علوم رایانش و فناوری اطلاعات
نشریه علمی انجمن کامپیوتر ایران
مجلد ۱۸، شماره ۱، ۱۳۹۹
صفحات ۱۰۴-۹۲
مقاله عادی

به‌کارگیری نرخ آموزش پایدار شبکه‌های عصبی خودرمزگذار به‌منظور تشخیص ناهنجاری برای دیواره آتش برنامه کاربردی وب

علی مرادی ورتونی[۱]، محمد تشنه‌لب[۲*]، سعید صدیقیان کاشی[۳]

*نویسنده مسئول، دریافت: ۹۹/۰۵/۲۰، بازنگری: ۹۹/۰۹/۱۸، پذیرش: ۹۹/۰۹/۲۹

[۱] دانشجوی دکتری، مهندسی کامپیوتر- هوش مصنوعی، دانشگاه صنعتی خواجه‌نصیرالدین طوسی، تهران، ایران
[۲] استاد، دانشکده مهندسی برق، دانشگاه صنعتی خواجه‌نصیرالدین طوسی، تهران، ایران
[۳] استادیار، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی خواجه‌نصیرالدین طوسی، تهران، ایران

چکیده

مهندسی ویژگی با استفاده از شبکه‌های عصبی ژرف پیشرفت چشم‌گیری داشته است. یکی از انواع شبکه‌های ژرف، شبکه عصبی خودرمزگذار پشته‌ای است. از این شبکه با قابلیت یادگیری بدون سرپرست برای استخراج ویژگی و تشخیص ناهنجاری استفاده می‌شود. در این مقاله نیز با تکیه بر این دو کاربرد مهم، از شبکه‌های خودرمزگذار پشته‌ای به‌منظور تشخیص حملات به عنوان راهکاری در دیواره آتش برنامه‌های کاربردی وب استفاده می‌نماییم. در این راهکار دسته‌بندهای تک کلاسه سعی در شناسایی درخواست‌های مخرب HTTP دارد. ویژگی‌هایی که با استفاده از مدل bigram مبتنی بر کاراکتر ساخته شده‌اند، زیاد و بسیاری از آن‌ها نامرتبط می‌باشند. ازاین‌رو با استفاده از شبکه‌های خودرمزگذار پشته‌ای ویژگی‌های مرتبط را استخراج می‌کنیم. همچنین پایداری آموزش شبکه‌های عصبی ژرف یکی از چالش‌های آموزش می‌باشد. به همین خاطر برای پایداری شبکه‌های خودرمزگذار، یک نرخ آموزش پایدار توسعه می‌دهیم. با استفاده از دو مجموعه دادگان CSIC-2010 و ECML/PKDD-2007 نتایج شبیه‌سازی شبکه خودرمزگذار و تشخیص حملات را مشاهده خواهیم کرد. همان‌گونه که خواهیم دید، با چنین رویکردی علاوه بر اینکه شبکه خودرمزگذار ناپایدار نمی‌شود، نتایج قابل قبولی هم در شناسایی حملات خواهد داشت.

کلمات کلیدی: شبکه‌های عصبی خودرمزگذار، یادگیری ژرف، بهینه‌سازی، پایداری شبکه‌های عصبی، تشخیص ناهنجاری، دیواره آتش برنامه کاربردی وب

۱- مقدمه

امروزه قابلیت‌های محاسباتی، یادگیری و نمایش دانش شبکه‌های عصبی مصنوعی بر کسی پوشیده نیست. اهمیت این شبکه‌ها با ظهور یادگیری ژرف رنگ بیشتری را به خود اختصاص داده است به‌گونه‌ای که شبکه‌های عصبی و به‌خصوص شبکه‌های عصبی ژرف به عنوان یکی از ابزارهای بسیار مهم یادگیری ماشین در دنیای واقعی مورد تحلیل قرار می‌گیرند. یکی از این شبکه‌ها، شبکه‌های عصبی خودرمزگذار هستند که با استفاده از یادگیری بدون سرپرست در فرایندهای استخراج ویژگی، کاهش ویژگی، بازیابی اطلاعات، فشرده‌سازی، پردازش تصویر و تشخیص ناهنجاری[۱] به‌وفور مورد استفاده قرار می‌گیرند [۱]. همچنین با یک رویکرد پشته‌ای قابلیت یادگیری ژرف را ممکن می‌سازند و بدین ترتیب ویژگی‌های موجود در سطوح بالاتر از ترکیب ویژگی‌های موجود در سطوح پایین‌تر ساخته می‌شوند.

مشخصه‌های کلی هر شبکه عصبی شامل ساختار و معماری شبکه، توابع فعال‌ساز نرون[۲]، تابع هزینه و الگوریتم بهینه‌سازی است که با این مشخصه‌ها می‌توان هر شبکه‌ای را از دیگری تفکیک کرد. کارایی شبکه عصبی بر پایه پس انتشار خطا به طور مشخص به استراتژی بهینه‌سازی بستگی دارد. برای این منظور از روش کاهش شیب[۳] مبتنی بر قانون مشتق زنجیره‌ای استفاده می‌شود. اهمیت چنین موضوعی را در شبکه‌های عصبی به‌خصوص یادگیری ژرف و دادگان[۴] با مقیاس بزرگ به خوبی می‌توانیم مشاهده کنیم.

انتخاب مقدار برای نرخ یادگیری سالیان زیادی است که چالش شبکه‌های عصبی محسوب می‌شود. نرخ همگرایی، نرخ یادگیری تطبیقی[۵]، پدیده زیگزاگ، نرخ یادگیری پایدار مسائل شبکه‌های عصبی است که با بهینه‌سازی در شبکه‌های عصبی ارتباط دارند. نرخ یادگیری اندازه گام‌های آموزشی را مشخص می‌کند و می‌تواند بر روی پایداری و همگرایی فرایند یادگیری تأثیر بگذارد. به طور خلاصه، انتخاب نرخ یادگیری در شبکه‌های عصبی بر پایه پس انتشار خطا، حساس و تأثیرگذار در فرایند یادگیری است. این موضوع در یادگیری ژرف که ابعاد دادگان به مراتب بیشتر است، اهمیت بیشتری خواهد داشت. بنجیو نقل می‌کند که: "مهم‌ترین فراپارامتر برای بیشتر الگوریتم‌های ژرف نرخ یادگیری است" [۲]. به‌علاوه، اگر داده‌ها همچون دادگان متن، خلوت باشند، این موضوع به طور خاص اهمیت خیلی بیشتری خواهند داشت. ما در مقاله [۱] نرخ یادگیری پایداری را برای شبکه‌های عصبی خودرمزگذار و در ادامه شبکه‌های خودرمزگذار پشته‌ای[۶] معرفی کردیم.

همان‌گونه که اشاره کردیم، دو کاربرد اساسی شبکه‌های خودرمزگذار استخراج و کاهش بعد ویژگی و درنهایت دسته‌بندی و نیز تشخیص ناهنجاری است. بنابراین در این مقاله سعی داریم از شبکه‌های عصبی خودرمزگذار پشته‌ای برای استخراج ویژگی‌ها با رویکرد بدون سرپرست و نیز برای تشخیص ناهنجاری در دیواره‌های آتش برنامه‌های کاربردی وب استفاده نماییم. یکی از چالش‌های داده‌های HTTP که پروتکل تبادل اطلاعات در برنامه‌های کاربردی وب هستند، استخراج ویژگی از این مجموعه داده‌هاست، از آنجایی که این دادگان شبه متن هستند از روش *n*-gram برای ساخت ویژگی متناظر با درخواست‌های HTTP استفاده می‌نماییم. بنابراین به دلیل حجم زیاد پردازشی و نیز خلوت بودن ماتریس دادگان، اهمیت نرخ پایداری در یادگیری شبکه خودرمزگذار برای تشخیص حملات وب بر روی دادگان HTTP مشاهده می‌نماییم.

به‌طورکلی نوآوری‌های مقاله به شرح ذیل می‌باشد:

- استفاده از روش‌های دسته‌بندی تک کلاسه به‌منظور تشخیص ناهنجاری و نیز روش *n*-gram برای ایجاد بردار ویژگی
- به‌کارگیری شبکه عصبی خودرمزگذار پشته‌ای به‌منظور کاهش فضای ویژگی و نیز به عنوان یک دسته‌بند تک کلاسه
- تعمیم نرخ آموزش پایدار در شبکه‌های خودرمزگذار برای بهینه‌سازی برخط تطبیقی مانند روش‌های آدام و RPSProp
- مقایسه رویکردهای تک کلاسه به همراه مقایسه انواع نرخ‌های آموزش بر روی دادگان HTTP

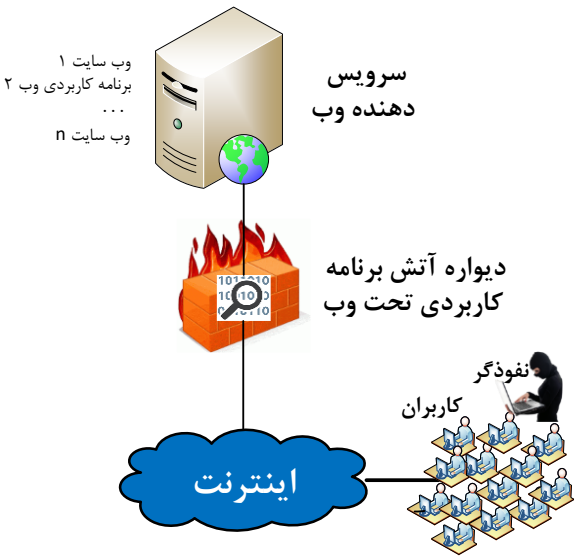
در ادامه، به تبیین دیواره آتش برنامه کاربردی وب در بخش دوم می‌پردازیم و سپس در بخش سوم مروری خواهیم داشت بر کارهای گذشته که با استفاده از رویکردهای یادگیری ژرف برای تشخیص حملات برنامه‌های کاربردی وب انجام شده است. و همچنین شرحی از روش‌های بهینه‌سازی مبتنی بر کاهش شیب به‌خصوص بهینه‌سازی برخط تطبیقی که مناسب روش‌های یادگیری ژرف هستند، خواهیم داشت. در ادامه بخش مرور کارهای گذشته به کار پیشین خود یعنی محاسبه نرخ آموزش پایدار برای شبکه عصبی خودرمزگذار اشاره می‌کنیم. در بخش چهارم نرخ آموزش پایدار تطبیقی برای شبکه‌های عصبی خودرمزگذار را معرفی می‌کنیم. همچنین ساختار ویژگی و نیز دسته‌بندهای تک کلاسه مدل دیواره آتش برنامه کاربردی وب را توضیح خواهیم داد. پس از آن نیز نتایج پیاده‌سازی مدل‌های مختلف تشخیص حملات وب با رویکرد شبکه‌های خودرمزگذار را در بخش پنجم به همراه مقایسه مدل‌ها ترسیم می‌کنیم. در آخر نیز، خلاصه و نتیجه روش‌های نرخ آموزش پایدار برای شبکه‌های خودرمزگذار را در بخش ششم بیان می‌کنیم.

۲- دیواره آتش برنامه کاربردی وب

تشخیص و ارزیابی آسیب‌پذیری، فرایندی است که حفره‌های امنیتی در شبکه‌های کامپیوتری، زیرساخت‌های برنامه‌های کاربردی و ارتباطات را معین می‌سازد. با رشد سریع اینترنت خدمات بسیاری ارائه شده که بخش وسیعی از زندگی روزانه ما را شامل شده است. به موازات رشد روزافزون اینترنت و شبکه‌های کامپیوتری، تهدیدات امنیتی نیز دوچندان شده‌اند که این امر باعث افزایش اهمیت به‌کارگیری ابزارهای امنیتی و هشدار دهنده شده است. اهمیت این موضوع هم به از بین رفتن حریم خصوصی و ارزش‌های مالی و هم به شکستن مسئولیت‌های مدنی و کیفری برمی‌گردد.

اقدامات فراوانی همچون دیواره آتش و سیستم تشخیص نفوذ در بعد پژوهشی و تجاری برای برقراری امنیت در شبکه‌های کامپیوتری به کار گرفته شده است [۳]. دیواره آتش وسیله‌ای است که کنترل دسترسی به یک شبکه را بنا بر سیاست امنیتی شبکه تعریف می‌کند. دیواره آتش به عنوان یک لایه امنیتی داده‌ها و ارتباطات را پالایش می‌کند. استفاده از دیواره‌های آتش از اقدامات اولیه برای ارتقای امنیت شبکه‌های کامپیوتری محسوب می‌شود. اما دیواره‌های آتش معمولی به دلیل ساختار خود معمولاً تا لایه‌های دوم و سوم مدل OSI را مورد محافظت قرار می‌دهند و کنترل دسترسی در این لایه‌ها را به عهده دارند. به‌محض باز شدن لاجرم بعضی دسترسی‌ها، محتوای بسته‌ها می‌تواند محل سوءاستفاده‌ی کاربران بدخواه قرار گیرد. تشخیص نفوذ به پروسه‌ی پایش[۷] رخدادها در یک سیستم یا شبکه‌ی کامپیوتری و تحلیل آن‌ها برای یافتن نشانه‌هایی از یک نفوذ اشاره دارد و سیستم تشخیص نفوذ سیستمی است که این فرآیند را اجرا می‌کند. معمولاً جلوگیری از نفوذ به فرآیندهایی نظیر یافتن نقاط ضعف شبکه‌ها و کاربردها و افزودن مکانیزم‌های امنیتی دلالت دارد [۳].

با آغاز web 2.0، اطلاعات از طریق شبکه‌های اجتماعی و مشاغل تجاری از طریق وب رونق گرفت. بنابراین وب‌سایت‌ها اغلب مورد حمله مستقیم قرار گرفته‌اند. درنتیجه صنایع توجه بیشتری را صرف امنیت برنامه‌های کاربردی وب علاوه بر امنیت تحت شبکه‌های کامپیوتری و سیستم‌عامل‌ها انجام داده‌اند. با توجه به رشد روزافزون حملات تحت وب و عدم کارایی سیستم‌های تشخیص نفوذ در جلوگیری از این حملات، نسل جدیدتری از محصولات در عرصه امنیت اطلاعات و ارتباطات با عنوان «دیواره آتش برنامه‌های کاربردی تحت وب (WAF[۸])» به‌منظور مقابله با این حملات توسعه یافته است. به این ترتیب سیستم امنیتی مطابق شکل ۱ خواهد بود.



شکل ۱: شمای یک سیستم امنیتی مجهز به دیواره آتش برنامه کاربردی وب

برای اینکه رایانه‌ها با یکدیگر در ارتباط باشند، روش‌های استانداردی از تبادل اطلاعات و پردازش توصیه می‌شود. این روش‌ها با نام پروتکل شناخته شده‌اند. صفحات وب بر روی ساختار مشهور HTTP انتقال می‌یابند. یک دیواره آتش برنامه کاربردی وب، درخواست‌های وب را محافظت می‌کند به‌خصوص سرویس‌دهنده‌هایی که پروتکل HTTP یا HTTPS[۹] ارائه می‌دهند.

به‌طورکلی روش‌های تشخیص نفوذ اعم از IDS و WAF به سه دسته مبتنی بر امضا[۱۰]، مبتنی بر مشخصه[۱۱] و تشخیص ناهنجاری تقسیم می‌شوند [۳].

روش مبتنی بر امضاء (تشخیص سوءاستفاده) به دنبال یافتن الگوهای از قبل شناخته شده در محتوای بسته‌ها است. این روش معمولاً ساده و در مورد حملات شناخته شده مؤثر است اما معایبی دارد. روش سیستم مبتنی بر امضا با حجم بزرگی از امضاها اشغال خواهد شد. بنابراین محافظت از این سیستم‌ها با انتشار روزانه آسیب‌پذیری‌های وب دشوار است. هم‌چنین آسیب‌پذیری‌ها ممکن است به‌وسیله برنامه‌های مبتنی بر وب خاصی تولید شوند. رویکرد مبتنی بر مشخصه که آن را روش تحلیل حالت‌های پروتکل نیز می‌نامند، IDS حالت‌های استاندارد پروتکل‌های مورد حفاظتش را به‌دقت می‌داند و تخطی از این حالات، به عنوان نفوذ تشخیص داده می‌شود. هم‌چنین ناهنجاری‌ها معمولاً به انحراف از یک رفتار شناخته شده مورد قبول گفته می‌شود. اما روش تشخیص ناهنجاری رفتار عادی کاربران را یاد می‌گیرد و با مشاهده الگوها نفوذها را که از رفتار طبیعی منحرف شدند شناسایی می‌کند. این رویکرد حمله روز نخست (حمله‌ای که هنوز به‌صورت عمومی ساخته نشده است) را تشخیص می‌دهد. و به دلیل شناسایی حملات روز نخست استفاده از رویکرد تشخیص ناهنجاری امری ضروری تلقی می‌شود.

۳- مرور ادبیات

در این بخش نیاز است هم مروری بر دیواره‌های آتش برنامه کاربردی وب و هم مروری بر الگوریتم‌های بهینه‌سازی شبکه‌های عصبی داشته باشیم. به‌منظور ایجاز و مروری مفید تنها دیواره‌های آتش مبتنی بر تشخیص ناهنجاری که از روش یادگیری ژرف استفاده نمودند مورد بحث قرار می‌دهیم و همچنین الگوریتم‌های بهینه‌سازی که امروزه بیشتر برای یادگیری ژرف به کار رفته است بررسی می‌کنیم و بعد از آن روش پیشنهادی که در مقاله [۱] برای پایداری شبکه‌های خودرمزگذار پشته‌ای به کار بردیم شرح می‌دهیم.

۳-۱- دیواره آتش برنامه کاربردی وب با استفاده از رویکرد یادگیری ژرف

رویکردهای مختلفی از یادگیری ژرف اعم از شبکه‌های عصبی بازگشتی[۱۲]، شبکه خودرمزگذار پشته‌ای، شبکه‌های باور ژرف[۱۳] و شبکه‌های عصبی پیچشی برای دیواره‌های آتش برنامه‌های کاربردی وب طراحی شده‌اند. در مقاله [۴] از شبکه‌های عصبی پیچشی برای تشخیص حملات وب استفاده شده است. آن‌ها از رویکرد یادگیری باناظر برای تشخیص حمله استفاده نموده‌اند به‌طوری‌که داده آموزشی را به دو دسته طبیعی و نامتعارف برچسب‌گذاری نموده‌اند. بنابراین از داده‌های غیرمتعارف یا همان حمله نیز در داده آموزشی استفاده نموده‌اند.

بسیاری از پژوهشگران از رویکردهای شبکه‌های خودرمزگذار در پیاده‌سازی دیواره آتش بهره برده‌اند. ما نیز از این روش برای کاهش ویژگی‌های bigramها استفاده نمودیم [۵, ۶]. ما علاوه بر روش خودرمزگذار پشته‌ای از شبکه باور ژرف برای استخراج ویژگی‌های bigram استفاده کردیم [۶]. پس از آن ۳ دسته‌بند تک کلاسه ماشین بردار پشتیبان تک کلاسه، جنگل مجزا و نیز پوشش بیضی‌گون برای تشخیص حمله به‌کاربرده شده است و روش‌های مختلف با یکدیگر مقایسه شده‌اند. همچنین استفاده از رویکرد شبکه‌های خودرمزگذار می‌توان به مقاله‌های [۷, ۸] اشاره نمود. به‌علاوه، مقاله [۹] از روش خودرمزگذار پشته‌ای با رویکرد حذف اغتشاش برای تشخیص حمله استفاده کرده است.

روش‌های شبکه‌های عصبی ژرف بازگشتی شامل LSTM[۱۴] و GRU[۱۵] برای چنین دیواره آتشی طراحی شده‌اند [۱۰]. چنین رویکردهایی برای داده‌های متوالی مانند درخواست‌های HTTP و ساختارهای سری زمانی مناسب می‌باشند. ورودی شبکه‌ها در این مقاله URLهای درخواست GET می‌باشد. پس از آن‌که URLها با استراتژی خاصی نشانه‌گذاری شدند، دو رویکرد شبکه بازگشتی، الگوهای متعارف را یاد می‌گیرند. همچنین یک شبکه عصبی مبتنی بر شبکه‌های پیچشی و LSTM به نام C-LSTM برای ترافیک‌های وب طراحی شده‌اند [۱۱]. همچنین از روش LSTM خودرمزگذار (AE-LSTM) پشته‌ای نیز به‌منظور استخراج ویژگی از ویژگی‌های HTTP طراحی کرده‌ایم [۱۲]. ساخت ویژگی به دو صورت bigram و one-hot دودویی در نظر گرفته شده است. در روش به کار رفته هم قابلیت کاهش ویژگی با استفاده از خودرمزگذار پشته‌ای و هم رویکردی بازگشتی با استفاده از شبکه‌های LSTM استفاده شده است. پس از آن نیز از دسته‌بند جنگل مجزا برای شناسایی حملات استفاده نمودیم.

۳-۲- بهینه‌سازی در شبکه‌های عصبی

کاهش شیب رویکردی برای کمینه کردن تابع هزینه $J(\theta)$ به‌وسیله به‌روزرسانی پارامترهای θ در جهت مخالف گرادیان تابع هزینه همانند (۱) است. به‌روزرسانی پارامترهای شبکه یا همان وزن‌های شبکه، قانون یادگیری دلتا نام دارد که نخستین بار در سال ۱۹۶۲ معرفی شد [۱۳].

$$\theta(k+1) = \theta(k) - \eta(k)\nabla_\theta J\big(\theta(k)\big) \quad (۱)$$

در مقاله‌ای که به‌منظور شبکه خودرمزگذار پایدار معرفی کردیم [۱]، کارهای گذشته از قبیل روش‌های اکتشافی، تطبیقی و تطبیقی برخط همراه با رویکردهای کاهش شیب طبیعی و پایدار به خوبی مرور شده است.

با توسعه الگوریتم‌های یادگیری ژرف و داده‌های با مقیاس بالا به‌خصوص داده‌های خلوت، به‌منظور کنترل مراحل گرادیان الگوریتم بهینه‌سازی کاهش شیب تصادفی و برخط، دسته جدیدی از روش‌های زیر گرادیان معرفی شده‌اند که نخستین تلاش در سال ۲۰۱۱ معروف به روش آداگراد[۱۶] ارائه شد [۱۴]. روش کار بدین صورت است که نرخ آموزش مطابق با پارامترها، برای به‌روزرسانی‌های غیرمکرر مقدار بزرگ‌تر و برای به‌روزرسانی‌های مکرر،مقدار کوچک‌تری را اختصاص می‌دهد. آداگراد نرخ یادگیری در هر تکراری را بر اساس گرادیان‌های قبلی تغییر می‌دهد. . رابطه این به‌روزرسانی را در (۲) مشاهده می‌کنیم.

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{G(k)+\varepsilon}}\nabla_\theta J\big(\theta(k)\big) \quad (۲)$$

همان‌گونه که مشاهده می‌کنیم در رابطه (۲) نرخ آموزشی بر عبارت $\sqrt{G(k)+\varepsilon}$ تقسیم شده است. $G(k)$ یک ماتریس قطری است که هر عضو آن مجموع مربعات گرادیان‌های قبلی تا مرحله kام می‌باشد و ε پارامتر هموارساز برای فرار از تقسیم بر صفر است.

روش آداگراد یک مشکل حاد دارد. مشکل اصلی این روش همین انباشته شدن مربعات گرادیان‌ها در مخرج است. به همین خاطر، موجب خیلی کوچک شدن نرخ یادگیری و به همراه آن ناپدید شدن اهمیت به‌روزرسانی پارامتر آموزشی خواهد شد. برای غلبه بر چنین مشکلی رویکردهای دیگری مطرح شدند. آدادلتا[۱۷] [۱۵] و RMSprop [۱۶] مشکل آداگراد را با محدود ساختن پنجره‌ای از انباشته گرادیان‌های گذشته با یک اندازه مشخص در (۳) حل کردند.

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{E[(\nabla_\theta J)^2](k)+\varepsilon}}\nabla_\theta J\big(\theta(k)\big) \quad (۳)$$

$E[(\nabla_\theta J)^2]$ میانگین کاهنده تمام گرادیان‌های گذشته است و به طریق (۴) قابل محاسبه است.

$$E[(\nabla_\theta J)^2](k) = \alpha E[(\nabla_\theta J)^2](k-1) + (1-\alpha)(\nabla_\theta J)^2(k) \quad (۴)$$

روش دیگری به نام آدام[۱۸] [۱۷] میانگین و واریانس‌های غیرمتمرکز گرادیان‌های آداگراد را به کار می‌برد ((۵) و (۶)).

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{\hat{v}(k)+\varepsilon}}\hat{m}(k), \quad (۵)$$

$$\begin{cases} \hat{m}(k) = \dfrac{m(k)}{1+\beta_1^k}\ , \qquad \hat{v}(k) = \dfrac{v(k)}{1+\beta_2^k}, \\ m(k) = \beta_1 m(k-1) + (1-\beta_1)\left(\nabla_\theta J(\theta(k))\right) \\ v(k) = \beta_2 v(k-1) + (1-\beta_2)\left(\nabla_\theta J(\theta(k))\right)^2 \end{cases} \quad (۶)$$

رویکردهای دیگری مانند نادام [۱۷] مطرح شدند که نسخه‌های پیچیده‌تر و خاصی از روش‌های آدام و RMPSProp می‌باشند. به عنوان مثال، AMSGrad همگرایی را تضمین می‌کند درحالی‌که در عمل مزایای آدام و RMSprop را هم دربر دارد [۱۸]. این روش از نرخ یادگیر کوچک‌تری نسبت به آدام خواهد داشت و در عین حال تأثیر گرادیان‌های گذشته بر روی یادگیری کند می‌شود. مراحل تغییرات روش آدام دیگر دیده نمی‌شود ((۷)).

$$\theta(k+1) = \theta(k) - \frac{\eta}{\sqrt{\hat{v}(k)+\varepsilon}}m(k), \quad \hat{v}(k) = \max\{\hat{v}(k-1), v(k)\} \quad (۷)$$

روش‌های دیگری به جز این رویکردها ارائه شده‌اند که اهمیت پژوهش در این حوزه را نشان می‌دهد. به نظر می‌رسد برای درک این روش‌ها همین مقدار مرور کافی باشد.

۳-۳- کاهش شیب تطبیقی پایدار خودرمزگذار

با تمام قابلیت‌های پسندیده روش‌های تعمیم و برخط پس انتشار خطا هنوز هم انتخاب پارامتر نرخ یادگیری معضل بسیاری از طراحان شبکه‌های عصبی است. ما به‌منظور ایجاد یک شبکه خودرمزگذار پشته‌ای پایدار یک مقدار پویا برای نرخ یادگیری در شبکه‌های عصبی خودرمزگذار پیشنهاد دادیم تا شبکه پایدار باشد. کد کننده و رمزگشای شبکه خودرمزگذار باید تمام متصل باشد. یعنی اینکه از شبکه‌های بازگشتی و پیچشی استفاده نمی‌شود. دیگر این‌که تابع هزینه‌ای که به کار می‌رود تابع هزینه مربعات خطاست. همچنین یک مقدار اکتشافی[۱۹] ثابتی برای نوع خاصی از شبکه‌های عصبی خودرمزگذار پشته‌ای تعریف کردیم.

اجزای هر لایه از شبکه خودرمزگذار در لیست زیر آورده شده است:

- x بردار ورودی شبکه عصبی (با بعد n)
- r خروجی یا همان نمایش بازتولید (n)
- e خطای هر خروجی
- h خروجی لایه نهان (میانی) یا همان لایه کد کننده (m)
- f تابع فعال‌ساز نرون لایه کدکننده
- g تابع فعال‌ساز نرون لایه رمزگشا (بازتولید)
- W^h ماتریس وزن لایه کدکننده ($m\times n$)
- W^o ماتریس وزن لایه بازتولید ($n\times m$)

با استفاده از توابع فعال‌ساز غیرخطی مانند تابع سیگموئید در لایه کد کننده و تابع درجه دوم برای کاهش تابع هزینه هم می‌توان زیرفضای آموزش را با در نظر گرفتن عبارت $W^h = (W^o)^t$ تغییر داد [۱۹]. این سناریو امری مرسوم در شبکه‌های عصبی خودرمزگذار است و باعث می‌شود حجم محاسبات به شدت کاهش یابد.

همان‌گونه که در مقاله [۱] ثابت کرده‌ایم، با استفاده از سری تیلور و نیز رابطه کاهش شیب نسبت به وزن‌های لایه بازتولید که در (۸) نشان داده شده:

$$\frac{\partial J}{\partial W^o}(k) = -\boldsymbol{e}(k)\boldsymbol{g}'(k)\boldsymbol{h}(k) \quad (۸)$$

نرخ یادگیری بردار نرون jام در لایه باز تولید برابر مقدار (۹) خواهد شد.

$$\eta_j^o(k) = \frac{1}{\left(g_j'(k)\right)^2\|\boldsymbol{h}(k)\|^2 + c^o(k)} \quad (۹)$$

$\|\boldsymbol{h}(k)\|$ نرم اقلیدسی نرون‌های میانی و ترم‌های مرتبه بالاتر سری تیلور با c^o نشان داده شده است. به‌منظور محاسبات کم‌تر می‌توانیم از ترم مرتبه بالاتر صرف نظر کنیم.

علاوه بر این مرسوم است که از توابع فعال‌ساز مشخصی مانند توابع سیگموئیدی، دوقطبی هایپربولیک که در بازه [-1,1] و یا [0,1] محدود هستند استفاده شود. در این صورت چنانچه بخواهیم بیشینه خروجی هر نرون در لایه نهان را مد نظر قرار بدهیم آنگاه برای هر نرون لایه نهان داریم: $|h_i| \le \mathbf{1}$. حال از آنجایی که تعداد نرون‌های لایه نهان برابر m است، درنتیجه $\|\boldsymbol{h}(k)\| \le \sqrt{m}$. و از طرفی، تابع هزینه در به‌روزرسانی متغیرها با دسته‌های کوچک با اندازه دسته‌های bsize برابر مجموع توابع هزینه همه نمونه‌های دسته است.

سرانجام می‌توانیم نرخ یادگیری پایدار برای الگوریتم کاهش شیب هر شبکه خودرمزگذار با مفروضات معینی تغییر دهیم. این مفروضات شامل تابع هزینه درجه دوم، توابع فعال‌سازی محدود در نرون‌های لایه کد کننده و توابع فعال‌ساز با مشتقات کوچک‌تر از یک در لایه باز تولید می‌باشد ((۱۰)).

$$\eta_{adapt-GD}(k) = \frac{1}{m \times bsize} \quad (۱۰)$$

۴- تعمیم کاهش شیب تطبیقی پایدار خودرمزگذار برای دسته‌های آموزشی کوچک و یادگیری برخط

روش پیشنهادی را برای بهینه‌سازی‌های تطبیقی برخط تعمیم خواهیم داد. با توجه به رابطه بهینه‌سازی در الگوریتم آداگراد (۲) باید اندازه‌ای برای ماتریس قطری مخرج در نظر گرفت. بنابراین، نرخ یادگیری تطبیقی پایدار برخط معادل (۱۱) خواهد شد.

$$\eta_{Adagrad}(k) = \frac{magnitude\left\{\sqrt{G(k)}\right\}}{\left(g'(k)\right)^2\|\boldsymbol{h}(k)\|^2 \times bsize} \quad (۱۱)$$

برای بزرگی[۲۰] ماتریس انواع گوناگونی از قبیل دترمینان یا نرم‌های مختلف می‌توان تعریف کرد. در این پژوهش ما از نرم فروبنیوس[۲۱] برای اندازه ماتریس استفاده نمودیم. نرم فروبنیوس مجذور مجموع مربع همه اعضا را محاسبه می‌نماید. از طرف دیگر از آنجایی که ماتریس $G(k)$ قطری است درنتیجه مجذور ماتریس $G(k)$ که همه درایه‌های آن مثبت هستند، با مثبت و منفی مجذور هر درایه ماتریس یعنی $\pm\sqrt{G_{p,p}(k)}$ برابر است. پیش از این در تعریف ماتریس $G(k)$ گفته شد که هر درایه آن مجموع مربعات گرادیان‌ها تا تکرار kام است. پس داریم: $G_{p,p}(k) = \sum(\nabla_{\theta_p} J)^2(k)$.

به طور خلاصه از آنچه گفته شد، داریم:

$$magnitude\left\{\sqrt{G(k)}\right\} = \left\|\pm\sqrt{G(k)}\right\|_F = \quad (۱۲)$$

$$\left\|\pm\sqrt{G_{p,p}(k)}\right\|_F = \sqrt{\sum_p G_{p,p}(k)} = \sqrt{\sum_p \sum_k \left(\nabla_{\theta_p} J\right)^2 (k)}$$

در مقاله [۱] بنا بر فرضیه محدب بودن تابع هزینه، یک رهیافت اکتشافی و ثابت در هر لایه خودرمزگذار در نظر گرفتیم و آن مقدار برای شبکه‌هایی که توابع محدود در بازه [-1,1] برای توابع فعال‌ساز لایه نهان در نظر گرفته شود برابر است با:

$$\eta_{adapt-layer}(k) = \frac{1}{\sqrt{m} \times bsize} \quad (۱۳)$$

از راهکارهای مناسب برای $G_{p,p}(k)$ که مقدار کم‌تری هم دارد می‌توان به آدام و RMSProp اشاره کرد. در این روش‌ها به ترتیب $\hat{v}(k) = \frac{v(k)}{1+\beta_2^k}$ و $E[(\nabla_\theta J)^2]$ قابل جایگزینی با $G_{p,p}(k)$ می‌باشد. هر کدام از این دو مقدار ترکیب خطی از دو مقدار مثبت دیگر می‌باشد. بنابراین به ازای هرکدام از روابط داریم:

در روش آدام؛

$$\begin{aligned}\sqrt{\sum_p G_{p,p}(k)} &= \sqrt{\sum_p \frac{v(k)}{1+\beta_2^k}} \\ &= \sqrt{\sum_p \frac{\beta_2 v(k-1) + (1-\beta_2)\left(\nabla_\theta J(\theta(k))\right)^2}{1+\beta_2^k}} \\ &\le \sqrt{\sum_p \frac{(1-\beta_2)\left(\nabla_\theta J(\theta(k))\right)^2}{1+\beta_2^k}} \\ &= \sqrt{\frac{(1-\beta_2)}{1+\beta_2^k}} \times \sqrt{\sum_p \left(\nabla_\theta J(\theta(k))\right)^2}\end{aligned} \quad (۱۴)$$

و در روش RMSProp؛

$$\begin{aligned}\sqrt{\sum_p G_{p,p}(k)} &= \sqrt{\sum_p E[(\nabla_\theta J)^2](k)} \\ &= \sqrt{\sum_p \alpha E[(\nabla_\theta J)^2](k-1) + (1-\alpha)(\nabla_\theta J)^2(k)} \\ &\le \sqrt{\sum_p (1-\alpha)(\nabla_\theta J)^2(k)} \\ &= \sqrt{(1-\alpha)} \times \sqrt{\sum_p \left(\nabla_\theta J(\theta(k))\right)^2}\end{aligned} \quad (۱۵)$$

باز با توجه به رابطه قبل و رهیافتی که در مقاله [۱] داشتیم، می‌توانیم از رابطه (۱۶) استفاده نماییم.

$$\sqrt{\sum_p \left(\nabla_\theta J(\theta(k))\right)^2} = \sqrt{\sum_{i,j} e_j^2(k) g_j'^2(k) h_i^2(k)} = \|\boldsymbol{e}(k) \cdot \boldsymbol{g}'(k)\| \|\boldsymbol{h}(k)\| \quad (۱۶)$$

درنتیجه نرخ یادگیری پایدار برای روابط آدام و RMSProp با استفاده از معادلات (۱۶) و به ترتیب (۱۴) و (۱۵) برابر خواهد شد با:

$$\eta_{st-Adam}(k) = \sqrt{\frac{1-\beta_2}{1+\beta_2^k}} \times \frac{\|\boldsymbol{e}(k) \cdot \boldsymbol{g}'(k)\|}{\left(g'(k)\right)^2 \|\boldsymbol{h}(k)\| \times bsize} \quad (۱۷)$$

$$\eta_{st-RMSProp}(k) = \sqrt{1-\alpha} \times \frac{\|\boldsymbol{e}(k) \cdot \boldsymbol{g}'(k)\|}{\left(g'(k)\right)^2 \|\boldsymbol{h}(k)\| \times bsize} \quad (۱۸)$$

به عنوان یک نتیجه همانند قبل اینکه اگر توابع محدود در بازه [-1,1] برای توابع فعال‌ساز لایه نهان در نظر گرفته شود و نیز توابع فعال‌ساز خروجی خطی باشد، نرخ‌های تطبیقی (۱۹) و (۲۰) را خواهیم داشت:

$$\eta_{Adam}(k) = \sqrt{\frac{1-\beta_2}{1+\beta_2^k}} \times \frac{\|\boldsymbol{e}(k)\|}{\sqrt{m} \times bsize} \quad (۱۹)$$

$$\eta_{RMSProp}(k) = \sqrt{1-\alpha} \times \frac{\|\boldsymbol{e}(k)\|}{\sqrt{m} \times bsize} \quad (۲۰)$$

این نشان می‌دهد که یک نرخ تطبیقی برخط پایدار برای هر لایه شبکه عصبی خودرمزگذار قابل تعریف است.

۵- تشخیص حملات برنامه‌های کاربردی وب

به‌منظور تشخیص حملات در برنامه‌های کاربردی وب از رویکرد مبتنی بر تشخیص ناهنجاری استفاده می‌کنیم. به دلیل نامتوازن بودن کلاس‌های متعارف و ناهنجار و نیز به دلیل آنکه بتوانیم مشاهدات ناهنجار را شناسایی کنیم از دسته‌بند تک کلاسه استفاده می‌شود. دسته‌بندهای تک کلاسه گوناگونی برای شناسایی و معین ساختن مرز کلاس متعارف ارائه شده‌اند. در ادامه چندین مدل از دسته‌بندهای مرسوم که از آن‌ها در کار خود استفاده نموده‌ایم را مرور می‌کنیم. پیش از آن روش ساخت ویژگی بر اساس مدل n-gram را شرح می‌دهیم. همچنین همان‌گونه که گفته شد از شبکه عصبی خودرمزگذار پشته‌ای برای کاهش بردار ویژگی استفاده شده است.

۵-۱- مدل ساخت ویژگی

برای ساخت ویژگی از مدل n-gram استفاده می‌کنیم. در چنین سناریویی نشانه‌ها در کنار یکدیگر برای مثال جفت نشانه در نظر گرفته می‌شود تا ترتیب حفظ شود [۲۰]. پارامتر n به تعداد نشانه‌هایی که در n-gram به کار رفته است برمی‌گردد تا اطلاعات آماری داده را دریافت کند و لیستی از توالی‌ها را تولید کند. مدل n-gram دودویی (n=2) یا همان bigram احتمال رخداد نشانه‌ای را نسبت به تمام نشانه‌های قبلی با استفاده از یک احتمال شرطی تقریب می‌زند. بنابراین با پیش بینی یک احتمال شرطی از نشانه بعدی، احتمال یک نشانه مشروط بر نشانه قبلی بر اساس فرضیه مارکوف در مدل bigram قابل استفاده است [۲۰].

دو چارچوب مطرح در مدل‌های n-gram یکی مبتنی بر کاراکتر و دیگری مبتنی بر کلمه است. طبق تجربه و ارزیابی بر اساس تابع هزینه نشان داده شده است که اطلاعات مبتنی بر کاراکتر جداسازی بهتری بین هرزنامه‌ها و زبان‌های قانونی داشته است [۲۱]. چنین فرایندی شبیه به کار ماست. مهم‌ترین خاصیت n-gram مبتنی بر کاراکتر این است که مستقل از زبان و نشانه‌هاست و این می‌تواند دلیل دوم ما در انتخاب مدل بر اساس کاراکتر باشد. همان‌گونه که می‌دانیم تعداد کل کاراکترها به مراتب کم‌تر از تعداد کلمات است. بنابراین به‌منظور فرار از تبدیل شدن به مسئله ناخوشایند این نیز دلیل دیگری می‌تواند در انتخاب مدل مبتنی بر کاراکتر باشد.

به طور خلاصه، ویژگی ساخته شده بر اساس مدل n-gram بدین گونه تعریف کرده‌ایم: ویژگی jاُم از بردار ویژگی iاُم یعنی x^{ij} با بسامد رخداد مورد (نشانه) jاُم در درخواست iاُم برابر است.

۵-۲- مدل‌های دسته‌بند تک کلاسه

همان‌گونه که پیش از این بیان شد از شبکه‌های عصبی خودرمزگذار برای تشخیص ناهنجاری می‌توان به مفهوم استخراج ویژگی به‌منظور کاهش ویژگی استفاده نمود و پس از آن دسته‌بندهای تک کلاسه برای شناسایی به کار برد.

همچنین می‌توان از این شبکه‌ها به طور مستقیم به عنوان مدل شناسایی ناهنجاری‌ها بهره‌مند شویم. بنابراین مدل‌های مفروض با روش‌های زیر مبتنی بر SAE پیاده‌سازی شده‌اند:

خطای بازتولید خودرمزگذار (AE[۲۲]): در این روش خطای شبکه خودرمزگذار محاسبه می‌شود [۷]. امتیاز ناهنجاری که همان خطای بازتولید همه لایه‌های پشته‌ای است برابر مجموع همه خطاهای بازتولید می‌باشد. حد آستانه برای این روش حساس است و مقاله [۷] حد آستانه‌ای مبتنی بر متوسط و انحراف معیار خطای بازتولید محاسبه کرده است. ما نیز حد آستانه متوسط به‌اضافه ۳ برابر انحراف معیار خطاهای بازتولید را در نظر گرفتیم.

تخمین چگالی کرنل (KDE[۲۳]): در این روش چگالی هر لایه SAE تخمین زده می‌شود [۲۲]. در این مقاله کرنل گوسی با مقدار $\sqrt{\frac{hidden-size}{2}}$ برای KDE استفاده شده است. این روش نیز همانند روش SAE-RE به مقدار آستانه حساس است. حد آستانه‌ای که برای این روش به کار بردیم برابر ۰٫۷ می‌باشد.

ماشین بردار پشتیبان تک کلاسه (OCSVM[۲۴]): در لایه انتهایی شبکه خودرمزگذار پشته‌ای از OC-SVM برای تخمین بردارهای پشتیبان توزیع ویژگی‌های لایه آخر استفاده کرده‌ایم [۶]. این روش ملزم انتخاب یک کرنل، یک پارامتر مقیاس برای تعریف مرزها و پارامتر حاشیه ϑ است چنانچه این پارامتر نسبت بیشینه نقاط خارج از محدوده در داده‌های آموزشی را مشخص می‌کند.

جنگل مجزای گروهی (EIF[۲۵]): جنگل‌های مجزا ناهنجاری‌ها را بنا بر تابع تصمیمی که بر روی ویژگی‌های آخرین لایه SAE استخراج شده است، شناسایی می‌کنند [۵, ۶].

پوشش بیضوی (Elliptic Envelope): روش پوشش بیضی‌گون نیز توزیع گوسی روی داده‌های آخرین لایه SAE را تخمین می‌زنند [۶].

۶- ارزیابی و پیاده‌سازی

پیاده‌سازی‌ها به‌منظور ارزیابی رویکرد نرخ یادگیری پایدار شبکه عصبی خودرمزگذار برای تشخیص ناهنجاری برنامه کاربردی وب و مقایسه با رویکرد نرخ یادگیری ثابت در این بخش نشان داده می‌شود.

۶-۱- دادگان

دو مجموعه داده عمومی به نام CSIC 2010 [۲۳] و ECML/PKDD 2007 [۲۴] در دسترس است. این دادگان تنها شامل داده‌های HTTP هستند و از طرفی قادریم تا برای تشخیص ناهنجاری تنها از مجموعه داده متعارف برای آموزش مدل‌ها بهره ببریم.

داده‌های CSIC شامل هزاران درخواست وب است که خودکار توسط انستیتوی امنیت اطلاعات اسپانیا فراهم شده است. این دادگان در ۳ فایل جداگانه شامل ۳۶۰۰۰ داده‌های متعارف برای آموزش، ۳۶۰۰۰ داده‌های ناهنجار (حملات) برای آزمایش و ۲۵۰۶۵ داده‌های ناهنجار برای آزمون تهیه شده‌اند. حملات مختلفی در این دادگان اعم از تزریق SQL، XSS، سرریز بافر، گردآوری اطلاعات، افشای فایل، تزریق CRLF، سمت سرور، دست‌کاری پارامتر و غیره است.

داده‌های ECML از بیش از ۵۰ هزار نمونه درخواست توسط کشور لهستان تشکیل شده است به‌طوری‌که ۳۰ درصد از این درخواست‌ها حمله می‌باشند. تمام این درخواست‌ها در یک فایل csv. قابل دسترسی است. مطابق دیگر مقالات، ۲۰ هزار درخواست به‌منظور داده‌های متعارف برای آموزش و بیش از ۱۵ هزار درخواست برای داده‌های متعارف و حملات جداگانه در نظر گرفته شده است. این داده‌ها در کنگره مشترک ECML[۲۶] و PKDD[۲۷] در سال ۲۰۰۷ تولید شد. نوع حملاتی که در این دادگان موجود است شامل XSS، تزریق SQL، تزریق LDAP، تزریق XPATH، پیمایش مسیر، اجرای دستورات و حملات سمت سرور می‌باشد.

۶-۲- معیارهای ارزیابی

یکی از چالش‌های اساسی رویکرد تشخیص ناهنجاری برنامه‌های کاربردی وب در کنار دقت ، صحت[۲۸] و بازیابی[۲۹] (نرخ تشخیص[۳۰]) روش آموزشی، مثبت‌های کاذب[۳۱] که با معیار نرخ مثبت کاذب[۳۲] روش آموزشی شناخته شده است، می‌باشد (جدول ۱). بنابراین تعمیم روش یادگیری برای نمونه‌های آموزشی لازم و ضروری است. هدف از یک سیستم تشخیص نفوذ مدلی است که رفتار متعارف را با دقت توصیف کند [۲۵]. اگر الگوریتم بیش از اندازه تعمیم داده شود، مجموعه طبیعی بسیار بزرگ می‌شود. در این حالت، تهدیدات نزدیک به داده‌های آموزشی ممکن است هنجار شناخته شوند (منفی کاذب) و کارایی سیستم را محدود می‌سازد از طرف دیگر، سیستمی که به‌سادگی مجموعه داده آموزشی را حفظ کند حافظه زیادی برای مجموعه کامل هنجارها نیاز است. چنین سیستمی برای مجموعه هنجار ناشناخته یا بی‌نهایت غیرممکن است. در این حالت الگوریتم تعمیم داده نشده است و رخدادهای طبیعی به اشتباه ناهنجار برچسب می‌خورند (مثبت کاذب). در یک سیستم تعمیم پائین نمونه‌های طبیعی که اندکی از مجموعه آموزشی فاصله دارند از دست می‌روند.

معمولاً برای سنجش عملکرد یک سیستم تشخیص نفوذ آن را شبیه یک دسته‌بند در نظر می‌گیرند. در این صورت به کمیت‌هایی نظیر مثبت صحیح (TP)، منفی صحیح (TN)، مثبت کاذب (FP) و منفی کاذب (FN) توجه می‌کنند و معیارهایی نظیر دقت، صحت و غیره را محاسبه می‌کنند. جدول ۱، تعدادی از معیارهای ارزیابی یک سیستم تشخیص نفوذ را با ادبیات یکسان شده نشان می‌دهد.

جدول ۱: نام‌ها و روش‌های متفاوت معیارهای ارزیابی

اختصار	نام معیار	رابطه
Acc.	Accuracy	$\frac{TP+TN}{TP+FP+TN+FN}$
DR	Detection rate, Recall	$\frac{TP}{TP+FN}$
PR	Precision	$\frac{TP}{TP+FP}$
Spec.	Specificity	$\frac{TN}{FP+TN}$
F1	F-score	$\frac{2*PR*DR}{PR+DR}$

با توجه به معیارهایی که معرفی شدند به نظر می‌رسد برای ارزیابی تعمیم مسئله تشخیص لازم است هر دو معیار حساسیت[۳۳] (DR) و اختصاصی[۳۴] (Spec.) بررسی شوند. علاوه بر این منحنی مشخصه عملکرد سیستم (ROC[۳۵]) که بر اساس نرخ‌های مثبت کاذب و مثبت صحیح به ازای آستانه‌های جداسازی مختلف ترسیم می‌شود حساسیت و اختصاصی یک سیستم را برای آستانه‌های مختلف ارزیابی می‌کند و به نظر می‌رسد نمایش خوبی برای بررسی تعمیم مسئله تشخیص است. همچنین برای ارزیابی عددی ROC می‌توانیم از سطح زیر نمودار (AUC[۳۶]) استفاده کنیم.

علاوه بر موارد بالا به‌منظور سنجش سرعت پردازش و محاسبات نرخ آموزش پایدار، زمان آموزش شبکه‌های عصبی خودرمزگذار پشته‌ای را با یکدیگر مقایسه می‌کنیم.

۶-۳- کتابخانه‌ها و مشخصات سیستم پیاده‌سازی

مسئله تشخیص حملات بر روی سیستم‌عامل لینوکس با مشخصات جدول ۲ پیاده‌سازی شده است.

جدول ۲: مشخصات سیستم پیاده‌سازی مدل‌های تشخیص حملات وب

سیستم‌عامل	X86-64 GNU/ Linux
CPU[۳۷]	Intel ® Core ™ i7-7700k CPU @4.20 GHz
GPU[۳۸]	GeForce GTX 1080 Ti GPU
RAM GPU	48 GB

در ضمن الگوریتم‌ها در زبان برنامه‌نویسی Python 3.X همراه با کتابخانه‌های پایتون اجرا شده است. علاوه بر کتابخانه‌های پایه پایتون، شبکه عصبی خودرمزگذار پشته‌ای با استفاده از کتابخانه تنسورفلو[۳۹] و مدل‌های دسته‌بند تک کلاسه با ابزارهای آماده کتابخانه Scikit طراحی شده‌اند.

۶-۴- نتایج ارزیابی

در حالت کلی تعداد کدهای اسکی[۴۰] برابر ۲۵۶ است. اما تنها کافی است که ۹۶ کاراکتر برای درخواست‌های HTTP مورد نقد و بررسی قرار بگیرد [۲۶]. بنابراین بردار مدل bigram دارای ۹۶×۹۶ = ۹۲۱۶ بعد است. علاوه بر این، از آنجایی که اندازه بردار ویژگی‌های bigram بزرگ است، مطابق روش‌های زیر از شبکه عصبی خودرمزگذار پشته‌ای با تعداد نرون‌های ۴۰۰۰، ۱۰۰۰، ۴۰۰، ۱۰۰، ۴۰ و ۱۰ به ازای هر لایه برای استخراج ویژگی‌ها استفاده شده است.

همچنین به‌منظور مقایسه نرخ‌های آموزش پایدار، انواع مختلفی از نرخ‌های آموزش را برای آموزش شبکه‌های خودرمزگذار مقایسه می‌کنیم. از مقادیر ثابت 10^{-4} و 10^{-5} و راهکار اکتشافی لایه‌ای برای بهینه‌سازی مختلف کاهش شیب، آدام، RMPSProp و AMSGrad استفاده می‌کنیم و این رویکردها را با راهکار پایداری به ازای این بهینه‌سازی‌ها به جز کاهش شیب مقایسه می‌نماییم.

از آنجایی که مقادیر پیش‌فرض α و β_2 در بهینه‌سازی‌های RMSProp و آدام در تنسورفلو به ترتیب برابر با ۰,۹ و ۰,۹۹۹ می‌باشد بنابراین از ضرایب $\frac{1}{\sqrt{10}}$ و $\frac{1}{\sqrt{1000}}$ پیش از رابطه $\frac{\|e(k)\|}{\sqrt{m}\times bsize}$ به ترتیب در روابط نرخ آموزش RMPSProp و آدام استفاده می‌نماییم. همان‌گونه که مشاهده می‌کنیم β_2 کوچک‌تر از ۱ است و توان‌های بالای این مقدار پس از چند تکرار نزدیک صفر خواهد شد. بنابراین می‌توان از این مقدار در مخرج کسر صرف نظر کرد. برای بهینه‌سازی AMSGrad نیز بنا بر تعریف از نرخ آموزش آدام استفاده می‌کنیم. همچنین از آنجایی که با دسته آموزشی سروکار داریم در هر دسته آموزشی نیز، کمینه مقدار $\|e(k)\|$ به کار می‌رود. بدین صورت تنها بار محاسباتی اضافه در آموزش شبکه خودرمزگذار با نرخ آموزش پایدار محاسبه مقدار کمینه $\|e(k)\|$ در هر دسته است.

عملکرد سیستم تشخیص نفوذ بر اساس معیارهای ذکر شده در جدول ۱ با استفاده از شبکه عصبی خودرمزگذار پشته‌ای و نیز دسته‌بندهای تک کلاسه به ازای نرخ‌های آموزش مختلف و بهینه‌سازی‌های کاهش شیب، آدام، RMSProp و AMSGrad در جدول تا جدول برای هر دو مجموعه دادگان CSIC و ECML آورده شده است. چنانچه مشاهده می‌کنیم برخی از این جداول مانند مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-4} و بهینه‌سازی کاهش شیب برای دادگان CSIC، وجود ندارند. این بدان دلیل است که چنین نرخ آموزشی برای این بهینه‌ساز شبکه را ناپایدار ساخته است و به‌اصطلاح خروجی این شبکه خودرمزگذار "nan" است. همچنین در این جداول مقادیر دیگری مشاهده می‌کنیم که زمان اجرا بر حسب ثانیه است.

پارامتر SAE_Time زمان اجرای شبکه عصبی خودرمزگذار پشته‌ای به ازای دادگان، بهینه‌ساز و نرخ آموزش مفروض می‌باشد. پس از آن TT و DT به ترتیب زمان آموزش و زمان تشخیص حملات با استفاده از دسته‌بند تک کلاسه را بیان می‌کند. بدین صورت علاوه بر توانایی در شناسایی حملات توسط دسته‌بند مورد نظر، سرعت اجرا و تشخیص این دسته‌بندها نیز مورد توجه و مقایسه قرار می‌گیرند. مطابق شکل ۱ یک سیستم دیواره آتش گلوگاهی در شبکه کامپیوتری محسوب می‌شود. درنتیجه سرعت شناسایی حملات پارامتر مهمی برای چنین سیستمی می‌باشد.

همچنین نتایج منحنی ROC به همراه مقدار AUC آن‌ها به ازای هر دسته‌بند در شکل ۲ تا شکل ۱۱ ترسیم شده است. در این شکل‌ها به‌منظور مقایسه عملکرد بهینه‌سازها منحنی‌ها را منوط به دسته یندها در نظر گرفتیم. همچنین از آنجایی که نمودارهای زیادی وجود دارند بنابراین به دلیل خوانایی تنها منحنی‌هایی که نتایج AUC بهتری دارند را ترسیم کردیم.

جدول ۳: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی کاهش شیب برای دادگان CSIC

CSIC	GD, $\eta = 10^{-5}$, SAE_Time(s)=۷۰۳,۲۴						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	58.68	73.86	62.7	36.89	67.82	22.65	13
EIF	69.86	71.61	75.9	67.34	73.69	0.2	1.02
Elliptic	63.99	70.09	69.21	55.22	69.65	4.27	0.005
KDE	64.56	70.14	69.86	56.53	70	834.21	147.94
SAE-RE	67.77	77.08	70.83	54.41	73.82	2.34	16.88

جدول ۴: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی کاهش شیب برای دادگان CSIC

CSIC	GD, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۶۹۸,۹۸						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	57.19	70.18	62.12	38.55	65.91	27.6	13.01
EIF	69.6	66.95	78.33	73.39	72.19	0.2	1
Elliptic	64.26	69.93	69.59	56.11	69.76	3.66	0.005
KDE	63.7	69.94	68.93	54.73	69.43	833.67	148.37
SAE-RE	71.55	72.67	77.65	69.96	75.08	1.66	16.95

جدول ۵: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی RMSProp برای دادگان CSIC

CSIC	RMSProp, $\eta = 10^{-5}$, SAE_Time(s)=۷۵۸,۷۵						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	59.24	70.02	64.14	43.76	66.95	26.48	12.93
EIF	68.45	71.24	74.21	64.44	72.69	0.2	1
Elliptic	64.33	70.04	69.64	56.13	69.84	5.23	0.005
KDE	63.37	70	68.53	53.84	69.26	894.43	149.23
SAE-RE	67.83	77.09	70.89	54.53	73.86	1.69	16.98

جدول ۶: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی RMSProp برای دادگان CSIC

CSIC	RMSProp, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۷۵۶,۱۴						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	46.18	39.28	56.23	56.09	46.25	14	12.93
EIF	71.62	82.3	72.99	56.27	77.37	0.19	0.99
Elliptic	null	null	null	null	null	null	null
KDE	67.25	70.03	73.24	63.26	71.6	879.74	149.3
SAE-RE	34.09	33.16	42.45	35.42	37.24	1.71	16.91

جدول ۷: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار RMSProp و بهینه‌سازی RMSProp برای دادگان CSIC

CSIC	RMSProp, $\eta = \eta_{RMSProp}$, SAE_Time(s)=۸۴۴,۱۲						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	55.63	68.97	60.93	36.48	64.7	24.72	12.92
EIF	73.73	82.41	75.34	61.26	78.71	0.19	0.99
Elliptic	64.15	69.99	69.44	55.76	69.72	4.01	0.005
KDE	70.09	70	77.15	70.23	73.4	975.84	147.28
SAE-RE	39.6	38.46	48.45	41.23	42.88	1.69	16.91

جدول ۸: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-4} و بهینه‌سازی Adam برای دادگان CSIC

CSIC	Adam, $\eta = 10^{-4}$, SAE_Time(s)=۷۶۷.24						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.56	69.98	61.58	37.29	65.51	25.69	13.03
EIF	67.82	70.16	73.93	64.47	72	0.2	1
Elliptic	63.99	70.03	69.25	55.33	69.63	4.83	0.005
KDE	62.98	70.07	68.07	52.79	69.05	902.11	149.4
SAE-RE	72.02	74.75	77.09	68.09	75.9	1.9	16.88

جدول ۹: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی Adam برای دادگان CSIC

CSIC	Adam, $\eta = 10^{-5}$, SAE_Time(s)=۷58.93						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	55.19	70.04	60.33	33.86	64.83	26.33	12.98
EIF	70.83	64.85	81.9	79.42	72.39	0.2	1.02
Elliptic	63.99	70.03	69.24	55.32	69.63	3.4	0.006
KDE	64.12	69.98	69.4	55.69	69.69	893.03	148.47
SAE-RE	67.95	76.53	71.24	55.62	73.79	2.58	16.94

جدول ۱۰: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی Adam برای دادگان CSIC

CSIC	Adam, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۷۶۰,۸۲						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.91	69.84	61.93	38.34	65.65	26.18	12.99
EIF	68.67	65.33	77.96	73.48	71.09	0.2	0.98
Elliptic	64.22	70.04	69.51	55.87	69.77	4.62	0.005
KDE	63.84	69.97	69.08	55.02	69.53	895.26	148.55
SAE-RE	75.07	86.46	75.05	58.71	80.35	1.69	16.89

جدول ۱۱: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار Adam و بهینه‌سازی Adam برای دادگان CSIC

CSIC	Adam, $\eta = \eta_{Adam}$, SAE_Time(s)=۸۴۵.۰۴						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.92	69.97	61.92	38.18	65.7	26	13
EIF	67.98	72.08	73.2	62.1	72.63	0.19	0.97
Elliptic	64.37	70.04	69.68	56.22	69.86	3.07	0.005
KDE	63.31	69.98	68.48	53.74	69.22	977.01	147.59
SAE-RE	74.98	81.97	77.06	64.94	79.44	1.7	16.89

جدول ۱۲: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-4} و بهینه‌سازی AMSGrad برای دادگان CSIC

CSIC	AMSGrad, $\eta = 10^{-4}$, SAE_Time(s)=۹۶۴,۸۹						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	61.2	100	61.2	0	75.93	18.15	8.69
EIF	73.68	100	73.68	0	84.84	0.19	1.02
Elliptic	65.23	100	65.23	0	78.95	4.24	0
KDE	64.55	69.9	69.95	56.86	69.92	1088.34	149.54
SAE-RE	null	null	null	null	null	null	null

جدول ۱۳: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی AMSGrad برای دادگان CSIC

CSIC	AMSGrad, $\eta = 10^{-5}$, SAE_Time(s)=۹۶۴,۹۶						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	52.8	65.33	59	34.81	62	22.89	12.99
EIF	68.81	72.47	74.07	63.55	73.26	0.2	1.04
Elliptic	null	null	null	null	null	null	null
KDE	64.48	70.08	69.79	56.44	69.93	1099.9	149.12
SAE-RE	67.63	75.91	71.12	55.74	73.44	1.74	17.05

جدول ۱۴: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی AMSGrad برای دادگان CSIC

CSIC	AMSGrad, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۹۶۴,۳۰						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.9	69.8	61.93	38.37	65.63	26.42	13
EIF	69.37	67.89	77.38	71.49	72.32	0.19	1.02
Elliptic	64.15	69.97	69.45	55.79	69.71	3.86	0.005
KDE	64.76	70.04	70.15	57.19	70.09	1095.5	143.81
SAE-RE	67.8	77.12	70.84	54.41	73.85	1.76	16.89

جدول ۱۵: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار Adam و بهینه‌سازی AMSGrad برای دادگان CSIC

CSIC	AMSGrad, $\eta = \eta_{Adam}$, SAE_Time(s)=۱۰۴۷,۸۲						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.66	69.87	61.69	37.68	65.52	25.82	12.96
EIF	72.6	66.53	83.65	81.32	74.11	0.19	1.03
Elliptic	64.23	69.99	69.54	55.96	69.76	4.66	0.005
KDE	64.29	69.93	69.63	56.19	69.78	1178.9	143.26
SAE-RE	67.87	77.04	70.95	54.69	73.87	1.75	16.95

جدول ۱۶: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی کاهش شیب برای دادگان ECML

ECML	GD, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۱۱۵,۹۷						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	55.57	59.12	55.04	52.04	57.01	6.97	3.74
EIF	61.71	87.98	57.57	35.62	69.6	0.15	0.5
Elliptic	65.7	74.7	63.18	56.77	68.45	2.93	0.002
KDE	59.06	70.52	57.24	47.69	63.19	157.94	40.11
SAE-RE	79.14	73.78	82.5	84.46	77.9	0.93	1.04

جدول ۱۷: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی RMSProp برای دادگان ECML

ECML	RMSProp, $\eta = 10^{-5}$, SAE_Time(s)=۱۵۰,۳۹						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	50.97	69.88	50.57	32.18	58.68	7.79	3.55
EIF	55.62	92.68	53.13	18.82	67.54	0.16	0.5
Elliptic	62.28	77.75	59.26	46.93	67.26	2.29	0.002
KDE	56.19	75.2	54.36	37.3	63.1	191.8	39.19
SAE-RE	79.45	74.82	82.33	84.06	78.39	0.97	1.05

جدول ۱۸: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار RMSProp و بهینه‌سازی RMSProp برای دادگان ECML

ECML	RMSProp, $\eta = \eta_{RMSProp}$, SAE_Time(s)=۱۸۳,۴۰						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	50.72	51.73	50.54	49.73	51.13	6.66	3.61
EIF	59.14	88.54	55.65	29.95	68.35	0.15	0.5
Elliptic	60.38	70.58	58.48	50.25	63.97	2.94	0.002
KDE	59.76	67.4	58.32	52.17	62.54	225.1	40.19
SAE-RE	79.16	73.75	82.56	84.53	77.9	1.05	1.04

جدول ۱۹: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی Adam برای دادگان ECML

ECML	Adam, $\eta = 10^{-5}$, SAE_Time(s)=۱۵۷,۸۲						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	56.94	59.29	56.46	54.6	57.84	7.31	3.56
EIF	57.55	94.16	54.27	21.2	68.85	0.15	0.49
Elliptic	59.82	76.99	57.19	42.77	65.63	2.51	0.002
KDE	60.18	76.08	57.6	44.38	65.56	199.82	40
SAE-RE	79.52	74.66	82.57	84.35	78.41	1.38	1.05

جدول ۲۰: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار Adam و بهینه‌سازی Adam برای دادگان ECML

ECML	Adam, $\eta = \eta_{Adam}$, SAE_Time(s)=۲۱۳,۴۵						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	45.95	69.48	47.12	22.57	56.16	7.67	3.56
EIF	59.18	87.5	55.76	31.06	68.11	0.15	0.5
Elliptic	61.76	77.11	58.88	46.52	66.77	2.83	0.002
KDE	57.9	72.88	55.95	43.03	63.31	255.77	40.06
SAE-RE	79.34	74.18	82.58	84.46	78.15	0.97	1.04

جدول ۲۱: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ ثابت 10^{-5} و بهینه‌سازی AMSGrad برای دادگان ECML

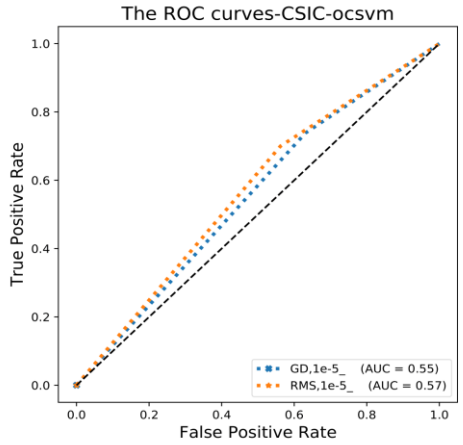
ECML	AMSGrad, $\eta = 10^{-5}$, SAE_Time(s)=۲۷۴,۶۶						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	54.17	17.78	64.55	90.31	27.88	6.91	3.55
EIF	59.44	95.42	55.4	23.71	70.1	0.15	0.49
Elliptic	63.8	76.74	60.84	50.95	67.87	2.13	0.002
KDE	62.05	75.93	59.31	48.28	66.6	315.47	39.96
SAE-RE	79.48	74.8	82.39	84.13	78.41	1	1.04

جدول ۲۲: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ تطبیقی لایه‌ای و بهینه‌سازی AMSGrad برای دادگان ECML

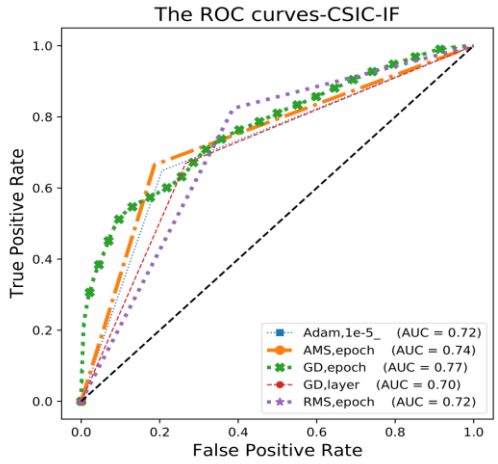
ECML	AMSGrad, $\eta = \eta_{adapt-layer}$, SAE_Time(s)=۲۷۴,۰۱						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	42.8	71.37	45.3	14.43	55.42	7.46	3.56
EIF	55.73	90.34	53.29	21.36	67.03	0.15	0.5
Elliptic	67.38	77.88	64.24	56.95	70.4	2.29	0.002
KDE	53.82	76.16	52.52	31.63	62.17	315.48	39.83
SAE-RE	79.47	74.81	82.37	84.1	78.41	1.01	1.04

جدول ۲۳: ارزیابی مدل‌های دسته‌بند تک کلاسه به همراه شبکه خودرمزگذار پشته‌ای با نرخ پایدار Adam و بهینه‌سازی AMSGrad برای دادگان ECML

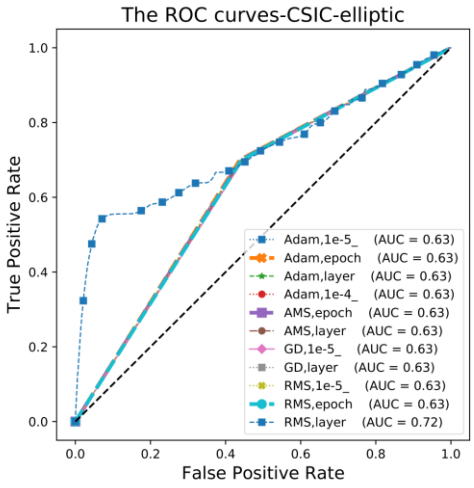
ECML	AMSGrad, $\eta = \eta_{Adam}$, SAE_Time(s)=۳۱۹,۸۷						
	Acc	DR	PR	Spec	F1	TT(s)	DT(s)
OCSVM	42.54	71.03	45.13	14.25	55.19	7.67	3.56
EIF	62.98	90.04	58.32	36.11	70.79	0.15	0.49
Elliptic	65.85	78.47	62.53	53.32	69.6	2.73	0.002
KDE	61.69	75.11	59.09	48.37	66.15	362.07	39.93
SAE-RE	79.46	74.72	82.42	84.17	78.38	1.02	1.05



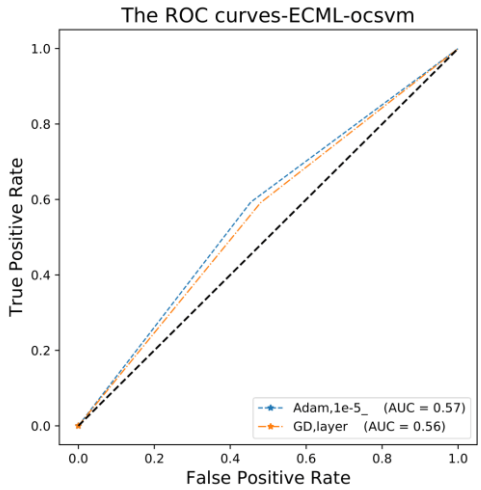
شکل ۲: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند OCSVM بر روی دادگان CSIC نتایج AUC بهتری دارند



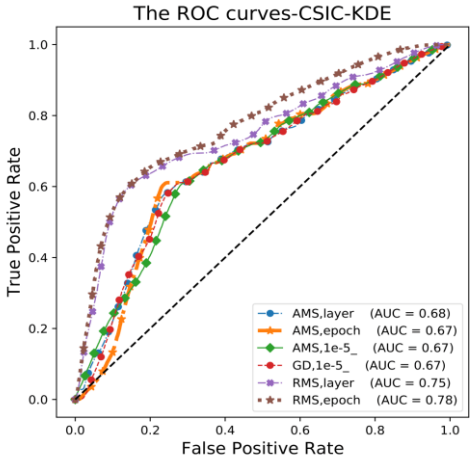
شکل ۳: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند EIF بر روی دادگان CSIC نتایج AUC بهتری دارند



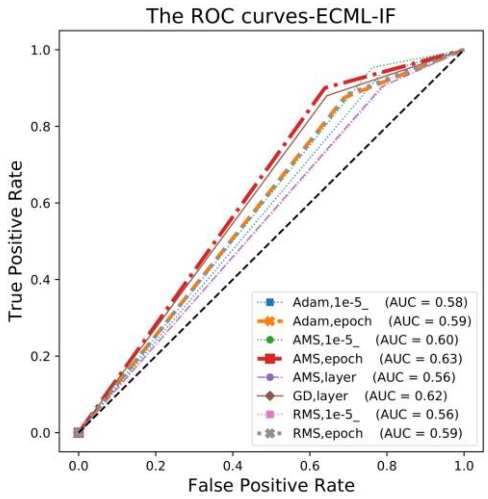
شکل ۴: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند Elliptic بر روی دادگان CSIC نتایج AUC بهتری دارند



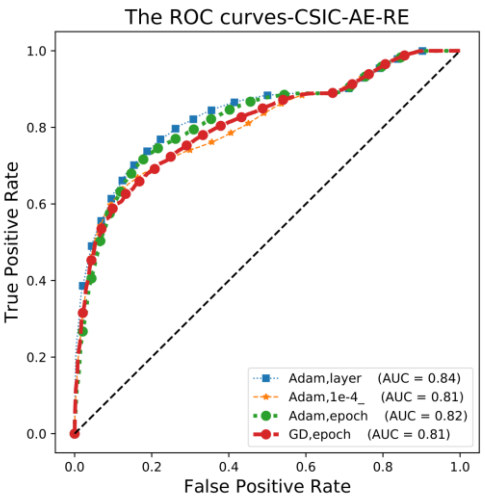
شکل ۷: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند OCSVM بر روی دادگان ECML نتایج AUC بهتری دارند



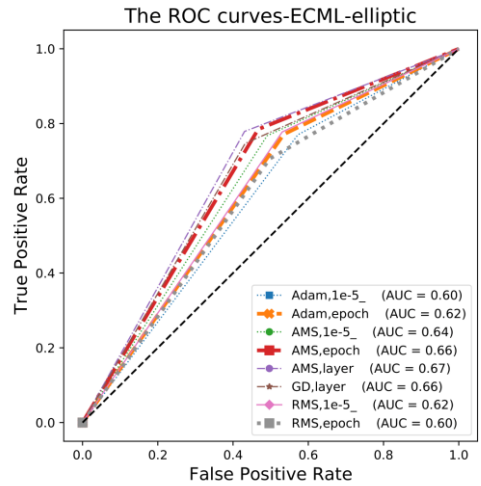
شکل ۵: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند KDE بر روی دادگان CSIC نتایج AUC بهتری دارند



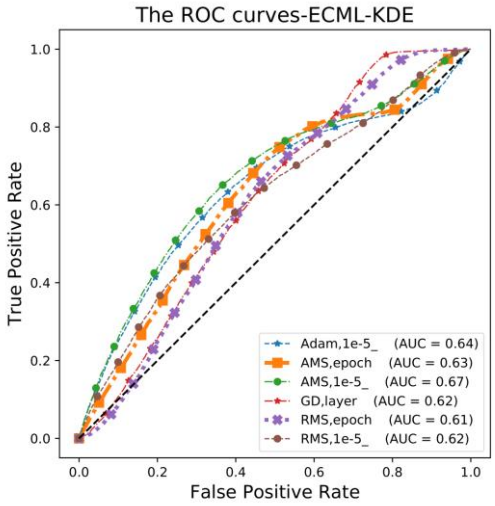
شکل ۸: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند EIF بر روی دادگان ECML نتایج AUC بهتری دارند



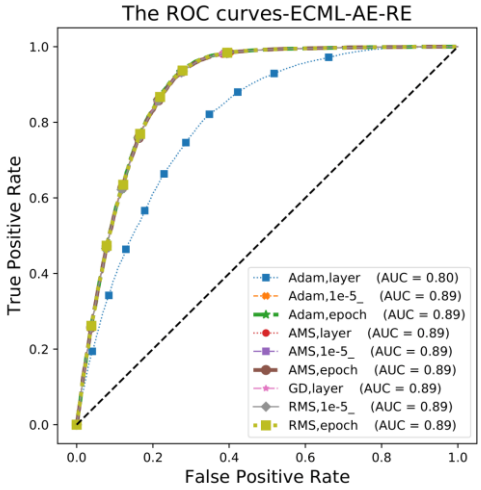
شکل ۶: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند RE بر روی دادگان CSIC نتایج AUC بهتری دارند



شکل ۹: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند Elliptic بر روی دادگان ECML نتایج AUC بهتری دارند



شکل ۱۰: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند KDE بر روی دادگان ECML نتایج AUC بهتری دارند



شکل ۱۱: منحنی ROC برای مدل‌هایی که با استفاده از دسته‌بند RE بر روی دادگان ECML نتایج AUC بهتری دارند

۶-۵- بحث و نتیجه‌گیری

در ابتدا باید ذکر کنیم که مقدار نرخ آموزش کوچک‌تر از 10^{-4} برای تمام شبکه‌های خودرمزگذار شبکه را ناپایدار می‌کرد و با توجه به عدم وجود برخی از جدول نتایج، نشان می‌دهد که مقدار 10^{-4} نیز برای بسیاری از الگوریتم‌ها برای مثال شبکه‌هایی که از بهینه‌سازی کاهش شیب استفاده نمودند و یا دادگان ECML باعث ایجاد ناپایداری کرده است و همین انتخاب مقدار درست نرخ آموزش، طراحان را سردرگم کرده است.

مقدار روش اکتشافی نرخ آموزش که در هر لایه تطبیقی است، روش بهتری برای تضمین پایداری در هر لایه شبکه عصبی است. در این صورت به ازای هر لایه شبکه عصبی مقدار مطابق همان لایه را در نظر می‌گیرد. بنابراین با هر لایه شبکه عصبی مطابق با تعداد نرون‌های لایه میانی نرخ آموزش مناسبی انتخاب می‌کند. اما از آنجایی که کار کردن با داده‌های بزرگ و خلوت حساسیت بیشتری دارد، این مقدار نیز برای برخی از شبکه‌های خودرمزگذار نیز باعث ناپایداری شده است. برای مثال چنین نرخ آموزش اکتشافی برای بهینه‌سازی‌های RMSProp و آدام بر روی دادگان ECML باعث ناپایداری شبکه عصبی خودرمزگذار پشته‌ای شده است. تأثیر میانگین متحرک[۴۱] از ماتریس G یعنی روابط (۴) و (۶) در بزرگی ماتریس و یا کوچک‌تر از یک شدن مقدار نرم بردار خطا باعث بروز یک چنین ناپایداری خواهد شد.

اما با توجه به وجود جدول نتایج بر روی نرخ‌های آموزش پایدار در هر لایه از شبکه خودرمزگذار، این نشان می‌دهد که پایداری در این شبکه‌ها تضمین شده است.

علاوه بر پایداری که هدف اصلی ما بود با توجه به نتایج جدول‌ها با معیارهای بیان شده در جدول ۱ و یا نمودارهای ROC نشان می‌دهد که نرخ آموزش پایدار نتایج قابل قبولی نیز در میزان بازنمایی فضای بردار ورودی شبکه‌های عصبی خودرمزگذار پشته‌ای داشته است. برای مثال اگر به جدول نگاه کنیم که با مقدار نرخ آموزش ثابت 10^{-5} آموزش داده شده است، دسته‌بند پوشش بیضوی با خطا مواجه شده است با آنکه نتایج عددی درستی برای بقیه دسته‌بندها ثبت شده است. این دسته‌بند به مقدار کواریانس ماتریس داده در هر تکراری از فرایند یادگیری حساس است. درنتیجه یک بازنمایی فضای ورودی بد می‌تواند این دسته‌بند را با خطا مواجه سازد. این‌گونه خطاها در همه نرخ‌های آموزش حتی نرخ آموزش اکتشافی تطبیقی هر لایه دیده می‌شود اما در نرخ آموزش پایدار با چنین خطایی برخورد نکردیم. همچنین با توجه بزرگی مقادیر دقت و حضور روش‌های پایدار در AUCهای برتر، می‌توان گفت روش‌های پایدار در رقابت دسته‌بندی‌ها نیز درخشش نسبتاً خوبی داشته‌اند.

قسمت سوم در بحث مقایسه روش‌های بهینه‌سازی و نرخ آموزش زمان بهینه‌سازی‌هاست. این به طور کامل واضح است که روش‌های پایدار به دلیل محاسبه نرم خطای نمونه‌ها در هر دسته کوچک، زمان زیادتری را برای آموزش مدل خودرمزگذار در نظر می‌گیرد. اما روش اکتشافی با مقدار ثابت به طور عملی تفاوتی نخواهد داشت. البته بد نیست گفته شود که این نتایج نشان می‌دهد که روش‌های برخط تطبیقی یعنی روش‌های مبتنی بر آداگراد نیز به دلیل محاسبات بیشتر، زمان بیشتری نسبت به کاهش شیب معمولی مصرف می‌کنند.

در آخر نیز مقایسه روش‌های دسته‌بند با یکدیگر از نظر میزان شناسایی و سرعت اجرا و تشخیص قابل توجه است. این مقادیر می‌تواند کمک شایانی برای طراحان در مسائل دیگر باشد. با توجه به نتایج به‌دست‌آمده نشان می‌دهد که روش خودرمزگذار پشته‌ای نتایج چشم‌گیرتری در شناسایی حملات داشته است. البته سرعت اجرای این روش با توجه به اینکه پیش از این خودرمزگذار پشته‌ای آموزش داده شده است تنها کافی است از روی لایه‌های خودرمزگذار، داده اولیه را بازتولید کرد و پس از آن با مقدار داده اولیه مقایسه کرد. بنابراین برای مقایسه سرعت اجرای این روش با دیگر دسته‌بندها نمی‌توان از این نتایج داوری کرد. اما سرعت دسته‌بندهای دیگر قابل مقایسه است. دسته‌بند تخمین چگالی احتمال چه در آموزش و چه در تشخیص کند و به‌اصطلاح تنبل می‌باشد. دسته‌بند SVM تک کلاسه نیز هم در اجرا و هم در تشخیص زمان‌بر است. همچنین دسته‌بند پوشش بیضوی با آنکه ممکن است در زمان آموزش زمان‌بر باشد اما سرعت بسیار بالایی در تشخیص دارد. در آخر نیز می‌توان گفت که دسته‌بند جنگل مجزا سرعت آموزش سریع و تشخیص خوبی دارد. از نظر تشخیص نیز تنها دسته‌بند SVM تک کلاسه نتایج خوبی نداشته است ولی سه دسته‌بند دیگر رفتارهای خوب ولی متفاوتی داشته‌اند.

به عنوان نتیجه باید گفت دسته‌بند مبتنی بر خودرمزگذار پشته‌ای، چه از لحاظ سرعت و چه از لحاظ دقت، دسته‌بند تک کلاسه بسیار قوی‌ای در تشخیص ناهنجاری‌ها می‌باشد.

۷- خلاصه و کارهای آتی

در این مقاله با استفاده از خواص استخراج ویژگی و تشخیص ناهنجاری در شبکه‌های عصبی خودرمزگذار پشته‌ای، حملات برنامه‌های کاربردی وب تشخیص

دادیم. برای طراحی چنین دیواره آتشی مبتنی بر تشخیص ناهنجاری، درخواست‌های HTTP نقد و بررسی کردیم. از رویکرد bigram مبتنی بر کاراکتر، ویژگی از داده‌های HTTP ساختیم ولی از آنجایی که تعداد ویژگی‌ها زیاد هستند و بسیاری از آن‌ها نامرتبط، با استفاده از شبکه‌های خودرمزگذار پشته‌ای ویژگی‌های مرتبط را استخراج کردیم و پس از آن از دسته‌بندهای تک کلاسه برای شناسایی حملات استفاده نمودیم. از آنجایی که ماتریس ویژگی‌های مدل bigram خلوت هستند، درنتیجه یادگیری شبکه عصبی با چنین دادگانی حساس خواهد بود. یکی از معضلات یادگیری انتخاب یک نرخ آموزش پایدار می‌باشد تا شبکه عصبی ژرف را مختل نسازد. به همین خاطر در این مقاله یک نرخ آموزش پایداری را توسعه دادیم. از دو مجموعه دادگان CSIC 2010 و ECML/PKDD 2007 شبکه خودرمزگذار پشته‌ای و دسته‌بندهای تک کلاسه به‌منظور تشخیص حملات به کار بردیم. نتایجی که مشاهده کردیم نشان داد که نرخ آموزش توسعه داده شده از ناپایداری شبکه خودرمزگذار پشته‌ای جلوگیری کرد. البته مقداری بر زمان آموزش شبکه ایجاد نمود. همچنین در این مقاله دسته‌بندهای تک کلاسه را نیز در توانایی جداسازی حملات از درخواست‌های متعارف همراه با سرعت اجرا و تشخیص حملات مقایسه نمودیم. آنچه که مشاهده شد گویای آن است که شبکه‌های خودرمزگذار پشته‌ای نه‌تنها در استخراج ویژگی ابزاری مناسب است که در مسائل تشخیص ناهنجاری بسیار موفق عمل می‌کند.

اقداماتی که در آینده در این حوزه می‌توان صرف کرد در چند بخش قابل تعمیم است. یکی آنکه از شبکه خودرمزگذار پشته‌ای با لایه‌های LSTM بر اساس خطای بازتولید تشخیص ناهنجاری انجام شود. شبکه LSTM برای مدل کردن داده‌های متوالی بسیار مفید می‌باشد. دیگر آنکه می‌توان از این نرخ آموزش پایدار برای مسائل دیگری که شبکه‌های عصبی خودرمزگذار به‌کاربرده‌اند، استفاده کرد. همچنین به کار بردن این نرخ آموزش برای توابع فعال‌ساز دیگر مانند ReLU که استفاده فراوانی در حوزه یادگیری ژرف دارد، می‌تواند انگیزش خوبی برای پژوهش در این حوزه باشد.

۸- مراجع

[1] A. M. Vartouni, M. Teshnehlab, and S. S. Kashi, "SAOSA: Stable Adaptive Optimization for Stacked Auto-encoders," *Neural Processing Letters,* pp. 1-26, 2020.

[2] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 17-36.

[3] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications,* vol. 36, no. 1, pp. 16-24, 2013.

[4] M. Zhang, B. Xu, S. Bai, S. Lu, and Z. Lin, "A deep learning method to detect web attacks using a specially designed cnn," in *International Conference on Neural Information Processing*, 2017, pp. 828-836: Springer.

[5] A. M. Vartouni, S. S. Kashi, and M. Teshnehlab, "An anomaly detection method to detect web attacks using Stacked Auto-Encoder," in *Fuzzy and Intelligent Systems (CFIS), 6th Iranian Joint Congress on*, pp. 131-134: IEEE, 2018.

[6] A. M. Vartouni, M. Teshnehlab, and S. S. Kashi, "Leveraging deep neural networks for anomaly-based web application firewall," *IET Information Security,* 2019.

[7] H. Mac, D. Truong, L. Nguyen, H. Nguyen, H. A. Tran, and D. Tran, "Detecting attacks on Web applications using autoencoder," in *Proceedings of the Ninth International Symposium on Information and Communication Technology*, 2018, pp. 416-421.

[8] A. Gurina and V. Eliseev, "Anomaly-based method for detecting multiple classes of network attacks," *Information,* vol. 10, no. 3, p. 84, 2019.

[9] Y. Pan, F. Sun, Z. Teng, J. White, D. C. Schmidt, J. Staples, and L. Krause, "Detecting web attacks with end-to-end deep learning," *Journal of Internet Services and Applications,* vol. 10, no. 1, pp. 1-22, 2019.

[10] J. Liang, W. Zhao, and W. Ye, "Anomaly-Based Web Attack Detection: A Deep Learning Approach," in *Proceedings of the 2017 VI International Conference on Network, Communication and Computing*, 2017, pp. 80-85: ACM.

[11] T.-Y. Kim and S.-B. Cho, "Web traffic anomaly detection using C-LSTM neural networks," *Expert Systems with Applications,* vol. 106, pp. 66-76, 2018.

[12] A. M. Vartouni, S. Mehralian, M. Teshnehlab, and S. S. Kashi, "Auto-encoder LSTM structure for anomaly-based Web Application Firewall," *International Journal of Information and Communication Technology Research,* 2020.

[13] B. Widrow and M. E. Hoff, "Associative storage and retrieval of digital information in networks of adaptive "neurons"," in *Biological Prototypes and Synthetic Systems*: Springer, 1962, pp. 160-160.

[14] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research,* vol. 12, no. Jul, pp. 2121-2159, 2011.

[15] M. Zeiler, "ADADELTA: an adaptive learning rate method. arXiv preprint arXiv: 12125701," 2012.

[16] T. Tieleman and G. Hinton, "Lecture 6.5-RMSProp, COURSERA: Neural networks for machine learning," *University of Toronto, Tech. Rep,* 2012.

[17] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980,* 2014.

[18] T. Tan, S. Yin, K. Liu, and M. Wan, "On the Convergence Speed of AMSGRAD and Beyond," in *IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, 2019, pp. 464-470: IEEE.

[19] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence,* vol. 35, no. 8, pp. 1798-1828, 2013.

[20] D. Jurafsky and J. H. Martin, *Speech and language processing*. Pearson London, 2014.

[21] I. Kanaris, K. Kanaris, I. Houvardas, and E. Stamatatos, "Words versus character n-grams for anti-spam filtering," *International Journal on Artificial Intelligence Tools,* vol. 16, no. 06, pp. 1047-1067, 2007.

[22] M. Nicolau and J. McDermott, "A hybrid autoencoder and density estimation model for anomaly detection," in *International Conference on Parallel Problem Solving from Nature*, 2016, pp. 717-726: Springer.

[23] C. Torrano-Gimnez, A. Prez-Villegas, and G. Alvarez, "The HTTP dataset CSIC 2010," ed: Instituto de Seguridad de la Información (ISI), 2010.

[24] C. Raıssi, J. Brissaud, G. Dray, P. Poncelet, M. Roche, and M. Teisseire, "Web analyzing traffic challenge: description and results," in *Proceedings of the ECML/PKDD*, 2007, pp. 47-52, 2007.

[25] K. L. Ingham, A. Somayaji, J. Burge, and S. Forrest, "Learning DFA representations of HTTP for protecting web applications," *Computer Networks,* vol. 51, no. 5, pp. 1239-1255, 2007.

[26] C. Torrano - Gimenez, H. T. Nguyen, G. Alvarez, and K. Franke, "Combining expert knowledge with automatic feature extraction for reliable web attack detection," *Security and Communication Networks,* vol. 8, no. 16, pp. 2750-2767, 2015.

علی مرادی ورتونی، دانش‌آموخته کارشناس مهندسی کامپیوتر گرایش سخت‌افزار از دانشگاه اصفهان در شهریورماه سال ۱۳۸۸ و کارشناسی ارشد علوم کامپیوتر گرایش سیستم‌های هوشمند از دانشگاه تبریز در بهمن‌ماه سال ۱۳۹۰ است. او اکنون دانشجوی دکتری دانشگاه صنعتی خواجه‌نصیرالدین طوسی در رشته مهندسی کامپیوتر گرایش هوش مصنوعی می‌باشد. زمینه پژوهشی موردعلاقه ایشان ارائه روش‌های یادگیری ژرف برای دیواره آتش برنامه کاربردی وب مبتنی بر تشخیص ناهنجاری می‌باشد. ایشان همچنین علاقه‌مند به پژوهش در راستای هوش مصنوعی، یادگیری ماشین و یادگیری ژرف و نیز تشخیص ناهنجاری است.

آدرس پست الکترونیکی ایشان عبارت است:

alimoradivartouni@ee.kntu.ac.ir

دکتر محمد تشنه‌لب، دانش‌آموخته دکتری مهندسی برق دانشگاه ساگا ژاپن در سال ۱۳۷۵ با موضوع کنترل شبکه‌های عصبی با استفاده از مدل‌های نرون انعطاف‌پذیر، دانش‌آموخته کارشناسی ارشد مهندسی برق از دانشگاه اویتا ژاپن در سال ۱۳۷۲ با موضوع بازشناسی الگوی پویا با استفاده از متغیر لحظه‌ای و شبکه‌های عصبی و دانش‌آموخته کارشناسی مهندسی برق از دانشگاه نیویورک استونی بروک امریکا در سال ۱۳۶۵ با موضوع بررسی بارگذاری محلی مصرف انرژی است. ایشان هم‌اکنون عضو هیئت‌علمی دانشکده مهندسی برق در دانشگاه صنعتی خواجه‌نصیرالدین طوسی می‌باشد. زمینه‌های پژوهشی موردعلاقه ایشان عبارت‌اند از: هوش مصنوعی شامل شبکه‌های عصبی، محاسبات فازی الگوریتم‌های تکاملی و بهینه‌سازی ذرات، شناسایی، پیش بینی و نیز بازشناسی الگو، تشخیص خطا، یادگیری ژرف، مهندسی پزشکی، محاسبات نرم بازه‌ای، شبکه‌های عصبی راف و فازی نوع اول و دوم.

آدرس پست الکترونیکی ایشان عبارت است:

teshnehlab@eetd.kntu.ac.ir

دکتر سعید صدیقیان کاشی، دانش‌آموخته دکترای تخصصی مهندسی کامپیوتر گرایش نرم‌افزار از دانشگاه علم و صنعت ایران در سال ۱۳۹۰ در موضوع هماهنگی در شبکه‌های حسگر کنشگر بی‌سیم. دانش‌آموخته کارشناسی ارشد مهندسی کامپیوتر گرایش نرم‌افزار از دانشگاه علم و صنعت ایران در سال ۱۳۸۴ با موضوع سامانه‌های سرویس گرا و مدل گرا و دانش‌آموخته کارشناسی مهندسی کامپیوتر از دانشگاه علم و صنعت ایران گرایش نرم‌افزار سال ۱۳۸۳ است. ایشان هم‌اکنون عضو هیئت‌علمی دانشکده برق و کامپیوتر در دانشگاه صنعتی خواجه‌نصیرالدین طوسی می‌باشد. زمینه‌های پژوهشی موردعلاقه ایشان عبارت‌اند از: سامانه‌های توزیع‌شده مانند رایانش ابری، اینترنت اشیا و شبکه‌های حسگر بی‌سیم و همچنین حوزه مهندسی نرم‌افزار .

آدرس پست الکترونیکی ایشان عبارت است:

sedighian@eetd.kntu.ac.ir

---

[1] Anomaly detection
[2] Neuron
[3] Gradient descent (GD)
[4] Dataset
[5] Adaptive learning rate
[6] Stack auto-encoders
[7] Monitoring
[8] Web Application Firewall
[9] Hypertext Transfer Protocol Secure
[10] Signature-based
[11] Specification-based
[12] Recurrent neural network (RNN)
[13] Deep Belief Network (DBN)
[14] Long Short-Term Memory
[15] Gated Recurrent Unit
[16] Adagrad
[17] Adadelta
[18] Adam
[19] Heuristic
[20] Magnitude
[21] Frobenius
[22] Reconstruction error
[23] Kernel density estimation (KDE)
[24] One-class support vector machine (OCSVM)
[25] Ensemble isolation forest (EIF)
[26] European Conference on Machine Learning (ECML)
[27] European Conference on Principles and Practice of Knowledge
[28] Precision
[29] Recall
[30] Detection rate
[31] False positive
[32] False Positive Rate (FPR)
[33] Sensitivity
[34] Specificity
[35] Receiver Operating Characteristic
[36] Area Under Curve
[37] واحد پردازش مرکزی
[38] واحد پردازش گرافیکی
[39] Tensorflow
[40] ASCII code
[41] Moving average

The CSI Journal on Computing Science and Information Technology

The CSI Journal on
Computing Science and
Information Technology
Vol. 18, No.1, 2020

Using stable learning rate of Auto-encoders to Anomaly detection in Web Application Firewall

Ali Moradi Vartouni[1], Mohammad Teshnehlab[2] , Saeed Sedighian Kashi[3]

[1,3] Faculty of Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran
[2] Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran

Abstract

Nowadays, use of deep neural networks for feature engineering is very popular. Stacked auto-encoder as one of deep models for neural network with the ability of unsupervised learning is widely used for feature extraction and anomaly detection. In this paper, based on these two issues, stacked auto-encoder is used to detect attacks in web applications. The firewall uses one-class classifiers to detect malicious HTTP requests. Due to the curse of dimensionality caused by character-based bigram model, related features are extracted using stacked auto-encoders. Also, one of the challenges of neural networks, especially deep learning, namely learning rate is addressed in the study. When feature matrix is sparse, the choice of learning rate is crucial. Therefore, a stable learning rate is also developed. Experiments conducted using stack auto-encoder and anomaly detection on the two datasets CSIC-2010 and ECML/PKDD-2007. Results show, in addition to stability, auto-encoders also have noticeable improvement in accuracy for attack detection in HTTP requests.

Keywords: Auto-encoder neural networks; Deep learning; Optimization; Stability of neural networks; Anomaly detection; Web application firewall.