سومین کنفرانس ملی مباحث نوین در کامپیوتر و فناوری اطلاعات
3ed National Conference on Advanced Topics in Computer and Information Technology
بیست و هشتم آذر ماه ۱۳۹۸

روشی جهت تشخیص بدافزار با استفاده از الگوریتم‌های داده‌کاوی و هوش مصنوعی

مجید پورافشار (۱) – عاطفه محمدی (۲)
(۱) گروه مهندسی کامپیوتر، واحد اهواز، دانشگاه آزاد اسلامی، اهواز، ایران
majidpourafshar@gmail.com
(۲) گروه مهندسی کامپیوتر، واحد اهواز، دانشگاه آزاد اسلامی، اهواز، ایران
Mohammadi289@yahoo.com

خلاصه: بدافزار به هرگونه برنامه کامپیوتری اطلاق می‌شود که دارای اهداف مخرب باشد. این برنامه‌ها مهمترین تهدید برای سیستم‌های کامپیوتری به حساب می‌آیند. تنوع این بدافزارها باعث محدود شدن راه کارهای مقابله با آنها شده است، به گونه‌ای که روزانه میلیون‌ها سیستم کامپیوتری بر اثر آسیب‌های ناشی از انواع ویروس‌ها، تروجان‌ها و کرم‌های اینترنتی و غیره آلوده می‌شوند. در سال های اخیر یکی از مهمترین چالش‌های امنیت اطلاعات و شبکه‌های ارتباطی، افزایش روز افزون انواع بدافزارها و به دنبال آن یافتن راه‌های مناسب جهت حفاظت سیستم‌ها در مقابل آنهاست که از مهمترین دغدغه‌های برنامه‌نویسان و متخصصین امنیت اطلاعات، شناخت به موقع و یافتن راه‌های مقابله با اثرات مخرب این‌گونه بدافزارها می‌باشد. در این راستا طی سال‌های اخیر استفاده از الگوریتم‌های داده‌کاوی و هوش مصنوعی بعنوان یکی از روش‌های نوظهور و امیدوار کننده توانسته است کاربرد بسیاری جهت شناسایی و تشخیص انواع بدافزارها داشته باشد. لذا در این تحقیق سعی کردیم با استفاده از شبکه عصبی مصنوعی و الگوریتم ازدحام ذرات، فایل‌های آلوده به بدافزار را تشخیص دهیم. پیاده‌سازی روش پیشنهادی نشان می‌دهد که توانسته است فایل‌های آلوده به بدافزار را با استفاده از مجموعه داده مربوط به فایل‌های سالم و آلوده به بدافزار با دقت ۰.۹۱درصد تشخیص دهد که نشان از عملکرد بالای آن دارد.

کلمات کلیدی: بدافزار، داده‌کاوی، هوش مصنوعی، الگوریتم ازدحام ذرات ، شبکه عصبی مصنوعی.

۱ – مقدمه

بدافزار به هرگونه برنامه کامپیوتری اطلاق می‌شود که دارای اهداف مخرب باشد. این برنامه‌ها مهمترین تهدید برای سیستم‌های کامپیوتری به حساب می‌آیند. تنوع این بدافزارها باعث محدود شدن راهکارهای مقابله با آنها شده است، به گونه‌ای که روزانه میلیون‌ها سیستم کامپیوتری بر اثر آسیب‌های ناشی از انواع ویروس‌ها، تروجان‌ها و کرم‌های اینترنتی آلوده می‌شوند[۲،۱]. روش‌های شناسایی بدافزار از دیدگاه‌های مختلفی به سه گروه روش‌های برپایه امضا، روش مبتنی بر محتوا و روش‌هایی برپایه رفتار تقسیم می‌شود. روش‌های مبتنی بر امضاء از الگوهای استخراج شده از بدافزارهای مختلف استفاده می‌کنند و این باعث می‌شود که آنها نسبت به دیگر روش‌های تشخیص ویروس بسیار سریع‌تر، موثر و کارا عمل کنند. این روش‌ها نمی‌توانند انواع بدافزار ناشناخته را شناسایی کنند و نیاز به مقدار زیادی نیروی انسانی، زمان و پول دارند تا امضاءهای منحصربفردی را بدست آورند که این از معایب این روش‌ها بحساب می‌آید. از دیگر نقص‌های این روش ناتوانی در مقابله با بدافزارهایی است که کدهایشان در حملات مختلف تغییر می‌کنند نظیر

پلی‌مورفیک و متامورفیک[۳].

به منظور مقابله با محدودیت‌های روش مبتنی بر محتوا، روش‌ها، تجزیه، تحلیل و طبقه‌بندی مبتنی بر رفتار مخرب‌ها مطرح شده است. در این روش نمونه رفتار بدافزار در زمان اجرا توسط یک مشخصه رفتاری ارائه شده است. پس از آن طبقه‌بندی رفتاری پروفایل‌ها برای تجزیه و تحلیل بیشتر استفاده می‌شود. روش‌های شناسایی براساس رفتار برنامه‌ای را جهت تشخیص اینکه یک نرم افزار بدخواه و مخرب است یا نه در نظر می‌گیرد. با توجه به اینکه روش‌های مبتنی بر رفتار؛ عملکرد فایل‌های اجرایی را در نظر می‌گیرد در نتیجه روی نقاط ضعف روش‌های مبتنی بر امضاء حساس نشدند. به عبارتی دیگر در روش شناسایی بر اساس رفتار الگوریتم تشخیص روی عملکرد حساس می‌شود نه گفته‌ها[۴]. مزیت عمده روش‌های شناسایی بدافزار بر اساس رفتار آن است که توانایی شناسایی بدافزارهای چندریخت که تکنیک‌های مبتنی بر امضاء نمی‌توانند آنها را تشخیص دهند را دارد و از طرفی دیگر طولانی بودن زمان اسکن از معایب اصلی روش‌های شناسایی بدافزار مبتنی بر رفتار می‌باشد[۵].

در طول سال‌های گذشته، روش‌های مختلفی به منظور تشخیص بدافزارها ارائه شده است از جمله در سال ۲۰۱۸ Sayadi و همکاران از روش Pipe-Line برای بهبود عملکرد زمانی جهت تشخیص بدافزار استفاده کردند[۶]. در سال ۲۰۱۸ Rehman و همکاران طبقه‌بندهای مختلفی را برای دسته‌بندی برنامه‌ها به عنوان یک فرایند مجاز یا مخرب درج نمودند[۷]. در سال ۲۰۱۸ Gupta به همراه همکارش مقاله‌ای با موضوع "ارزیابی تشخیص نرم‌افزارهای مخرب با استفاده از شمارنده عملکرد سخت‌افزار" را ارائه کردند[۸]. در سال ۲۰۱۷ Gulmezoglu و همکارانش در مقاله "چگونگی نقص حریم خصوصی وب سایت با رخداد ویژگی سخت‌افزاری"، نشان دادند که با استفاده از تکنیک‌های پیشرفته یادگیری ماشین‌ها محدوده بیکار معماری مرورگرها را بررسی و آنالیز می‌کنند[۹]. در سال ۲۰۱۷ Patel و همکارانش در مقاله‌ای با عنوان "تجزیه و تحلیل شناسایی بدافزار مبتنی بر سخت افزار"، بر این باور هستند اثربخشی این روش‌های یادگیری عمدتا متکی به اطلاعاتی است که توسط تعداد محدود شمارنده‌های عملکرد سخت‌افزاری ارائه می‌شود[۱۰].

در سال ۲۰۱۷ Singh و همکاران در مقاله‌ای بعنوان "شناسایی روت‌کیت در ریشه‌های سطح هسته با استفاده از شمارنده عملکرد سخت‌افزاری"، یک تحلیل جامع‌تر از کاربرد استفاده از یادگیری ماشین و شمارنده‌های عملکرد سخت‌افزاری را برای یک زیرمجموعه خاص از نرم‌افزارهای مخرب ارائه دادند[۱۱]. در سال ۲۰۱۶ Fan و همکاران در مقاله خود بر اساس دنباله‌ای دستورات استخراج شده از مجموعه نمونه فایل، الگوریتمی موثری برای الگوهای متوالی مخرب کشف کردند[۱۲]. در سال ۲۰۱۶ Ozsoy به همراه همکارانش در مقاله‌ای با نام "تشخیص بدافزار مبتنی بر سخت‌افزار با استفاده از ویژگی‌های معماری سطح پایین"، یک موتور تشخیص بدافزار سخت‌افزاری به نام map ارائه دادند[۱۳]. در سال ۲۰۱۶ Huda و همکاران در مقاله خود چارچوبی ترکیبی برای تشخیص بدافزار با استفاده از ترکیب ماشین بردار پشتیبان، فیلتر معیار حداکثر افزونگی حداقل ارائه کردند[۱۴].

در سال ۲۰۱۶ Khammas و همکاران در مقاله خود ویژگی‌های تغییر داده نشده در ساختار بدافزارهای فراریخت را برای استفاده در فرایند شناسایی بدافزار با استفاده از ماشین بردار پشتیبان ارائه دادند[۱۵]. در سال ۲۰۱۶ Grosse و همکارانش از روش یادگیری عمیق برای تشخیص بدافزار استفاده نمودند[۱۶]. در سال ۲۰۱۶ Kolosnjaji و همکاران در تحقیقشان از مدل‌های اکتشافی نظیر الگوریتم فاخته و روش جستجوی حریصانه برای تشخیص بدافزار استفاده نمودند[۱۷].

باتوجه به اینکه امروزه یکی از مهمترین دغدغه‌های برنامه‌نویسان و متخصصین امنیت اطلاعات، افزایش روزافزون بدافزارها و به دنبال آن یافتن راه‌های مناسب جهت حفاظت سیستم‌ها در مقابل آنهاست؛ لذا استفاده از الگوریتم‌های هوش مصنوعی یکی از روش‌های نوظهور و امیدوار کننده برای تشخیص بدافزار می‌باشد که می‌توان از آنها استفاده کرد. لذا در این تحقیق سعی داریم با استفاده از شبکه عصبی مصنوعی و الگوریتم ازدحام ذرات، فایل‌های آلوده به بدافزار را تشخیص دهیم.

۲ – شبکه عصبی مصنوعی

شبکه‌های عصبی مصنوعی یکی از گرایش‌های هوش مصنوعی به حساب می‌آیند که بعنوان یکی از روش‌های مهم در تجزیه و تحلیل داده‌ها استفاده می‌شوند و در سال ۱۹۴۳ میلادی توسط مک‌کالاچ و پیتر معرفی شدند[۱۸]. مغز و سیستم عصبی انسان دارای پیچیدگی بالایی است و این پیچیدگی این امکان را به انسان می‌دهد تا محاسبات پیچیده‌ای را به خوبی انجام دهد. فعالیت و پردازش مغز انسان در واقع حاصل ارتباطات این سلول‌ها است که شبکه پیچیده‌ای را ایجاد می‌نماید. بنابراین شبکه عصبی مصنوعی با الگوبرداری از شیوه پردازش اطلاعات در سلول‌های عصبی یک روش کارآمد برای تشخیص الگو و طبقه‌بندی[1] اطلاعات می‌باشد[۱۹].

۳ – الگوریتم ازدحام ذرات

در سال ۱۹۹۵ کندی و ابرهارت برای اولین بار مفهوم بهینه‌سازی ازدحام ذرات را مطرح کردند [۲۰]. الگوریتم ازدحام ذرات یک الگوریتم جستجوی جمعی است که از روی رفتار اجتماعی دسته‌های پرندگان مدل شده است. در ابتدا این الگوریتم به منظور کشف الگوهای حاکم بر پرواز همزمان پرندگان و تغییر ناگهانی مسیر آنها و تغییر شکل بهینه دسته به کار گرفته شد. دراین الگوریتم ذرات در فضای جستجو جاری می‌شوند. تغییر مکان ذرات در فضای جستجو تحت تأثیر تجربه و دانش خودشان و همسایگانشان است؛ بنابراین موقعیت دیگر توده ذرات روی چگونگی جستجوی یک ذره اثر می‌گذارد. نتیجه مدلسازی این رفتار اجتماعی فرایند جستجویی است که ذرات به سمت نواحی موفق میل

[1] Classification

می‌کنند. ذرات از یکدیگر می‌آموزند و بر مبنای دانش بدست آمده به سمت بهترین همسایگان خود می‌روند. اساس کار این الگوریتم بر این اصل استوار است که در هر لحظه هر ذره مکان خود را در فضای جستجو با توجه به بهترین مکانی که تاکنون در آن قرار گرفته است و بهترین مکانی که در کل همسایگانش وجود دارد، تنظیم می‌کند.

۴ – ارائه روش پیشنهادی

در این بخش از تحقیق به بررسی روشی جهت تشخیص فایل‌های آلوده به بدافزار براساس الگوریتم ازدحام ذرات و شبکه عصبی مصنوعی می‌پردازیم. در روش پیشنهادی می‌خواهیم اوزان و آستانه مناسب شبکه عصبی مصنوعی را با استفاده از الگوریتم ازدحام ذرات تعیین کنیم.

۵ – مجموعه داده مورد استفاده

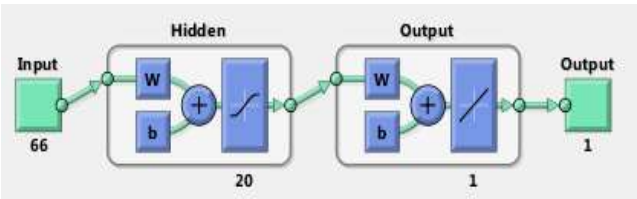
مجموعه داده مورد استفاده در این تحقیق از سایت virusign.com دریافت شد که شامل ۵۵۴ نمونه و ۶۷ ویژگی است که ۶۶ ویژگی آن از نوع اصلی و ویژگی آخر آن نشان‌دهنده وجود و یا عدم وجود بدافزار در فایل است. باتوجه به اینکه مقادیر داده‌ها در یک بازه مشابه قرار ندارند لذا در این تحقیق از روش نرمالیزاسیون آماری $max-min$، در بازه [1,-1] جهت نرمالیزه کردن داده‌ها طبق رابطه(۱) استفاده می‌کنیم[۲۱] :

$$V_{norm} = \frac{V - min(V)}{max(V) - min(V)} * 2 - 1 \qquad (۱)$$

همانطور که در رابطه(۱) آمده است V مقدار داده مورد نظر جهت نرمال شدن، $min(V)$ کمینه بردار ورودی V و $max(V)$ بیشینه بردار ورودی V بوده و V_{norm} مقدار نرمال شده V است.

۶- بهبود شبکه عصبی توسط الگوریتم ازدحام ذرات

ساختار شبکه عصبی پیشنهادی در این تحقیق که از نوع پیشخور می‌باشد در شکل(۱) نشان داده شده است.



شکل۱: ساختار شبکه عصبی پیشنهادی

میزان کیفیت شبکه عصبی پیشنهادی در تشخیص فایل‌های آلوده به بدافزار با متوسط خطای طبقه‌بندی[۲] تشخیص بدافزار طبق رابطه(۲) سنجیده می‌شود:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad (۲)$$

در رابطه(۲) y_i شماره واقعی کلاس نمونه iام و $\hat{y}_i$ شماره کلاس پیش‌بینی نمونه iام که توسط شبکه عصبی تشخیص داده شده است و n تعداد نمونه‌های آزمون بکار رفته در روش پیشنهادی است. در این تحقیق سعی خواهیم داشت که بوسیله الگوریتم ازدحام ذرات با انتخاب اوزان و آستانه‌های مناسب و بهینه مقدار این خطا را کاهش دهیم. بنابراین پس از ایجاد شبکه عصبی پیشنهادی و تقسیم‌بندی داده‌ها به روش 10Fold، می‌بایست شبکه عصبی پیشنهادی را در قالب یک ذره از جمعیت الگوریتم ازدحام ذرات طبق رابطه(۳) کدگذاری کنیم.

$$(w,b) = [w_1, w_2, \ldots, w_n, b_1, \ldots, b_m] \qquad (۳)$$

که در این رابطه w_n و b_m به ترتیب اوزان و آستانه شبکه عصبی مصنوعی در نظر گرفته می‌شود و بردار (w,b) نیز بعنوان یک ذره از جمعیت الگوریتم ازدحام ذرات تعریف می‌شود. پس از قرار گرفتن ذرات بصورت تصادفی در فضا حلقه الگوریتم آغاز می‌گردد. در هر تکرار حلقه، ذرات به جستجوی مقادیر بهینه برای وزن‌ها می‌پردازند. این حلقه با توجه به تعداد تکرارهای در نظر گرفته شده جهت آموزش شبکه عصبی، اجرا می‌شود و در نهایت ذره‌ای که کمترین هزینه را دارد انتخاب شده و شبکه عصبی بر اساس مقدار ذره با کمترین هزینه ساخته می‌شود. در نتیجه با اتمام مرحله آموزش، عملکرد مدل پیشنهادی توسط این وزن‌های بهینه شده توسط مجموعه داده‌های آزمون مورد ارزیابی قرار می‌گیرد.

۷- ارزیابی روش پیشنهادی

در این تحقیق با استفاده از ماتریس کانفیوشن از معیارهای دقت، حساسیت و ویژگی طبق روابط (۴) تا (۶) جهت ارزیابی عملکرد روش پیشنهادی جهت تشخیص فایل‌های آلوده به بدافزار استفاده می‌کنیم [۲۲] :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (۴)$$

$$Sensitivity = \frac{TP}{TP+FN} \qquad (۵)$$

$$Specificity = \frac{TN}{FP+TN} \qquad (۶)$$

۶ – یافته‌ها

در این بخش از تحقیق نتایج بدست آمده از پیاده‌سازی روش پیشنهادی مورد بررسی قرار می‌گیرد. در این تحقیق روش پیشنهادی بوسیله الگوریتم ازدحام ذرات ۱۰۰۰ بار مورد آموزش قرار گرفت که در جدول(۱) میانگین نتایج داده‌های آزمایشی روش پیشنهادی با K=10 آمده است:

جدول ۱: میانگین نتایج داده‌های آزمایشی روش پیشنهادی

[2] Mean Square Error (MSE)

ویژگی (Specificity)	حساسیت (Sensitivity)	دقت (Accuracy)
0.93 %	0.89 %	0.91 %

۷- نتیجه‌گیری

امروزه با توجه به ضرورت استفاده از اینترنت، رشد چشمگیر شبکه‌ها و زیرساخت‌های رایانه‌ای و همچنین طراحی بدافزارهای پیچیده و پویایی که دائم در حال بروزرسانی خود می‌باشند، حفظ امنیت و نظارت بر ترافیک شبکه‌ها یکی از مهمترین ملزومات فضای سایبری می‌باشد. لذا همواره نیاز به روشی می‌باشد که بتواند به شناسایی و جلوگیری از نفوذ بدافزارها بپردازد. در این تحقیق روشی به منظور تشخیص فایل‌های آلوده به بدافزار با استفاده از شبکه عصبی مصنوعی بهینه شده بوسیله الگوریتم ازدحام ذرات ارائه گردید که برای انجام آزمایشات از مجموعه داده‌ای حاوی فایل‌های سالم و آلوده به بدافزار استفاده شد. نتایج بدست آمده از پیاده‌سازی روش پیشنهادی، نشان داد روش پیشنهادی با دقت۰,۹۱درصد فایل‌های آلوده به بدافزار را تشخیص می‌دهد. بنابراین می‌توان گفت روش پیشنهادی عملکرد خوبی در کاهش خطا و افزایش دقت تشخیص شبکه عصبی مصنوعی داشته است و در نتیجه به خوبی توانسته است فایل‌های آلوده به بدافزار را تشخیص دهد.

مراجع

[1] Milosevic, N., Dehghantanha, A., & Choo, K. K. R. "Machine Learning Aided Android Malware Classification". Computers & Electrical Engineering, 61, pp.266-274. (2017).

[2] Ali Alatwi, H., Oh, T., Fokoue, E., & Stackpole, B" .Android Malware Detection Using Category-Based Machine Learning Classifiers." In Proceedings Of The 17th Annual Conference On Information Technology Education (Pp. 54-59). ACM. . (2016).

[3] Damodaran., Anusha., et al., "A comparison of static, dynamic, and hybrid analysis for malware detection." Journal of Computer Virology and Hacking Techniques, pp.1-12. (2015).

[4] Ahmadi, M., Sami, A., Rahimi, H., & Yadegari, B.."Malware detection by be-havioural sequential patterns". Computer Fraud & Security, pp.11–19. (2013).

[5] Egele, M., Scholte, T., Kirda, E., & Kruegel, C. "A survey on automated dynamic malwareanalysis". techniques and tools.Computing Surveys,44,6. (2012).

[6] Sayadi, H., Patel, N., PD, S. M., Sasan, A., Rafatirad, S., & Homayoun, H. "Ensemble learning for effective run-time hardware-based malware detection: A comprehensive analysis and classification". In 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC) (pp. 1-6). (2018).

[7] Rehman, Z. U., Khan, S. N., Muhammad, K., Lee, J. W., Lv, Z., Baik, S. W., & Mehmood, I. 'Machine learning-assisted signature and heuristic-based detection of malwares in Android devices". Computers & Electrical Engineering, 69, 828-841. (2018).

[8] Gupta, A., "Assessing Hardware Performance Counters For Malware Detection" (Doctoral Dissertation, Boston University). (2017).

[9] Gulmezoglu, B., Zankl, A., Eisenbarth, T. And Sunar, B., "Perfweb: How To Violate Web Privacy With Hardware Performance Events". In European Symposium On Research In Computer Security (Pp. 80-97). Springer, Cham. (2017).

[10] Patel, N., Sasan, A. And Homayoun, H, "Analyzing Hardware Based Malware Detectors". In Proceedings Of The 54th Annual Design Automation Conference. (P. 25). ACM. (2017).

[11] Singh, B., Evtyushkin, D., Elwell, J., Riley, R. And Cervesato, I., "On The Detection Of Kernel-Level Rootkits Using Hardware Performance Counters". In Proceedings Of The 2017 ACM On Asia Conference On Computer And Communications Security (PP. 483-493). ACM. (2017).

[12] Fan, Y., Ye, Y. And Chen, L "Malicious Sequential Pattern Mining For Automatic Malware Detection".Expert Systems With Applications, 52, Pp.16-25. .,(2016).

[13] Ozsoy, M., Khasawneh, K.N., Donovick, C., Gorelik, I., Abu-Ghazaleh, N. And Ponomarev, D., "Hardware-Based Malware Detection Using Low-Level Architectural Features". IEEE Transactions On Computers, 65(11), Pp.3332-3344. (2016).

[14] Huda, S., Abawajy, J., Alazab, M., Abdollalihian, M., Islam, R. And Yearwood, J., "Hybrids Of Support Vector Machine Wrapper And Filter Based Framework For Malware Detection". Future Generation Computer Systems, 55, Pp.376-390. (2016).

[15] Khammas, B.M., Monemi, A., Ismail, I., Nor, S.M. And Marsono, M.N., "Metamorphic Malware Detection Based On Support Vector Machine Classification Of Malware Sub-Signatures".

Telkomnika (Telecommunication Computing Electronics And Control), 14(3), Pp.1157-1165. (2016).

[16] Grosse, K., Papernot, N., Manoharan, P., Backes, M., & McDaniel, P. (2016). "Adversarial perturbations against deep neural networks for malware classification". arXiv preprint arXiv:1606.04435.

[17] Kolosnjaji, B., Zarras, A., Lengyel, T., Webster, G., & Eckert, C. "Adaptive semantics-aware malware classification".In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (pp. 419-439). Springer, Cham. (2016).

[18] Mcculloch, W. S., & Pitts, W "A Logical Calculus Of The Ideas Immanent In Nervous Activity". The Bulletin Of Mathematical Biophysics, 5(4), pp.115-133. . (1943).

[19] Soltani.Z , Jafarian.A , ,"A New Artificial Neural Networks Approach For Diagnosing Diabetes Disease Type II ",International Journal Of Advanced Computer Science And Applications, 7(6). (2016)

[20] J. Kennedy and R. C. Eberhart, , "Particle swarm optimization", *Proceedings of the IEEE International Conference on Neural Networks*, 4, 1942–1948. (1995)

[21] Han, j., Kamber, M., Pei, j.,"Data Mining Concepts and Techniques", Morgan Kaufmann publishers is an imprint of Elsevier .(2012).

[22] Yerima, S. Y., & Sezer, S. "Droidfusion: A novel multilevel classifier fusion approach for android malware detection". IEEE transactions on cybernetics, 49(2), 453-466. (2018).