Electronic & Cyber Defense
Vol. 12, No. 4, Winter 2025, Serial No. 48,pp.45-54
ISSN: 2322-4347, E-ISSN: 2980-8979

Host-based Anomaly Malware Detection Using Deep Learning

N. Alaei, A. Tajoddin*

*Assistant Professor, University of Zanjan, Zanjan, Iran

(Received: 2024/10/15, Revised: 2024/12/03, Accepted: 2025/01/12, Published: 2025/02/01)

DOR: https://dor.isc.ac/dor/20.1001.1.23224347.1403.12.4.4.8

Abstract:

Windows operating system, as the most widely used operating system of desktop computers, is still one of the main targets of malware writers. For this reason, many researches have recently been conducted to detect Windows malware. Due to the emergence and application of deep learning, although researchers have been able to use it to detect Windows malware, but there are still various challenges such as the detection of new and zero-day malwares and lack of evolution of processes of the feature engineering that increase the false positive rate. Currently, deep learning based malware detection approaches are either two class or multi classes, which fail to detect anomaly and zero-day malware. in this research, in addition to using a combination of various features of static and dynamic including file, registry, network, calls and PE import names, we also have increased the number and variety of normal datasets using the conditional tabular generative adversarial model for more accurate training, then we made it possible to detect anomalies and zero-day malware by presenting the deep approach of one-class generative adversarial network model. The result of the research includes a false alarm rate of approximately 1% with a high detection rate of 99% that compared to similar methods, indicates the success of the proposed method.

Keywords: Anomaly Detection, Windows Malware, One-Class Deep Learning, Deep Generative Adversarial Network

"پدافند الکترونیکی و سایبری"
سال دوازدهم، شماره ۴، زمستان ۱۴۰۳، ص۵۴-۴۵
شاپا الکترونیکی: 2980-8979 شاپا چاپی: 2322-4347
علمی – پژوهشی

تشخیص بدافزارهای ناشناخته در سطح میزبان مبتنی بر یادگیری عمیق

نسرین اعلایی [۱]، اصغر تاج الدین [۲*]
۱-کارشناسی ارشد ، ۲- استادیار، دانشگاه زنجان، زنجان، ایران

(دریافت: ۱۴۰۳/۰۷/۲۴، بازنگری: ۱۴۰۳/۰۹/۱۳، پذیرش: ۱۴۰۳/۱۰/۲۳، انتشار: ۱۴۰۳/۱۱/۱۳)
DOR: https://dor.isc.ac/dor/20.1001.1.23224347.1403.12.4.5.9

چکیده

سیستم‌عامل ویندوز به‌عنوان پرکاربردترین سیستم‌عامل رایانه‌های رومیزی، کماکان یکی از اهداف اصلی بدافزار نویسان است. به همین دلیل طی سال‌های اخیر تحقیقات و پژوهش‌های متعددی برای تشخیص بدافزارهای ویندوزی انجام‌شده است. با ظهور و کاربرد یادگیری عمیق، هرچند محققان توانستند از آن برای تشخیص بدافزارهای ویندوزی بهره گیرند، اما هنوز چالش‌های مختلفی ازجمله تشخیص بدافزارهای جدید و روز صفر و عدم تکامل فرآیند مهندسی ویژگی وجود دارد که موجب افزایش نرخ هشدار نادرست می‌شود. در حال حاضر روش‌های تشخیص بدافزار ارائه‌شده به کمک یادگیری عمیق دو یا چند کلاسه هستند که امکان تشخیص ناهنجاری و بدافزارهای روز صفر را ندارند. بنابراین به‌منظور بهبود این چالش‌ها، از یک رویکرد تشخیص بدافزار مبتنی بر ناهنجاری به کمک یادگیری عمیق، در این پژوهش استفاده‌شده است. درواقع در این پژوهش ترکیبی از انواع ویژگی‌های ایستا و پویا ازجمله ویژگی‌های فایل، رجیستری، شبکه، فراخوانی‌های سیستمی و نام‌های درج PE را به‌کاربرده و پس‌ازآن با ارائه رویکرد مدل شبکه خصمانه تک کلاسه عمیق، امکان تشخیص ناهنجاری و شناسایی بدافزارهای روز صفر را فراهم کردیم. همچنین برای ارزیابی روش پیشنهادی، از دو مجموعه داده استفاده‌شده است که نقطه قوت این مجموعه داده‌ها وجود اکثر نمونه‌های بدافزارهاست و نقطه‌ضعف آن نیز تعداد کم نمونه داده‌های عادی است و ازآنجاکه یک مدل شبکه عمیق نیاز به حجم زیاد داده با تعداد رکورد فراوان دارد تا با دقت بالا آموزش ببیند و خروجی با درصد خطای کمتر را نمایش دهد، تصمیم گرفتیم به کمک مدل مولد متخاصم جدول شرطی، تعداد و تنوع مجموعه داده‌های عادی را برای آموزش دقیق‌تر افزایش دهیم و به نتایج پژوهش شامل نرخ هشدار نادرست تقریبی ۱٪ به همراه نرخ تشخیص بالای ۹۹٪ در مقایسه با روش‌های مشابه و روش‌های چند کلاسه، برسیم که بیانگر موفقیت روش پیشنهادی است.

واژگان کلیدی: تشخیص ناهنجاری، بدافزار ویندوز، یادگیری عمیق تک کلاسه، شبکه مولد خصمانه عمیق

۱- مقدمه

یک بدافزار، نرم‌افزار و یا میان‌افزاری است که قصد انجام فرآیندی غیرمجاز باهدف نقض امنیت یک سیستم اطلاعاتی را دارد. بسته به اهداف و سیستم‌های تکثیر، بدافزارها را می‌توان به انواع مختلف ازجمله ویروس، کرم، تروجان، باج افزار، روت کیت و غیره تقسیم کرد. به‌طورکلی، روش‌های شناسایی بدافزار به دودسته اصلی تقسیم می‌شوند: ۱) تشخیص سوءاستفاده یا تشخیص مبتنی بر امضا و ۲) تشخیص ناهنجاری. روش‌های تشخیص سوءاستفاده معمولاً از تطبیق الگو با پایگاه داده‌ای از امضاهای بدافزار شناخته‌شده، استفاده می‌کنند و بنابراین نمی‌توانند نمونه‌های بدافزار جدید را تشخیص دهند. برای غلبه بر این نقص، روش‌های تشخیص ناهنجاری وجود دارد که با رویکردی معکوس، رفتار طبیعی سیستم را تعریف و هر رفتار دیگری را غیرطبیعی تلقی می‌کند [۱]. برای تشخیص یک بدافزار دو نوع رویکرد تحلیل ایستا و پویا وجود دارد. تحلیل ایستا شامل بررسی یک نرم‌افزار بدون اجرای آن است. از سوی دیگر، تحلیل پویا شامل بررسی نرم‌افزار با اجرای آن است. تحلیل ایستا سریع‌تر است، اما اگر بدافزار با استفاده از فناوری‌های مبهم سازی کد[1]، پنهان شود، آنگاه امکان تشخیص آن وجود ندارد. برعکس،

[1] Obfuscation.

استناد: اعلایی، نسرین، تاج الدین، اصغر "تشخیص بدافزارهای ناشناخته در سطح میزبان مبتنی بر یادگیری عمیق"، پدافند الکترونیکی و سایبری، ۱۲(۴)، ۵۴-۴۵، ۱۴۰۳. https://dor.isc.ac/dor/20.1001.1.23224347.1403.12.4.5.9
ناشر: دانشگاه جام امام حسین(ع).

* رایانامه نویسنده مسئول: tajoddin@znu.ac.ir

فناوری‌های پنهان‌سازی کد و بدافزار چندریختی[1] را می‌توان به‌وسیله تحلیل پویا شناسایی کرد، چراکه در زمان اجرای برنامه، بدافزار رفتار مخربانه خود را آشکار می‌کند. پژوهشگران امنیتی در تحقیقات خود به این نتیجه رسیدند که تشخیص بدافزارها تنها با استفاده از رویکرد مبتنی بر امضا به علت عدم توانایی در تشخیص بدافزارهای ناشناخته، ازجمله بدافزارهای روز صفر کافی نیست. برای حل این چالش از روش‌های مبتنی بر تشخیص ناهنجاری استفاده شد. متأسفانه، روش‌های تشخیص ناهنجاری به دلیل عدم توانایی در ساخت مدل‌های پیچیده از برنامه‌های عادی، هشدار مثبت کاذب بالایی دارند. ازاین‌رو، توسعه روش‌های تشخیص ناهنجاری که نمونه‌های بدافزار جدید را کشف می‌کنند درحالی‌که میزان هشدار کاذب کمی دارند، یک زمینه تحقیقاتی چالش‌برانگیز است [۱]. یکی از مهم‌ترین چالش‌های موجود در تشخیص ناهنجاری و تشخیص بدافزارهای ناشناخته، به دست آوردن یک مجموعه آموزشی بزرگ از داده‌های برچسب خورده است که کار زمان‌بری است. بنابراین، می‌توان از روش‌های تشخیص ناهنجاری نیمه نظارت‌شده و نظارت‌نشده در این زمینه استفاده کرد. در این پژوهش یک مدل نظارت‌نشده مبتنی بر یادگیری عمیق پیشنهاد می‌شود که در این مدل تنها از داده‌های عادی برای آموزش سیستمِ تشخیص استفاده شد. همچنین به‌منظور بهبود عملکرد از فرایند مهندسی ویژگی و انتخاب ویژگی‌های مناسب استفاده می‌شود. به‌طور خلاصه مشارکت و نوآوری‌های ارائه‌شده در این پژوهش به‌صورت زیر است:

۱- استخراج و انتخاب ویژگی‌های ترکیبی که شامل کلیدها و مقادیر رجیستری، ویژگی‌های مرتبط با فایل، عملیات شبکه، فراخوانی‌های سیستمی و به‌طورکلی ویژگی‌های استخراجی سطح بالا و همچنین ویژگی‌های ایستا که شامل pe import name‌ها است. مزیت اصلی نظارت بر ویژگی‌های سطح بالا این است که می‌توان رفتار عادی را از رفتار مخرب با سربار محاسباتی کم تشخیص داد [۲].

۲- تولید و افزایش تعداد داده‌های مجموعه داده ارائه‌شده به کمک شبکه متخاصم عمیق.

۳- ارائه یک مدل تشخیص ناهنجاری با رویکرد یادگیری عمیق تک کلاسه[2] با نرخ مثبت کاذب پایین و نرخ تشخیص بالا و به‌ویژه تشخیص بدافزارهای روز صفر. مزیت استفاده از روش تک کلاسه، آن است که می‌تواند در تشخیص بدافزارهای جدید و ناشناخته مؤثر باشد.

در بخش بعد، بخش دوم، پژوهش‌های مشابه این حوزه بیان‌شده است. در بخش سوم به معرفی مفاهیم پایه پرداخته‌شده است. در بخش چهارم سامانه تشخیص بدافزار پیشنهادی معرفی‌شده است. در بخش پنجم نتایج شبیه‌سازی‌ها و مقایسه‌ها ارائه‌شده است و در انتهای مقاله نیز به نتیجه‌گیری نهایی پرداخته‌شده است.

۲. مروری بر مقالات پیشین

در طول سال‌های گذشته، روش‌های متنوعی به‌منظور شناسایی فعالیت‌های بدافزارها ارائه‌شده است که در ادامه به‌اختصار به چند مورداشاره می‌شود.

جیاکن لی و همکارانش در سال ۲۰۱۳ میلادی [۳] یک رویکرد یادگیری ماشین تک کلاسه به نام FENOC برای تشخیص بدافزار ارائه کردند که در این رویکرد از ویژگی‌های ترکیبی از لایه‌های معنایی چندگانه، شامل ویژگی‌های ایستا، ویژگی‌های پویا و نمودارهای رفتاری استفاده‌شده که توانست در تشخیص بدافزار، به‌ویژه در داده‌های نامتعادل، نسبت به مدل باینری بهتر عمل کند و نرخ مثبت کاذب کمتری داشته باشد. آدریان تانگ و همکارانش در سال ۲۰۱۴ میلادی [۴] از ویژگی‌های سطح پایین‌تر پشتیبانی شده از سخت‌افزار را برای شناسایی بدافزار در یک مدل مبتنی بر ناهنجاری استفاده کردند که این کار موجب می‌شد تا طیف وسیع‌تری از بدافزارها، حتی بدافزارهای روز صفر را شناسایی کنند. کیگوانگ و همکارانش در سال ۲۰۱۶ میلادی [۵] یک روش تحلیل معنایی دنباله‌های API پیشنهاد دادند که عملیات بر روی منابع حساس سیستم و رفتارهای پیچیده به روشی قابل تفسیر در لایه‌های معنایی مختلف انتزاع می‌شوند. علاوه بر این، برای مقابله با این مشکل که بدافزارها و برنامه‌های عادی به‌شدت نامتعادل هستند، یک ماشین بردار پشتیبانی یک کلاس بهبودیافته به نام OC-SVM-Neg پیشنهاد شد که توانستند به‌دقت ۹۴٪ برسند. محمود یوسفی آذر و همکارانش در سال ۲۰۱۷ میلادی [۶] یک روش یادگیری ویژگی بدون نظارت برای طبقه‌بندی بدافزار و تشخیص ناهنجاری مبتنی بر شبکه با استفاده از رمزگذار خودکار ارائه دادند. این روش از حداقل ویژگی‌ها در مقایسه با الگوریتم‌های پیشرفته دیگر استفاده می‌کند که باعث می‌شود مدل ازنظر محاسباتی برای حفاظت از زمان واقعی کارآمدتر باشد. جین یانگ کیم و همکارانش در سال ۲۰۱۸ میلادی [۷] روشی را برای پیش آموزش مدل GAN با استفاده از یادگیری انتقال[3] پیشنهاد کرده‌اند که این روش بهترین عملکرد را در مقایسه با سایر مدل‌های معمولی نشان داد و توانست بدافزارها را حتی با مقدار کمی داده شناسایی کند. برای این مدل از مجموعه داده Microsoft Malware Classification Challenge استفاده کرده و به‌دقت ۹۶/۳۹ درصد رسیدند و در سال ۲۰۱۸ میلادی [۸] روشی به نام شبکه متخاصم مولد عمیق کانوولوشن انتقالی[4] پیشنهاد کرده‌اند که در این روش، آشکارساز ویژگی‌های مختلف بدافزار را با استفاده از داده‌های واقعی و داده‌های اصلاح‌شده تولیدشده توسط tDCGAN بر اساس رمزگذار خودکار عمیق[5] می‌آموزد. برای این روش از مجموعه داده Microsoft Malware Classification

[1] Polymorphic Malware.
[2] One Class Classification.
[3] transfer learning.
[4] transferred deep-convolutional GAN (tDCGAN).
[5] deep autoencoder (DAE).

Challenge استفاده‌شده و به‌دقت ۹۵/۷۴٪ درصد رسیدند. تاج‌الدین و همکارش در سال ۲۰۱۹ میلادی [۲] یک رویکرد جدید تشخیص بدافزار مبتنی بر رجیستری ارائه دادند که از طبقه‌بندی گروهی متشکل از چند طبقه‌بندی تک کلاسه برای شناسایی بدافزار شناخته‌شده و ناشناخته استفاده می‌کند. ایجاز و همکارانش در سال ۲۰۱۹ میلادی [۹] از ویژگی‌های پویا و ایستا در کار خود استفاده کردند که به‌صورت جداگانه ویژگی‌های پویا و ایستا را با چندین مدل یادگیری ماشین ارزیابی کردند و به نتایج بهتری با ویژگی ایستا رسیدند. خاتونی و همکارانش در سال ۲۰۱۹ میلادی [۱۰] به بررسی رفتار بدافزارها در شبکه پرداختند و از ویژگی‌هایی همچون اطلاعات مربوط به‌TCP، UDP، URLs و پورت‌ها استفاده کردند. ویژگی‌ها را به مدل درخت تصادفی برای طبقه‌بندی دادند که توانستند به‌دقت ۹۷ درصد برسند. الحیدری و همکارانش در سال ۲۰۲۲ میلادی [۱۱] از دو ویژگی‌های API Call پویا و PE Imports ایستا استفاده کردند و با مدل جنگل تصادفی و درخت تصمیم برای ویژگی ایستا به‌دقت ۹۷/۸۸ درصد و برای ویژگی پویا به‌دقت ۸۷/۹۰ درصد رسیدند. احسان اله شقاقی و همکارانش در سال ۲۰۲۲ میلادی [۱۲] یک روش کاشف که شامل دو ماژول ایستا، برای استخراج ویژگی‌های سرآیند فایل اجرایی، و دو ماژول رفتاری برای استخراج ویژگی‌هایی برای تولید امضاء و مدل رفتاری بداندیش بر اساس روش‌های یادگیری ماشین است، ارائه دادند و توانستند با جنگل تصادفی به‌دقت ۹۹ درصد برسند. هدف ما در این تحقیق ارائه معیاری دقیق‌تر و کاراتر به‌منظور تشخیص بدافزارهای ویندوز به‌ویژه بدافزارهای روز صفر است.

۳. مفاهیم پایه

در این بخش به‌طور مختصر به معرفی مفاهیم به کار گرفته‌شده در این مقاله پرداخته می‌شود.

۳-۱. شبکه‌ی عصبی مولد خصمانه‌ی عمیق

یک شبکه مولد خصمانه عمیق[1]، شامل دو بخش اصلی مولد[2] و بخش متمایز گر[3] است. مولد یک بردار نویزی تصادفی از توزیع گوسی[4] یا هر توزیع تصادفی دیگری را به‌عنوان ورودی دریافت کرده و با الگو قرار دادن نمونه‌های آموزشی واقعی، سعی در نگاشت توزیع داده‌های تصادفی به توزیع داده‌های واقعی مدنظر، می‌کند. سپس نمونه داده‌های جعلی[5] تولیدشده توسط مولد به همراه نمونه داده‌های واقعی، به بخش متمایز گر داده می‌شود تا از یکدیگر تفکیک شوند. تابع هدف شبکه مولد خصمانه طبق رابطهٔ (۱) است:

$$\text{Min Max } V(D,G) = \text{Log}(D(x)) + \text{Log}(1- D(G(z))) \quad (۱)$$

که در آن D و G به ترتیب به شبکه‌های متمایز گر و مولد اشاره دارند.

۳-۲. شبکه متخاصم مولد جدول شرطی

شبکه متخاصم مولد جدول شرطی[6]، برای داده‌های عددی پیوسته و گسسته مناسب است که مولد باتوجه‌به یک شرط که به‌صورت یک بردار شرطی است و بر روی یکی از ستون‌های گسسته است، داده مصنوعی تولید می‌کند. این بردار شرطی نشان‌دهنده بردارهای one-hot به‌هم‌پیوسته‌ای از تمام ستون‌های گسسته است که فقط مشخصات یک دسته از مجموعه داده انتخاب می‌شود. در CTGAN، از نرمال‌سازی[7] حالت خاص برای غلبه بر ستون‌هایی با توزیع پیچیده استفاده می‌شود. در این نوع نرمال‌سازی هر ستون به‌طور مستقل محاسبه می‌شود. ازآنجایی‌که ستون‌های یک ردیف مستقل از هم هستند، از شبکه‌های کاملاً متصل[8] در مولد و متمایز گر استفاده‌شده است تا تمام همبستگی‌های ممکن بین ستون‌ها در نظر گرفته شود. در مولد از تابع فعال‌ساز Relu استفاده می‌شود. در متمایز گر، از دولایه مخفی کاملاً متصل با تابع relu و dropout در هر لایه پنهان استفاده‌شده است. مدل با استفاده از Cross Entropy Loss آموزش داده‌شده و همچنین از بهینه‌ساز Adam با نرخ یادگیری[9] $۲/۱۰^{-۴}$ استفاده‌شده است [۱۳].

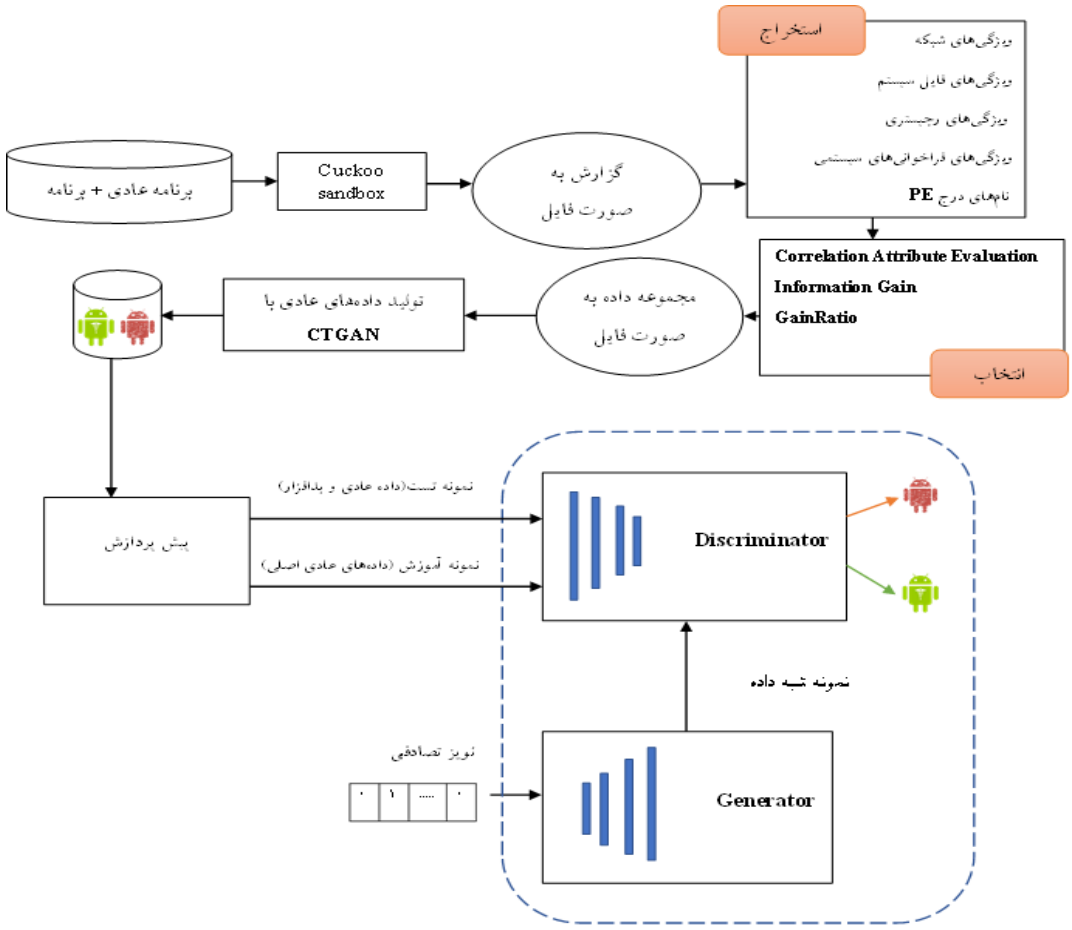
۴. ارائه روش پیشنهادی

در این بخش به معرفی و تشریح رویکرد پیشنهادی به‌منظور تشخیص خودکار بدافزار پرداخته خواهد شد. معماری کلی روش پیشنهادی در شکل (۱) آورده شده است. ابتدا مراحل پیش‌پردازش داده و فرایند استخراج ویژگی را شرح داده و درنهایت مدل آموزشی را بررسی می‌کنیم.

۴-۱. استخراج ویژگی‌ها

اولین مرحله از فرآیند شناسایی بدافزار استخراج ویژگی‌های داده‌ها است. تبدیل مجموعه بزرگ و مبهم ورودی‌ها به مجموعه ویژگی‌ها که الگوهای مورداستفاده برای نشان دادن نمونه فایل‌ها است، استخراج ویژگی نامیده می‌شود. ویژگی‌های پویا از مجموعه داده در جدول (۱) آورده شده است. ویژگی‌های ایستا نیز شامل توابع ورودی است که برای تحلیل اینکه یک فایل اجرایی چه انجام می‌دهد، حدود ۴۰۰۰۰ از این توابع را که در کل برنامه‌های موجود در مجموعه داده خود وجود داشت را استخراج کردیم.

[1] generative adversarial network (GAN).
[2] Generator.
[3] Discriminator.
[4] Gaussian Distribution.
[5] Fake Sample.
[6] Conditional Tabular GAN (CTGAN).
[7] Normalization.
[8] Fully Connected.
[9] Learning rate.



شکل (۱). معماری روش پیشنهادی

جدول (۱): ویژگی‌های پویای استخراج‌شده

ویژگی‌ها	توضیحات	ویژگی‌ها	توضیحات
F2	تعداد بسته‌های tcp/udp دریافتی و ارسالی با شماره پورت مقصد یکتا در شبکه	F1	تعداد کل بسته‌های tcp/udp دریافتی و ارسالی در شبکه
F4	تعداد فایل‌هایی که باز شدند	F3	تعداد بسته‌های tcp/udp دریافتی و ارسالی با آدرس مقصد یکتا در شبکه
F6	تعداد فایل‌هایی که در آن‌ها نوشته‌شده	F5	تعداد فایل‌هایی که خوانده شدند
F8	تعداد فایل‌هایی که خاتمه یافتند و بسته شدند	F7	تعداد فایل‌هایی که مجدد ایجاد شدند
F10	تعداد کلیدهای رجیستری که باز شدند	F9	تعداد فایل‌هایی که خراب شدند
F12	تعداد کلیدهای رجیستری که نوشته شدند	F11	تعداد کلیدهای رجیستری که خوانده شدند
F14	تعداد کل فراخوانی‌ها	F13	تعداد کلیدهای رجیستری که حذف شدند
F16	تعداد کل فراخوانی‌هایی با status =۱	F15	تعداد کل فراخوانی‌هایی که در یک دسته‌بندی هستند
F18	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی رجیستری هستند	F17	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی سیستمی هستند
F20	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی شبکه‌ای هستند	F19	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی پردازشی هستند
F22	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی منابع هستند	F21	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی فایل هستند
F24	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی Ui هستند	F23	تعداد کل فراخوانی‌هایی که در دسته‌بندی فراخوانی مدیریتی(مثل زمان) هستند

۴-۲. انتخاب ویژگی

انتخاب ویژگی‌های مناسب از داده‌ها متناسب با نوع مسئله و مدل یکی دیگر از گام‌های مهم در فرآیند شناسایی و طبقه‌بندی بدافزار است. ویژگی‌های ایستای استخراج‌شده، نام توابع ورودی فایل‌های pe است که بسیاری از این توابع، توابعی است که توسط کامپایلر برای تمام برنامه‌ها فراخوانی می‌شوند که این توابع برای تمیز کردن بدافزار از برنامه عادی اهمیتی ندارند و در عوض توابعی که توسط خود برنامه‌نویس به برنامه‌ها اضافه می‌شوند و ارتباط بیشتری با بدافزار دارند، حائز اهمیت هستند و باید تحلیل شوند. لذا به‌منظور انتخاب این نوع توابع برتر نسبت به‌کل توابع موجود، به‌صورت مجزا ویژگی‌ها را به کمک سه روش رتبه‌بندی: ۱) correlation Attribute، ۲) GainRatio Attribute، ۳) InfoGain Attribute، رتبه‌بندی کردیم و درنهایت با تحلیل و اشتراک گرفتن از خروجی‌های این سه رتبه‌اند، ۱۲۸۰ ویژگی برتر انتخاب شد.

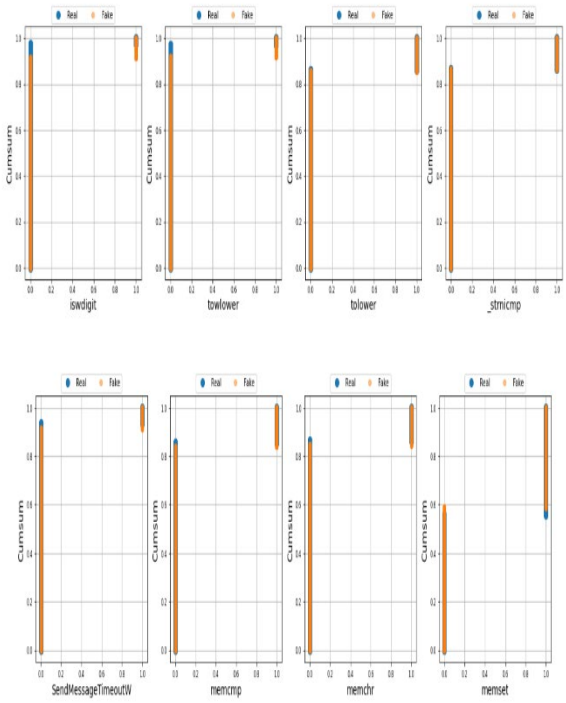
۴-۳. تولید داده‌های عادی

باتوجه‌به اینکه یک مدل شبکه عمیق نیاز به حجم زیاد داده با تعداد رکورد فراوان دارد تا با دقت بالا آموزش ببیند و خروجی با درصد خطای کمتر را نمایش دهد و باتوجه‌به اینکه تعداد مجموعه داده‌های بیان‌شده کم است و همچنین عدم وجود مجموعه داده مناسب دیگر، تصمیم گرفتیم از شبکه‌های مولد تخاصمی برای تولید مجموعه داده موردنیاز خود استفاده کنیم. برای افزایش تعداد داده‌های مجموعه داده خود از مدل CTGAN استفاده کردیم که این مدل از میان انواع مدل‌های تولید برای مجموعه داده‌های عددی جدولی کاربرد بیشتری دارد و نتیجه بهتری می‌دهد. باتوجه‌به اینکه روش ما تک کلاسه بودن مدل است نیازی به‌اضافه کردن تعداد داده‌های بدافزار نبود، بنابراین داده‌های عادی را از بدافزار جدا کرده و مدل با روش hold out cross validation با نسبت ۳۰ -۷۰، ۷۰ برای آموزش و ۳۰ برای آزمون، تقسیم‌شده و از ۲۲۸۶ داده عادی، ۶۰۰۰۰ داده عادی نمونه و مشابه داده اصلی تولید کردیم و کل داده‌های عادی ما با احتساب داده‌های اولیه عادی به ۶۲۲۸۶ رسید و داده‌های بدافزار هم به همان تعداد قبل، ۷۶۰۷، است که جمعاً مجموعه داده ما یک فایل csv با تعداد ۱۳۰۶ ویژگی و ۶۹۸۹۴ فایل داده است.

باتوجه‌به اینکه مجموعه داده دوم نیز مشابه مجموعه داده مرجع ما تعداد برنامه سالم کمی برای آموزش طبقه‌بندی ما دارد، این مجموعه داده را نیز مشابه به مجموعه داده اول به مدل CTGAN داده و تعداد داده‌های سالم را به ۶۱۶۰۰ رساندیم. میزان شباهت چند نمونه از ویژگی واقعی با ویژگی تولیدشده توسط مدل در شکل (۲) نشان داده‌شده است.

۴-۴. پیش‌پردازش

یکی از موضوعات پراهمیت در بخش آماده‌سازی داده‌ها، موضوع تغییر مقیاس[1] داده‌ها است که معمولاً توسط دو روش نرمال‌سازی و استانداردسازی[2] صورت می‌گیرد. مجموعه داده ما نیز به علت وجود داده‌ها در بازه‌های خیلی متفاوت از همدیگر نیاز به نرمال‌سازی داشت که از روش متداول Min-Max برای نرمال‌سازی مقادیر مجموعه داده استفاده کردیم. مقادیر در بازهٔ ۰ تا ۱ نرمال شده‌اند.



شکل (۲): مقایسه چند نمونه از داده‌های عادی تولیدشده و داده‌های عادی اصلی

۴-۵. مدل طبقه بند

شبکه‌های متخاصم مولد عمیق یک نوع شبکه عصبی عمیق از نوع مدل‌های مولدی هستند که با استفاده از دو بخش متخاصم بهبود می‌یابند. این دو بخش شبکه‌های عصبی چندلایه هستند که شامل یک مولد و یک متمایزکننده است. مولد در این شبکه مسئول تولید داده‌های جدید است. هدف آن تولید داده‌های جدید به‌گونه‌ای است که به نظر برسند که از توزیع دادگان واقعی گرفته‌شده‌اند که درواقع مولد الگوها و ویژگی‌های موجود در دادگان واقعی را در دادگان تولیدشده بازتولید می‌کند. متمایزکننده یک طبقه‌بندی باینری است که وظیفه تشخیص بین دادگان واقعی و دادگان تولیدشده را دارد. هدف آن این است که بتواند با تشخیص صحیح بین این دو نوع داده، مولد را بهبود بخشید. به‌طور خلاصه، تمیز دهنده بین یک نمونهٔ حقیقی و یک نمونهٔ ساختگی تفکیک ایجاد می‌کند. شبکه متخاصم مولد تک کلاسه نیز یک نوع از

[1] Re-scaling.
[2] Standardization.

شبکه‌های متخاصم عمیق است که برای تشخیص داده‌های ناشناخته و نامعتبر استفاده می‌شود. در این شبکه، هدف اصلی این است که تشخیص دهد که آیا یک نمونه ورودی به توزیع داده‌های آموزش‌دیده شده متعلق است یا خیر. بنابراین در مدل طبقه‌بندی پیشنهادی ما، هدف مولد، تولید نمونه‌های شبه داده سالم است و هدف متمایزکننده جداسازی داده‌های سالم و بدافزار از همدیگر است. مولد با دریافت یک بردار نویز به‌عنوان ورودی، داده‌های جدیدی را تولید می‌کند که به کلاس موردنظر شبیه باشد. این بردار نویز که یک بردار چندبعدی است، به‌عنوان منبع تصادفی برای تولید داده‌های جدید استفاده می‌شود. برای آموزش مدل، بردار نویز از توزیع گاوسی یا توزیع یکنواخت نمونه‌برداری می‌شود. این بردار نویز معمولاً به‌صورت تصادفی در طول آموزش ایجاد می‌شود و مولد با تلاش برای بهبود کیفیت داده‌های تولیدشده، بهترین بردار نویز را پیدا می‌کند. ابعاد بردار تعریف‌شده به این صورت است که خانه اول آن تعداد داده‌های ورودی شبکه مولد را نشان می‌دهد و خانه دوم نشان‌دهنده ابعاد داده است.

از طرفی دیگر یک دسته از داده‌های هنجار را جدا کرده و همراه با بردار نویزی تولیدشده توسط مولد، به شبکه متمایز گر می‌دهیم. ابعاد بردارهای هنجار نیز همانند ابعاد بردارهای نویزی است. در شبکه متمایز گر، برچسب داده به‌صورت یک کلاس وجود دارد. این برچسب برای تمام داده‌های سالم و معتبر که به‌عنوان داده ورودی استفاده می‌شوند، یک برچسب مشخص (مثلاً ۰) استفاده می‌شود. اما برای بدافزارهای جدیدی که توسط شبکه تولید می‌شوند، برچسبی وجود ندارد؛ زیرا این داده‌ها ناشناخته هستند و هدف اصلی شبکه استفاده از مولد است. درواقع، این شبکه به‌عنوان یک روش تشخیص بدافزار روز صفر، بدون نیاز به داشتن برچسب برای بدافزارهای جدید عمل می‌کند. در حین فرایند آموزش و در هر مرحله از اجرا، شبکه متمایز گر وزن‌ها را روی‌داده‌های هنجار واقعی یاد می‌گیرد و آموخته‌های خود را با توزیع داده‌های شبه هنجار مقایسه می‌کند. درنهایت تابع هزینه برای شبکه متمایز گر محاسبه می‌شود. ساختار شبکهٔ متمایز گر، یک شبکه پرسپترون با چندلایه است. در لایهٔ نهایی، تابع فعال‌ساز سیگموید[1] با یک واحد مخفی[2]، احتمال مربوط به واقعی یا مصنوعی بودن هر نمونه را برمی‌گرداند سپس طبق خروجی متمایز گر، تابع هزینه محاسبه‌شده و پارامترهای متمایز گر که شامل وزن‌ها و بایاس است، طی فرایند پس انتشار به‌روزرسانی می‌شوند. تابع هزینه متمایز گر Binary Cross Entropy است. ساختار شبکه مولد، یک شبکه پرسپترون سه‌لایهٔ معکوس با تابع فعال‌ساز relu است. این شبکه، یک بردار با توزیع تصادفی اعداد بین ۰ و ۱ دریافت می‌کند و پس از گذر از دو لایهٔ مخفی، سعی می‌کند یک توزیع احتمالی از داده‌های آموزشی هنجار را تخمین بزند تا نمونه‌هایی شبیه به نمونه‌های هنجار تولید کند. ابعاد بردارهای خروجی شبکه مولد با ابعاد نمونه‌های هنجار اصلی همخوانی دارند. درنهایت نمونه-های تولیدشده به همراه نمونه‌های واقعی به شبکه متمایز گر داده می‌شود تا هر نمونه را به کلاس موردنظر طبقه‌بندی کند.

۵. ارزیابی

در این بخش به ارزیابی روش پیشنهادی به‌منظور تشخیص بدافزارها به‌ویژه بدافزار روز صفر، پرداخته می‌شود. بدین منظور در ابتدا به معرفی مجموعه داده پرداخته و سپس ماتریس درهم‌ریختگی و معیارهای دقت، حساسیت، صحت، F1 معرفی می‌شوند.

۵-۱. معرفی مجموعه داده

در این پژوهش از دو مجموعه داده استفاده کردیم. مجموعه داده مرجع در این پژوهش، مجموعه داده ارائه‌شده در [۲] است. این مجموعه داده توسط cuckoo sandbox در سال ۲۰۱۹ تولیدشده که شامل ۲۲۸۶ برنامه عادی و ۷۶۰۷ بدافزار است. نمونه بدافزارهایی که در این مجموعه داده وجود دارد عبارت‌اند از: بدافزارهای درب پشتی، کرم‌ها، برنامه‌های مخرب تبلیغاتی، برنامه‌های مخرب بانکی، بدافزارهای ایمیل، جاسوس‌ها و انواع خانواده‌های اسب تروا. از فایل مجموعه داده خام که به فرمت json است استفاده کرده و ویژگی‌های پویا که شامل کلیدها و مقادیر رجیستری، ویژگی‌های مرتبط با فایل، عملیات شبکه، فراخوانی‌های سیستمی است و تحلیل ایستا که شامل نام‌های درج PE است که مجموعاً ۱۳۰۶ ویژگی است، استخراج و درنهایت به فایلcsv تبدیل کردیم. مقادیر ویژگی‌های پویا مقادیر عددی مبتنی بر تکرار است و مقادیر ویژگی‌های ایستا به‌صورت وجود یا عدم وجود است. دومین مجموعه داده‌ی استفاده‌شده، مجموعه داده‌ی Top_1000_PE_Imports است. این مجموعه داده، داده‌های تحلیل ایستا است که شامل ۱۰۰۰ تابع ورودی برتر استخراج‌شده به کمک Cuckoo Sandbox است. مقادیر ویژگی‌های ایستا به‌صورت وجود یا عدم وجود است. این مجموعه داده ۱۹۳۰ نمونه برنامه‌ی سالم و ۴۵۶۵۱ نمونه برنامه‌ی مخرب دارد.

۵-۲. سنجه‌های ارزیابی:

به‌منظور ارزیابی کارائی روش پیشنهادی که در بخش چهارم شرح داده شد، نیاز به معیارهای استانداردی داریم. ماتریس درهم‌ریختگی، یک روش اندازه‌گیری عملکرد، برای مسائل طبقه‌بندی است که در آن خروجی می-تواند دو یا چند کلاس باشد. این ماتریسِ دو در دو، شامل پارامترهای TP، FP، TN، FN است. با استفاده از این پارامترها، می‌توان عملکرد مدل را در کلاس‌های اقلیت و اکثریت برای مجموعه داده‌های متوازن و نامتوازن مقایسه کرد. ماتریس درهم‌ریختگی به‌صورت شکل (۳) است.

		پیش‌بینی‌شده	
		منفی	مثبت
حقیقی	منفی	TN	FP
	مثبت	FN	TP

شکل (۳): ماتریس درهم‌ریختگی

[1] Sigmoid Activation Function.
[2] Hidden Unit.

جدول (۲): توضیح پارامترهای ماتریس درهم‌ریختگی

نام پارامتر	توضیح
منفی واقعی یا TN	برنامه‌های سالمی که به‌درستی در کلاس برنامه‌های سالم قرارگرفته‌اند
منفی کاذب یا FN	بدافزارهایی که ندانسته به‌عنوان برنامه سالم در نظر گرفته‌شده‌اند
مثبت کاذب یا FP	برنامه‌های سالمی که ندانسته در دسته بدافزارها قرارگرفته‌اند
مثبت واقعی یا TP	بدافزارهایی که به‌درستی در کلاس بدافزار طبقه‌بندی‌شده‌اند

جدول (۳): معیارهای ارزیابی

نام معیار	روش محاسبه
صحت یا Accuracy	(TP+TN) /(TN+FN+TP+FP)
دقت یا Precision	(TP) / (TP+FP)
یادآوری یا Recall	(TP) / (TP+FN)
F-Score	2× (Precision × Recall) / (Precision + Recall)
FPR	(FP) / (FP+TN)

۵-۳. نتایج آزمایش‌های انجام‌شده روی مجموعه داده اول

به علت کمبود تعداد داده‌های هنجار، مدل نتوانست به‌خوبی آموزش ببیند و نتیجه مناسبی به دست نیامد. بنابراین با افزایش تعداد داده‌های هنجار به کمک مدل CTGAN توانستیم به نتایج خوبی برسیم. برای اطمینان از این‌که مدل CTGAN توانسته داده‌های شبه هنجار را با دقت بالا تولید کند، در ابتدای کار با روش hold out cross validation با نسبت ۳۰ -۷۰، تقسیم کرده و از ۲۲۸۶ داده عادی ۳۰ درصد از داده‌های هنجار واقعی که ۶۸۶ است را از مجموعه داده اولیه خود جدا کرده، برای آزمون مدل کنار گذاشتیم و فقط ۷۰ درصد از داده‌ها را به مدل CTGAN داده و داده‌های هنجار را با احتساب ۷۰ درصد داده هنجار واقعی به ۶۱۶۰۰ رساندیم. در جدول (۴) نسبت نمونه‌های داده‌های اولیه در مجموعاً آموزش و آزمایش و همچنین پارامترهای آموزشی شبکه مولد خصمانه تک کلاسه در جدول (۵) نمایش داده‌شده است.

جدول (۴): نسبت نمونه‌های داده تولیدشده در دو مجموعه‌ی آموزش و آزمایش با داده‌های آزمون واقعی

	داده‌های هنجار	داده‌های بدافزار
آموزش	۶۱۶۰۰	۰
آزمایش	۶۸۶	۷۶۰۷

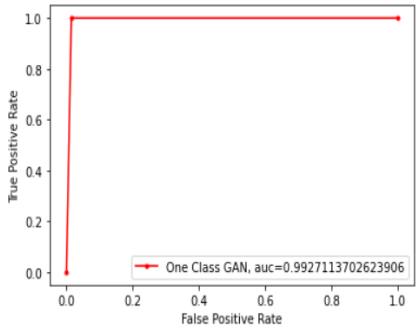
جدول (۵): پارامترهای در نظر گرفته‌شده برای آموزش داده اولیه

Activation Function	Relu
Loss Function	Binary Cross Entropy
Optimizer	Adam (۰/۰۰۰۱)
Batch Size	۱۰۰
epochs	۱۰۰
Number of Hidden Layer in Discriminator	۳
Number of Hidden Layer in Generator	۲

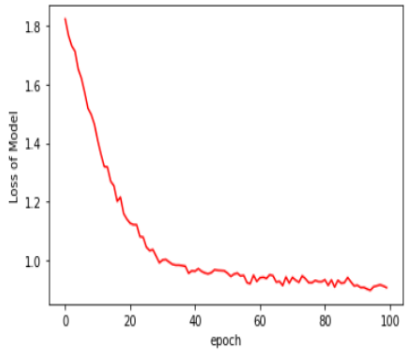
مولد خصمانه، در مرحله آموزش، توانست با خطای مناسبی توزیع داده‌های هنجار واقعی را به‌خوبی یاد بگیرد. در مرحله آزمایش، شبکه متمایز گر را با نمونه بدافزارهای واقعی آزمایش کرده‌ایم که منحنی ROC-AUC و خطای یادگیری آن به ترتیب در شکل‌های (۴) و (۵) و همچنین نتایج مرحله آموزش و آزمایش و ماتریس درهم‌ریختگی نیز به ترتیب در جداول (۶) و (۷) آورده شده است.

جدول (۶): نتایج طبقه‌بندی مولد خصمانه‌ی تک کلاسه در مرحله‌ی آموزش و آزمایش برای مجموعه داده اول

	Precision	Recall	F1-Score	Accuracy
Train	۹۹/۰۸۳	۹۹/۰۹۸	۹۹/۰۹۱	۹۹/۰۹۶
Test	۹۹/۰۹۳	۹۹/۰۲۷	۹۹/۰۶۰	۹۹/۰۸۸



شکل (۴): منحنی ROC-AUC برای مجموعه داده اول



شکل (۵): خطای یادگیری ویژگی‌ها برای داده‌های تولیدشده از مجموعه داده اول در ۱۰۰ دوره آموزش

جدول (۷): ماتریس درهم‌ریختگی برای داده‌های واقعی از مجموعه داده اول

		True Class	
		Benign	Malware
Predicted class	Benign	۶۷۶	۱۰
	Malware	۰	۷۶۰۷

مجموعه داده Top_1000_PE_Imports نیز مشابه مجموعه داده مرجع ما تعداد برنامه سالم کمی برای آموزش طبقه‌بندی ما دارد، به‌منظور مقایسه عادلانه، این مجموعه داده را نیز مشابه به مجموعه داده اول به مدل CTGAN داده و تعداد داده‌های سالم را به ۶۱۶۰۰ رساندیم. در جدول (۸) نسبت نمونه‌های داده‌های اولیه در مجموعه‌ی آموزش و آزمایش و پارامترهای آموزشی شبکه مولد خصمانه تک کلاسه نمایش داده‌شده است. برای مجموعه داده دوم نیز مولد خصمانه، در مرحله‌ی آموزش، توانست با خطای مناسبی توزیع داده‌های هنجار واقعی را به‌خوبی یاد بگیرد. در مرحله آزمایش، شبکه‌ی متمایز گر را با نمونه بدافزارها و داده‌های سالم واقعی آزمایش کرده‌ایم. طبقه‌بندی پیشنهادی توانسته است به‌درستی نمونه‌های بدافزار را شناسایی کند و در کلاس موردنظر طبقه‌بندی کند.

جدول (۸): نسبت نمونه‌های داده Top_1000_PE_Import

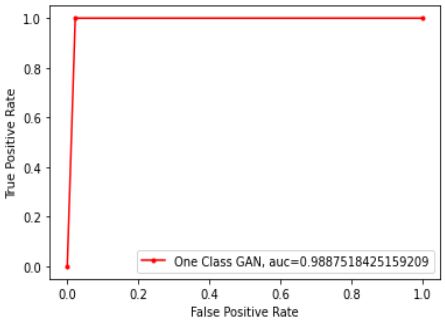
	هنجار	بدافزار
آموزش	۶۱۶۰۰	۰
آزمایش	۵۷۹	۴۵۶۵۱

جدول (۹): پارامترهای در نظر گرفته‌شده برای داده دوم

Activation Function	Relu
Loss Function	Binary Cross Entropy
Optimizer	Adam (۰/۰۰۰۱)
Batch Size	۱۸۰
epochs	۱۰۰
Number of Hidden Layer in Discriminator	۳
Number of Hidden Layer in Generator	۲

جدول (۱۰): نتایج طبقه‌بندی مولد خصمانه‌ی تک کلاسه در مرحله‌ی آموزش و آزمایش برای مجموعه داده دوم

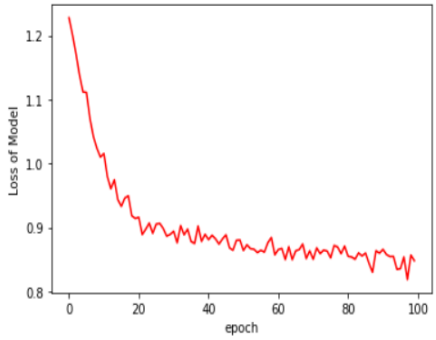
	Precision	Recall	F1-Score	Accuracy
Train	۹۹/۰۸۶	۹۹/۰۹۰	۹۹/۰۸۹	۹۹/۰۸۸
Test	۹۹/۰۸۱	۹۸/۰۸۸	۹۹/۰۳۴	۹۹/۰۹۴



شکل (۶): منحنی ROC-AUC داده‌های واقعی مجموعه داده دوم

جدول (۱۱): ماتریس درهم‌ریختگی داده‌های واقعی مجموعه داده دوم

		True Class	
		Benign	Malware
Predicted class	Benign	۵۶۶	۱۳
	Malware	۱۰	۴۵۶۴۹



شکل (۷): خطای یادگیری ویژگی‌ها برای داده‌های تولیدشده از مجموعه داده دوم در ۱۰۰ دوره آموزش

به‌منظور ارزیابی و مقایسه دقت مجموعه داده تولیدی و روش پیشنهادی، از طبقه‌بندی ماشین بردار پشتیبان تک کلاسه و همچنین طبقه‌بندی‌های چند کلاسه درخت تصمیم[1] و جنگل تصادفی استفاده کرده‌ایم. در ادامه نتایج مربوط به ارزیابی و مقایسه در جدول (۱۲) نشان داده‌شده است و روی مجموعه Top_1000_PE_Imports، از روش ZeVigilante [۱۱] و مدل ماشین بردار تک کلاسه استفاده کردیم. رویکرد روش ZeVigilante طبقه‌بندی باینری بوده و متفاوت از رویکرد روش ارائه‌شده است اما مجموعه داده‌ی استفاده‌شده در این پژوهش یکسان است. طبقه‌بندی‌های انتخاب‌شده از این پژوهش، درخت تصمیم، جنگل تصادفی، K نزدیک‌ترین همسایه[2]، مدل بیز ساده[3]، شبکه‌های عصبی[4] است که بالاترین دقت برای RF است که در جدول (۱۳) نتایج مربوط به ارزیابی و مقایسه نشان داده‌شده است.

[1] Decision Tree.
[2] k - Nearest Neighbor (knn)
[3] Naive Bayes (NB)
[4] Neural Network (NN)

جدول (۱۲): نتایج انجام‌شده روی مجموعه داده‌ی مرجع در مرحله‌ی آزمایش

	Accuracy	Precision	Recall	F1-score	FPR	TPR
One Class GAN	**۹۹/۸۸ ٪**	**۹۹/٪۹۳**	**۹۹/٪۲۷**	**۹۹/٪۶۰**	**۰۱/٪۴۵**	٪۱۰۰
One Class SVM	۸۳/٪۹۷	۸۳/٪۳۹	۷۵/٪۸۹	۷۸/٪۳۰	۰۴/٪۹۲	۵۶/٪۷۱
Random Forest	۹۸/٪۵۰	۹۷/٪۹۹	۹۸/٪۵۰	۹۸/٪۴۵	۰۲/٪۰۴	۹۶/٪۸۹
Decision Tree	۹۵/٪۸۹	۸۷/٪۵۷	۹۴/٪۲۲	۹۰/٪۵۱	۰۳/٪۶۳	۹۲/٪۰۷

جدول (۱۳). نتایج انجام‌شده روی مجموعه داده‌ی Top_1000_PE_Imports در مرحله‌ی آزمایش

	Accuracy	Precision	Recall	F1-score	FPR	TPR
One Class GAN	**۹۹/٪۹۴**	**۹۹/٪۸۱**	**۹۸/٪۸۸**	**۹۹/٪۳۴**	۰۲/٪۲۵	**۹۹/٪۹۹**
One Class SVM	۵۶/٪۸۷	۶۷/٪۴۷	۶۸/٪۲۱	۵۶/٪۸۵	۰۵/٪۱۸	۴۱/٪۴۴
[۱۱]**ZeVigilante**	۹۸/٪۱۷	٪۹۸	٪۹۸	٪۹۸	**۰۱/٪۶۹**	۸۶/٪۹۳

۶- نتیجه‌گیری

سیستم‌عامل ویندوز به دلیل تسلط خود در بین سیستم‌عامل‌های رومیزی، هدف محبوبی برای نویسندگان بدافزار بوده است که با سرعت چشمگیری درحال‌رشد هستند. در این پژوهش، یک مدل تشخیص ناهنجاری بدافزارهای ویندوز مبتنی بر رویکرد یادگیری عمیق ارائه‌شده است که این رویکرد از مدل شبکه خصمانه تک کلاسه، برای شناسایی بدافزار به کمک تحلیل پویا و تحلیل ایستا استفاده می‌کند. مزیت استفاده از روش تک کلاسه، آن است که می‌تواند در تشخیص بدافزارهای جدید و ناشناخته مؤثر باشد. درواقع با ارائه این مدل، چالش عدم وجود نمونه‌های کافی از بدافزار را حل کرده‌ایم. علاوه بر این، ویژگی‌های متمایزکننده و سطح بالا همچون فراخوانی‌های سیستمی، رجیستری ویندوز، فایل‌های سیستمی و تعاملات شبکه را استخراج کردیم و به‌منظور بهبود آموزش طبقه‌بندی نیز ویژگی‌های ایستا که شامل نام توابع ورودی فایل اجرایی بود، به ویژگی‌های سطح بالا افزودیم که موجب بهبود دقت طبقه‌بندی در تفکیک داده‌های سالم از بدافزار می‌شوند. درنهایت مدل پیشنهادی را با مدل یادگیری ماشین تک کلاسه و مدل‌های دو کلاسه مقایسه کردیم. به‌منظور اثبات صحت و درستی مدل پیشنهادی از دو مجموعه داده با ویژگی‌ها و خصوصیات مختلف استفاده کرده‌ایم. نتایج به‌دست‌آمده نشان می‌دهد توانایی تفکیک دو کلاس بدافزار و سالم در مدل مولد خصمانه عمیق در مقایسه با روش‌های پیشین بیشتر است. مدل مولد خصمانه تک کلاسه باتوجه‌به شرایط و ویژگی‌هایی که دارا است می‌تواند به‌عنوان یک سیستم تشخیص‌دهنده بدافزارهای ویندوز به کار گرفته شود.

در ادامه پژوهش می‌توان با تحلیل بیشتر ویژگی‌های PE و همچنین ویژگی‌های فایل، رجیستری و شبکه به سطح بالاتری از رفتار بدافزارها دست‌یافت. همچنین می‌توان به کمک Ensemble در سطح یادگیری عمیق تک کلاسه از مزیت‌های شورایی و رأی حداکثری نیز بهره برد. همچنین می‌توان رویکرد مدنظر را در حوزه اینترنت اشیاء نیز به کار گرفت.

۷- منابع

[1] Chalapathy, R. and S. Chawla, Deep learning for anomaly detection: A survey. arXiv preprint arXiv:1901.03407, 2019.

[2] Tajoddin, A. and M. Abadi, RAMD: registry-based anomaly malware detection using one-class ensemble classifiers. Applied Intelligence, 2019. **49**(7): p. 2641-2658.

[3] Liu, J., et al. FENOC: an ensemble one-class learning framework for malware detection. in 2013 Ninth International Conference on Computational Intelligence and Security. 2013. IEEE.

[4] Tang, A., S. Sethumadhavan, and S.J. Stolfo. Unsupervised anomaly-based malware detection using hardware features. in Research in Attacks, Intrusions and Defenses: 17th International Symposium, RAID 2014, Gothenburg, Sweden, September 17-19, 2014. Proceedings 17. 2014. Springer.

[5] Miao, Q., et al., Malware detection using bilayer behavior abstraction and improved one-class support vector machines. International Journal of Information Security, 2016. **15**: p. 361-379.

[6] Yousefi-Azar, M., et al. Autoencoder-based feature learning for cyber security applications. in 2017 International joint conference on neural networks (IJCNN). 2017. IEEE.

[7] Kim, J.-Y., S.-J. Bu, and S.-B. Cho. Malware detection using deep transferred generative adversarial networks. in Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part I 24. 2017. Springer.

[8] Kim, J.-Y., S.-J. Bu, and S.-B. Cho, Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders. Information Sciences, 2018. **460**: p. 83-102.

[9] Ijaz, M., M.H. Durad, and M. Ismail. Static and dynamic malware analysis using machine learning. in 2019 16th International bhurban conference on applied sciences and technology (IBCAST). 2019. IEEE.

[10] Khatouni, A.S., et al. Exploring nat detection and host identification using machine learning. in 2019 15th International Conference on Network and Service Management (CNSM). 2019. IEEE.

[11] Alhaidari, F., et al., ZeVigilante: Detecting Zero-Day Malware Using Machine Learning and Sandboxing Analysis Techniques. Computational Intelligence and Neuroscience, 2022. **2022**.

[12] R. Jalayi, E. Shaghaghi, and M. A. Javadzadeh, "Kashef: A Two-step detector of Windows-based Malicious executable files," Imam Hossein University, 2021. (In Persion)

[13] Xu, L., et al., Modeling tabular data using conditional gan. Advances in Neural Information Processing Systems, 2019. **32**.