

علمی-پژوهشی

تشخیصی شبکه‌بات نظیر به نظیر با استفاده از روش یادگیری عمیق

مهدی اسدی^۱، سعید پارسا^{۲*}، محمدعلی جبرئیل جمالی^۳، وحید مجیدنژاد^۴

۱- دانشجوی دکتری، گروه مهندسی کامپیوتر، واحد شبستر، دانشگاه آزاد اسلامی، شبستر، ایران، ۲- دانشیار، گروه مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران، ۳ و ۴- استادیار، گروه مهندسی کامپیوتر، واحد شبستر، دانشگاه آزاد اسلامی، شبستر، ایران

(دریافت: ۹۸/۲/۳، پذیرش: ۹۸/۷/۱۰)

چکیده

یک شبکه‌بات، شبکه‌ای از رایانه‌های آلوده و دستگاه‌های هوشمند بر روی اینترنت است که توسط مدیرات بدافزار از راه دور کنترل می‌شود تا فعالیت‌های بدخواهانه مختلفی نظیر اجرای حملات منع خدمات، ارسال هرزنامه، سرقت کلیک و غیره را انجام دهند. زمانی که مدیرات با بات‌های خود ارتباط برقرار می‌کند، ترافیکی تولید می‌کند که تجزیه و تحلیل این ترافیک برای شناسایی ترافیک شبکه‌بات می‌تواند یکی از عوامل تاثیر گذار برای سامانه‌های تشخیص نفوذ باشد. در این مقاله، روش یادگیری عمیق با حافظه کوتاه مدت ماندگار (LSTM) جهت طبقه‌بندی فعالیت‌های شبکه‌بات نظیر به نظیر پیشنهاد می‌شود. رویکرد پیشنهادی بر اساس ویژگی‌های بسته‌های پروتکل کنترل انتقال بوده و کارایی روش با استفاده از دو مجموعه داده ISCX و ISOT ارزیابی می‌شود. نتایج آزمایش‌های انجام یافته، توانایی بالای رویکرد پیشنهادی برای شناسایی فعالیت‌های شبکه‌بات نظیر به نظیر را بر اساس معیارهای ارزیابی نشان می‌دهد. روش پیشنهادی نرخ دقت ۹۹/۶۵ درصد، نرخ صحت ۹۶/۳۲ درصد و نرخ بازخوانی ۹۹/۶۳ درصد را با نرخ مثبت کاذب برابر ۰/۶۷ ارائه می‌کند.

کلیدواژه‌ها: شبکه‌بات، تشخیص شبکه‌بات، یادگیری عمیق، شبکه عصبی بازگشتی، حافظه کوتاه مدت ماندگار

۱- مقدمه

اینترنت یک کانال عمومی است که به کاربران اجازه می‌دهد تا با یکدیگر ارتباط برقرار کنند. خدمات برخط^۱ با مهاجمان مواجه می‌شوند و از مهم‌ترین مسائل امنیتی الکترونیکی، نفوذ به دستگاه‌ها است. بسیاری از دستگاه‌ها، از جمله ادارات دولتی و سازمان‌های تجاری، از این مشکل متضرر می‌شوند؛ بنابراین، همه آن‌ها بر روی افزایش سطح امنیت دستگاه‌های خود به جهت نگرانی‌های خود در مورد مسائل امنیتی فعالیت می‌کنند. زیرساخت اطلاعات برای پشتیبانی از فعالیت‌های بحرانی در دستگاه‌های بزرگ مانند مخابرات و سامانه‌های بانکی، امری ضروری است. نفوذ توسط ابزارهای مختلف امنیت، یک سامانه اطلاعاتی را به خطر می‌اندازد، در نتیجه جوامع را به طور قابل توجهی تهدید می‌کند. هدف از نفوذ مربوط به هر رویه‌ای است که تلاش می‌کند محرمانه بودن^۲، یکپارچگی^۳ و

دسترسی پذیری^۴ (CIA) منابع را تهدید کند [۱].

با رشد سریع اتصال کامپیوترها به شبکه‌های قابل دسترسی، انتظار مصونیت در مقابل نفوذ به شبکه برای دستگاه‌های کامپیوتری غیرممکن است، لذا اقدامات اولیه و تشخیص برای جلوگیری از هرگونه آسیب زیادی بسیار با اهمیت است زیرا هیچ راه‌حل کاملی برای جلوگیری از وقوع نفوذ وجود ندارد. هدف ما ایجاد مدلی است که مجموعه‌ای از ویژگی‌های پروتکل کنترل انتقال را دریافت کرده و مشخص کند که آیا این ویژگی‌ها متعلق به یک شبکه معمولی و یا یک شبکه‌بات نظیر به نظیر^۵ است. مجموعه ویژگی‌های ورودی شامل خصیصه‌های پروتکل کنترل انتقال شبکه کاربری است و نتیجه خروجی نشان می‌دهد که دسترسی به یک شبکه به صورت عادی بوده و یا از طریق یک شبکه‌بات نظیر به نظیر صورت گرفته است.

مدل‌های شبکه‌عصبی و فن‌های خوشه‌بندی، روش هوش مصنوعی مؤثرتری برای بهبود شناسایی رفتارهای مخرب و

* رایانامه نویسنده پاسخگو: Parsa@iust.ac.ir

¹ Online

² Confidentiality

³ Integrity

⁴ Accessibility

⁵ Peer to Peer Botnet (P2P Botnet)

دهد. در همین حال، در مرحله تشخیص، شبکه حافظه کوتاه مدت ماندگار به عنوان یک ابزار طبقه بندی برای تشخیص نوع شبکه به کار می رود. روش پیشنهادی مجموعه آزمون^۶ را دریافت کرده و آن را با مجموعه‌ای از بردارهای ساخته شده (مجموعه آموزش) در مرحله اول مقایسه می کند.

طرح جدید به ما اجازه می دهد تعداد زیادی از داده های برچسب گذاری^۷ را به دست آوریم. این رویکرد سبب می شود تا از شبکه های عصبی عمیق برای انجام طبقه بندی استفاده شود که مزایای زیادی را نسبت به روش های سنتی حاصل می کند. در سال های اخیر، روش های یادگیری عمیق از لحاظ دقت پیش بینی بر روی انواع وظایف پیچیده و نیز طبقه بندی قابل توجهی که برای یک طرح تشخیص خطی لازم است، پیشرفته ترین نتایج را حاصل کرده اند. در مقابل بسیاری از روش های سنتی، شبکه های عصبی عمیق می توانند ویژگی های خود را به صورت خودکار و بدون تلاش مهندسی ویژگی ها توسط انسان، یاد گیرند. استفاده خودکار از ویژگی ها برای تشخیص شبکه بات دارای اهمیت است زیرا سازندگان بدافزار می توانند با استفاده از دانش ویژگی های مهندسی انسان، ترافیک شبکه بات را با ترافیک معمولی ترکیب کنند. بسیاری از مدل های سنتی باید از شروع کار توانایی این را داشته باشند که با اطلاعات جدید خود را به روز کنند. وزن شبکه های عصبی، پس از آن، می تواند به طور مداوم تنظیم شود؛ بنابراین یک شبکه عصبی را می توان برای تشخیص شبکه بات استفاده کرد و به طور هم زمان برای ترافیک برخط^۸ توسعه داد. در این مقاله با استفاده از شبکه عصبی بازگشتی (LSTM) مدل پایه ای ایجاد و آموزش داده می شود. همچنین برای ارزیابی مدل خود از مدل های یادگیری ماشین سنتی پیاده سازی شده (مانند روش درخت تصمیم گیری و روش جنگل تصادفی) و نیز روش ارائه شده در [۱۱] توسط آلتامان و همکاران و روش ارائه شده توسط اوبیدات [۱۲] با توجه به شناسایی شبکه های بات نظیر به نظیر و نیز به کارگیری مجموعه داده یکسان استفاده می کنیم.

ادامه این مقاله بر این اساس سازماندهی شده است که در بخش ۲ به طور خلاصه کارهای مرتبط ارائه شده است. سپس روش پیشنهادی در بخش ۳ توضیح داده شده است. نتایج تجربی در بخش ۴ ارائه گردیده و در نهایت نتایج و پیشنهادهایی برای کارهای آتی در بخش ۵ بیان شده است.

حملات تهاجمی در شبکه های کامپیوتری ارائه داده است [۱-۲]. یک شبکه بات به عنوان ابزار عمومی برای رسیدن به اهداف مهاجم محسوب می شود. به علاوه، این شبکه برای ارسال دستورات و دریافت نتایج کنترل شده توسط مدیر بات^۱ مورد استفاده قرار می گیرد. در این مقاله، چارچوب یادگیری عمیق برای شناسایی شبکه عادی و شبکه بات پیشنهاد می شود. روش های طبقه بندی متفاوتی برای تعیین نوع نفوذ استفاده می شود که برخی از آن ها عبارت اند از: تحلیل آماری [۳-۴]، تحلیل مبتنی بر قانون [۵]، داده کاوی [۶-۷] و شبکه عصبی [۸]. در تحلیل آماری، سامانه رفتار طبیعی و تکرار فعالیت ها را ثبت کرده و سپس آن ها را با اقدامات نفوذ مقایسه می کند تا تعیین کند که آیا این رفتارها عادی هستند و یا مخرب. شبکه عصبی یک مدل با ناظر^۲ یا بدون ناظر^۳ را از طریق آموزش سامانه رفتار نرمال و غیر نرمال به منظور ردیابی آن ها در آینده ایجاد می کند. در تحلیل مبتنی بر قانون، کارشناسان امنیتی کامپیوتر، مجموعه ای از قوانین را برای فعالیت های کامپیوتری امن و ناامن ایجاد می کنند، برای نمونه، شبکه بیزی هر داده ورودی جدید را با توجه به احتمال وقوع رفتارها و وقایع درون سامانه خودش طبقه بندی می کند. فن های داده کاوی از ویژگی های فیلدها برای ایجاد برچسب ها یا خوشه ها جهت تعیین نوع یا برچسب آیتم جدید داده استفاده می کنند. رویکرد پیشنهادی برای ارائه یک سامانه تشخیص از تحلیل رفتاری جریان شبکه [۹] با استفاده از شبکه حافظه کوتاه مدت ماندگار^۴ (LSTM) برای بهبود دقت تشخیص شبکه بات و یافتن بهترین ویژگی ها، استفاده می کند. مدل پیشنهادی شامل مرحله استخراج ویژگی و مرحله تشخیص است.

در مرحله استخراج ویژگی، روش پیشنهادی ابتدا الگوریتمی جهت جداسازی بسته های پروتکل کنترل انتقال از سایر بسته ها انجام می دهد. این کار به دلیل استفاده اغلب بات های نظیر به نظیر از این پروتکل جهت ارتباط خود با سایر نظیرها، به کار گرفته می شود. سپس بر روی مجموعه داده ها از الگوریتم درخت طبقه بندی و رگرسیون^۵ (CART) [۱۰] برای انجام فرآیند استخراج ویژگی از مجموعه داده های در دسترس استفاده می شود تا با حذف ویژگی های با تأثیرگذاری کم، ابعاد فضای جستجو را کاهش داده و سرعت تشخیص را افزایش دهد. شبکه یادگیری عمیق یک نسخه از بردارها و ویژگی های استخراج شده را دریافت می کند تا بردارهای ارائه شده جدید را به دست آورده و بهبود

¹ Botmaster

² Supervised

³ Unsupervised

⁴ Long Short Term Memory

⁵ Classification and Regression Tree

⁶ Test Set

⁷ Labelled Data

⁸ Online Traffic

۲- کارهای مرتبط

مبتنی بر گره که به مرحله آموزش تعلق دارند، بر اساس ویژگی‌های استخراج شده از بات‌های تعیین شده ساخته می‌شوند. روش‌های محاوره‌ای مبتنی بر رفتار غیرعادی شبکه از جمله، تعداد بسته‌های مبادله شده در محاوره و مدت زمان محاوره هستند. با توجه به اینکه ترافیک کانال فرمان و کنترل^۹ (C&C) معمولاً رفتار غیرعادی را تعیین نمی‌کند و نیز از رفتار ترافیکی معمول جدا نیست، در این موارد، فن‌های مبتنی بر محاوره ممکن است ناموفق باشند. روش‌های مبتنی بر کاوش، زمانی که به‌عنوان فن‌های یادگیری ماشین استفاده می‌شوند، برای استخراج الگوهای تصادفی شبکه مناسب هستند. روش‌های مبتنی بر امضا با توجه به اینکه از امضاهای کشف شده بات‌های پیشین و قدیمی جهت تشخیص استفاده می‌کند ممکن است برای شناسایی انواع جدیدی از شبکه‌های بات ناموفق باشند.

یکی دیگر از ابعاد بررسی روش تشخیص شبکه‌بات استفاده از طبقه‌بندی الگوریتم‌های تشخیص است [۱، ۲۵]. این الگوریتم‌ها عبارت‌اند از:

۱. یادگیرنده مبتنی بر دانش (از جمله K-نزدیک‌ترین همسایگی: KNN)
۲. بیزی ساده (NB)^{۱۰}،
۳. ماشین بردار پشتیبان (SVM)،
۴. درخت تصمیم‌گیری^{۱۱}،
۵. جنگل تصادفی^{۱۲}.

علاوه بر این، روش‌های ترکیبی طبقه‌بندی نیز آزمایش و ارزیابی شده‌اند. الگوریتم تقویتی^{۱۳} الگوریتمی است که از ماشین بردار پشتیبان^{۱۴} (SVM)، درخت تصمیم‌گیری و شبکه بیزی ساده استفاده می‌کند. در سال ۲۰۱۴، ژانگ و همکاران [۲۷] رویکردی برای بهبود قابلیت و کارایی عملکرد سامانه تشخیص شبکه بات را پیشنهاد کردند. این روش شامل دو مرحله اصلی است:

- (۱) تشخیص این‌که تمام دستگاه‌ها در ارتباطات نظیربه‌نظیر دخالت دارند و استخراج آثار آماری از ترافیک نظیربه‌نظیر.
- (۲) تجزیه و تحلیل ترافیک میزبان نظیربه‌نظیر برای طبقه‌بندی به‌عنوان میزبان سالم نظیربه‌نظیر و یا میزبان بات‌های نظیربه‌نظیر.

در سال‌های گذشته، ردیابی و تشخیص شبکه‌بات یکی از موضوعات اصلی تحقیقات زمینه امنیت دستگاه‌ها و شبکه‌های کامپیوتری بوده است. با این حال روش‌های مختلفی در تحقیقات موجود در [۲۳-۱۳] وجود دارد، بسیاری از روش‌ها نمی‌توانند شبکه‌بات را به‌طور مؤثر تشخیص دهند. کارهای اولیه در تشخیص شبکه‌بات، عمدتاً بر مبنای تجزیه و تحلیل داده^۱ بسته‌ها است که روشی برای بررسی و آزمودن امضاهای مخرب در پروتکل دیتاگرام کاربر^۲ (UDP) و بسته‌های پروتکل کنترل انتقال (TCP) است. روش‌های بررسی داده بسته‌ها معمولاً از روش‌های متمرکز تشخیص شبکه‌بات استفاده کرده و بسیار کند هستند زیرا نیاز به تجزیه بسته‌های داده بزرگ دارند. بات‌های جدید اغلب از رمزنگاری و سایر روش‌ها برای پنهان‌سازی ارتباط بین بات‌ها و بسته‌های ردوبدل شده استفاده می‌کنند. با توجه به محدودیت‌های روش‌های موجود، روش‌های تشخیص شبکه‌بات بر اساس تحلیل جریان مابین مدیر بات و بات‌ها پیشنهاد شده است [۲۴]. در مجموع روش‌های تشخیص شبکه‌بات را می‌توان در ۶ کلاس طبقه‌بندی کرد [۲۵]:

- ۱- تشخیص مبتنی بر جریان^۳،
- ۲- مبتنی بر منابع^۴،
- ۳- مبتنی بر گره^۵،
- ۴- مبتنی بر محاوره^۶،
- ۵- مبتنی بر کاوش^۷،
- ۶- مبتنی بر امضا^۸.

در روش‌های مبتنی بر جریان [۲۶] دو محدودیت اصلی وجود دارد. اولاً، جریان‌های مختلف بین هر دو گره شبکه باید تحلیل شود. با این حال، اغلب این جریان‌ها در یک شبکه نرمال و غیر مخرب ادغام می‌شوند. ثانیاً، ویژگی‌های جریان باید در زمان اجرا استخراج شوند که بیان می‌کند تحلیل مبتنی بر جریان، نیاز به هزینه‌های قابل توجه محاسباتی در زمان اجرا دارد. در هر نمونه مشخص، تعداد فراوانی از جریان‌ها در شبکه موجود است و این شرایط می‌تواند محدودیت فوق را بدتر کند. روش‌های مبتنی بر منابع در فاز آموزش ساخته می‌شوند که در آن مدل نرمال منابع غیر مخرب ممکن است شامل تمام شرایط نباشد. روش‌های

¹ Payload

² User Datagram Protocol

³ Flow-based

⁴ Recourse-based

⁵ Node-based

⁶ Conversation-based

⁷ Mining-based

⁸ Signuter-based

⁹ Command and Control Channel

¹⁰ Naive Bayes

¹¹ Decision Tree

¹² Random Forest

¹³ Boosting

¹⁴ Support vector machine

نرخ مثبت کاذب (FPR) بالا را تولید می‌کند. ونک‌تاش و همکاران [۳۱] روش شناسایی شبکه‌بات مبتنی بر پروتکل انتقال ابرمتن^۲ با استفاده از نرخ یادگیری تطبیقی با استفاده از شبکه‌های عصبی پیشرو چندلایه^۳ ارائه داده‌اند. برای تشخیص، ویژگی‌های مربوط به اتصال پروتکل کنترل انتقال در فواصل زمانی مشخص استخراج گردیده است. وانگ و همکاران [۳۲] یک روش تشخیص شبکه‌بات مبتنی بر رفتار^۴ بر پایه روش‌های تشخیص الگوی فازی را پیشنهاد کرده‌اند. در صورتی که این روش در زمان ردیابی شبکه با فعالیت‌های عادی شبکه منظم (به‌عنوان مثال، بررسی به‌روزرسانی نرم‌افزار جدید) مواجه گردد نرخ مثبت کاذب افزایش خواهد یافت.

هوانگ [۳۳] یک روش شناسایی شبکه‌بات مبتنی بر میزبان^۵ را بر اساس مدل خرابی شبکه^۶ طراحی کرد. شکست شبکه به‌عنوان عامل تفکیک‌ناپذیر برای ترافیک شبکه‌بات در نظر گرفته شده است، بدین معنی که شکست^۷ ناشی از گم‌شدگی نظیرها^۸، خدمات‌دهنده فرمان و کنترل و یا اهداف حمله است. ویژگی‌های استخراج شده از جریان‌های شکست به ۶ کلاس طبقه‌بندی شده‌اند. هوانگ مدلی معرفی کرد که می‌تواند بات‌ها را با دقت ۹۹٪ تشخیص دهد؛ با این حال، در مواردی که بات‌ها موفق به ایجاد شکست نشوند، درست تشخیص داده نمی‌شوند.

دایال و کومار در [۳۴] بر روی یک چارچوب تشخیص دولایه^۹ برای تشخیص شبکه‌های بات نظیربه‌نظیر نوپز دار کار کرده‌اند. رویکرد آن‌ها می‌تواند شبکه‌های بات را در مرحله انتظار و بدون نیاز به امضای بات‌ها شناسایی کند. روش آن‌ها از سه ویژگی استفاده کرده است: (i) نظیرهایی با طول حیات طولانی و درخواست‌های جستجو، (ii) شدت ارتباطات و (iii) رفتار وابسته و موقت. این آزمایش برای ارزیابی تنها یکی از هر دو نوع ترافیک بات و ترافیک عادی است که ممکن است نتایج دقیق ارائه ندهد. چن و همکاران [۳۵] سامانه تشخیص شبکه‌باتی را طراحی کردند که ترافیک بدخواه شبکه‌بات را با استفاده از یادگیری ماشین با ناظر^{۱۰} و تحلیل ترافیک مبتنی بر محاوره تشخیص می‌دهد. آن‌ها کارایی عملکرد پنج الگوریتم شناخته شده یادگیری ماشین با ناظر را ارزیابی کرده و روش آن‌ها میزان دقت کمی به میزان ۹۳/۶٪ را حاصل کرد. علاوه بر این، ارزیابی آن‌ها بر اساس رده خاصی از بات‌ها در مجموعه داده بوده است.

در این آزمایش، چهار برنامه کاربردی نظیربه‌نظیر و دو بات به‌عنوان مجموعه داده استفاده شده است. جریان‌های نظیربه‌نظیر که به‌صورت سلسله مراتبی خوشه‌بندی شده بودند، برای تعیین ترافیک شبکه‌بات نظیربه‌نظیر از ترافیک نظیربه‌نظیر سالم با سرعت بالا استفاده شده است.

رهبری‌نیا و همکاران [۲۸] روش PeerRush را پیشنهاد دادند که از طبقه‌بندی یک کلاسه برای طبقه‌بندی انواع مختلف ترافیک عادی و ترافیک مخرب نظیربه‌نظیر استفاده می‌کند. در این روش ابتدا، یک پروفایل کاربردی از نمونه‌های ترافیک برنامه‌های شناخته شده نظیربه‌نظیر ساخته شده و ویژگی‌هایی همچون مدت زمان جریان و تأخیر زمانی بین ارسال بسته‌ها برای طبقه‌بندی کاربردهای نظیربه‌نظیر مورد استفاده قرار گرفته است. بر اساس ویژگی‌های انتخاب شده، رویکرد فوق به‌دقت بالا برای طبقه‌بندی کاربرد نظیربه‌نظیر دست می‌یابد، اما این روش به‌وضوح روش تشخیص شبکه‌بات نظیربه‌نظیر را نشان نمی‌دهد. بعلاوه، می‌توان به‌راحتی از طریق اصلاح تأخیر در بین بسته‌ها از این نوع از تشخیص اجتناب کرد. سعد و همکاران [۱۴] ویژگی‌ها و رفتار ترافیک شبکه را برای تشخیص مرکز فرمان و کنترل بات نظیربه‌نظیر بر اساس ایمیل‌های مخرب، وب‌سایت‌ها، شبکه‌های اشتراک‌گذاری فایل و شبکه‌های بی‌سیم اقتضایی^۱ مورد بررسی قرار داده‌اند. با استفاده از مجموعه داده ISOT، پنج الگوریتم مختلف یادگیری ماشین برای جداسازی ترافیک شبکه‌بات از ترافیک نرمال استفاده شده و بیشترین دقت به‌دست آمده توسط این روش مطالعه ۸۹٪ بوده است.

ژائو و تریور [۲۹] روشی برای تشخیص شبکه‌بات نظیربه‌نظیر را بر اساس شناخت رفتار مخرب شبکه‌های Fast-Flux ارائه دادند. آن‌ها معیارهای ترافیک شبکه دریافتی را محاسبه کرده و سپس برای تعیین ترافیک شبکه‌بات استفاده کردند. رویکرد آن‌ها بر اساس الگوریتم درخت تصمیم‌گیری با دقت بالا است. یک درخت تصمیم‌گیری به‌عنوان یک مجموعه ویژگی (به‌عنوان مثال، روش‌های کاهش داده) استفاده می‌شود تا ویژگی‌های بی‌ارزش شبکه را حذف کند. تعداد داده‌های مورد نیاز کاهش می‌یابد، به این ترتیب امکان بهبود میزان یادگیری و دقت طبقه‌بندی و کاهش زمان محاسبات فراهم می‌شود. ژائو و همکاران [۳۰] یک روش تشخیص شبکه‌بات را بر اساس تجزیه و تحلیل رفتار ترافیکی و فواصل جریان معرفی کردند. الگوریتم درخت با برش خطای کاسته شده (REPTree) برای طبقه‌بندی ترافیک بات و ترافیک سالم مورد استفاده قرار گرفته است. با این حال، این روش تشخیص

² Hyper Text Transfer Protocol (HTTP)

³ Multilayer FeedForward Neural Network

⁴ Behavior-based

⁵ Host-based

⁶ Network Failure Model

⁷ Failure

⁸ Peers

⁹ Two-tier

¹⁰ Supervised Machine Learning

¹ Adhoc Network

مفهوم تقارن گرافیکی^۴، ترافیک شبکه را با انتخاب صرفاً ترافیک پروتکل کنترل انتقال کاهش داده و ویژگی‌های تأثیرگذار را با روش انترپبی اطلاعات جریان و تعیین ویژگی‌های با همبستگی بالاتر در شبکه‌های حسگر بی‌سیم را انتخاب نمایند. همچنین آن‌ها سه الگوریتم یادگیری ماشین: N-نزدیک‌ترین همسایه، ماشین بردار پشتیبان و الگوریتم درخت با برش خطای کاسته شده را برای دستیابی به دقت بالا در تشخیص حملات منع خدمات توزیع شده (DDoS) و شبکه‌های بات نظیر به نظیر رمزگذاری شده استفاده کرده‌اند.

اهداف روش پیشنهادی برای شناسایی ترافیک بدخواهانه بات جهت کاهش آسیب حملات آن است. مجموعه ویژگی‌های ورودی شامل خصیصه‌های پروتکل کنترل انتقال است. فعالیت‌های شبکه‌ای انواع مختلف بات‌ها مانند Zeus، Storm و Waledac بررسی و ویژگی‌های قابل توجهی از جریان‌های ترافیکی با فن طبقه‌بندی و رگرسیون برای کاهش تعداد ویژگی‌ها و بهبود تأثیر آن‌ها برای جدا کردن جریان‌های عادی ترافیک از شبکه‌های بات استخراج شده است. روش پیشنهادی شامل انتخاب ۱۵ ویژگی بر اساس میزان اثرگذاری آن‌ها در تشخیص شبکه‌بات از بین ۲۹ ویژگی استخراج شده از مجموعه داده ISCX است. روش پیشنهادی به دلیل استفاده از شبکه عصبی بازگشتی LSTM جهت تشخیص بر اساس داده‌های واقعی ترافیک به‌عنوان یک روش جدید مطرح می‌شود. روش پیشنهادی از یادگیری عمیق برای تشخیص شبکه‌بات و نیز از ترافیک واقعی بجای ترافیک مصنوعی، برای آموزش شبکه استفاده می‌کند.

۳- رویکرد پیشنهادی

چارچوب پیشنهادی به دو مفهوم اصلی متکی است. ابتدا، به‌صورت غیرفعال بر ترافیک شبکه نظارت دارد [۳۷] و ثانیاً، از این حقیقت استفاده می‌کند که بات‌ها در طول مرحله انتشار، رفتارهای ارتباطی را اغلب با خدمات‌دهنده فرمان و کنترل و نظیرهای خود نشان می‌دهند تا بتوانند نظیرهای دیگر را کشف کرده و آخرین به‌روزرسانی فعالیت‌ها را از طریق روش‌های از پیش‌برنامه‌ریزی شده خود دریافت کنند [۳۸-۳۹]. بات‌ها با دیگر نرم‌افزارهای مخرب متفاوت‌اند، زیرا آن‌ها گروهی کار می‌کنند و عمدتاً به یک کانال ارتباطی برای هماهنگ کردن فعالیت‌های بدخواهانه و مخرب خود نیاز دارند. این اتصالات، راهی است که مهاجم با بات‌های خود ارتباط برقرار می‌کند. روش پیشنهادی از یک شبکه عصبی عمیق ترکیبی استفاده می‌کند،

آلتامان و همکاران [۱۱] طرح تشخیص شبکه‌بات نظیر به نظیر مبتنی بر درخت تصمیم‌گیری و شبکه‌های عصبی چندلایه تطبیق‌پذیر را ارائه داده‌اند که تمرکز آن‌ها بر روی رفتارهای ارتباطی بین بات‌ها و خدمات‌دهنده فرمان و کنترل آن‌ها بوده است. آن‌ها ۶ قاعده^۱ را برای انتخاب بسته‌های موردنظر خود مشخص کرده‌اند تا تعداد بسته‌های موردبررسی را کاهش دهند. در مجموع ۲۹ ویژگی پروتکل کنترل انتقال بر اساس مدت‌زمان اتصال ۳۰ ثانیه‌ای استخراج کرده و یک درخت طبقه‌بندی و رگرسیون^۲ ارائه دادند، ناخالصی انترپبی در یک گره مشخص برای تعیین گره بعدی موردبررسی قرار گرفته و الگوریتم Relief جهت کشف تأثیر ویژگی‌های مختلف در شناسایی ارزش هر یک از ویژگی بکار برده شده است. ویژگی‌های رتبه‌بندی شده توسط این دو الگوریتم انتخاب ویژگی و زیرمجموعه‌های مختلف از ویژگی‌ها تعیین شده است. مجموعه داده ISCX و مجموعه داده ISOT برای فرایند ارزیابی مورد استفاده قرار گرفته و رویکرد پیشنهادی سبب شناسایی شبکه‌های بات با میزان صحت ۹۹/۲٪، میزان دقت ۹۸/۳۲٪، بازخوانی ۹۷/۸٪ و نرخ مثبت کاذب ۰/۷۵ شده است. این رویکرد قادر به تشخیص شبکه‌های باتی که از پروتکل دیتاگرام کاربر (UDP) برای برقراری ارتباط استفاده می‌کنند، نیست.

اوبیدات [۱۲] تحقیق جدیدی برای شناسایی شبکه‌های بات را پیشنهاد کرد که از ترکیب خوشه‌بندی (داده‌کاوی) و شبکه عصبی برای ایجاد سیستم خود استفاده می‌کند. در این راستا رویکرد او ابتدا مجموعه داده‌ها را خوانده و شباهت بین آن‌ها را محاسبه می‌کند. سپس، این روش یک طبقه‌بندی SVM را با استفاده از الگوریتم‌های خوشه‌بندی K-medoids و K-means ایجاد می‌کند. در نهایت، نتایج فرآیند خوشه‌بندی در شبکه عصبی هاپفیلد^۳ به‌عنوان یک ماشین یادگیری برای تشخیص انواع مختلف بات‌ها استفاده می‌شود. نتایج نشان می‌دهد که الگوریتم K-medoids و تغییرات آن نسبت به الگوریتم K-mean و تغییرات مربوطه آن، بهبود یافته است. رویکرد ایشان مبتنی بر ویژگی‌های بسته‌های پروتکل کنترل انتقال است. عملکرد و دقت سامانه با استفاده از یک مجموعه داده از پیش تعیین شده مورد ارزیابی قرار گرفته و سامانه دقت ۹۵/۷٪ را با نرخ مثبت کاذب برابر ۳ نشان می‌دهد.

یانگ و وانگ [۳۶] در تحقیق خود سعی کرده‌اند با استفاده از

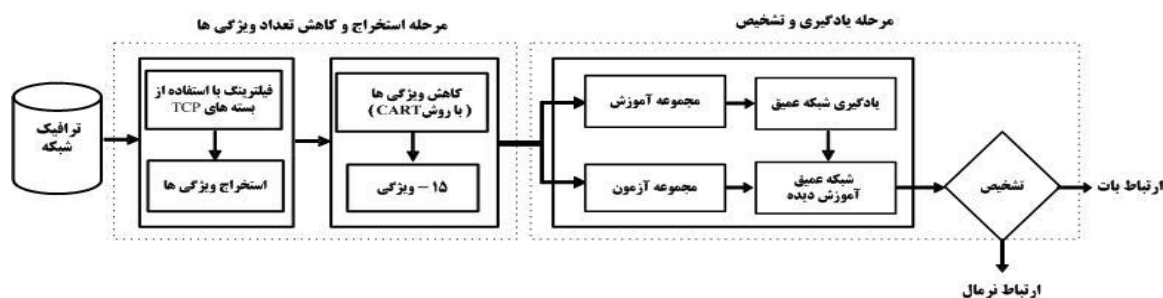
¹ Rule

² Classification and Tree Regression

³ Hopfield

⁴ Graphic Symmetry Concept

روش یادگیری عمیق (LSTM) برای بهبود عملکرد شبکه عصبی مصنوعی در تشخیص شبکه‌هاست استفاده می‌کند. روش پیشنهادی در دو مرحله کار می‌کند. همان‌طور که در مقدمه نیز اشاره شد، مدل پیشنهادی شامل مراحل استخراج ویژگی و تشخیص است. چارچوب کلی مدل پیشنهادی در زیر بخش‌های بعدی مورد بحث قرار می‌گیرد. شکل (۱) مدل پیشنهادی را نشان می‌دهد.



شکل (۱): نمودار بلوک مدل پیشنهادی.

زیرا شبکه‌های بات شناخته‌شده با استفاده از اتصالات پروتکل کنترل انتقال [۴۰] از جمله بات Waledac [۴۱]، بات Storm [۴۲]، بات Conficker [۴۳] و بات Zeuse [۴۴] ارتباط برقرار می‌کنند. از این‌رو، در مدل ارائه‌شده، ویژگی‌های مربوط به اتصال پروتکل کنترل انتقال بر اساس بسته‌های کنترلی پروتکل کنترل انتقال استخراج شده است. رویکرد پیشنهادی از

الگوریتم (۱): کاهش ترافیک شبکه.

```

1: Procedure reduction (packets)
2: ArrayList <Packet> TCP_Control_Packets_List ;
3: For i=1 to size(Packets)
4:   IF Packets(i) has (TCP header) then
5:     IF Packets (i) has (TCP. payloadSize==0) then
6:       pktheadr= packet.getHeader(Packets(i));
7:       IF((pktheadr.flags.syn=1 OR pktheadr.flags.ack=1 OR
          pktheadr.flags.rest=1 OR pktheadr.flags.fin=1)
          AND NOT (pktheadr.flags.cwr=1 OR
          pktheadr.flags.ecn=1 OR pktheadr.flags.push=1
          OR pktheadr.flags.urg=1)) then
8:         TCP_Control_Packets_List.Add(packets(i));
9:       ELSE
10:        Discard the Packet;
11:      End IF
12:    End IF
13:  End For
14: For i=1 to size(TCP_Control_Packets_List)-1
15:  TimeInterval[i]=TCP_Control_Packets_List[i+1].Timestamp -
    TCP_Control_Packets_List[i].Timestamp
16: End For
17: Return TCP_Control_Packets_list;
18: End Procedure
    
```

در مجموع الگوریتم (۱) شامل ۶ قاعده برای انتخاب بسته‌های مطلوب است:

- قاعده ۱: بسته‌های حاوی (SYN flag)
- قاعده ۲: بسته‌های حاوی (SYN-ACK flag)
- قاعده ۳: بسته‌های حاوی (ACK flag)
- قاعده ۴: بسته‌های حاوی (FIN-ACK flag)

۳-۱-۱- مرحله استخراج و کاهش تعداد ویژگی‌ها

۳-۱-۱-۱- کاهش ترافیک

مرحله استخراج ویژگی با خواندن داده‌ها یا رکوردهای مجموعه داده‌های آموزشی و استخراج ویژگی‌های مبتنی بر جریان شروع می‌شود. یک ویژگی^۱، به مشخصه^۲ یک جریان بسته در پنجره زمانی^۳ T اشاره دارد که می‌تواند یک مقدار عددی یا غیر عددی داشته باشد. اگرچه بسیاری از ویژگی‌های مبتنی بر جریان برای شناسایی انواع مختلف شبکه‌های بات پیشنهاد شده‌اند، اما نمی‌توان به‌طور قطعی در مورد ارزش واقعی و تعداد مناسب این ویژگی‌ها نتیجه‌گیری کرد. در این مقاله رفتارهای مختلف شبکه‌های بات معروف مانند Storm، Waledac، Nugache و Zeus بررسی شده و ویژگی‌های مختلف از بسته‌های نمونه به‌عنوان یک بردار استخراج می‌شود. منطق اصلی روش مورد استفاده در این مقاله بر مبنای آن است که شبکه‌های بات دارای الگوهای ترافیکی هستند که می‌توانند با روش‌های یادگیری عمیق شناخته شوند و می‌توانند ترافیک شبکه‌ها را از ترافیک شبکه معمولی جدا سازند. در ادامه، منطق تولید ویژگی‌ها از بسته‌های نمونه توصیف می‌شود. در اینجا، از فیلتر ترافیک بسته‌های کنترلی پروتکل کنترل انتقال جهت کاهش حجم ترافیک شبکه برای افزایش کارایی روش پیشنهادی استفاده می‌شود. فیلترینگ شامل دو مرحله است: اولاً، فیلتر کردن تمام ترافیک شبکه برای پروتکل کنترل انتقال؛ ثانیاً، استخراج بسته‌های کنترلی پروتکل کنترل انتقال. الگوریتم (۱) رویه کاهش ترافیک شبکه را از فایل ردیابی شبکه نشان می‌دهد (فایل با پسوند Pcap).

¹ Feature
² Characteristic
³ Time Window

- قاعده ۵: بسته‌های حاوی (Rest-ACK flag)
- قاعده ۶: بسته‌های حاوی (Rest-SYN flag)

۳-۱-۲- استخراج ویژگی

داخلی با دو فرزند و (۲) گره‌های برگ بدون فرزند. گره داخلی مرتبط با یک تابع تصمیم است که نشان می‌دهد کدام گره برای ملاقات بعدی مدنظر است. برای ایجاد درخت، نمونه‌های آموزشی که حاوی مجموعه‌ای از ویژگی‌ها و برچسب‌های کلاس خود هستند، موردنیاز است. مجموعه آموزش به دلیل ایجاد درخت، مجدداً به زیرمجموعه‌های کوچک‌تر تقسیم می‌شود. بر اساس ماتریس تصمیم‌گیری حاصل از توزیع کلاس‌ها در مجموعه آموزشی، هر گره به یک کلاس پیش‌بینی شده تخصیص می‌یابد. آزمون در گره‌های داخلی بر اساس اندازه‌گیری ناخالص آن مشخص می‌شود تا انتخاب کند که کدام ویژگی و چه مقدار آستانه‌ای انتخاب شده است.

جدول (۱): ویژگی‌های انتخاب شده از ارتباطات ترافیک شبکه.

ویژگی	توضیحات (محاسبه مقادیر در یک‌فاصله زمانی ۳۰ ثانیه‌ای برای هر جریان)
F1	تعداد بسته‌های کنترلی
F2	تعداد بسته‌های انتقال داده شده
F3	تعداد بسته‌های دریافت شده
F4	تعداد بایت‌های انتقال داده شده
F5	تعداد بایت‌های دریافت شده
F6	تعداد بسته‌های SYN انتقال داده شده
F7	تعداد بسته‌های SYN دریافت شده
F8	تعداد بسته‌های ACK انتقال داده شده
F9	تعداد بسته‌های ACK دریافت شده
F10	تعداد بسته‌های ACK تکراری انتقال داده شده
F11	تعداد بسته‌های ACK تکراری دریافت شده
F12	میانگین طول بسته‌های کنترلی انتقال داده شده
F13	میانگین طول بسته‌های کنترلی دریافت شده
F14	میانگین طول بسته‌های کنترلی
F15	تعداد ارتباطات شکست خورده انتقال داده شده
F16	تعداد ارتباطات شکست خورده دریافت شده
F17	تعداد بسته‌های ACK انتقال داده شده از هر جریان در یک‌فاصله زمانی که دارای توالی هستند
F18	تعداد بسته‌های ACK دریافت شده از هر جریان در یک‌فاصله زمانی که دارای توالی هستند
F19	تعداد بسته‌های SYN-ACK انتقال داده شده
F20	تعداد بسته‌های SYN-ACK دریافت شده
F21	کل تعداد بایت‌ها
F22	نرخ بسته‌های کنترلی ورودی
F23	نرخ اندازه متوسط بسته‌های خروجی بر اندازه متوسط بسته‌های کنترلی
F24	حاصل تفریق مقدار ویژگی F6 از مقدار ویژگی F20
F25	تعداد بسته‌های FIN-ACK انتقال داده شده
F26	تعداد بسته‌های FIN-ACK دریافت شده
F27	تعداد بسته‌های RST-ACK انتقال داده شده
F28	تعداد بسته‌های RST-ACK دریافت شده
F29	متوسط زمان تلاش برای ایجاد ارتباط

در مرحله استخراج ویژگی‌ها، ویژگی‌هایی که در شناسایی رفتار بدخواهانه و مخرب بات قابل توجه هستند، استخراج می‌شوند و این ویژگی‌ها به صورت ویژگی‌های ۲۹ تایی بر اساس ارتباط ۳۰ ثانیه‌ای (جدول (۱)) جمع‌آوری می‌شوند. این ویژگی‌ها بر اساس اتصال مابین دو میزبان مختلف به‌عنوان گروهی از بسته‌های مبادله شده استخراج می‌شوند که توسط آدرس آی‌پی مبدأ، آدرس آی‌پی مقصد، درگاه مبدأ و درگاه مقصد مشخص می‌شود. در روش پیشنهادی، تمام ویژگی‌ها از سرآیند بسته کنترلی استخراج می‌شوند. (روش‌های قبلی با استفاده از کنترل عمیق محتوای بسته این کار را انجام داده‌اند [۴۷-۴۵]). از این‌رو، کارایی عملکرد افزایش یافته و استفاده از منابع سامانه کاهش می‌یابد. جدول (۱)، ۲۹ ویژگی انتخاب شده و مورد استفاده از تمام ویژگی‌های رویکرد تشخیص شبکه‌بات را نشان می‌دهد. این ویژگی‌ها از یک اتصال ۳۰ ثانیه‌ای (فاصله زمانی) ساخته می‌شوند و یک بردار ویژگی برای نشان دادن ویژگی‌های یک اتصال ۳۰ ثانیه‌ای در نظر گرفته شده است. هر ویژگی دارای معنای خاصی است، همان‌طور که در جدول (۱) نشان داده شده است.

۳-۱-۳- کاهش ویژگی‌ها

فن کاهش ویژگی‌ها برای کاهش تعداد صفات مورد استفاده قرار می‌گیرد و حذف این ویژگی‌ها در الگوریتم یادگیری تأثیر کمی بر موضوع طبقه‌بندی دارد. کاهش ویژگی‌ها برای کاهش مشکل بیش‌برازش^۱ [۴۸] استفاده می‌شود و مشکل جمع‌آوری داده‌های نامتوازن^۲ را رفع می‌کند [۴۹]؛ بنابراین، عملکرد فن کاهش ویژگی یکی از عوامل مهم تأثیرگذار بر دقت الگوریتم‌های طبقه‌بندی است. در این مقاله، هدف از کاهش ویژگی‌ها، انتخاب یک زیرمجموعه مناسب از ویژگی‌ها است که عملکرد شبکه عصبی را بهبود می‌بخشد و پیچیدگی یک مدل طبقه‌بندی را بدون کاهش میزان دقت آن، کاهش می‌دهد. در این مقاله، درخت طبقه‌بندی و رگرسیون [۱۰] به‌عنوان فن کاهش ویژگی برای حذف ویژگی‌های نامناسب باهدف کاهش میزان اطلاعات موردنیاز برای به دست آوردن نرخ بهتر یادگیری شبکه‌های عصبی و دقت طبقه‌بندی استفاده می‌شود. قدرت اصلی درخت طبقه‌بندی و رگرسیون سریع بودن و مقیاس‌پذیر بودن آن برای مجموعه داده‌های بزرگ است.

درخت تصمیم‌گیری تولیدشده توسط الگوریتم درخت طبقه‌بندی و رگرسیون شامل دو نوع گره است: (۱) گره‌های

¹ Over-Fitting

² Imbalance

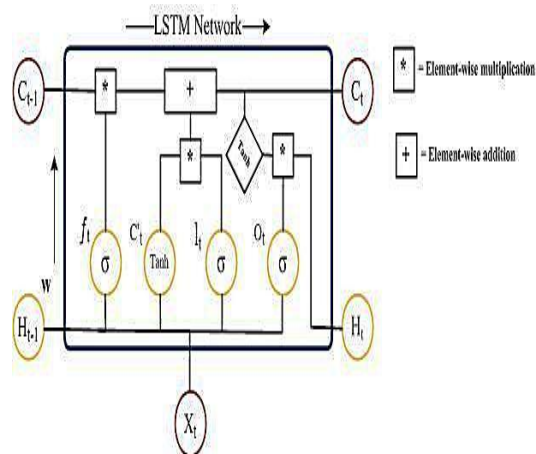
آموزش و مجموعه آزمون نامیده می‌شوند. مجموعه آموزش برای یادگیری است که شامل ۷۰ درصد از کل داده‌های مجموعه داده و مجموعه آزمون شامل ۳۰ درصد باقی‌مانده از کل داده‌های مجموعه داده برای ارزیابی مدل استفاده می‌شود. پیش از توصیف مدل پیشنهادی برای تشخیص شبکه‌ها، مرور کلی از عملکرد شبکه عصبی حافظه کوتاه مدت ماندگار ارائه داده می‌شود.

۳-۲-۱- حافظه کوتاه مدت ماندگار

شبکه عصبی بازگشتی^۱ برای یادگیری وابستگی‌ها در یک توالی ورودی طراحی شده است، که یک حلقه اطلاعات را نشان می‌دهد و به عنوان ورودی یک دنباله در زمان جاری و یک خروجی از مرحله قبلی را می‌گیرد. حافظه کوتاه مدت ماندگار [۵۰] یک نوع خاص از شبکه‌های عصبی بازگشتی به اصطلاح گیت شده^۲ است. به جای نورون معمولی، شبکه از سلول‌های حافظه کوتاه مدت ماندگار ساخته می‌شود که شامل یک خود حلقه^۳ و یک سامانه دارای واحدهای دروازه‌ای است که جریان اطلاعات را کنترل می‌کند. یک شبکه حافظه کوتاه مدت ماندگار دارای سه ویژگی زیر است که آن را از یک نورون رایج در یک شبکه عصبی بازگشتی متمایز می‌کند:

۱. تصمیم‌گیری در هنگام ورود به نورون را کنترل می‌کند.
۲. بر تصمیم‌گیری در مورد یادآوری محاسبات مرحله قبل، کنترل دارد.
۳. تصمیم‌گیری در هنگام ارسال خروجی به مرحله بعد را کنترل می‌کند.

شکل (۲) معماری و روابط یک سلول حافظه کوتاه مدت ماندگار را نمایش می‌دهد.



شکل (۲): معماری و روابط موجود در یک سلول LSTM.

بهترین معیار شناخته شده ناخالصی برای درخت طبقه‌بندی و رگرسیون، ناخالصی انتروپی است که توسط رابطه (۱) تعریف می‌شود [۱۱].

$$E(t) = -\sum_j^c p\left(\frac{j}{t}\right) \log_2 p\left(\frac{j}{t}\right) \quad (1)$$

جایی که $E(t)$ ناخالصی انتروپی در گره t است، فرکانس نسبی $p\left(\frac{j}{t}\right)$ کلاس j در گره t است و c تعداد کلاس‌ها در طبقه‌بندی است. بهترین مقدار گره تقسیم شده (t) از مجموعه‌ای از تمام تقسیم ارزش‌های (X) انتخاب می‌شود، به طوری که حداکثر کاهش ناخالصی، تفاوت بین ناخالصی در گره ریشه و ناخالصی در گره‌های فرزندان است:

$$\Delta E(X, t) = E(t) - (P_L E(t_L) + P_R E(t_R)) \quad (2)$$

جایی که $\Delta E(X, t)$ ناخالصی است، $E(t_L)$ و $E(t_R)$ ناخالصی گره‌های شاخه چپ و راست هستند، P_L و P_R درصدی از اشیاء هستند که به سمت چپ (t_L) یا راست (t_R) گره‌های فرزند است. جدول (۲) رتبه‌بندی از ۲۹ ویژگی پراهمیت انتخاب شده توسط الگوریتم درخت طبقه‌بندی و رگرسیون را ارائه می‌دهد. ویژگی‌های $F_3, F_13, F_23, F_14, F_21, F_12, F_29, F_4, F_15, F_6, F_5, F_27, F_10, F_8, F_17$ بهترین شناسایی رفتار ارتباطات را دارند و سایر ارتباطات مقدار صفر داشته و هیچ شناسایی بین ارتباطات نرمال و مخرب ندارند.

جدول (۲): رتبه‌بندی اهمیت ویژگی‌ها با استفاده از الگوریتم CART.

اهمیت	ویژگی	اهمیت	ویژگی
۹e-۰۹	F17	۱۰۰	F3
.	F2	۶۹/۷۷	F13
.	F7	۵۸/۸۳	F23
.	F9	۱۴/۹۴	F21
.	F11	۲/۹۰	F14
.	F16	۰/۷۹	F29
.	F18	۰/۳۸	F12
.	F19	۰/۱۲	F1
.	F20	۰/۰۸	F4
.	F22	۰/۰۷	F15
.	F24	۰/۰۱۲	F6
.	F25	۰/۰۱۱	F5
.	F28	۰/۰۱۱	F27
.	F26	۰/۰۰۰۵	F10
-	-	۳e-۰۶	F8

۳-۲-۲- مرحله تشخیص

پس از اتمام مرحله استخراج و کاهش ویژگی‌ها، روش ارائه شده فرآیند ارزیابی را از طریق مرحله تشخیص آغاز می‌کند. در این مرحله مجموعه داده‌ها به دو بخش تقسیم می‌شود که مجموعه

^۱ Recurrent Neural Networks

^۲ Gated

^۳ Self-loop

فاصله داشته باشد، الگوریتمی که بیش‌برازش شده باشد، نمی‌تواند به درستی پاسخی برای این داده‌های جدید پیدا کند و آن‌ها را با اشتباه زیادی طبقه‌بندی می‌کند. روش‌های مختلفی برای جلوگیری از بیش‌برازش در الگوریتم‌های یادگیری ماشین و یادگیری عمیق مطرح شده است که عبارت‌اند از مقایسه مدل^۷، اعتبارسنجی متقابل^۸، تنظیم^۹، توقف اولیه^{۱۰}، هرس کردن^{۱۱} و حذف تصادفی گره‌ها^{۱۲}

در این مقاله از روش حذف تصادفی گره‌ها در شبکه برای بهبود عملکرد مدل پیشنهادی استفاده شده است که با حذف گره‌ها به صورت تصادفی در مرحله آموزش، سبب غلبه بر بیش‌برازش می‌شود [۵۴]. این کار فقط در مرحله آموزش صورت می‌پذیرد و در مرحله آزمون و انتشار، تمام گره‌ها فعال هستند. در این مقاله میزان حذف تصادفی با مقدار ۵۰٪ تنظیم شده است. مدل در کراس^{۱۳} به شرح زیر تعریف شده است:

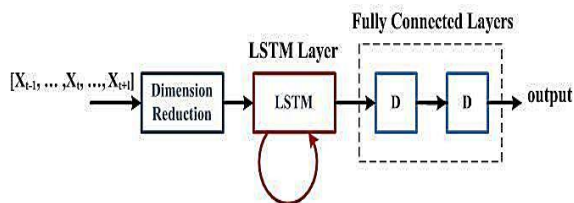
LSTM-۱-۲-۲-۳

```
model = Sequential()
model.add(LSTM(100, input_shape = (1, look_back)))
model.add(Dropout(0.5))
model.add(Dense(100))
model.add(Dense(1))
model.add(Activation('sigmoid'))
model.compile(loss='binary_crossentropy', optimizer =
'adam', metrics=['acc', precision, recall, f1])
```

۳-۳- جزئیات پیاده‌سازی

سیستم استفاده شده برای آزمایش شبکه دارای مشخصات زیر است:

- پردازنده Intel Core i7-2670QM، ۲/۲۰ گیگاهرتز
- حافظه اصلی ۸ گیگابایت
- کارت گرافیکی GPU NVIDIA GeForce GT540 با ۲ گیگابایت رم.



شکل (۳): مدل تشخیص روش پیشنهادی.

$$f_t = \sigma(X_t * U_f + H_{t-1} * W_f)$$

$$C'_t = \tanh(X_t * U_c + H_{t-1} * W_c)$$

$$l_t = \sigma(X_t * U_i + H_{t-1} * W_i)$$

$$O_t = \sigma(X_t * U_o + H_{t-1} * W_o)$$

$$C_t = f_t * C_{t-1} + l_t * C'_t$$

$$H_t = O_t * \tanh(C_t) \quad (۳)$$

$X_t = \text{Input Vector}$

$H_{t-1} = \text{Previous cell output}$

$C_{t-1} = \text{Previous cell Memory}$

$H_t = \text{Current cell output}$

$C_t = \text{Current cell Memory}$

$W, U = \text{Weight vectors for forget gate (f), Candidate (c), i/p gate(i) and o/p gate (o)}$

۳-۲-۲- مدل یادگیری عمیق

در اینجا کاربرد شبکه LSTM که در روش یادگیری عمیق آموزش داده می‌شود ارائه می‌شود. هر مدل با استفاده از اندازه دسته‌ای^۱ برابر با ۶۴، مورد آموزش قرار می‌گیرد. LSTM از معماری طراحی شده در [۵۱] با اندکی اصلاح برگرفته شده است. شبکه LSTM شامل یک لایه تعبیه شده^۲ است. لایه تعبیه شده ورودی را در قالب شاخص‌ها می‌پندارد (هر شاخص به یک لایه تعبیه شده منحصر به فرد نگاشت می‌شود). پس از لایه تعبیه شده، معماری LSTM از یک لایه (۱۰۰ سلول LSTM با فعال‌سازی Tanh پیش‌فرض)، یک لایه اتصال کامل^۳ (۱۰۰ گره با یک فعال‌سازی خطی پیش‌فرض) و در نهایت تنها یک لایه خروجی تابع فعال‌سازی سیگمید^۴ تشکیل شده است. به جای استفاده از RMSProp [۵۲]، با استفاده از Adam [۵۳] به عنوان یک الگوریتم بهینه‌سازی، به نتایج خوبی در همگرایی گم‌شدگی^۵ می‌توان دست یافت. یعنی با وجود عدم اطلاع از تعداد تکرار اجرا برای رسیدن به نتایج بهینه می‌توان با استفاده از کاهش یافتن صحت یادگیری یا همچنین افزایش خطای یادگیری، یادگیری را متوقف کرده و تعداد بهینه تکرار اجرای الگوریتم را بدست آورد.

بیش‌برازش^۶ به معنای این است که الگوریتم فقط داده‌هایی را که در مجموعه آموزشی یادگیری کرده است را می‌تواند به درستی پیش‌بینی کند ولی اگر داده‌ای کمی از مجموعه آموزشی

⁷ Model comparison

⁸ Cross-validation

⁹ Regularization

¹⁰ Early stopping

¹¹ Pruning

¹² Drop out

¹³ Keras

¹ Batch Size

² Embedding layer

³ Fully Connected

⁴ Sigmoid

⁵ Loss convergence

⁶ Over-fitting

به K نمونه تقسیم شده و ارزیابی برای K تکرار اجرا می‌شود. در هر تکرار، K-1 نمونه برای آموزش انتخاب شده و از ۱ نمونه برای ارزیابی صحت طبقه‌بندی استفاده شده است. در اینجا مقدار K جهت انجام آزمایش‌ها برابر با ۱۰ انتخاب شده است. عملکرد مدل پیشنهادی با ۴ روش اشاره شده دیگر مقایسه شده است. برای ارزیابی عملکرد سامانه تشخیص شبکه عصبی عمیق، معیارهای اندازه‌گیری از قبیل دقت^۴، صحت^۵، بازخوانی^۶، میانگین هارمونی و نرخ مثبت کاذب^۷ با استفاده از روابط ۴ الی ۸ محاسبه می‌شود:

- مثبت واقعی (TP): تعداد رفتارهای بدخواهی (بات) که به درستی به‌عنوان بات شناسایی شده‌اند.
- مثبت کاذب (FP) تعداد رفتارهای عادی که به‌عنوان بات شناسایی شده‌اند.
- منفی واقعی (TN) تعداد رفتارهای عادی که به‌طور صحیح به‌عنوان فعالیت‌های عادی شناسایی شده‌اند.
- منفی کاذب (FN) تعداد رفتارهای بات که به‌عنوان فعالیت‌های عادی شناسایی شده‌اند.
- دقت: نشان‌دهنده درصد نمونه‌هایی است که به‌درستی به‌عنوان نمونه مثبت طبقه‌بندی می‌شوند.

$$Precision(RPC) = \frac{TP}{TP + FP} \quad (۴)$$

- صحت: درصد پیش‌بینی‌های صحیح تمام نمونه‌ها را نشان می‌دهد.

$$Accuracy(ACC) = \frac{TP + TN}{TP + TN + FP + FN} \quad (۵)$$

- بازخوانی: نشان‌دهنده درصد نمونه‌های باتی است که به‌عنوان یک نمونه بات پیش‌بینی شده است. این معیار، میزان تشخیص^۸ نیز نامیده می‌شود.

$$Recall(RCL) = \frac{TP}{TP + FN} \quad (۶)$$

- معیار میانگین هارمونی: یک اندازه‌گیری از دقت آزمون است. این معیار هر دو معیار بازخوانی و دقت آزمون را برای محاسبه امتیاز بررسی می‌کند.

شبکه عصبی عمیق LSTM با استفاده از پایتون ورژن ۳/۶ در محیط ژوپیتر نوت‌بوک^۱ و تنسورفلو^۲ [۵۴] آموزش داده شده و از کراس نسخه ۲.۲.۲ [۵۵] که یک API برای نمونه سریع از تنسورفلو نسخه ۱.۱۰.۰ در CPU است، استفاده می‌شود. سیستم عامل مورد استفاده برای شبیه‌سازی و یادگیری، ویندوز ۷ است. بسیاری از آزمایش‌ها برای تعیین پارامترهای مناسب و یافتن ساختار مناسب شبکه انجام یافته است.

۴- نتایج و اعتبارسنجی

۴-۱- مجموعه داده‌ها

در این مقاله مجموعه داده‌های ISCX [۵۶] و ISOT [۱۴] که شامل ترافیک بات و ترافیک سالم هستند برای ارزیابی سامانه پیشنهادی مورد استفاده قرار گرفته‌اند. این مجموعه داده‌ها برای آموزش، اعتبارسنجی مدل و ارزیابی عملکرد آن مورد استفاده قرار گرفته است. مجموعه داده‌های آموزشی شامل ۱۹۷۸۱۱ رکورد است و بات‌های Conficker، Waledac و Storm از انواع اصلی بات‌ها هستند که در فرآیند ارزیابی از مجموعه داده‌های اصلی به‌دست آمده‌اند. استفاده از داده‌های جدید خارج از مجموعه آموزشی برای ساخت داده‌های آزمون واقعی‌تر قابل توجه است و شامل ۹۷۴۲۱ رکورد است.

برای ایجاد یک مجموعه داده تجربی با ترافیک شبکه‌های بات و ترافیک سالم، فایل‌های ردیابی با پسوند pcap. با استفاده از نرم‌افزار وایرشارک^۳ برای ارزیابی مورد بررسی قرار گرفت. سپس، Microsoft Network Monitor برای ساخت ارتباطات و استخراج ویژگی‌ها از فایل PCAP، استفاده شده و ارتباطات در دو کلاس ارتباط بات و ارتباطات سالم برچسب‌گذاری شدند. در اینجا جریان ارتباطی شبکه با یک ۴ تایی، آدرس آی‌پی مبدأ، آدرس آی‌پی مقصد، شماره درگاه مبدأ و شماره درگاه مقصد مشخص شده است که حداقل یک بسته بین دو جهت منتقل شده باشد.

۴-۲- ارزیابی عملکرد و نتایج

برای ارزیابی نرخ دقت تشخیص و ارزیابی نرخ خطای روش یادگیری عمیق پیشنهادی، الگوریتم‌های جنگل تصادفی و درخت تصمیم‌گیری در این مقاله از اعتبارسنجی K-fold استفاده می‌شود. در اعتبارسنجی K-fold، مجموعه داده‌ها به‌طور تصادفی

^۴ Precision

^۵ Accuracy

^۶ Recall

^۷ False Positive Rate

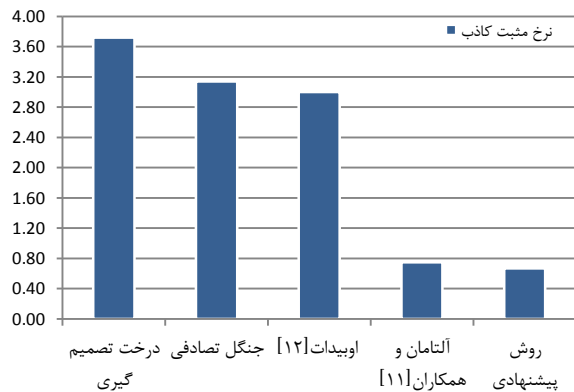
^۸ Detection Rate

^۱ Jupyter Notebook

^۲ Tensorflow

^۳ Wireshark

است. بسیاری از مدل‌ها و تغییرات در جهت کاهش میزان بیش‌برازش و تعمیم یادگیری مورد آزمایش قرار گرفتند. جدول (۴) ساختار، پارامترها و صحت مدل پیشنهادی را در آزمایش‌ها نشان می‌دهد و شکل (۶) معیارهای ارزیابی محاسبه‌شده مدل پیشنهادی را در تکرارهای مختلف نشان می‌دهد.



شکل (۵): مقایسه میزان مثبت کاذب روش پیشنهادی با ۴ روش دیگر.

جدول (۳) و شکل (۴) نتایج مقایسه روش پیشنهادی با سایر روش‌های مبتنی بر تحلیل ترافیک شبکه برای تشخیص شبکه‌های بات را نشان می‌دهد. این جدول همچنین نشان می‌دهد که میزان دقت، بازخوانی، میانگین هارمونی و نرخ مثبت کاذب با استفاده از رویکرد ترکیبی ارائه‌شده بهتر از میزان به‌دست‌آمده توسط روش‌های ذکر شده است و در معیار صحت نیز با وجود پایین بودن این مقدار صحت از مقادیر روش‌های مرجع [۱۱] و روش جنگل تصادفی، ولی اختلاف زیادی وجود نداشته است.

جدول (۴): ساختار، پارامترها و صحت مدل پیشنهادی در آزمایش.

صحت	بهبودسازی		نرخ حذف تصادفی ادغامی ^۲	تعداد واحد LSTM	استفاده از نرمال‌سازی دسته‌ای ^۳	نرخ حذف تصادفی شاخه‌ای ^۲	مدل
	نرخ یادگیری	نوع					
۹۶/۳۲	۰/۵	Adam	۰	۱۰۰	بله	۰/۰۰۱	LSTM

مقایسه روش‌های مختلف تشخیص شبکه‌بات آسان نیست زیرا هر یک از آن‌ها از مجموعه داده‌ها و نمونه‌های شبکه‌بات مختلفی در آزمایش‌های خود استفاده می‌کنند. از این‌رو، رویکرد

$$F - measure(F1) = \frac{2 * PRC * RCL}{PRC + RCL} \quad (7)$$

• نرخ مثبت کاذب (FPR): درصد نمونه‌های سالم را که به‌اشتباه به‌عنوان نمونه‌های بات طبقه‌بندی شده‌اند، نشان می‌دهد.

$$FPR = \frac{FP}{FP + TN} \quad (8)$$

جدول (۳) و شکل (۴)، صحت، دقت، بازخوانی، میانگین هارمونی و نرخ مثبت کاذب روش پیشنهادی را در مقایسه با الگوریتم‌های پیاده‌سازی شده جنگل تصادفی، درخت تصمیم‌گیری، روش مطالعه شده توسط آلتامان و همکاران [۱۱] و روش ارائه‌شده توسط اوپیدات [۱۲] نشان می‌دهند.

جدول (۳): مقایسه ۵ روش از نظر معیارهای ارزیابی عملکرد.

روش معیار	روش پیشنهادی	آلتامان و همکاران [۱۱]	اوپیدات [۱۲]	جنگل تصادفی	درخت تصمیم‌گیری
صحت	۹۶/۳۲	۹۹/۲۰*	۹۵/۷۸	۹۶/۶	۹۵/۷۱
دقت	۹۹/۶۵*	۹۸/۳۲	۸۹/۷۴	۹۷/۷	۹۶/۸
بازخوانی	۹۹/۶۱*	۹۷/۸۰	۹۹/۴۷	۹۸/۳	۹۷/۸
میانگین هارمونی	۹۹/۶۳*	۹۸/۰۶	۹۴/۳۵	۹۸	۹۷/۳
نرخ مثبت کاذب	۰/۶۷	۰/۷۵	۳	۳/۱۴	۳/۷۲

* بهترین نتایج

بررسی‌ها نشان می‌دهد که روش پیشنهادی شبکه عصبی عمیق دارای بالاترین میزان صحت و دقت با نرخ کم مثبت کاذب (شکل (۵)) است.



شکل (۴): معیارهای ارزیابی روش ارائه‌شده در مقایسه با ۴ روش دیگر.

این دقت به‌رغم چالش اصلی بیش‌برازش^۱ به‌دست آمده

^۲ Branch Dropout

^۳ Batch Normalization

^۴ Merge Dropout

^۱ Over-fitting

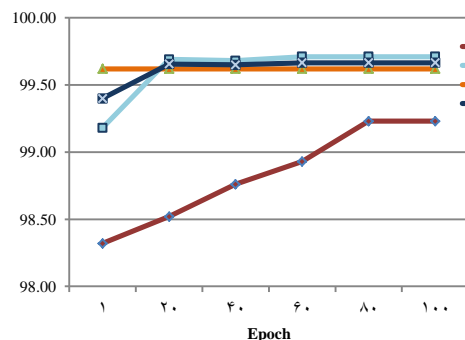
۵- نتیجه‌گیری

کارهای متعددی در تشخیص شبکه‌بات انجام یافته است، با این حال، نوآوری روش ما در استفاده از شبکه عصبی یادگیری عمیق و استفاده از الگوریتم درخت طبقه‌بندی و رگرسیون در استخراج ویژگی‌های پراهمیت از مجموعه داده‌ها در ارتباط با شناسایی شبکه‌های بات است. روش پیشنهادی به ویژگی‌های استخراج شده از سرآیندهای بسته کنترل پروتکل کنترل انتقال در طول ارتباط ۳۰ ثانیه‌ای بین دو میزبان متکی است، بنابراین، می‌تواند برای تشخیص بات بدون تکیه بر داده بسته از جمله آدرس آی‌پی مبدأ، آدرس آی‌پی مقصد، شماره درگاه مبدأ، شماره درگاه مقصد و ترافیک رمزگذاری شده مورداستفاده قرار گیرد. عملکرد روش تشخیص پیشنهاد شده با مراجع [۱۱]، [۱۲]، الگوریتم جنگل تصادفی و الگوریتم درخت تصمیم‌گیری پیاده‌سازی شده مقایسه گردیده است. نتایج نشان می‌دهد که روش پیشنهادی، دقت، بازخوانی و میانگین هارمونی بالاتر با نرخ مثبت کاذب پایین‌تری نسبت به آن‌ها دارد و در مقدار صحت محاسبه شده کاهش کمتری نسبت به دو روش جنگل تصادفی و مرجع [۱۱] نشان می‌دهد. با توجه به بهبود مقادیر بیشتر معیارهای ارزیابی می‌توان پیشرفت محسوسی در تشخیص شبکه‌بات نظریه‌نظیر مشاهده کرد. مطالعات بعدی در راستای گسترش و به‌کارگیری این رویکرد در سامانه‌های بلادرنگ از طریق افزودن یک روش یادگیری خوب برای انتخاب مهم‌ترین ویژگی‌های مرتبط است که سبب افزایش کارایی و میزان دقت تشخیص شود.

۶- مراجع

- [1] M. Abu-Khalaf, EE 5322 Neural Networks Notes, Personal Study. [online] Arri.uta.edu /acs /abumurad /EE5322 /EE5322_NN_notes.pdf, 2004.
- [2] M. Botha, V. R. Solms, K. Perry, E. Loubser, G.Y. Port, and E. Technikon, "The utilization of Artificial Intelligence in a Hybrid Intrusion Detection System," Annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology, 2002.
- [3] V. Vapnik, "Statistical Learning Theory," John Wiley & Sons Inc., New York, 1998.
- [4] J. Zhang, R. Perdisci, W. Lee, U. Sarfraz, and X. Luo, "Detecting stealthy P2P botnets using statistical traffic fingerprints," 2011 IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN), Jun. 2011.
- [5] K. Ilgun, R. A. Kemmerer, and P. A. Porras, "State Transition Analysis: A Rule-Based Intrusion Detection Approach," IEEE Trans. Soft. Eng., vol. 21, pp. 181-199, 1995.
- [6] X. Yu, X. Dong, G. Yu, Y. Qin, and D. Yue, "Data-Adaptive Clustering Analysis for Online

پیشنهادی با رویکردهای تشخیص دیگری از جمله روش ارائه شده در [۱۱] و [۱۲] و روش‌های درخت تصمیم‌گیری و جنگل تصادفی که بر روی این مجموعه داده‌ها در این مقاله پیاده‌سازی شده‌اند با توجه به معیارهای صحت، دقت، بازخوانی، میانگین هارمونی و نرخ مثبت کاذب مقایسه می‌شود.



شکل (۶): معیارهای ارزیابی روش پیشنهادی در طول دوره‌ها.

در روش پیشنهادی، تنها اطلاعات مربوط به ارتباطات شبکه موردنیاز بوده و نیاز به بررسی محتوای داده بسته وجود ندارد؛ بنابراین، روش ما در مقابل شبکه‌های باتی که از شیوه‌های رمزنگاری استفاده می‌کنند، ایمن است. اگرچه روش ما می‌تواند بات‌ها را تشخیص دهد و ارتباطات میزبان را به‌عنوان سالم و بدخواه طبقه‌بندی کند، محدودیت روش پیشنهادی ما این است که تنها ترافیک پروتکل کنترل انتقال را برای تشخیص ترافیک شبکه‌بات در نظر می‌گیرد. در تشخیص رفتار شبکه‌بات، چهار مشکل اصلی وجود دارد: اولاً، ترافیک شبکه مداوم است که نشان می‌دهد ترافیک ناپایدار است و ویژگی‌ها در طول زمان تغییر می‌کنند. علاوه بر این، شبکه‌های بات پس از دریافت دستورالعمل از مدیر بات، به‌صورت هوشمند از طریق به‌روزرسانی بات‌ها و یا اصلاح چرخه‌حیات خود، تغییر می‌یابند. ثانیاً، خطر ظهور شبکه‌بات جدید در شبکه و نیز گسترش رفتار مخفی وجود دارد. علاوه بر این، رفتار یک میزبان آلوده ممکن است به نظر رفتار سالمی باشد و اگر طبقه‌بندی برای این رفتار قبلاً آموزش ندیده باشد، شناسایی فعالیت‌های بدخواهانه و مخرب دشوار است. ثالثاً، ارزیابی ترافیک ورودی شبکه در زمان واقعی، به دلیل سرعت و حجم بالای ترافیک شبکه، یک کار محاسباتی پرهزینه است. درنهایت، دسترسی به مجموعه داده ترافیک شبکه‌بات به‌روزرسانی شده، چالشی در تشخیص شبکه‌های بات است. دقت الگوریتم‌های طبقه‌بند بستگی به کیفیت و درستی مجموعه داده‌های آموزشی دارد [۵۷]. قابلیت دسترسی به مجموعه داده‌های شبکه‌بات به دلیل مسئله امنیتی و حریم خصوصی به‌عنوان یک منبع آزمایشگاهی علمی توسعه‌یافته است و برای محققان دستیابی به سایر منابع مانند شبکه‌های سازمان‌های امنیتی و شرکتی جهت ردیابی شبکه‌بات دشوار است.

- [19] S. C. Guntuku, P. P. Narang, and C. Hota, "Real-time Peer-to-Peer Botnet Detection Framework based on Bayesian Regularized Neural Network," arXiv preprint arXiv:1307.7464.
- [20] H. Huy, X. Wei, M. Faloutsos, and T. Eliassi-Rad, "Entelecheia: Detecting p2p botnets in their waiting stage," IFIP Networking Conference, IEEE, pp. 1-9, 2013.
- [21] L. Xu, X. Xu, and Y. Zhuo, "P2P Botnet Detection Using Min-Vertex Cover," *Journal of Networks*, vol. 7, no. 8, Aug. 2012.
- [22] P. Narang, V. Khurana, and C. Hota, "Machine-learning approaches for P2P botnet detection using signal-processing techniques," *Proceedings of the 8th ACM International Conference on Distributed Event-Based Systems - DEBS'14*, 2014.
- [23] Y. Qiao, Y. Yang, J. He, C. Tang, and Y. Zeng, "Detecting P2P bots by mining the regional periodicity," *Journal of Zhejiang University Science C*, vol. 14, no. 9, pp. 682–700, Sep. 2013.
- [24] S. García, A. Zunino, and M. Campo, "Survey on Network-based Botnet Detection Methods," *Security and Communication Networks*, vol. 7, no. 5, pp. 878–903, Jun. 2013.
- [25] A. A. Obeidat, "Analysis the P2P Botnet Detection Methods," *International Journal of Computer Science (IJCS)*, vol. 4, no. (3), pp. 1-11, 2016.
- [26] W. Tarnag, L-Z. Den, K-L. Ou, and M. Chen, "The Analysis and Identification of P2P Botnet's Traffic Flows," *International Journal of Communication Networks and Information Security*, vol. 3, no. 2, pp. 138-148, 2011.
- [27] J. Zhang, R. Perdisci, W. Lee, X. Luo, and U. Sarfraz, "Building a Scalable System for Stealthy P2P-Botnet Detection," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp. 27–38, Jan. 2014.
- [28] B. Rahbarinia, R. Perdisci, A. Lanzi, and K. Li, "PeerRush: Mining for unwanted P2P traffic," *Journal of Information Security and Applications*, vol. 19, no. 3, pp. 194–208, Jul. 2014.
- [29] D. Zhao and I. Traore, "P2P Botnet Detection through Malicious Fast Flux Network Identification," *2012 Seventh International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Nov. 2012.
- [30] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. A. Ghorbani, and D. Garant, "Botnet detection based on traffic behavior analysis and flow intervals," *Computers & Security*, vol. 39, pp. 2–16, Nov. 2013.
- [31] G. Kirubavathi Venkatesh and R. Anitha Nadarajan, "HTTP Botnet Detection Using Adaptive Learning Rate Multilayer Feed-Forward Neural Network," *Lecture Notes in Computer Science*, pp. 38–48, 2012.
- [32] K. Wang, C-Y. Huang, S-J. Lin, and Y-D. Lin, "A fuzzy pattern-based filtering algorithm for botnet detection," *Computer Networks*, vol. 55, no. 15, pp. 3275–3286, Oct. 2011.
- Botnet Detection," *2010 Third International Joint Conference on Computational Science and Optimization*, 2010.
- [7] O. Y. Al-Jarrah, O. Alhussain, P. D. Yoo, S. Muhaidat, K. Taha, and K. Kim, "Data Randomization and Cluster-Based Partitioning for Botnet Intrusion Detection," *IEEE Transactions on Cybernetics*, vol. 46, no. 8, pp. 1796–1806, Aug. 2016.
- [8] W. Lu, M. Tavallaei, G. Rammidi, and A. A. Ghorbani, "BotCop: An Online Botnet Traffic Classifier," *2009 Seventh Annual Communication Networks and Services Research Conference*, May 2009.
- [9] S. Parsa, H. Mortazi, "Botnet Detection with Flow Behavior Analysis Approach," *Journal of Electronical & Cyber Defence*, vol. 5, no. 4, 2017. (In Persian)
- [10] M. Razi and K. Athappilly, "A Comparative Predictive Analysis of Neural Networks (NNs), Nonlinear Regression and Classification and Regression Tree (CART) Models," *Expert Systems with Applications*, vol. 29, no. 1, pp. 65–74, Jul. 2005.
- [11] M. Alauthaman, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks," *Neural Computing and Applications*, vol. 29, no. 11, pp. 991–1004, Oct. 2016.
- [12] A. A. Obeidat, "Hybrid Approach for Botnet Detection Using K-Means and K-Medoids with Hopfield Neural Network," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 9, no. 3, pp. 305-313, 2017.
- [13] W. Xianglin, J. Fan, M. Chen, A. Tarem, and A. S. K. Pathan, "SMART: A Subspace based Malicious Peers Detection Algorithm for P2P Systems," *International Journal of Communication Networks and Information Security*, vol. 5, pp. 1-9, 2013.
- [14] S. Saad, I. Traore, A. A. Ghorbani, B. Sayed, D. Zhao, Wei Lu, J. Felix, and P. Hakimian, "Detecting P2P Botnets through Network Behavior Analysis and Machine Learning," *2011 Ninth Annual International Conference on Privacy, Security and Trust*, Jul. 2011.
- [15] N. Kheir and C. Wolley, "BotSuer: Suing Stealthy P2P Bots in Network Traffic through Netflow Analysis," *Lecture Notes in Computer Science*, pp. 162–178, 2013.
- [16] J. Kang, Y-Z. Song, and J-Y. Zhang, "Accurate Detection of Peer-to-Peer Botnet using Multi-Stream Fused Scheme," *Journal of Networks*, vol. 6, no. 5, May 2011.
- [17] T. Cholez, I. Chrisment, O. Festor, and G. Doyen, "Detection and Mitigation of Localized Attacks in a Widely Deployed P2P Network," *Peer-to-Peer Networking and Applications*, vol. 6, no. 2, pp. 155–174, May 2012.
- [18] Y. Fan and N. Xu, "A P2P Botnet Detection Method Used On-line Monitoring and Off-line Detection," *International Journal of Security and Its Applications*, vol. 8, no. 3, pp. 87–96, May 2014.

- [46] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "BotMiner: Clustering Analysis of Network Traffic for Protocol and Structure-independent Botnet Detection," USENIX security symposium, 2008.
- [47] T-F. Yen and M. K. Reiter, "Traffic Aggregation for Malware Detection," Lecture Notes in Computer Science, pp. 207–227.
- [48] C. Livadas, R. Walsh, D. Lapsley, and W. T. Strayer, "Using Machine Learning Techniques to Identify Botnet Traffic," Proceedings, 2006 31st IEEE Conference on Local Computer Networks, Nov. 2006.
- [49] P. van der Putten and M. van Someren, "A Bias-Variance Analysis of a Real World Learning Problem: The CoIL Challenge 2000," Machine Learning, vol. 57, no. 1/2, pp. 177–195, Oct. 2004.
- [50] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
- [51] J. Woodbridge, H. S. Anderson, A. Ahuja, and D. Grant, "Predicting Domain Generation Algorithms with Long Short-Term Memory Networks," preprint arXiv:1611.00791, 2016.
- [52] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," preprint arXiv:1412.6980, 2014.
- [53] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 2014.
- [54] M. Abadi, et al, "TensorFlow: Large-scale machine learning on heterogeneous systems," Accessed: 2017-05-28 [Online]. Available: <http://tensorflow.org/>
- [55] F. Chollet, "Keras," Accessed: 2017-05-28 [Online]. Available: <https://github.com/fchollet/keras>
- [56] A. Shiravi, H. Shiravi, M. Tavallae, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," Computers & Security, vol. 31, no. 3, pp. 357–374, May 2012.
- [57] J-D. Wang and H-C. Liu, "An approach to evaluate the fitness of one class structure via dynamic centroids," Expert Systems with Applications, May 2011.
- [33] C-Y. Huang, "Effective bot host detection based on network failure models," Computer Networks, vol. 57, no. 2, pp. 514–525, Feb. 2013.
- [34] H. Dhayal and J. Kumar, "Peer-to-Peer Botnet Detection based on Bot Behaviour," International Journal of Advanced Research in Computer Science, vol. 8, no. 3, pp. 172-175, 2017.
- [35] R. Chen, W. Niu, X. Zhang, Z. Zhuo, and F. Lv, "An Effective Conversation-Based Botnet Detection Method," Mathematical Problems in Engineering, vol. 2017, pp. 1–9, 2017.
- [36] Z. Yang and B. Wang, "A Feature Extraction Method for P2P Botnet Detection Using Graphic Symmetry Concept," Symmetry, vol. 11, no. 3, p. 326, Mar. 2019
- [37] H. R. Zeidanloo, M. J. Z. Shooshtari, P. V. Amoli, M. Safari, and M. Zamani, "A Taxonomy of Botnet Detection Techniques," 2010 3rd International Conference on Computer Science and Information Technology, Jul. 2010.
- [38] K-S. Han, K-H. Lim, and E-G. Im, "The Traffic Analysis of P2Pbased Storm Botnet Using Honeynet," Journal of the Korea Institute of Information Security and Cryptology, vol. 19, no. 4,
- [39] S-K. Noh, J-H. Oh, J-S. Lee, B-N. Noh, and H-C. Jeong, "Detecting P2P Botnets Using a Multi-phased Flow Model," 2009 Third International Conference on Digital Society, Feb. 2009.
- [40] C. Li, W. Jiang, and X. Zou, "Botnet: Survey and Case Study," 2009 Fourth International Conference on Innovative Computing, Information and Control (ICICIC), Dec. 2009.
- [41] G. Sinclair, C. Nunnery, and B. B. Kang, "The Waledac Protocol: The How and Why," 2009 4th International Conference on Malicious and Unwanted Software (MALWARE), Oct. 2009.
- [42] S. Shin, G. Gu, N. Reddy, and C. P. Lee, "A Large-Scale Empirical Study of Conficker," IEEE Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 676–690, Apr. 2012.
- [43] T. Holz, M. Steiner, F. Dahl, E. Biersack, and F. Freiling, "Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm," LEET journal, vol. 8, no. 1, pp. 1-9, 2008.
- [44] H. Binsalleeh, T. Ormerod, A. Boukhtouta, P. Sinha, A. Youssef, M. Debbabi, and L. Wang, "On the analysis of the Zeus botnet crimeware toolkit," 2010 Eighth International Conference on Privacy, Security and Trust, Aug. 2010.
- [45] G. Gu, J. Zhang, and W. Lee, "BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic," 15th Annual Network and Distributed System Security Symposium, 2008.

P2P Botnet Detection Using Deep Learning Method

M. Asadi, S. Parsa*, M. A. Jabreil Jamali, V. Majidnezhad

*Science and Technology University, Tehran, Iran

(Received: 23/04/2019, Accepted: 02/10/2019)

ABSTRACT

A Botnet is a set of infected computers and smart devices on the Internet that are controlled remotely by a Botmaster to perform various malicious activities like distributed denial of service attacks(DDoS), sending spam, click-fraud and etc. When a Botmaster communicates with its own Bots, it generates traffic that analyzing this traffic to detect the traffic of the Botnet can be one of the influential factors for intrusion detection systems (IDS). In this paper, the long short term memory (LSTM) method is proposed to classify P2P Botnet activities. The proposed approach is based on the characteristics of the transfer control protocol (TCP) packets and the performance of the method is evaluated using both ISCX and ISOT datasets. The experimental results show that our proposed approach has a high capability in identifying P2P network activities based on evaluation criteria. The proposed method offers a 99.65% precision rate, a 96.32% accuracy rate and a recall rate of 99.63% with a false positive rate (FPR) of 0.67%.

Keywords: Botnet, Botnet Detection, Deep Learning, Recurrent Neural Network (RNN), Long Short Term Memory (LSTM)