روی میزان قدرت شناسایی بات‌نت‌های از نوع جدید که برای اولین بار در مرحله‌ی آزمون به سیستم وارد می‌شوند، آزمایش دوم و سوم انجام می‌شود.

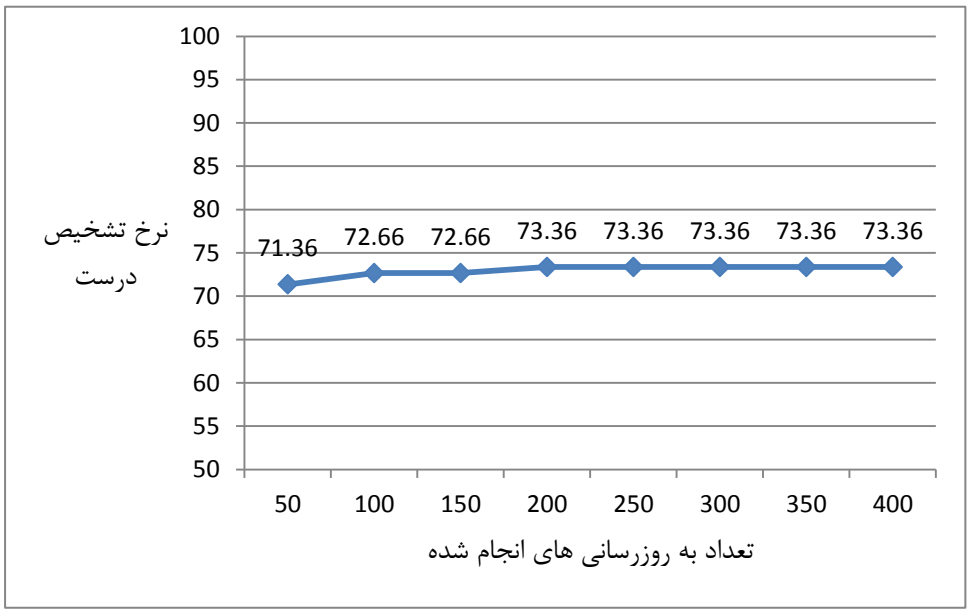
آزمایش دوم: در آزمایش دوم، داده‌های که به سیستم وارد می‌شوند همان داده‌های ورودی در آزمایش اول می‌باشند. اما برای بررسی عملکرد سیستم در حالتی که برای اولین بار با باتنت‌هایی از نوع جدید مواجه می‌شود، ارزیابی بر اساس دسته آزمون تعمیم‌پذیری انجام می‌گردد. نتایج این آزمایش در نمودارهای شکل ۴-۴ و ۴-۵ و ۴-۶ مشاهده می‌شود.



شکل ۴-۴) نمودار نرخ تشخیص(باتنت)سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تعمیم‌پذیری



شکل ۴-۵) نمودار نرخ هشدار نادرست سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تعمیم‌پذیری



شکل ۴-۶) نمودار نرخ تشخیص درست سیستم (٪) بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تعمیم‌پذیری

در نمودار شکل ۴-۴، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص

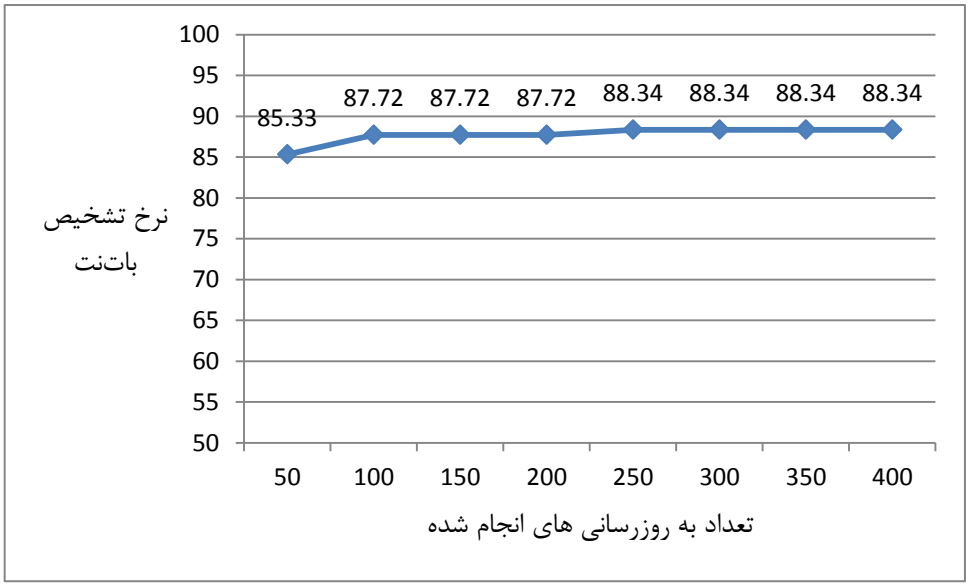
سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۵ نیز، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم را نشان می‌دهد و محور عمودی نرخ هشدار نادرست سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۶، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص درست سیستم (بات‌نت‌ها و سالم‌ها) بعد از هر یک از این به روزرسانی‌ها می‌باشد.

با مقایسه نمودارهای شکل ۴-۱ و ۴-۴، می‌توان مشاهده نمود که در آزمایش دوم، که سیستم در آن برای اولین بار با بات‌نت‌هایی از انواع دیده نشده و جدید مواجه می‌شود، نرخ تشخیص بات‌نت کاهش یافته و از حد نهایی ۹۰.۳۳٪ در آزمایش اول به ۸۷.۷۲٪ در آزمایش دوم رسیده است. نرخ هشدار نادرست نیز در آزمایش دوم افزایش یافته است. هم‌چنین نرخ تشخیص درست (بات‌نت و سالم) از ۸۴.۴٪ به ۷۳.۳۶٪ کاهش یافته است.

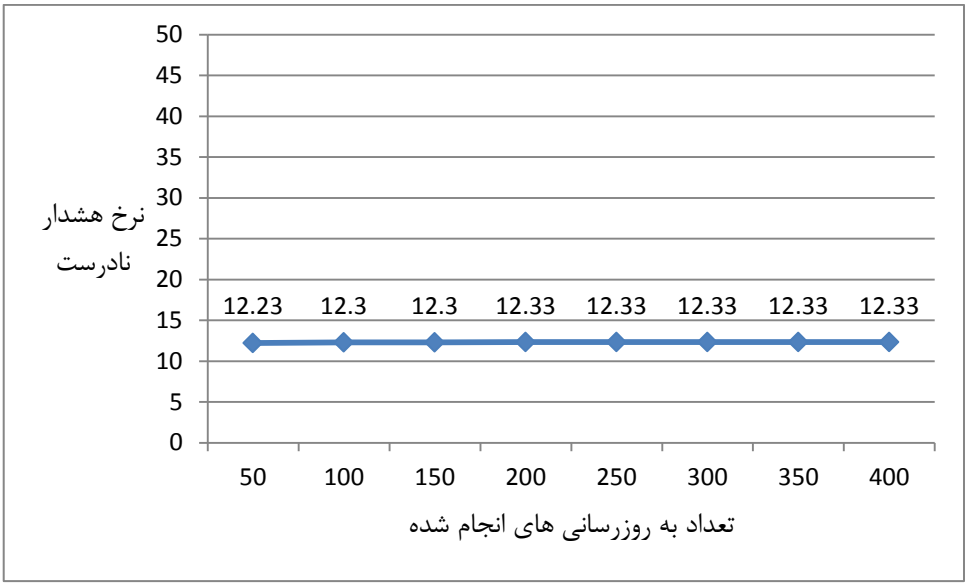
آزمایش سوم: جهت بررسی تاثیر یادگیری افزایشی بر سیستم، آزمایش سومی انجام می‌شود. در این آزمایش، ۵۰ بسته داده‌ی اول مجموعه آموزشی‌ای که مورد استفاده قرار می‌گیرد، تقریبا مشابه آزمایش قبلی می‌باشد، اما از دسته ۵۰ ام به بعد، در میان داده‌ها، بات‌نت‌های نوع جدیدی که قبلا تنها در میان داده‌های آزمون وجود داشتند نیز قرار می‌گیرند. سپس مشابه آزمایش قبل، سیستم بر اساس دسته آزمون تعمیم‌پذیری مورد ارزیابی قرار می‌گیرد.

در واقع با این آزمایش، سعی بر بررسی و شبیه‌سازی حالتی است که سیستم برای اولین بار با بات‌نت‌های جدیدی روبرو می‌شود و پس از تشخیص برچسب آن‌ها تعدادی از آن‌ها را در مجموعه خود ذخیره می‌کند. بنابراین انتظار می‌رود که در دفعات بعد چنانچه مجددا از این نوع بات‌نت‌ها وارد

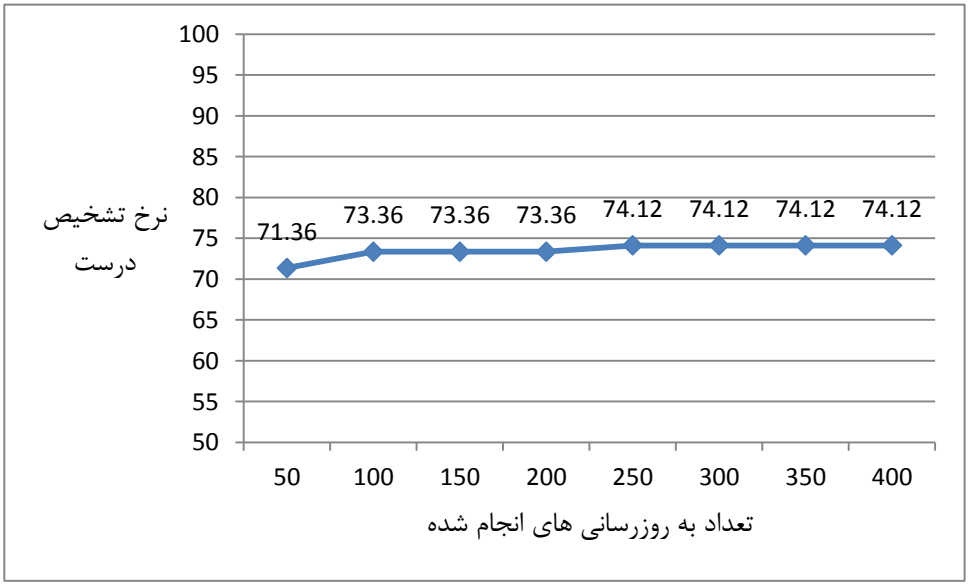
سیستم شود، با دقت بیشتری قادر به شناسایی آن‌ها باشد؛ در نتیجه ارزیابی بر اساس دسته آزمون تعمیم‌پذیری انجام می‌شود. نتایج این آزمایش در نمودار شکل ۴-۷ و ۴-۸ و ۴-۹ نمایش داده شده است.



شکل ۴-۷) نمودار نرخ تشخیص(باتنت) سیستم (٪) بعد از به روزرسانی‌های متوالی با آموزش توسط مجموعه آموزشی متفاوت و ارزیابی بر اساس دسته آزمون تعمیم‌پذیری



شکل ۴-۸) نمودار نرخ هشدار نادرست سیستم (٪) بعد از به روزرسانی‌های متوالی با آموزش توسط مجموعه آموزشی متفاوت و ارزیابی بر اساس دسته آزمون تعمیم‌پذیری



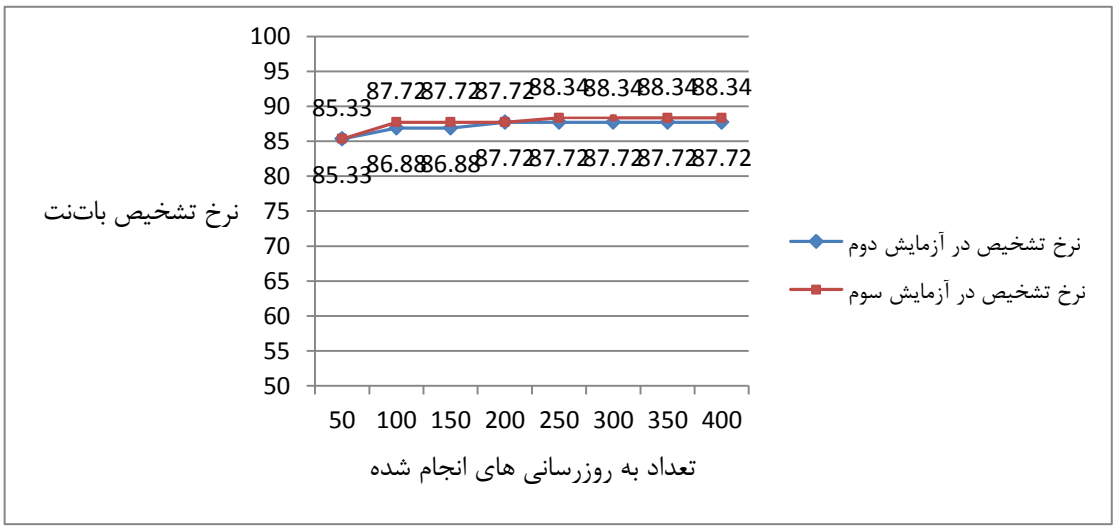
شکل ۴-۹) نمودار نرخ تشخیص درست سیستم (٪) بعد از به روزرسانی‌های متوالی با آموزش توسط مجموعه آموزشی متفاوت و ارزیابی بر اساس دسته آزمون تعمیم‌پذیری

در نمودار شکل ۴-۷، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۸ نیز، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم را نشان می‌دهد و محور عمودی نرخ هشدار نادرست سیستم بعد از هر یک از این به روزرسانی‌ها می‌باشد. در نمودار شکل ۴-۹ ، محور افقی تعداد بسته‌های داده هزارتایی وارد شده به سیستم که در واقع معادل تعداد به روزرسانی‌های انجام شده می‌باشد را نشان می‌دهد و محور عمودی نرخ تشخیص درست سیستم (باتنت و سالم) بعد از هر یک از این به روزرسانی‌ها می‌باشد

همانطور که در نمودارهای شکل ۴-۵و ۴-۶ مشاهده می‌شود، مقدار نهایی نرخ تشخیص باتنت و نرخ هشدار نادرست و نرخ تشخیص درست (باتنت و سالم) در این آزمایش به ترتیب ۸۸.۳۴٪ ، ۱۲.۳۳٪ و ۷۴.۱۲٪ شده است؛ که نمایانگر افزایش در نرخ تشخیص و عدم تغییردر نرخ هشدار نادرست، نسبت به آزمایش قبل می‌باشد.

با مقایسه این نتایج، با نتایج آزمایش دوم، که در آن باتنت‌های نوع جدید وارد سیستم نشده بودند تا در یادگیری افزایشی مورد استفاده قرار گیرند، مشاهده می‌شود که دقت عملکرد سیستم در آزمایش سوم افزایش پیدا کرده است. این موضوع بیان‌گر این است که سیستم قادر بوده است از داده‌های جدیدی که جهت آزمون و شناسایی وارد سیستم شده بودند، برای یادگیری افزایشی و به روزرسانی خود بهره برده و با ذخیره تعدادی از آن‌ها در مجموعه آموزشی پویای خود، دقت عملکرد را در شناسایی این نوع جدید در دفعات بعدی (که در این آزمایش معادل با مرحله ارزیابی با دسته آزمون تعمیم‌پذیری است) افزایش دهد.

برای مشاهده واضح‌تر تفاوت این دو حالت، نمودارهای نرخ تشخیص بات‌نت در آزمایش دوم و سوم، در نمودار شکل ۴-۱۰ ، در کنار یکدیگر قرار گرفته‌اند. همانطور که مشاهده می‌شود، در به روزرسانی ۵۰ ام و ۵۰ ام دقت‌ها یکسان بوده، اما از به روزرسانی ۵۰ام به بعد، دقت در آزمایش سوم، به دلیل اعمال بات‌نت‌های جدید به سیستم و بهره‌گیری از آن‌ها جهت به روزرسانی مجموعه آموزشی، افزایش یافته است.



شکل ۴-۱۰)مقایسه دو نمودار نرخ تشخیص بات‌نت (٪) در آزمایش دوم و سوم

با توجه به نتایج حاصل از تمامی آزمایش‌ها، می‌توان نرخ تشخیص و نرخ هشدار نادرست نهایی سیستم را به ترتیب ۸۷.۷۲٪ و ۱۲.۳۳٪ اعلام نمود.

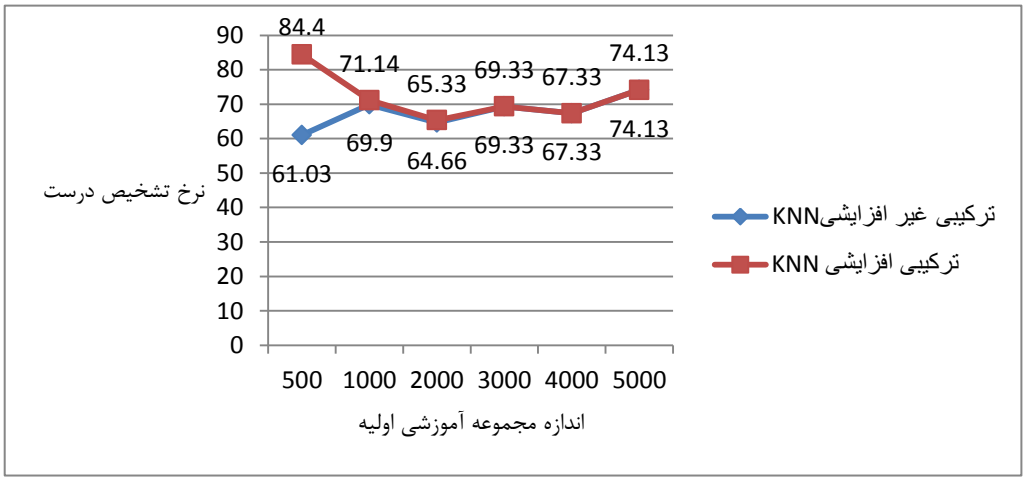
۴-۳- مقایسه

در این بخش، دقت عملکرد سیستم طراحی شده با سه حالت ممکن دیگر مقایسه می‌گردد. لازم به ذکر است که همگی این مقایسه‌ها، به جز آخرین مقایسه، بر اساس نرخ تشخیص درست (باتنت و سالم)که در این پژوهش بر اساس رابطه ۴-۳ مطرح شد، انجام شده است. در آخرین مقایسه، که سیستم پیشنهادی با سیستم ارائه شده در پژوهش [۹] مقایسه می‌گردد، از نرخ تشخیص باتنت استفاده شده است، زیرا در این پژوهش و نیز اغلب پژوهش‌های مشابه دیگر، از رابطه ۴-۱ جهت اعلام نرخ تشخیص استفاده نموده‌اند.

حالت اول زمانی است که یادگیری افزایشی وجود نداشته باشد و تنها از رده‌بند ترکیبی KNN استفاده شود. حالت دوم، مشابه سیستم ارائه شده است، با این تفاوت که در آن به جای استفاده از یادگیری ترکیبی مبتنی بر KNN، تنها از یک رده‌بند KNN استفاده شده است. حالت سوم نیز کاملا مشابه با سیستم ارائه شده می‌باشد، اما در قسمت تعیین صلاحیت، شرط محدودیت بازه‌ی تغییرات وارایانس داده‌های مجموعه پویا اعمال نمی‌شود. در نهایت نیز نتایج حاصل با نتایج ارائه شده در پژوهش[۹]، مقایسه می‌گردد. در پژوهش [۹] نیز جهت تشخیص باتنت‌ها از مجموعه داده باتنت ISCX استفاده شده است و نتایج قابل قبول و معتبرتری نسبت به سایر پژوهش‌های مشابه حاصل شده است.

۴-۳-۱- مقایسه با سیستم با یادگیری ترکیبی غیر افزایشی

مقایسه عملکرد سیستم پیشنهادی، با حالتی که در آن یادگیری افزایشی وجود ندارد و تنها از رده‌بند ترکیبی KNN استفاده می‌شود، به این صورت باید انجام شود که دقت نهایی به دست آمده توسط سیستم پیشنهادی، با دقتی که از طریق یادگیری ترکیبی بر اساس مجموعه آموزشی اولیه‌ی ایستا حاصل می‌شود مقایسه گردد. برای این منظور، برای مجموعه آموزشی اولیه اندازه‌های مختلفی در نظر گرفته شده و هربار سیستم پیشنهادی با مجموعه آموزشی اولیه با اندازه‌های مختلف اجرا می‌شود و سپس دقت نهایی با دقت حاصل از اجرای یادگیری ترکیبی مبتنی بر KNN، بر روی این مجموعه‌های اولیه مقایسه می‌گردد. نمودار شکل۴-۱۱، نتایج حاصل را نمایش می‌دهد. در این‌جا معیار نرخ تشخیص به عنوان معیار دقت در نظر گرفته شده است.

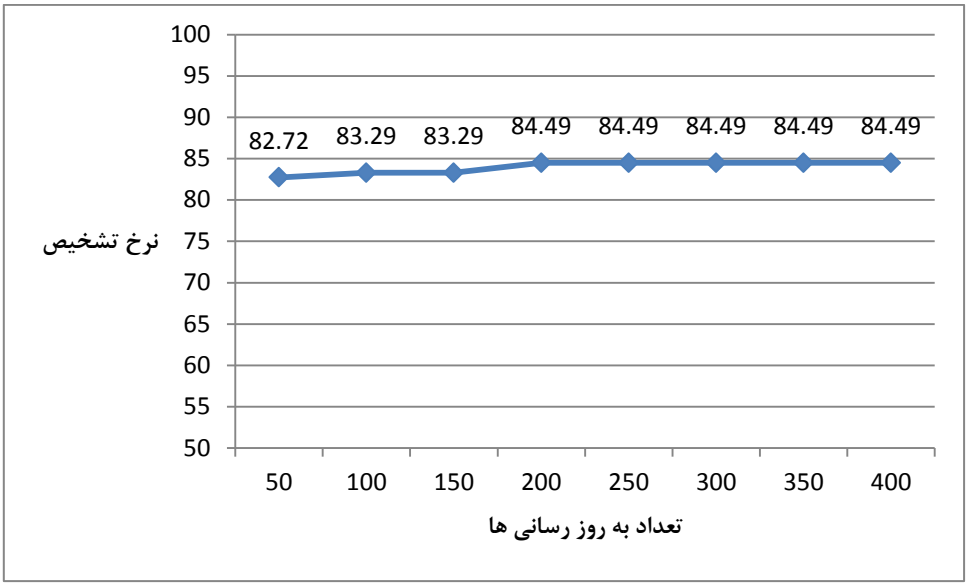


شکل ۴-۱۱) نمودار مقایسه نرخ تشخیص (٪) در دوحالت یادگیری افزایشی و یادگیری بر اساس KNN غیرافزایشی

همان‌طور که در شکل ۱۱-۴ مشاهده می‌شود، زمانی که یک مجموعه آموزشی اولیه با اندازه کوچک (در اینجا ۵۰۰و۱۰۰۰و۲۰۰۰) در اختیار باشد، چنانچه تنها از مجموعه آموزشی اولیه ایستا و به کمک الگوریتم ترکیبی مبتنی بر KNN، جهت تشخیص بات‌نت‌ها استفاده شود، نرخ تشخیص سیستم کمتر از حالتی خواهد شد که در آن از یادگیری افزایشی استفاده شود و مجموعه آموزشی به تدریج گسترش داده شود. بنابراین می‌توان مشاهده نمود که با بکارگیری آموزش افزایشی، سیستم پیشنهادی کمک می‌کند تا با داشتن یک مجموعه داده برچسب گذاری شده‌ی بسیار کوچک هم بتوان به دقت خوبی در تشخیص بات‌نت دست یافت.

۴-۳-۲- مقایسه با سیستم با یادگیری افزایشی غیرترکیبی

در این بخش عملکرد سیستم در حالتی که در آن به جای استفاده از یادگیری ترکیبی و رده‌بندی بر اساس رای‌گیری میان نتیجه سه رده‌بند KNN، از یک رده‌بند KNN ساده استفاده می‌شود، مقایسه می‌گردد. در این حالت با توجه به جدول ۴-۴، مقدار K=۹ انتخاب می‌شود. مقایسه میان نرخ تشخیص در این حالت با سیستم پیشنهادی در نمودار شکل ۴-۱۲ قابل مشاهده می‌باشد.



شکل ۴-۱۲) نمودار نرخ تشخیص سیستم (٪) با یادگیری غیر ترکیبی بعد از به روزرسانی‌های متوالی با ارزیابی بر اساس دسته آزمون تصادفی

با مقایسه نمودار شکل ۴-۱۲ و نمودار ۴-۱، می‌توان مشاهده نمود که استفاده از یادگیری افزایشی ترکیبی منجر به افزایش دقت تشخیص می‌شود. یادگیری افزایشی غیرترکیبی در این آزمایش در نهایت به نرخ تشخیص ۸۴.۴۹٪ می‌رسد، در حالی که این مقدار در یادگیری افزایشی ترکیبی ۹۰.۳۳٪ می‌باشد.

۴-۳-۳- مقایسه با سیستم بدون شرط محدودیت تغییرات واریانس مجموعه پویا

از آن‌جایی که تا کنون برای تشخیص باتنت سیستمی با رویکرد یادگیری افزایشی ارائه نشده است، سیستم مناسبی جهت مقایسه با روش پیشنهادی در این پژوهش در دست نیست. در نتیجه این روش پیشنهادی با سیستم خودفراگیر ارائه شده در [۴۰] که برای رده‌بندی ترافیک شبکه بر اساس پروتکل‌های کاربردی پیشنهاد شده است، مقایسه می‌شود. این رده‌بند، نسبت به رده‌بند

طراحی شده در این پایان‌نامه عملکرد ساده‌تری داشته و به علاوه دارای سربار ذخیره‌سازی بیشتری می‌باشد.

نحوه انتخاب داده‌ها برای افزوده شدن به مجموعه آموزشی پویا در رده‌بند ارائه شده در [۴۲] مشابه عملکرد تابع Qualification در این پایان‌نامه می‌باشد، اما با این تفاوت که شرط آخر، یعنی تعیین محدوده مجاز تغییرات واریانس داده‌های محموعه پویا، اعمال نمی‌شود. اعمال این شرط در پژوهش حاضر با هدف صرفه‌جویی در حافظه مصرفی، در عین حفظ دقت نهایی می‌باشد.

در این بخش هدف مقایسه تعداد داده‌های ذخیره شده در مجموعه آموزشی پویا بعد از هر به روزرسانی در دو حالت اعمال (معادل عملکرد پژوهش حاضر) و عدم اعمال (معادل عملکرد پژوهش [۴۲]) شرط محدودیت تغییرات واریانس مجموعه پویا می‌باشد. زیرا همان‌گونه که پیش از این بیان شد، یکی از مزیت‌های اعمال این شرط کاهش میزان حافظه مصرفی می‌باشد. این کاهش زمانی با ارزش است که دقت به دست آمده در حالتی که داده‌ها بدون در نظر گرفتن این محدودیت به مجموعه اضافه شوند با دقتی که با وجود این محدودیت به دست می‌آید برابر باشد؛ اما با اعمال این محدودیت، تعداد داده‌های ذخیره شده بسیار کمتر شده باشند.

برای این منظور سیستم در هر دو شرایط (با اعمال شرط و بدون اعمال شرط) اجرا می‌شود. در جدول ۴-۴ تعداد داده‌ی منتخب جهت افزوده شدن به مجموعه پویا در هر دو حالت، در چند به‌روزسانی متناظر نشان داده شده است. باید توجه داشت که در هر دو شرایط نرخ نهایی تشخیص سیستم پس از ۲۰۰ مرتبه به‌روزرسانی، ۸۴٪ است. در این آزمایش‌ها از الگوریتم رده‌بندی KNN با K= ۹ استفاده شده است.

جدول ۴-۴- مقایسه تعداد داده‌های ذخیره شده در دو حالت اعمال و عدم اعمال شرط محدودیت تغییر واریانس

تعداد داده‌های افزوده شده به مجموعه پویا در هر به روز رسانی (با اعمال شرط محدودیت تغییر واریانس)	تعداد داده‌های افزوده شده به مجموعه پویا در هر به روز رسانی (بدون اعمال شرط محدودیت تغییر واریانس)
۱. ۲۰۳	۱. ۶۵۰
۲. ۲۳۵	۲. ۶۶۱
۳. ۲۱۷	۳. ۶۵۶
۴. ۲۲۱	۴. ۶۶۶
۵. ۲۲۸	۵. ۶۵۹
۶. ۲۲۱	۶. ۶۷۹
۷. ۲۰۶	۷. ۶۶۹
۸. ۲۱۲	۸. ۶۶۵
۹. ۲۴۱	۹. ۶۹۲
۱۰. ۲۳۰	۱۰. ۶۵۲
۱۱. ۲۳۴	۱۱. ۶۹۰
۱۲. ۲۶۳	۱۲. ۶۹۰
۱۳. ۲۲۰	۱۳. ۶۷۶
۱۴. ۲۳۴	۱۴. ۶۳۹
۱۵. ۲۵۵	۱۵. ۶۵۴
۱۶. ۲۶۱	۱۶. ۶۶۰
۱۷. ۲۳۶	۱۷. ۶۷۰

با مقایسه مقادیر جدول ۴-۵ در هر دو حالت می‌توان دریافت که اعمال این شرط منجر به کاهش چشم‌گیر تعداد داده‌های ذخیره شده می‌شود، و این در حالی است که نرخ تشخیص ثابت مانده است. در نتیجه با اعمال این شرط، سیستم توانسته است با ذخیره فقط تعداد کمی داده‌ی جدید و تقریبا یک سوم حالت عادی، به دقت خوبی در تشخیص برسد و در حافظه مصرفی کاهش چشم‌گیری ایجاد شود.

۴-۳-۴- مقایسه با پژوهش مشابه

همانطور که در بخش ۲-۲ در مورد مقایسه‌ی دقت گزارش شده در پژوهش [۹] با سایر پژوهش‌های انجام شده بیان شد، نتایج سیستم ارائه شده در این پزوهش نیز به دلیل استفاده از مجموعه داده‌ی جامع جهت ارزیابی، نمی‌توانند با اغلب سیستم‌های طراحی شده در این زمینه مقایسه گردد. بنابراین در این بخش نتایج حاصل شده با نتایج گزارش شده در پژوهش [۹] که از همین مجموعه داده استفاده نموده است، مقایسه می‌گردد. در پژوهش [۹]، بالاترین دقت گزارش شده جهت شناسایی بات‌نت‌ها ۷۵٪ می‌باشد. سیستم ارائه شده در این پایان‌نامه، با بکارگیری یادگیری افزایشی و به‌روزرسانی، با ارزیابی بر اساس مجموعه داده‌ی شامل انواع بات‌نت‌های جدید به دقت ۸۸٪ دست یافت. این موضوع بیانگر تاثیر به‌سزای یادگیری افزایشی بر میزان قابلیت تعمیم‌پذیری سیستم می‌باشد.

فصل پنجم

نتیجه‌گیری و پژوهش‌های آینده

۵-۱- نتیجه‌گیری

در این پژوهش با تمرکز بر مساله‌ی پویا بودن محیط واقعی اینترنت و ضرورت وجود سیستم‌های تشخیص بات‌نت با قابلیت سازگاری با چنین محیطی، یک سیستم تشخیص بات‌نت با یادگیری افزایشی طراحی شد که قابلیت به روزرسانی خود در زمان بهره‌برداری را دارا می‌باشد.

برای این منظور از یک رده‌بند ترکیبی مبتنی بر kNN استفاده شد. به منظور بالا بردن قابلیت اطمینان نتایج گزارش شده ازدقت این سیستم، پیاده‌سازی و ارزیابی این سیستم به کمک یکی از جامع‌ترین مجموعه داده‌های موجود در این زمینه انجام گرفت.

با مقایسه‌ی نتایج به دست آمده در این پژوهش با نتایج گزارش شده در پژوهش‌هایی که جهت ارزیابی سیستم خود از مجموعه داده‌ی قابل مقایسه‌ای با مجموعه داده‌ی به‌کارگرفته شده در این پژوهش، استفاده نموده‌اند، مشاهده می‌شود که سیستم پیشنهادی، با ذخیره تعدادی از داده‌های جدید وارد شده به سیستم و به روزرسانی رده‌بندهای خود بر اساس این داده‌ها، قادر به شناسایی انواع زیادی از بات‌نت‌ها با نرخ تشخیص بالاتری می‌باشد و نسبت به روش‌های مشابه، ظرفیت بیشتری در تشخیص بات‌نت‌های جدید و مشاهده نشده دارد.

بنابراین، به دلیل بهره‌گیری از داده‌های جدید وارد شده به سیستم و یادگیری افزایشی برخط در این سیستم، می‌توان نتیجه گرفت سیستم مورد نظر نسبت به سیستم‌های مشابه موجود تطابق بیشتری با محیط واقعی و پویا داشته و از توانایی بالایی در تصمیم‌گیری در زمان مواجه با انواع جدید بات‌نت‌ها برخوردار می‌باشد.

۵-۲- پژوهش‌های آینده

با توجه به اینکه دسته‌بندی سیستم ما بر مبنای kNN است، پیچیدگی محاسباتی بازسازی رده‌بند سیستم به شدت به اندازه‌ی مجموعه‌ی آموزشی وابسته می‌باشد. هم‌چنین با توجه به روند الگوریتم، چنان‌چه افزودن داده‌های جدید به مجموعه پویا بدون کنترل ادامه پیدا کند، مجموعه پویا می‌تواند به صورت نامحدودی بزرگ شود. در آزمایش‌های انجام شده در این پژوهش، حدود ۴۰۰۰۰۰ داده به سیستم اعمال می‌شود که کمتر از ۸۰۰۰۰ آن‌ها به مجموعه پویا افزوده می‌گردند و بنابراین کارایی سیستم به طور قابل توجهی کاهش نمی‌یابد. اما قبل از به کارگیری این سیستم در دنیای واقعی، لازم است این مساله مورد توجه قرار گیرد و در واقع برای حداکثر اندازه‌ی ممکن برای مجموعه پویا محدودیتی تعیین کرد. به این ترتیب مجموعه بیش از حد بزرگ نشده و سیستم نیز کارایی خود را پس از مدتی از دست نخواهد داد.

حداکثر اندازه مناسب برای مجموعه پویا می‌تواند به صورت تجربی و بر اساس کارایی و سرعت عمل سیستم با ابعاد مختلف این مجموعه تعیین شود؛ و زمانی که اندازه مجموعه در طول اجرا به این حد تعیین شده برسد تعدادی از داده‌ها از این مجموعه حذف گردند. بهتر است این داده‌های منتخب برای حذف شدن از میان داده‌های اولیه مجموعه، که دارای برچسب واقعی بوده‌اند، نباشد.

برای این منظور پیشنهاد می‌شود این داده‌ها بر اساس زمان مورد استفاده قرار گرفتن انتخاب گردند. به این معنی که از میان داده‌ها، تعدادی داده که نسبت به سایر داده‌ها در رده‌بندی‌های قدیمی‌تری به عنوان نزدیکترین همسایه‌ها مورد استفاده قرار گرفته‌اند حذف گردند. بنابراین داده‌هایی که اخیرا بیشتر مورد استفاده بوده و نقش بیشتری در پیش‌بینی برچسب‌های داده‌های جدید داشته‌اند در مجموعه باقی خواهند ماند.

یکی از پیشنهادها برای بهبود تشخیص سیستم ارائه شده، به کارگیری روش‌های شناسایی با رویکردهای متفاوت‌تر در کنار روش شناسایی کنونی می‌باشد. به عنوان مثال می‌توان به صورت ترکیبی، علاوه بر استفاده از الگوریتم مبتنی بر KNN که در این پژوهش براساس فاصله اقلیدسی عمل می‌کند، از الگوریتمی هم‌چون بیزین ساده[۳۶] که دارای رویکرد احتمالاتی است نیز بهره برد. اما همان‌طور که در فصل چهارم بیان شد، الگوریتم بیزین ساده دقت خوبی نشان نداد. در راستای افزایش دقت، پیشنهاد می‌شود از روش‌های قدرتمندتری جهت گسسته‌سازی داده‌ها استفاده شود. هم‌چنین می‌توان از روش بیزین ساده با داده‌های پیوسته استفاده نمود.

یکی از ضعف‌های سیستم ارائه شده، افزایش نرخ هشدار نادرست در طول زمان می‌باشد. یکی از عوامل تاثیرگذار بر این امر می‌تواند ویژگی‌های مورد استفاده جهت شناسایی باشد که اگرچه نرخ تشخیص خوبی را ارمغان می‌آورد، اما در طول زمان با افزوده شدن انواع مختلف بات‌نت به مجموعه آموزشی، منجر به افزایش نرخ هشدار نادرست می‌گردد. انتظار می‌رود با بکارگیری ویژگی‌های بهتر بتوان این مساله را بهبود داد. هم‌چنین می‌توان از ویژگی‌های وزن‌دار استفاده نمود؛ به این معنی که ویژگی‌های موثرتر، دارای وزن و اهمیت بیشتری باشند.

علاوه بر موارد اشاره شده، همانطور که پیش‌تر بیان شد، بخش تعیین صلاحیت (qualification) نقش به سزایی در کارایی سیستم از نظر سرعت، دقت و میزان حافظه مصرفی ایفا می‌کند؛ بنابراین اعمال شرط‌ها و محدودیت‌های موثرتر در این مرحله، می‌تواند تاثیر چشم‌گیری در کارایی سیستم داشته باشد. به عنوان مثال، در شرط سوم موجود در این سیستم، بهتر است بازه‌ی مورد استفاده برای تعیین محدوده تغییرات واریانس ثابت نباشد و در طول زمان با توجه به رشد مجموعه داده، متغیر باشد. زیرا در حالت کلی با افزایش تعداد نمونه‌ها، تغییرات واریانس کاهش می‌یابد، بنابراین بهتر است به تدریج بازه‌ی مورد نظر کوچک‌تر شود.

[36] Naïve Bayes

منابع

[1] Jose, S. (2008), “WAN and application optimization solution guide”, *Cisco Systems, Inc*, chapter 5.

[2] Zhang, J. and Chen, Ch. And Xiang, Y. and Zhou, W. (2013), “Robust network traffic identification with unknown applications”, ACM SIGSAC symposium on Information, computer and communications security, 8, pp 405-414.

[3] Yahyazadeh M. and Abadi M, (2015) “BotGrab: A negative reputation system for botnet detection,” *Computer and Electrical Engineering*, 41, pp 68-85.

[۴] عزمی، ر. و قلی‌نژاد، م. و صابری، م. (۱۳۹۴)، " تشخیص بات‌نت برای شبکه‌های نظیر به نظیر، " *مجله علمی-پژوهشی پدافند الکترونیکی و سایبری*، سال سوم، شماره۴، دوره ۳، ص ۶۰-۴۳.

[5] Liu, J. and Xiao, Y. and Ghaboosi, K. and Deng, H. and Zhang, J. (2009), “Botnet: Classification, Attacks, Detection, Tracing, and Preventive Measures,” *EURASIP Journal on Wireless Communications and Networking*, 9, pp 1-11.

[6] Wang P, Sparks S, Zou CC. (2010), “An advanced hybrid peer-to-peer botnet,” *IEEE Transactions on Dependable and Secure Computing*,7, 2, pp 113–127.

[7] Mahmoud, M. and Nir, M. and Matrawy, A. (2015), “A Survey on Botnet Architectures, Detection and Defences,” *International Journal of Network Security*, 17, 3, pp 272-289.

[8] Garcia, S. and Grill, M. and Stiborek, J. and Zunino, A. (2014), “An empirical comparison of botnet detection methods,” *Computers & Security*, 45, pp 100-123.

[9] Biglar Beigi, E. and Hadian Jazi, H. and Stakhanova, N. and Ghorbani, A. A. (2014), “Towards effective feature selection in machine learning-based botnet detection approaches,” *IEEE Conference on Communications and Network Security (CNS)*, pp 247-255, San Francisco, CA, USA.

[10] Haddadi, F. and Nur Zincir-Heywood, A. (2015), “Botnet Detection System Analysis on the Effect of Botnet Evolution and Feature Representation,” *Conference on Genetic and Evolutionary Computation*, pp 893-900, Madrid, Spain.

[11] Yu, X. and Dong, X. and Yu, G. and Qin, Y. and Yue, D. and Zhao, Y. (2010), “Online Botnet Detection Based on Incremental Discrete Fourier Transform*,” Journal of Networks*, 5, 5, pp 568-576.

[۱۲] یحیی‌زاده م، (۱۳۹۰)، پایان‌نامه ارشد: " روشی مستقل از ساختار و پروتکل فرمان و کنترل برای تشخیص بات‌نت‌ها"، دانشکده مهندسی برق و الکترونیک، دانشگاه تربیت مدرس.

[13] Li, W. and Abdin, K. and Dann, R. and Moore, A. (2006), “Approaching real-time network traffic classification,” *Technical report RR-06-12*, Department of Computer Science, Queen Mary, University of London.

[۱۴] یحیی‌زاده م، و شریف‌نیا ر، (۱۳۹۱)، "بات‌نت‌ها: انواع، چرخه حیات و روش‌های تشخیص"، *آزمایشگاه تخصصی آپا در حوزه امنیت سرویس‌های شبکه و تجهیزات بی‌سیم*، دانشکده فنی و مهندسی، دانشگاه فردوسی مشهد.

[15] Li, X. and Duan, H. and Liu, W. and Wu, J. (2009), “Understanding the construction mechanism of botnets,” *Symposia and Workshops on Ubiquitous Autonomic and Trusted Computing*, pp 508-512, United State.

[16] Tsiatsikas, Z. and Anagnostopoulos, M. and Kambourakis, G. and Lambrou, S. and Geneiatakis, D. (2015), "Hidden in plain sight. SDP-based covert channel for botnet communication*,"International Conference on Trust, Privacy & Security in Digital Business*, pp 48-59, Valencia, Spain.

[17] Silva, s. and Silva, R. and Pinto, R. and Salles, R. (2013), "Botnets: A survey**",**Computer Networks, 2, 57, pp. 378 – 403.

[18] Zhuge, J. and Han, X. and Guo, J. and Zou, W. and Holz, T. and Zhou, Y. (2007), "Characterizing the IRC-based botnet phenomenon," Peking University & University of Mannheim Technical Report.

[19] Wang, P. and Wu, L. and Aslam, B. and Zou, C.C. (2009) "A systematic study on peer-to-peer botnets," *International Conference on Computer Communications and Networks*, pp. 1–8, San Francisco, CA.

[20] Wang, P. and Sparks, S. and Zou, C.C. (2010), "An advanced hybrid peer-to-peer botnet," *Dependable and Secure Computing*, 7, 2, pp. 113–127.

[21] Rodriguez-Gomez, R. and Macia-Fernandez, G. and Garcia-Teodoro, P. (2012), "Survey and taxonomy of botnet research through life-cycle," *ACM Computing Surveys (CSUR)*, 45, 4, pp 1-33.

[22] Feily, M. and Shahrestani, A. and Ramadass, S. (2009), "A survey of botnet and botnet detection," International Conference on Emerging Security Information, Systems and Technologies, pp 268-273, Athens/Glyfada, Greece.

[23] Zhu, Z. and Lu, G. and Chen, Y. and Fu, Z.J. and Roberts, P. and Han, K. (2008), "Botnet Research Survey," *IEEE International Conference on Computer Software and Applications*, pp.967–972, Turku, Finland.

[24] Lu, W. and Rammidi, G. and Ghorbani, A. (2011), "Clustering Botnet Communication Trace Based on N-gram Feature Selection", *Computer Communications*, 34, 3, pp 502–514.

[25] Stevanovic, M. and Pederson, J.M. (2013),"Machine learning for identifying botnet network traffic", Aalborg University, Denmark.

[26] Abdullah, R. S. and Abdullah, F. M. and Noh, Z. A. and Masud, M. Z. and Selamat, S. R. and Yusof, R. (2016), "Revealing the Criterion on Botnet Detection Technique," *International Journal of Computer Science*, 13, 3.

[27] Yin, C. and Zhang, S. and Yin, Z. and Wang, J. (2015), "An Algorithm of Clustering by Density Peaks Using in Anomaly Detection," *International Journal of Security and Its Applications*, 9, 12, pp 115-128.

[28] Cherubin, G. and Nouretdinov, I. and Gammerman, A. and Jordaney, R. and Wang, Z. and Papini, D. and Cavallaro, L. (2015), "Conformal Clustering and Its Application to botnet *Traffic," Statistical Learning and Data Science, Lecture Notes in Computer Science*, pp 313-322.

[29] Lu, W. and Rammidi, G. and Ghorbani, A. A. (2011), "Clustering botnet communication traffic based on n-gram feature selection," *Computer Communications*, 34, 3, pp 502-514.

[30] Stevanovic, M. and Pedersen, J. M. (2015), "An analysis of network traffic classification for botnet detection," *International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, pp 1-8, London, United Kingdom.

[31] Raghava, N. S. and Sahgal, D. and Chandna, S. (2012), "Classification of Botnet Detection Based on Botnet Architechture," *International Conference on Communication systems and Network Technologies (CSNT)*, pp 569-572, Gujarat, India.

[32] Stevanovic, M. and Pedersen, J. M. (2014), "An efficient flow-based botnet detection using supervised machine learning," *International conference on Computing, Networking and Communications (ICNC)*, pp 797-801, Honolulu, Hawaii, USA.

[33] Aviv, A. j. and Haeberlen, A. (2011), "Challenges in experimenting with botnet detection systems," *Conference on Cyber Security experimentation and test*, pp 6-6, San Francisco, CA.

[34] Tavallaee, M. and Stakhanova, N. and Ghorbani, A. A. (2010), "Toward Credible Evaluation of Anomaly-Based Intrusion-Detection Methods," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40, 5, pp 516-524.

[35] Garcia, S. and Zunino, A. and Campo, M. (2013), "Survey on network-based botnet detection methods," *Security and Communication Networks*, 7, 5, pp 878-903.

[36] Karim, A. and Salleh, R. and Khurram Khan, M. and Siddiqa, A. and Raymond Choo, K. (2016), "On the analysis and detection of mobile botnet applications," Journal of Universal Computer Science, 22, 4, 567-588.

[37] Zhao, D. and Traore, I. and Sayed, B. and Lu, W. and Saad, S. and Ghorbani, A. and Garant, D. (2013), "Botnet detection based on traffic behavior analysis and flow intervals," *Computers & Security*, 39, pp 2-16.

[38] Li, Z. and Blaich, A. and Striegel, A. (2010), "Fighting botnets with economic uncertainty," *Security and Communication Networks*, 4, 10, pp 1104–1113.

[39] Yahyazadeh, M. and Abadi, M. (2012), "BotOnus: an Online Unsupervised Method for Botnet Detection," The ISC Int'I Journal of Information Security, 4, 1, pp 51-62.

[40] Yu, X. and Dong, X. and Yu, G. and Qin, Y. and Yue, S. and Zhao, Y. (2010), "Online Botnet Detection Based on Incremental Discrete Fourier Transform," *Journal of Networks*, 5, 5, pp 568-576.

[41] Hyslip, T. S. and Pittman, J. M. (2015), ",A survey of botnet detection techniques by command and control infrastructure *" the Journal of Digital Forensics, Security and Law(JDFSL)*, 10, 1, pp 7-26.

[42] Divakaran, D. M. and Su, L. and Liau Y. S. and Thing V.L.L. (2015), "SLIC: Self-Learning Classifier for network traffic," *Computer Networks*, 91, pp 283-2.

Abstract

The increasing growth of Internet usage, has considerably motivated attackers to develop cybercrime. In recent decade, network security and network traffic monitoring have been significantly important due to these expanded new threats. In general, traffic flows can occur due to two main reason; malicious purpose and attacks, or benign purpose. Traffic flows are classified regarding their purposes. Botnets have been recently recognized as the most formidable threats on the Internet. Different approaches have been proposed for detecting these types of attacks; the most effective approaches are based on machine learning. One of the main reasons for the trend towards these methods is their strength of generalization to identify new types of botnets.

Because of the importance of botnets in the recent decade, in this research, a self-learning botnet detection system consisting of incremental learning, has been proposed based on traffic classification. This system classifies traffic flows according to their application; botnet or benign. An incremental training is conducted and the system updates its classifier continuously, regarding the new samples to obtain more capacity of generalization. In addition to pursue the learning process, like the other online methods, this system is capable of using the new incoming samples in its classifier, without knowing their real label; this is because it can predict the labels fairly precisely. Moreover, in order to achieve a valid evaluation of the system performance, which is rarely found in previous studies, the system has been evaluated with a comprehensive data set that has a wide variety of botnets. The experiments results and comparisons demonstrate that the proposed system can perform properly in a dynamic environment. The Maximum improvement rate which is provided by this system is 13% in botnet detection rate.

Keywords: traffic classification, machine learning, botnet detection, incremental learning

Shahrood University of Technology
Faculty of Computer Engineering and IT
MSc Thesis in Computer Engineering Artificial Intelligence and Robotics

Automated application based classification and processing of network traffic using hybrid learning methods

By: Mahsa Nazemi Gelian

Supervisor:
Dr. Hoda Mashayekhi

September 2016