آشکارسازی بدافزارها با استفاده از دسته‌بندی دنباله‌های با طول متغیر

فاطمه حسینی[۱]، میترا میرزارضایی[۲*] و آرش شریفی[۳]
دانشگاه آزاد اسلامی، واحد علوم و تحقیقات ، گروه مهندسی کامپیوتر، تهران، ایران

چکیده

در این مقاله روشی مبتنی بر گراف به‌عنوان استخراج ویژگی برای دنباله‌های با طول متغیر پیشنهاد می‌شود. روش پیشنهادی بدون ثابت‌کردن طول دنباله‌ها، با تعیین پر تکرارترین دستورها و گذاشتن باقی دستورها در مجموعۀ 'other' از لحاظ سرعت و حافظه صرفه‌جویی می‌کند. با توجه به میزان شباهت ویژگی‌ها، هر نمونه امتیازی می‌گیرد و از امتیازات جهت دسته‌بندی استفاده می‌شود. برای بهبود نتایج، دو رویکرد پیشنهاد می‌شود. در رویکرد نخست، ویژگی‌های استخراج‌شده از روش‌های امتیازدهی بر روی آپکد، هگزادسیمال و فراخوانی سیستمی در ورودی دسته‌بندها ترکیب می‌شوند. در رویکرد دوم، خروجی دسته‌بندهای مختلف ترکیب شده و از رأی اکثریت استفاده می‌شود. رویکرد پیشنهادی با دقت ۹۷ ٪ بدافزارهای دگرگون‌شدۀ رایانه‌ای از مجموعۀ vxheaven را نه‌تنها شناسایی، بلکه دستۀ بدافزارها را نیز تعیین می‌کند؛ درحالی‌که روش‌های SSD و HMM تحت شرایط یکسان با دقت ۸۴ ٪ و ۸۰ ٪ توانستند بدافزارها را شناسایی کنند.

واژگان کلیدی: آشکارسازی بدافزارها، روش‌های مبتنی بر گراف، ترکیب دسته‌بندها، دسته‌بندی با طول متغیر، ماشین بردار پشتیبان

Malware Detection using Classification of Variable-Length Sequences

Fatemeh Hosseini [1], Mitra Mirzarezaee[2] & Arash Sharifi[3]
Islamic Azad University, Science and Research Branch, Department of Computer Eng., Tehran, Iran

Abstract

In this paper, a novel method based on the graph is proposed to classify the sequence of variable length as feature extraction. The proposed method overcomes the problems of the traditional graph with variable length of data, without fixing length of sequences, by determining the most frequent instructions and insertion the rest of instructions on the set of "other", save speed and memory. According to features and the similarities of them, a score is given to each sample and that is used for classification. To improve the results, the method is not used alone, but in the two approaches, this method is combined with other existing Technique to get better results. In the first approach, which can be considered as a feature extraction, extracted features from scoring techniques (Hidden Markov Model, simple substitution distance and similarity graph) on op-code sequences, hexadecimal sequences and system calls are combined at classifier input. The second approach consists of two steps, in the first step; the scores which obtained from each of the scoring Technique are given to the three support vector machine. The outcomes are combined according to the weight of each Technique and the final decision is taken based on the majority vote. Among the components of the support vector machine, when given a higher weight in the similarity graph method (the proposed method), the result is better, Because the similarity graph method is more accurate than the other two methods. Then, in the second section, considering the strengths and benefits of each classifier, classifier outputs are combined and the majority voting is used. Three methods have been tested for group combinations, including Ensemble

* Corresponding author

*نویسندۀ عهده‌دار مکاتبات

تاریخ ارسال: ۹۶/۰۹/۰۸ ● تاریخ پذیرش: ۹۸/۰۳/۲۹ ● نوع مطالعه: پژوهشی

Averaging, Bagging, and Boosting. Ensemble Averaging consisting of the combination of four classifiers of random forests, a support vector machine (as obtained in the previous section), K nearest neighbors and naive Bayes, and the final decision is taken based on the majority vote; therefore, it is used as the proposed method. The proposed approach could detect metamorphic malware from Vxheaven set and also determines categories of malware with accuracy of 97%, while the SSD and HMM methods under the same conditions could detect malware with an accuracy of 84% and 80% respectively.

Keywords: Malware Detection, Graph Techniques, Combining Classifiers, Variable Length Classification, Support vector machine

۱- مقدمه

با توجه به اهمیت و جایگاه رایانه‌ها در عصر فناوری و این موضوع که تهدید و در معرض خطر قرارگرفتن رایانه‌ها می‌تواند تهدید جدی برای جامعه باشد؛ عرضهٔ روشی هوشمند جهت آشکارسازی بدافزارها امری ضروری است. در حوزه آشکارسازی بدافزارها، اغلب با داده‌ها یا کدهای دارای طول متغیر سروکار داریم. ماهیت این نوع داده‌ها باعث می‌شود مقایسه، تجزیه و تحلیل و درنهایت دسته‌بندی آنها با چالشی بزرگ مواجه باشد. از آنجا که روش‌های دسته‌بندی موجود، اغلب با مجموعه ویژگی‌های با طول ثابت کار می‌کنند؛ باید به‌دنبال راه‌کاری بود تا داده‌های با طول نابرابر به قالبی مناسب برای این دسته‌بندها تبدیل شوند. ساده‌ترین راه برای رسیدن به این منظور، نرمال‌سازی طول بردار ویژگی است. اگر چه به‌طورمعمول به‌دلیل سادگی این روش از آن زیاد استفاده می‌شود، اما نرمال‌سازی طول بردار ویژگی باعث تخریب داده‌ها و افت شدید کارایی تشخیص می‌شود. اگر بتوان یک خصوصیت رایج برای همه اعضای یک خانواده دگرگون‌شده تعیین کرد، پس می‌توان از این ویژگی برای تشخیص و دسته‌بندی آنها استفاده کرد. از آنجا که هیچ الگوریتمی وجود ندارد که برای تمامی شرایط و در تمامی زمان‌ها بهترین یادگیر را به‌وجود آورد و هیچ دسته‌بندی قادر به تشخیص صحیح همه الگوها در تمام شرایط نیست، استفاده از نتایج چند دسته‌بند با عنوان یادگیری دسته جمعی، یک روش مؤثر است که در آن به‌منظور بهبود دقت یادگیری، نتایج دسته‌بندها با یکدیگر ترکیب شده و یک سامانهٔ مرکب شکل می‌گیرد. ترکیب نتایج دسته‌بندهایی که از ویژگی‌های متفاوتی استفاده می‌کنند می‌تواند کارایی سامانه تشخیص را بهبود بخشد [1]. به‌منظور جلوگیری ازدست‌رفتن اطلاعات داده‌های ورودی، هدف، یافتن رویکردی است که الگوی مناسب‌تری برای داده‌های با طول متغیر و دسته‌بندی آن‌ها نمایش دهد؛ با استفاده از روش‌های جستجوی شباهت و امتیازدهی، روشی برای تجزیه و تحلیل، دسته‌بندی و آشکارسازی بدافزارها پیشنهاد می‌شود که براساس ماهیت متغیر بردارهای ویژگی عمل می‌کند. از آنجایی که کدهای بدافزار ساختار برنامه‌نویسی استانداردی ندارند، لذا دارای گراف‌های متفاوت با طول‌های متفاوتی هستند و نمی‌توان آن‌ها را با هم مقایسه کرد، مگر با استفاده از روش‌های نرمال‌سازی، طول تمام گراف‌ها را به یک اندازه در آورد. مشکل دیگر این است که به‌طورمعمول بدافزارهای امروزی، شیوه‌نامه‌های بیشتری دارند؛ لذا حجم گراف تولیدشده نیز بالا می‌رود. در این صورت گراف‌های مورد آزمایش بسیار بزرگ و پرتعداد می‌شوند و الگوریتم‌هایی که از حافظه استفاده می‌کنند، پس از چندین ساعت، با خطای حافظه متوقف می‌شوند؛ ازاین‌رو در این پژوهش، الگوریتمی طراحی شده که بر این مشکلات غلبه کند.

روش پیشنهادی مبتنی بر گراف است و بر این ایده بنا شده که دنباله‌های آپکد، هگزا دسیمال و فراخوانی‌های سامانه‌ای در کنار یکدیگر در فایل‌های بدافزار می‌توانند دانش خوبی از میزان شباهت فایل‌های اجرایی نمایش دهند. دنباله‌های موجود، حاوی تعداد زیادی آپکد، دستورهای هگزا دسیمال و تابع فراخوانی سامانه‌ای مجزا هستند؛ از این دستورها، تنها تعداد خاصی از آنها به‌طور مداوم ظاهر می‌شوند و در طبیعت ذاتی بدافزار یا فایل خوش‌خیم سهیم هستند؛ لذا روش پیشنهادی در مواجه با داده‌های با طول متغیر، بدون ثابت‌کردن طول دنباله‌ها، با تعیین پرتکرارترین دستورها و گذاشتن باقی دستورها در مجموعهٔ **'other'** از لحاظ سرعت و حافظه صرفه‌جویی می‌کند. برای بهبود نتایج و پیداکردن یک الگوی مناسب شباهت بین دنباله‌های با طول متغیر، از روش ارائه‌شده به‌تنهایی استفاده نمی‌شود؛ بلکه از دو رویکرد ترکیب ویژگی‌ها در ورودی و ترکیب دسته‌بندها در خروجی استفاده می‌شود. در رویکرد نخست، از ترکیب روش‌های مدل مخفی مارکوف، فاصله جایگزینی ساده و روش مبتنی بر گراف در ورودی دسته‌بندها استفاده می‌شود. همچنین با بهره‌گیری از ترکیب نتایج دسته‌بندها می‌توان کارایی سامانه آشکارسازی بدافزارها را بهبود بخشید و نرخ خطای تشخیص را کاهش داد. در رویکرد دوم، ترکیب دسته‌بندها در خروجی مورد آزمایش قرار گرفته است که شامل دو مرحله است. در گام نخست، امتیازات حاصل از هر یک از روش‌های امتیازدهی به سه دسته‌بند ماشین بردار پشتیبان داده شده است. خروجی‌ها بر

اساس وزن‌دهی به هر یک از روش‌ها ترکیب شده و تصمیم نهایی بر اساس رأی اکثریت گرفته شده است. از میان ترکیبات ماشین بردار پشتیبان با وزن‌دهی متفاوت، زمانی که به روش گراف تشابه (روش ارائه‌شده) وزن بیشتری داده می‌شود، نتیجه بهتری می‌دهد؛ چون روش گراف تشابه، نسبت به دو روش دیگر معیارهای ارزیابی بالاتری دارد؛ سپس در بخش دوم با توجه به قدرت و مزایای هر دسته‌بند، از ترکیب خروجی دسته‌بندها استفاده شده است. این تنوع باعث شده است که از قدرت و مزایای سامانه‌های مختلف استفاده کند؛ بنابراین به‌عنوان روش پیشنهادی از آن استفاده شده است. روش پیشنهادی می‌تواند در دیگر حوزه‌هایی که با ویژگی‌های با طول متغیر سر و کار دارد به‌کار گرفته شود.

در ادامه در بخش ۲ به کارهای انجام‌شده اشاره می‌شود. معماری کامل روش پیشنهادی در بخش ۳ بیان شده است. در بخش ۴ شبیه‌سازی و نتایج آزمایش‌های انجام‌شده برای ارزیابی روش پیشنهادی مورد بررسی قرار می‌گیرد. درنهایت در بخش ۵ نتیجه‌گیری به‌عمل می‌آید.

۲- پیشینهٔ پژوهش

در طول این سال‌ها، روش‌های زیادی برای آشکارسازی بدافزارها پیشنهاد شده است. *الذئاب* و همکاران [2]، از تجزیه و تحلیل n–گرام از محتوای دودویی، برای دسته‌بندی استفاده کرده‌اند. تجزیه و تحلیل اولیه، با استفاده از ماشین بردار پشتیبان[1] توسط ارزش‌های مختلف n از ۱ تا ۵، صورت گرفته است. پایگاه داده آنها شامل ۲۴۲ بدافزار و ۷۲ فایل سالم بوده است و با دقت ۰/۹۶۵ بدافزارها را شناسایی کرده‌اند. *لین* و همکاران [3]، روشی عرضه کرده‌اند که ترکیبی از انتخاب و استخراج ویژگی است و به‌طور قابل توجهی ابعاد ویژگی‌های آموزش و دسته‌بندی را کاهش داده است. بر اساس رفتارهای بدافزارها که در محیط سندباکس جمع آوری شده، روش آنها از پنج مرحله تشکیل شده است: ۱) استخراج n-گرام ویژگی از فضای داده؛ ۲) ایجاد یک دسته‌بند ماشین بردار پشتیبان برای دسته‌بندی بدافزارها؛ ۳) انتخاب یک زیرمجموعه از ویژگی‌ها؛ ۴) تبدیل بردارهای ویژگی با بُعد بالا به بردار ویژگی با بعد کوچک‌تر و ۵) انتخاب مدل. آنها ۴۲۸۸ نمونه بدافزار را گردآوری کردند. دقت، حساسیت و F-measure روش ارائه‌شده به‌ترتیب با اندازه ۰/۷۹۶، ۰/۷۷۸۵ و ۰/۷۸۴۴ بوده است.

یو و همکاران [4]، یک روش آشکارسازی ویروس، بر اساس شناسایی دنباله فراخوانی سامانه‌ای تحت محیط ویندوز ارائه داده‌اند. آنها برای نخستین‌بار با استفاده از تجزیه و تحلیل ایستا، فراخوانی‌های سیستمی را از فایل‌های بدافزار استخراج کرده و از الگوریتم نایو بیز به‌عنوان تعیین‌کننده تقریبی ویروس استفاده کرده‌اند. در این پژوهش چهار گونهٔ مختلف بدافزار مورد آزمایش قرار گرفته است. نتایج حاصل از روش پیشنهادی در مقایسه با هشت آنتی ویروس معرفی‌شده بهبود قابل توجهی را در شناسایی ویروس Win32 نشان می دهد که نشان از کارایی مناسب روش پیشنهادی دارد.

لندیچ و همکاران [5]، از سه روش استخراج ویژگی برای استخراج آپکد استفاده کرده‌اند. در روش انتخاب ویژگی، داده‌های بی‌ربط حذف شده و از الگوریتم‌های دسته‌بندی Ripper ، C4.5،Ibk و K–نزدیک‌ترین همسایه استفاده شده است. خروجی دسته‌بندهای متعدد ترکیب‌شده و پیش‌بینی نهایی بر اساس رأی‌گیری حق وتو است. استراتژی تصمیم‌گیری حق وتو با معرفی رأی‌گیری حق وتو مبتنی بر اعتماد بهبود یافته است.

وانگ و همکاران [6]، نشان داده‌اند که ابزارهای بر اساس مدل پنهان مارکوف در آشکارسازی بدافزارهای دگرگون‌شده مؤثر هستند. آنها دویست بدافزار از خانواده دگرگون‌شده را جمع‌آوری کرده، نرخ تشخیص ۱۰۰ ٪ بوده است. در آشکارسازی بدافزارها استراتژی جدیدی از مدل مخفی مارکوف، معرفی شده که نتایج برتری نسبت به روش‌های مبتنی بر آستانه، به‌دست آورده است. روش مبتنی بر آستانه با استفاده از یک مدل پنهان مارکوف، یک فایل را به‌عنوان فایل آلوده معرفی می‌کند، اگر احتمال به‌دست‌آمده بیش از آستانه تعیین شده باشد. *کلهر* و همکاران [7]، نشان داده‌اند که چگونه رویکرد آستانه می‌تواند با این استراتژی جدید، برای کاهش سربار عملکرد ترکیب شود و در دقت و صحت نتایج هیچ کاهشی نباشد. آنها مدل پنهان مارکوف را برای چهار کامپایلر مختلف کد اسمبلی دست‌نوشته، سه کیت ساخت و ساز ویروس، و دو خانواده بدافزار دگرگون‌شده مورد بررسی قرار دادند. کاربرد پروفایل مدل پنهان مارکوف، که اطلاعات مربوط به موقعیت و حالات را در نظر می‌گیرد، می‌تواند برای آشکارسازی انواع خاصی از ویروس‌های دگرگون‌شده مؤثر باشد؛ اما برای ویروس‌هایی که بلاک‌های کد را دور از هم و با فاصله تغییر می‌دهند، به‌خوبی انجام نمی‌گیرد [8]. تحلیل مدل پنهان مارکوف برای دسته‌بندی بدافزارها نیز می‌تواند مؤثر واقع شود. *آناچاترا* و همکاران [9]، آزمایش خود را بر روی چهار کامپایلر مختلف پیاده‌سازی کرده و ۹۴۴۲ بدافزار گردآوری کردند. تعداد خوشه‌ها بین دو تا

[1] Support Vector Machines (SVM)

پانزده مورد آزمایش قرار گرفت و با نه خوشه نتیجه بهتری به‌دست آمده است و با انحراف معیار ۰/۹۵ بدافزارها را خوشه‌بندی کرده‌اند. *جاس* و همکاران [10]، کاربرد روش‌های بیزین را در کنار مدل پنهان مارکوف برای آشکارسازی بدافزارها شرح داده‌اند. سینگ [11]، از مزایای روش‌های امتیازدهی با استفاده از مدل پنهان مارکوف استفاده و آن را با ماشین بردار پشتیبان ترکیب کرده است. مجموعه بدافزارها شامل درب پشتی از خانواده هربوت[1]، تروجان از خانواده پوشش امنیتی[2]، ویروس‌های لوح سخت[3]، خانواده متامورفیسم[4]، ضدبات [5]، تروجان از زیرو اکسس[6] بوده است. هر خانواده به‌صورت جداگانه در برابر فایل‌های سالم آموزش داده شده است. AUC برای هر پنج خانواده بدون استفاده از کد دگرگون‌شده برابر یک بوده است درحالی‌که AUC برای ویروس‌های لوح سخت، پوشش امنیتی، هربوت، ضدبات، زیرو اکسس با نسبت دگرگون‌کردن کد ۵۰٪ به ترتیب ۰/۷۹، ۰/۷۰، ۰/۹۵، ۱ و ۰/۹۳ بوده است.

استفاده از نتایج چند دسته‌بند با عنوان یادگیری دسته‌جمعی یک رویکرد مؤثر در یادگیری ماشین است که در آن به‌منظور بهبود دقت یادگیری نتایج دسته‌بندها با یکدیگر ترکیب شده و یک سامانه مرکب شکل می‌گیرد. قائمی و همکاران [12]، یک معماری جدید برای دسته‌بندی ترکیبی پرسش‌ها ارائه کرده‌اند. نتایج هر یک از دسته‌بندها توسط پنج روش رأی‌گیری وزن‌دار، فضای دانش رفتار، بیز ساده، کلیشه تصمیم و دمپستر شفر ترکیب شده و خروجی نهایی را شکل می‌دهد. این روش ترکیبی متشکل از دو دسته‌بند مبتنی بر یادگیری ماشین (ماشین بردار پشتیبان و نمایش پراکنده) و یک دسته‌بند مبتنی بر قانون استفاده شده است. در پایان نتایج حاصل از دسته‌بندها با روش معمول در ترکیب دسته‌بندهای تک‌طبقه ترکیب شده‌اند و نتایج حاصل بیان‌کننده بهبود عملیات دسته‌بندی نسبت به روش‌های موجود است.

در نگاه نخست ویژگی‌های با طول ثابت، به‌دلیل سادگی و سرعت بالا و کاهش زمان، روشی کارا به‌نظر می‌آید، اما هیچ بدافزاری فقط از کدهای بدخواه ساخته نشده است. بدافزارها همیشه ترکیبی از قسمت‌های بی‌خطر و خطرناک هستند؛ حتی در بیش‌تر موارد قسمت‌های خطرناک نسبت به قسمت‌های بی‌خطر در اقلیت قرار دارند؛ درنتیجه، استفاده‌کردن از یک زیرمجموعه از کدها امکان این که

[1] Harebot
[2] Sccurity Shield
[3] Smart HDD
[4] NGVCK
[5] Zbot
[6] ZeroAccess

قسمت‌های خطرناک در مدل‌های بدافزار دیده نشود، بالا می‌رود. اگرچه روش‌های هوشمند زیادی معرفی شده، اما توانایی‌های بدافزارها روزبه‌روز پیچیده‌تر شده است؛ به‌صورتی‌که هر روشی که برای آشکارسازی و شناسایی بدافزارها ابداع می‌شود، بدافزار با استفاده از استراتژی‌هایی جهت مخفی‌سازی کدهای خود، نوعی هوشمند ایجاد می‌کنند تا این روش‌ها قادر به آشکارسازی نباشند و بتوانند راه‌های امنیتی اخیر را کنار زنند و آنها را خنثی کنند. از آنجا که رویکردهای شناسایی و آشکارسازی بدافزارها نیز باید به‌روز شده و از هوشمندی برخوردار باشند، عرضه و توسعه روش‌های جدید و هوشمند امری ضروری است؛ لذا این مقاله، پژوهشی است بر روی یک الگوریتم جدید و هوشمند تا به‌کمک آن بتوان یک الگوی مناسب‌تری برای داده‌های با طول متغیر و دسته‌بندی آن‌ها ضمن افزایش نرخ تشخیص پیشنهاد دهد.

۳- روش پیشنهادی

در حوزه آشکارسازی بدافزارها، ویژگی‌های مختلف، بازنمایی‌های متفاوتی از فایل‌های اجرایی هستند که هرکدام حاوی یک نوع اطلاعات مفید در مورد آن فایل هستند. برای تشخیص یک الگوی مناسب از بدافزارها به‌طورمعمول به استخراج ویژگی‌های متفاوتی نیاز است. در این مقاله روشی مبتنی بر گراف به‌عنوان استخراج ویژگی برای دنباله‌های با طول متغیر پیشنهاد می‌شود. این روش با توجه به میزان شباهت ویژگی‌ها، بدون نیاز به ثابت‌کردن طول دنباله‌ها، یک امتیاز به هر نمونه می‌دهد و از این امتیازات جهت دسته‌بندی استفاده می‌شود. سه روش مدل مخفی مارکوف[7]، فاصله جایگزینی ساده[8] و روش پیشنهادی گراف تشابه به‌عنوان استخراج ویژگی بررسی می‌شوند و از ترکیب نتایج آنها در ورودی استفاده خواهد شد.

بر خلاف کارهای پیشین که به‌طورمعمول از یک دسته ویژگی استفاده شده و فقط دو طبقه بدافزار و فایل‌های سالم وجود داشته است، در این مقاله از سه ویژگی دنباله‌های آپکد، هگزا دسیمال و فراخوانی‌های سامانه‌ای در کنار یکدیگر استفاده خواهد شد و چهار طبقه مختلف داریم که سه طبقه آن مربوط به گونه‌های مختلف بدافزار است و سعی در تشخیص و تعیین دسته مطلوب برای هر یک از آنها هستیم.

از آنجا که هر دسته‌بند تا حد خاصی قادر به تشخیص صحیح الگوها است و هیچ تک‌الگوریتم آموزشی مشخصی وجود ندارد که بتواند برای تمامی کاربردها و درتمامی شرایط بهترین و دقیق‌ترین باشد، لذا در رویکرد دوم، از ترکیب

[7] Hidden Markov Model (HMM)
[8] Simple Substitution Distance (SSD)

دسته‌بندها در خروجی استفاده می‌شود. انگیزهٔ اصلی بر توسعه چنین روشی، کاهش نرخ خطا است. انواع ترکیب دسته‌بندی‌کننده‌ها شامل ترکیب ایستا[1] و ترکیب پویا هستند. ترکیب ایستا شامل سه روش میانگین گروه[2]، بگینگ[3] و بوستینگ[4] است [1].

۱-۳- روش پیشنهادی مبتنی بر گراف

در این بخش، روش پیشنهادی مبتنی بر گراف (**شبه‌کد**) برای دسته‌بندی داده‌های با طول متغیر شرح داده می‌شود. این روش شامل چهار مرحله تعیین کلید، ایجاد کد میانی، ایجاد گراف تشابه از روی ماتریس توزیع و تعیین امتیاز است.

```
Procedure Graph Similarity
Begin
    Create key
    for each sample do
        Create middle code based on key
    End for
    for each sample in train set do
        Built e matrix on their middle code
    End for
    E=(matrix e0ij + matrix e1ij+ … + matrix e(m-1)ij) / number of samples on train set
    Built E graph based on E matrix
    for each file in test set do
        Built D matrix for it
        Built D graph based on D matrix
        Calculate score for it
    End for
    Return scores
End procedure
```

(شبه‌کد۱): روش پیشنهادی مبتنی بر گراف تشابه

(Pseudo Code 1): the proposed method based on the graph

مرحله نخست، در بین داده‌های آموزشی پرتکرارترین دستورها را پیدا کرده و n تا، به‌عنوان کلید انتخاب می‌شوند.

در مرحله دوم، تمام داده‌ها بر اساس کلید بازنویسی می‌شوند؛ هر یک از دستورها، اگر متعلق به کلید بودند آن را نوشته در غیر این‌صورت از واژه "other" استفاده خواهد شد. بدین ترتیب یک کد میانی ایجاد خواهد شد که از لحاظ سرعت و حافظه صرفه جویی خواهد شد.

در مرحله سوم، از روی کد میانی ماتریسی ایجاد می‌شود که حاوی تعداد وقوع هر دستور به‌دنبال دستور دیگر در دنباله داده شده است؛ سپس با تقسیم هر مقدار به مجموع هر سطر در ماتریس، ماتریس احتمالی ایجاد می‌شود. برای هریک از دنباله‌های مجموعه آموزش این ماتریس ساخته شده و با استفاده از رابطه (۲) ماتریس E ایجاد می‌شود (m تعداد داده‌های مجموعه آموزش):

$$E=(matrix_{0ij} + matrix_{1ij}+\ldots+ matrix_{(m-1)\,ij})/m \qquad (۱)$$

گراف مجموعه آموزش از روی این ماتریس ساخته می‌شود. برای هر یک از فایل‌های مجموعه آزمون، ماتریس D از روی ماتریس احتمالی دنباله مورد نظر ایجاد می‌شود و گراف تشابه D از روی این ماتریس برای هر دنباله ساخته می‌شود.

در مرحله چهارم، امتیاز دنباله مورد نظر، با توجه به رابطه (۳) تعیین می‌شود:

$$Score(k) = d(D,E) = \frac{1}{N2}\sum |E - D1|^2 \qquad (۲)$$

امتیاز حاصل بیان‌گر میزان شباهت دنباله‌ها به مجموعه آموزش است.

فرض کنید برای ویژگی آپکد، کلید مورد نظر به‌صورت زیر باشد:

MOV, CALL, ADD, XOR

حال به‌عنوان نمونه، اگر دنباله آپکد برای امتیازدهی به‌صورت زیر باشد:

JMP,MOV,MOV,ADD,INC, INC, INC

با مقایسه کلید و دنبالهٔ مورد نظر، کد میانی به‌صورت زیر بازنویسی می‌شود:

Other, MOV, MOV, ADD, Other, Other, Other

از روی این کد، ماتریس توزیع گراف که حاوی تعداد وقوع هر آپکد به‌دنبال آپکد دیگر در فایل داده شده است، (جدول ۱) ایجاد می‌شود؛ سپس با تقسیم هر مقدار به مجموع هر سطر در ماتریس، ماتریس احتمالی (جدول۲) ایجاد شده و گراف تشابه آپکد از روی این جدول ساخته می‌شود (شکل ۱).

(جدول-۱): ماتریس توزیع گراف

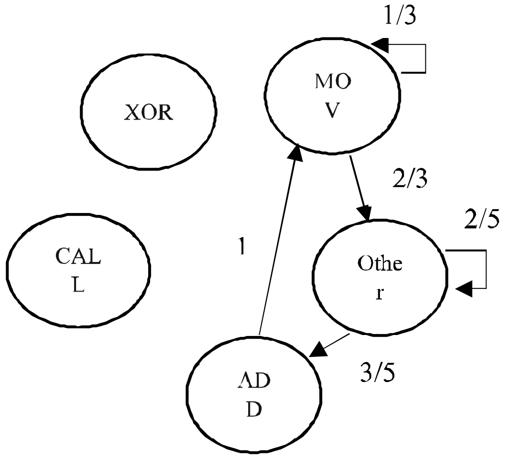
(Table-1): digraph distribution matrix

	MOV	CALL	ADD	XOR	Other
MOV	1	0	0	0	2
CALL	0	0	0	0	0
ADD	1	0	0	0	0
XOR	0	0	0	0	0
Other	0	0	3	0	2

[1] Static Structure
[2] Ensemble Averaging
[3] Bagging
[4] Boosting

(جدول-۲): ماتریس احتمالی جدول (۱)
(Table-2): Probabilities matrix from Table1

Other	XOR	ADD	CALL	MOV	
2/3	0	0	0	1/3	MOV
0	0	0	0	0	CALL
0	0	0	0	1	ADD
0	0	0	0	0	XOR
2/5	0	3/5	0	0	Other



(شکل-۱):گراف پیشنهادی بر روی آپکد
(Figure-1): The proposed graph for opcode

۴- شبیه‌سازی

در این بخش مجموعه‌داده، معیارهای ارزیابی و نتایج آزمایش‌های انجام‌شده برای ارزیابی کارایی روش پیشنهادی شرح داده می‌شود.

۴-۱- مجموعه‌داده

مجموعه بدافزارهای آزمایش‌ها از مجموعه VX Heaven Virus Collection[1] گردآوری شده و شامل سه گونه مختلف ویروس‌ها، تروجان و درب پشتی است. برنامه پاک و بی‌خطر از فایل‌های اجرایی ویندوز از پوشه Program Files یا System32 جمع آوری شده است. بدین ترتیب چهار طبقه مختلف شامل فایل‌های سالم، ویروس، تروجان و درب پشتی (از هر طبقه دویست نمونه) وجود دارد.

استخراج ویژگی مبتنی بر محتوای اسمبلی فایل‌ها صورت گرفته است. سه ویژگی آپکد، هگزا دسیمال و فراخوانی سامانه‌ای برای هر نمونه از روی کد اسمبلی آنها استخراج خواهد شد.

[1] http://vxheaven.org

۴-۲- معیارهای ارزیابی

از آنجا که سامانه تشخیص، چهار طبقه است، معیارهای ارزیابی سامانه پیشنهادی بر اساس سامانه چندطبقه است [13,14].

تعاریف مورد نیاز برای معیارهای ارزیابی به‌صورت زیر هستند:

Tp (True Positive) – تعداد بدافزارهایی که مشکوک شناخته شده‌اند.

Tn (True Negative) - تعداد برنامه‌های سالمی که سالم شناخته شده‌اند.

Fp (False Positive) - تعداد برنامه های‌سالمی که مشکوک شناخته شده‌اند.

Fn (False Negative) – تعداد بدافزارهایی که سالم شناخته شده‌اند.

مقدار صحت سامانه[2]: میزان نزدیکی آن معیار به اندازه حقیقی سامانه گفته می‌شود.

$$\text{Average Accuracy}=\frac{\sum_{i=1}^{l}\frac{tpi+tn\,i}{tp\,i+fp\,i+tn\,i+fn\,i}}{1} \quad (۳)$$

معیار دقت [3] میزانی است که نشان می‌دهد اگر شرایط عوض نشود سامانه تا چه حدی همان نتیجه را به‌دست خواهد آورد و براساس فرمول (۵) محاسبه می‌شود:

$$\text{Precision}_{\mu}=\frac{\sum_{i=1}^{l}TPi}{\sum_{i=1}^{l}TPi+FPi} \quad (۴)$$

معیار شفافیت یا اختصاصی بودن [4] نشان‌دهندهٔ نسبتی از حالت منفی است که درست تشخیص داده شده‌اند.

$$\text{Specificity}_{\mu}=\frac{\sum_{i=1}^{l}TNi}{\sum_{i=1}^{l}TNi+FPi} \quad (۵)$$

معیار حساسیت[5] نشان‌دهندهٔ نسبتی از حالت مثبت است که درست شناخته شده‌اند.

$$\text{Sensitivity}_{\mu}\,(\text{Recall})=\frac{\sum_{i=1}^{l}TPi}{\sum_{i=1}^{l}TPi+FNi} \quad (۶)$$

F-measure دو معیار دقت و حساسیت را هم‌زمان مورد بررسی قرار می‌دهد. از این رو این معیار را به‌عنوان مجموع وزن‌دار این دو کمیت نیز می‌توان در نظر گرفت. درحقیقت به این معیار می‌توان به عنوان میانگین هارمونیک بین این دو معیار دقت و حساسیت نگاه کرد. بهترین پاسخ این معیار یک و بدترین پاسخ، صفر است.

$$F1_{\mu}=2.\frac{\text{Precision}\mu * \text{Recall}\,\mu}{\text{Precision}\mu+\text{Recall}\mu} \quad (۷)$$

[2] Accuracy
[3] Precision (Positive Predictive Value)
[4] Specificity
[5] Sensitivity

برای هر آزمایش، از اعتبارسنجی ضرب‌دری با پنج فولد استفاده می‌شود. در این نوع اعتبارسنجی داده‌ها به پنج زیرمجموعه افراز می‌شوند. از این پنج زیرمجموعه، هر بار یکی برای اعتبارسنجی و چهارتای دیگر برای آموزش به‌کار می‌روند. این روال پنج‌بار تکرار می‌شود و همهٔ داده‌ها به‌طوردقیق یک‌بار برای آموزش و یک‌بار برای اعتبارسنجی به‌کار می‌روند. درنهایت میانگین نتیجه این پنج بار اعتبارسنجی به‌عنوان یک تخمین نهایی برگزیده می‌شود.

۳-۴- نتایج

تابع امتیازدهی و اندازه کلید در روش پیشنهادی گراف تشابه، پارامترهایی هستند که باید مقدار بهینهٔ آنها تعیین شود. تغییرات در هر یک از این پارامترها اثر قابل توجهی بر عملکرد الگوریتم خواهد داشت.

مقادیر مختلف اندازه کلید برای هر یک از ویژگی‌های فراخوانی سامانه‌ای، آپکد و هگزا دسیمال به‌صورت جداگانه مورد آزمایش قرار گرفته است. دنباله‌های موجود، حاوی تعداد زیادی آپکد، دستورهای هگزا دسیمال و تابع فراخوانی سامانه‌ای مجزا هستند؛ از این دستورها، تنها تعداد خاصی از آنها به‌طور مداوم ظاهر می‌شوند و در طبیعت ذاتی بدافزار یا فایل خوش‌خیم سهیم هستند. در جداول (۳،۴و۵) تأثیر مقادیر مختلف N بر کارایی روش گراف تشابه نمایش داده شده است.

با توجه به جدول (۳)، مقادیر ۲۵ و ۱۰۰ برای آپکد نتایج بهتری داشته است. برای اینکه اندازه ماتریس بزرگ نشود و مقایسه آسان‌تر شود، مقدار ارزش ۲۵ برای کلید در نظر گرفته می‌شود. ازاین‌رو، ماتریس D و E برای آپکد در اندازه ۲۶*۲۶ ساخته شده است. این تعجب‌آور نیست که تعداد به‌نسبه کمی از دستورها نتایج بهتری از مقادیر بزرگ‌تر دارند، زیرا پرتکرارترین آپکدها می‌توانند رفتار کل آپکدها در نمونه‌های بدافزار را توصیف کنند.

با توجه به نتایج جدول (۴)، مقدار ده برای ویژگی هگزا دسیمال نتایج بهتری داشته است. ازاین‌رو، ماتریس D و E برای ویژگی هگزا دسیمال در اندازهٔ ۱۱*۱۱ساخته شده است. با توجه به ماهیت دستورهای هگزا دسیمال که اطلاعات خام برای شیوه‌نامهٔ مورد نظر هستند، تعداد کمی از آنها به‌عنوان پرتکرارترین دستورها می‌تواند کل دستورهای هگزا دسیمال را توصیف کند.

(جدول-۳): تأثیر مقادیر مختلف کلید بر کارایی روش گراف تشابه آپکد

(Table-3): The effect of different values of key on the performance of proposed Opcode graph

F1	شفافیت	حساسیت	دقت	صحت	
0.57	0.86	0.57	0.57	0.78	**N=10**
0.68	0.89	0.68	0.68	0.84	**N=25**
0.52	0.84	0.52	0.52	0.76	**N=50**
0.68	0.89	0.68	0.68	0.84	**N=100**
0.66	0.89	0.66	0.66	0.83	**N=120**
0.65	0.88	0.65	0.65	0.83	**N=150**
0.56	0.85	0.56	0.56	0.78	**N=200**

(جدول-۴): تأثیر مقادیر مختلف کلید بر کارایی روش گراف پیشنهادی هگزا دسیمال

(Table-4): The effect of different values of key on the performance of proposed Hexadecimal graph

F1	شفافیت	حساسیت	دقت	صحت	
0.61	0.87	0.61	0.61	0.80	**N=10**
0.48	0.83	0.48	0.48	0.74	**N=25**
0.50	0.83	0.50	0.50	0.75	**N=50**
0.46	0.82	0.46	0.46	0.73	**N=100**
0.41	0.80	0.41	0.41	0.71	**N=120**
0.48	0.83	0.48	0.48	0.74	**N=150**
0.59	0.86	0.59	0.59	0.79	**N=200**

(جدول-۵): تأثیر مقادیر مختلف کلید بر کارایی روش گراف پیشنهادی فراخوانی سیستمی

(Table-5): The effect of different values of key on the performance of proposed system call graph

F1	شفافیت	حساسیت	دقت	صحت	
0.59	.0.86	0.59	0.59	0.79	**N=10**
0.61	0.87	0.61	0.61	0.81	**N=25**
0.58	0.86	0.58	0.58	0.79	**N=50**
0.63	0.88	0.63	0.63	0.81	**N=100**
0.61	0.87	0.61	0.61	0.80	**N=120**
0.62	0.87	0.62	0.62	0.81	**N=150**
0.53	0.84	0.53	0.53	0.77	**N=200**

با توجه به جدول (۵)، مقادیر ۲۵ و ۱۰۰ برای ویژگی فراخوانی سامانه‌ای نتایج بهتری داشته است؛ اما ارزش ۱۰۰ از صحت و حساسیت بهتری برخوردار است. با توجه به این که فراخوانی سامانه‌ای می‌تواند رفتار یک قطعه کد بدافزار را نشان دهد، برای تمایز بهتر بین گونه‌های مختلف بدافزارها،

ماتریس D و E برای ویژگی فراخوانی سامانه‌ای در اندازه ۱۰۰*۱۰۰ساخته شده است.

مقدار بهینهٔ تابع امتیازدهی در روش گراف تشابه نیز باید تعیین شود. به‌منظور تعیین تابع امتیازدهی مؤثر برای ارزیابی روش گراف تشابه، چهار تابع به‌صورت جداگانه در روش گراف تشابه اعمال شده است.

$$\text{Score1}: \sum_{i,j} |\boldsymbol{dij - eij}| \quad (۸)$$

$$\text{Score2}: \frac{\boldsymbol{1}}{\boldsymbol{n2}} \sum_{i,j} |\boldsymbol{dij - eij}|^2 \quad (۹)$$

$$\text{Score3}: \frac{\boldsymbol{1}}{\boldsymbol{n2}} \sum_{i,j} |\boldsymbol{dij - eij}| \quad (۱۰)$$

$$\text{Score4}: \sum_{i,j} |\boldsymbol{dij - eij}|^2 \quad (۱۱)$$

نتایج حاصل از اعمال هر یک از این توابع در جدول (۶) آمده است.

(جدول-۶): مقایسه توابع امتیازدهی در روش گراف پیشنهادی
(Table-6): Compare Scoring functions in the proposed graph method

تابع	صحت	دقت	حساسیت	شفافیت	F1
Score1	0.87	0.74	0.74	0.91	0.74
Score2	0.89	0.79	0.79	0.93	0.79
Score3	0.87	0.74	0.74	0.91	0.74
Score4	0.89	0.78	0.78	0.93	0.78

از آنجا که فاصلهٔ اقلیدسی فاصلهٔ مستقیم بین دو نقطه است که درحقیقت اندازه کوتاه‌ترین خط بین دو نقطه است، با توجه به نتایج به‌دست‌آمده، مربع فاصله اقلیدسی دو گراف تشابه (تابع۲ و تابع۴) نتایج بهتری نسبت به دیگر توابع داشته‌اند. تابع۲ با توجه به اینکه تعداد گره‌ها در آن دخیل است با تفاوت اندکی نتایج بهتری نسبت به تابع۴ در بر دارد. بابراین از این تابع در روش گراف تشابه استفاده شده است.

با توجه به امتیازات به‌دست‌آمده، یک الگوی جدید از مجموعه‌داده ایجاد شده است که به جای استفاده‌ی مستقیم از داده‌ها و ویژگی‌های استخراج‌شده با طول متغیر و حجم زیاد، می‌توان از این الگو برای ورودی دسته‌بندها استفاده کرد. در این مقاله از دسته‌بند ماشین بردار پشتیبان با روش یکی در مقابل یکی برای تعیین دسته مطلوب نمونه‌ها استفاده شده است.

در رویکرد نخست، هر سه روش‌های امتیازدهی شامل مدل مخفی مارکوف، فاصله جایگزینی ساده و گراف تشابه بر روی هر سه ویژگی اعمال و از ترکیب نتایج آنها در ورودی دسته‌بندها استفاده شده است (جدول۷).

(جدول-۷): ترکیب سه روش امتیازدهی بر روی هر سه ویژگی
(Table-7): The combination of three scoring functions on all three features

صحت	دقت	حساسیت	شفافیت	F1	خطا
0.93	0.86	0.86	0.95	0.86	0.07

پس از ترکیب سه روش امتیازدهی بر روی هر سه ویژگی، نسبت به هر یک از روش‌های امتیازدهی به‌صورت فردی نتایج پیشرفت چشم‌گیری داشته است؛ لذا در سامانه پیشنهادی از تمام ترکیبات ویژگی استفاده شده است. بدین صورت می‌توان از مزایای هر روش و هر ویژگی استفاده کرد و به تشخیص بدافزارهای ناشناخته کمک کرد.

در رویکرد دوم، ترکیب دسته‌بندها در خروجی مورد آزمایش قرار گرفته است. در بخش نخست، امتیازات حاصل از هر یک از روش‌های مدل مخفی مارکوف، فاصله جایگزینی ساده و گراف تشابه به سه دسته‌بند ماشین بردار پشتیبان داده شده است. خروجی‌ها بر اساس وزن‌دهی به هر یک از روش‌ها ترکیب شده و تصمیم نهایی بر اساس رأی اکثریت گرفته شده است. نتایج حاصل از این ترکیبات در جدول (۸) آورده شده است. سطر نخست این جدول مربوط به حالتی است که به روش گراف تشابه وزن سه و به دو رویکرد دیگر وزن یک داده شده است. سطر دوم، به روش فاصله جایگزینی ساده وزن سه و به دو روش دیگر وزن یک و سطر سوم به مدل مخفی مارکوف وزن سه و به دو رویکرد دیگر وزن یک داده شده است. سطر چهارم حالتی است که به هر سه روش وزن یکسان داده است.

(جدول-۸): ترکیب نتایج ماشین بردار پشتیبان
(Table-8): Combine results support vector machine

	صحت	دقت	حساسیت	شفافیت	F1
1	0.94	0.88	0.88	0.96	0.88
2	0.91	0.81	0.81	0.94	0.81
3	0.89	0.78	0.78	0.93	0.78
4	0.92	0.84	0.84	0.95	0.84

از میان ترکیبات ماشین بردار پشتیبان با وزن‌دهی متفاوت، زمانی که به روش گراف تشابه وزن بیشتری داده می‌شود، نتیجه بهتری می‌دهد؛ چون روش گراف تشابه نسبت به دو روش دیگر معیارهای ارزیابی بالاتری دارد.

در بخش دوم با توجه به قدرت و مزایای هر دسته‌بند، از ترکیب خروجی دسته‌بندها استفاده شده است. سه روش ترکیب دسته‌بندها شامل میانگین گروه، روش بگینگ و آدابوست مورد آزمایش قرار گرفته است (جدول۹).

(جدول-۹): ترکیب دسته‌بندها
(Table-9): Combine Classifications

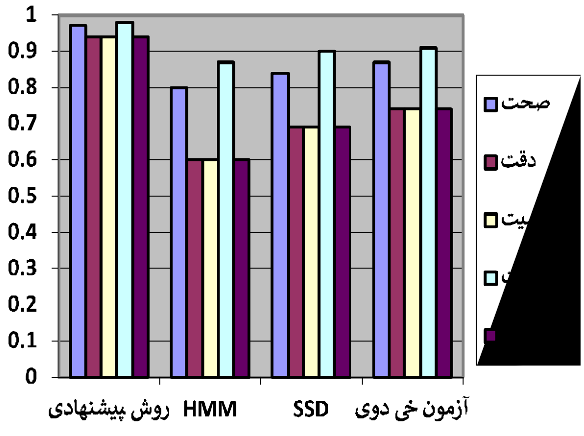
	صحت	دقت	حساسیت	شفافیت	F1	خطا
میانگین گروه	0.97	0.94	0.94	0.98	0.94	0.03
بگینگ	0.93	0.86	0.86	0.95	0.86	0.07
آدابوست	0.84	0.68	0.68	0.89	0.68	0.16

میانگین گروه از ترکیب چهار دسته‌بند جنگل تصادفی، ماشین بردار پشتیبان، K-نزدیک‌ترین همسایه و نایو بیز ایجاد شده و تصمیم بر اساس رأی اکثریت گرفته شده است؛ این تنوع باعث شده که از قدرت و مزایای سامانه‌های مختلف استفاده کند. بنابراین به عنوان روش پیشنهادی از آن استفاده شده است.

در جدول (۱۰) و شکل (۲) کارایی روش پیشنهادی با روش‌های مرسوم که در مقالات دیگر به دفعات مورد استفاده قرار گرفته، مقایسه شده است.

(جدول-۱۰): مقایسه روش پیشنهادی با روش‌های استفاده‌شده
(Table-10): Comparison of proposed methods with the methods used

	صحت	دقت	حساسیت	شفافیت	F1	خطا
روش پیشنهادی	0.97	0.94	0.94	0.98	0.94	0.03
HMM	0.80	0.60	0.60	0.87	0.60	0.18
SSD	0.84	0.69	0.69	0.90	0.69	0.20
آزمون خی‌دوی	0.87	0.74	0.74	0.91	0.74	0.14



(شکل-۲): مقایسه روش پیشنهادی با روش‌های استفاده‌شده
(Figure-2): Comparison of proposed methods with the methods used

رویکرد پیشنهادی با صحت ۰/۹۷و نرخ خطای تشخیص ۰/۰۳ بدافزارهای ناشناخته را شناسایی کرد، درحالی‌که روش فاصله جانشینی ساده، مدل مخفی مارکوف و آزمون خی دوی تحت شرایط یکسان با دقت ۰/۸، ۰/۸۴، ۰/۸۷ و نرخ خطای تشخیص ۰/۱۸، ۰/۲۰ و ۰/۱۴توانستند بدافزارها را شناسایی کنند. رویکرد پیشنهادی دارای عملکرد بهتری نسبت به دیگر روش‌های پایه دارد.

۵- نتیجه‌گیری

در این مقاله، یک روش پیشنهادی مبتنی بر گراف برای دسته‌بندی دنباله‌های با طول متغیر بدون ثابت‌کردن طول دنباله‌ها در حوزه آشکارسازی بدافزارها پیشنهاد شد. روش پیشنهادی شامل چهار مرحله تعیین کلید، ایجاد کد میانی، ایجاد گراف تشابه E و D از روی ماتریس توزیع و تعیین امتیاز است. روش پیشنهادی با تعیین پرتکرارترین دستورها و گذاشتن باقی دستورها در مجموعهٔ ‘other’ از لحاظ سرعت و حافظه صرفه‌جویی کرده است. آزمایش‌ها نشان می‌دهند گراف تشابه پیشنهادی، ضمن غلبه بر مشکلات روش‌های گراف سنتی در مواجه با داده‌های با طول متغیر، بهتر از روش‌های پایه همچون مدل مخفی مارکوف، روش فاصله جایگزینی ساده و آزمون خی‌دوی است. همچنین در این مقاله، از ترکیب دسته‌بندها در آشکارسازی بدافزارها با استفاده از دسته‌بندی دنباله‌های با طول متغیر استفاده شده است. روش پیشنهادی شامل دو رویکرد بود؛ در رویکرد نخست، روش‌های استخراج ویژگی شامل سه روش مدل مخفی مارکوف، فاصله جایگزینی ساده و روش گراف تشابه در ورودی ترکیب شدند؛ درحالی‌که در رویکرد دوم، از ترکیب دسته‌بندها در خروجی استفاده شد. در ترکیب روش‌های استخراج ویژگی، نسبت به هر یک از روش‌های امتیازدهی به‌صورت فردی نتایج پیشرفت چشم‌گیری داشته است. از میان ترکیبات دسته‌بندها در خروجی، میانگین گروه با ترکیب خروجی چهار دسته‌بند K-نزدیکترین همسایه، جنگل تصادفی، ماشین بردار پشتیبان و نایو بیز با دقت ۹۷٪ بالاترین نرخ تشخیص و کم‌ترین نرخ خطای تشخیص را داشت. در رویکرد پیشنهادی گراف می‌توان اشاره کرد که تغییرات جزئی در توابع امتیازدهی اثرات بسیار زیادی در نتایج در بر خواهد داشت. برای کشف دقت بیشتر، می‌توان به‌دنبال یک تابع امتیازدهی قوی‌تری بود. همچنین با استفاده از گراف روند کنترلی، در کنار گراف فراخوانی سامانه‌ای می‌توان گراف قدرتمندتری به‌دست آورد. روش پیشنهادی می‌تواند در دیگر حوزه‌هایی که با ویژگی‌های با طول متغیر سر و کار دارد به‌کار گرفته شود.

6- References ۶- مراجع

[1] J. Quinlan, "Bagging, Boosting and C4.5," 2006.

[2] M. Alazab, R. Layton, S. Venkataraman and P. Watters, "Malware Detection Based on Structural and Behavioural Features of API Calls," Perth, WA, 2010.

[3] C. T. Lin, N.-J. Wang, H. Xiao and C. Eckert, "Feature Selection and Extraction for Malware Classification," *journal of Information Science and Engineering 31*, vol. 31, no. 3, pp. 965-992, 2015.

[4] J. Xu, A. H. Sung, S. Mukkamala ,and Q. Liu, "Obfuscated Malicious Executable Scanner," *Journal of Research and Practice in Information Technology*, vol. 39, pp. 181-197, 2007.

[5] M. J. Landage and P. M.P.Wankhade, "Malware Detection with Different Voting Schemes," *COMPUSOFT, An international journal of advanced computer technology*, vol. 3, no. 1, pp. 450-456, 2014.

[6] W. Wong and M. Stamp, "Hunting for metamorphic engines," *Journal in Computer Virology*, vol. 2, no. 3, pp. 211–229, 2006.

[7] T. Kalbhor, "Dueling hidden Markov models for virus analysis," *Journal of Computer Virology and Hacking Techniques*, vol. 11, no. 2, pp. 103–118, 2015.

[8] S. Attaluri, S. McGhee and M. Stamp, "Profile hidden Markov models and metamorphic virus detection," *Journal in Computer Virology*, vol. 5, no. 2, pp. 151–169, 2009.

[9] C. Annachhatre and M. Stamp, "Hidden Markov models for malware classification," *Journal of Computer Virology and Hacking Techniques*, vol. 11, no. 2, pp. 59–73, 2015.

[10] S. Josse and E. Filiol, "New Trends in Security Evaluation of Bayesian Network-Based Malware Detection Models," Maui, Hawaii USA, 2012.

[11] T. Singh, F. D. Troia, V. A. Corrado, T. H. Austin and M. Stamp, "Support vector machines and malware detection," *Journal of Computer Virology and Hacking Techniques*, vol. 12, no. 4, pp. 203–212, 2016.

[12] H.Ghaemi and M.Kahani, "Question classification using ensemble classifiers," *Quarterly Journal Signal and Data Processing,* vol. 29, number.3, pp.99, 1395.

[۱۲]هادی قائمی و محسن کاهانی، "دسته‌بندی پرسش ها با استفاده از ترکیب دسته‌بندها" ، فصل‌نامه علمی-پژوهشی پردازش علائم و داده‌ها، شماره ۳، پیاپی ۲۹، صفحه ۹۹، سال ۹۵.

[13] V. Asch, "Macro- and micro-averaged evaluation measures [[BASIC DRAFT]]," univercity of Antwerp, 2013.

[14] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, pp. 427–437, 2009.

فاطمه حسینی مدرک کارشناسی خود را در رشته مهندسی کامپیوتر-نرم‌افزار از دانشگاه آزاد واحد ارسنجان در سال ۱۳۸۸ دریافت کرد. در حال حاضر ایشان به‌عنوان دانشجوی کارشناسی ارشد رشته هوش مصنوعی در واحد علوم و تحقیقات تهران مشغول به تحصیل هستند. پژوهش‌های ایشان بر روی امنیت اطلاعات و شبکه، داده‌کاوی و یادگیری ماشین متمرکز است.

نشانی رایانامه ایشان عبارت است از:

fatima.hosseini@srbiau.ac.ir

میترا میرزارضایی مدرک کارشناسی خود را در رشته مهندسی کامپیوتر-نرم‌افزار از دانشگاه آزاد واحد تهران مرکزی و همچنین مدارک کارشناسی ارشد و دکترای خود را در رشته مهندسی کامپیوتر-هوش مصنوعی از دانشگاه آزاد اسلامی واحد علوم و تحقیقات دریافت کرد. درحال حاضر ایشان عضو هیأت علمی تمام‌وقت واحد علوم و تحقیقات هستند. زمینه‌های پژوهشی مورد علاقه ایشان شناسایی الگو و کاربردهای آن است.

نشانی رایانامه ایشان عبارت است از:

mirzarezaee@srbiau.ac.ir

آرش شریفی مدرک کارشناسی خود را در رشته مهندسی کامپیوتر-سخت‌افزار از دانشگاه آزاد واحد تهران جنوب در سال ۱۳۸۳ دریافت و همچنین مدارک کارشناسی ارشد و دکتری خود را در رشته مهندسی کامپیوتر-هوش مصنوعی در سال‌های ۱۳۸۶ و ۱۳۹۰ از دانشگاه آزاد واحد علوم و تحقیقات دریافت کرد. در حال حاضر ایشان عضو هیات علمی تمام‌وقت واحد علوم و تحقیقات هستند. زمینه‌های پژوهشی مورد علاقه ایشان عبارتند از: محاسبات نرم، یادگیری عمیق و پردازش تصویر.

نشانی رایانامه ایشان عبارت است از:

a.sharifi@srbiau.ac.ir