دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده‌ی مهندسی کامپیوتر

تشخیص نفوذ شبکه‌های کامپیوتری مبتنی بر یادگیری‌ماشین

نگارش
بهار کاویانی

استاد راهنما
دکتر رضا صفابخش

اردیبهشت‌ماه ۱۴۰۰


بسم الله الرحمن الرحيم

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده‌ی مهندسی کامپیوتر

تشخیص نفوذ شبکه‌های کامپیوتری مبتنی بر یادگیری‌ماشین

نگارش
بهار کاویانی

استاد راهنما
دکتر رضا صفابخش

اردیبهشت‌ماه ۱۴۰۰

سپاس‌گزاری

اینجانب بهار کاویانی مراتب تقدیر و تشکر خود را نسبت به استاد راهنمای خود، آقای دکتر رضا صفابخش که طی تدوین این گزارش نوشتاری همواره مرا یاری نموده‌اند، ابراز می‌دارم.

بهار کاویانی

اردیبهشت‌ماه ۱۴۰۰

چکیده

امروزه محبوبیت فراگیر و استفاده‌ی روزانه از اینترنت، به دنبال خود مشکلات امنیتی فراوانی را به وجود آورده که به تنهایی یکی از مسایل پیچیده و بسیار مهم حوزه‌ی شبکه‌های کامپیوتری است. با گسترش روز افزون این دانش و تکنولوژی، حملات سایبری و نفوذ به شبکه‌های کامپیوتری نیز گسترده‌تر شده‌است. در چنین شرایطی برای ایجاد امنیت کامل، تنها راه‌کارهای مقابله با نفوذ همانند استفاده از دیواره‌های آتش[۱]، نمی‌توانند راه‌گشای ما در این مسیر باشند و نیاز به راه‌حل‌ها و الگوریتم‌هایی برای شناسایی و محدود کردن نفوذ به سیستم‌ها و شبکه‌ی کامپیوترها احساس می‌شود. در حقیقت سیستم‌های تشخیص نفوذ[۲]، تا جای ممکن رفتارهای خراب‌کارانه را پیش‌بینی و از خود در برابر این حملات محافظت می‌کنند.

الگوریتم‌های یادگیری ماشین می‌توانند در این کار، دقت بهتر و سرعت تشخیص بیش‌تری را برای ما به ارمغان آورند. از طرفی یکی دیگر از نتایج استفاده از یادگیری ماشین این است که دیگر برای تشخیص نفوذ شبکه به تجربه و دانش کارشناسان و متخصصین نیازی نخواهیم داشت. بنابراین باید از الگوریتم‌های مختلف در این زمینه شناخت کافی داشته باشیم تا بتوانیم با توجه به نیازهای سیستمی خود بهترین الگوریتم را استفاده کنیم.

این پژوهش سعی دارد تا معرفی کوتاهی از الگوریتم‌های تشخیص نفوذ مبتنی بر یادگیری ماشین به عمل آورد و هم‌چنین به کمک اطلاعات و ویژگی‌های جمع‌آوری شده از آن‌ها، این الگوریتم‌ها را از جهت‌های مختلف همانند نظارت‌شده یا نظارت‌نشده بودن، کم‌عمق یا عمیق بودن و همچنین انواع داده‌هایی که هر یک از الگوریتم‌ها با آن‌ها سر و کار دارند، با یکدیگر مقایسه کند.

واژه‌های کلیدی:

سیستم تشخیص نفوذ، شبکه‌های کامپیوتری، یادگیری ماشین، امنیت، الگوریتم

[۱] Firewalls

[۲] Intrusion Detection System (IDS)

فهرست مطالب

عنوان صفحه

فهرست اشکال و جداول

عنوان صفحه

فصل اول

مقدمه

۱-۱ مقدمه

حتما تاکنون بارها و بارها درباره‌ی افزایش حملات و رویدادهای نفوذ به اینترنت و شبکه‌های محلی شنیده‌اید. در چنین شرایطی که هر لحظه ارتباط روزانه‌ی ما با اینترنت بیش‌تر می‌شود، وجود چنین مشکلاتی می‌تواند خطرات جبران‌ناپذیری را برای سازمان‌ها یا افراد در پی داشته باشد. بنابراین وجود یک سیاست و سیستم امنیتی با هدف کاهش خطرات مربوط به محرمانه بودن اطلاعات و در دسترس بودن آن‌ها بسیار ضروری خواهد بود.

البته موضوع امنیت یک موضوع تازه نیست و سال‌هاست که سازمان‌ها راه‌حل‌های مختلفی را از جمله استفاده از دیواره‌های آتش برای صاف کردن[۱] ترافیک‌های ورودی، استفاده از احراز هویت برای کنترل کردن اطلاعات و داده‌ها، استفاده از ضد ویروس برای جلوگیری کردن از انتشار کرم[۲] و به کارگیری فناوری‌هایی چون VPN برای رمزگذاری داده‌ها و ... برای جلوگیری از نفوذ و مقابله با آن ایجاد کرده‌اند. با این همه، باز هم مشکلات امنیتی بسیاری وجود دارد که مهاجمان با دور زدن این راه‌کارهای امنیتی به سازمان‌ها و سیستم‌ها تحمیل می‌کنند.

در این شرایط سیستم‌های تشخیص نفوذ (IDS) و جلوگیری از نفوذ[۳] (IPS) می‌توانند در تشخیص تلاش‌های نفوذ به شبکه و همچنین جلوگیری از آن‌ها کمک کنند. در ادامه‌ی این فصل توضیحات بیش‌تری در مورد این سیستم‌ها داده شده‌است. در فصل‌های آینده الگوریتم‌هایی در این زمینه بر پایه‌ی یادگیری ماشین جمع‌آوری شده‌اند که به اختصار توضیح داده می‌شود. سپس با بررسی ویژگی‌های هر یک از این الگوریتم‌ها، موارد استفاده‌ی هر کدام مشخص شده‌است.

۱-۲ سیستم تشخیص نفوذ

همانطور که کمی پیش‌تر گفته شد، سامانه‌های تشخیص نفوذ وظیفه دارند تا هرگونه استفاده‌ی غیرمجاز یا خراب‌کارانه از سیستم‌ها را شناسایی کنند. این وظیفه تنها در برابر نفوذهای خارجی مطرح نیست بلکه باید آسیب‌هایی که به‌طور عمد یا غیر عمد از سمت کاربران داخلی به سیستم تحمیل می‌شود نیز تشخیص داده شوند.

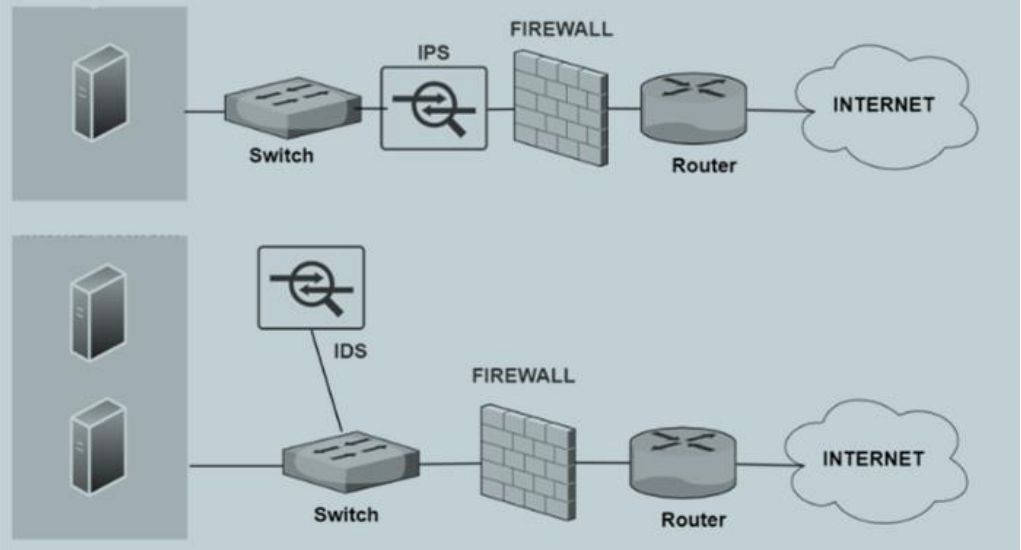
برای ایجاد وضوح بیش‌تری از نحوه‌ی کار این سیستم‌ها لازم که تفاوت آن را با سیستم پیشگیری از نفوذ (IPS) بررسی کنیم. یک سیستم تشخیص نفوذ سه وظیفه‌ی پایش، تشخیص و واکنش را انجام می‌دهد. در واقع مانند یک سیستم شنود، ترافیک شبکه را تجزیه و تحلیل می‌کند. اگر تلاشی برای نفوذ به شبکه انجام گیرد، پس از تشخیص، سیستم پیشگیری از نفوذ حملات را از بین می‌برد. بنابراین IDS مانع از انجام حملات نمی‌شود، اما به ما این امکان را می‌دهد تا هنگام وقوع آن‌ها مطلع شویم و IPS جلوی حملات و نفوذهایی که توسط IDS شناسایی شده را می‌گیرد. در شکل ۱ نیز می‌توانید تفاوت این دو سیستم را مشاهده کنید.

---

[۱] filtering

[۲] worm

[۳] Intrusion Prevention System (IPS)



شکل ۱ - تفاوت سیستم تشخیص نفوذ و سیستم پیشگیری از نفوذ

۱-۲-۱ مزایای استفاده از سیستم تشخیص نفوذ

سیستم‌های IDS با جمع‌آوری اطلاعات مفید در مورد حملات و نفوذهای رخ داده شده، امکان عیب‌یابی و شناخت آسیب‌پذیری‌ها را فراهم می‌آورند. هم‌چنین با هشدار دادن در مورد حملات کشف شده، می‌توانند سبب جلوگیری از تکرار حملات مشابه شوند و یا با استفاده از الگوهای به دست آمده، از اجرای کامل برخی حملات جلوگیری کنند.

۱-۲-۲ معایب استفاده از سیستم تشخیص نفوذ

این سیستم‌ها چون بر پایه‌ی اطلاعات و آمارهای جمع‌آوری شده می‌توانند نتیجه‌گیری کنند، ممکن است یک ترافیک خوب را به عنوان حمله قلمداد کنند و یا برعکس یک ترافیک حمله را اگر با الگوهای قبلی هم‌خوانی نداشته باشد، آن را نفوذ در نظر نگیرند. به علاوه باید میزان حساسیت این سیستم‌ها به درستی تنظیم گردد؛ زیرا اگر میزان حساسیت آن‌ها بالا باشد، می‌تواند موجب به وجود آمدن هشدارها و اختلالات زیادی شود که بسیاری از آن‌ها به علت استفاده‌های روزانه و عادی کاربران سازمان بوده است. در صورتی هم که این حساسیت خیلی پایین باشد، بسیاری از حملات تشخیص داده نخواهند شد.

۱-۲-۳ دسته‌بندی کلی رویکردهای تشخیص نفوذ

به طور کلی سیستم‌های IDS را می‌توان از نظر روش تحلیل نفوذ به دو دسته‌ی کلی تقسیم کرد. روش تشخیص رفتار غیر عادی[۱] و روش تشخیص مبتنی بر امضا[۲] [۱].

[۱] Anomaly-based

[۲] Signature-based

روش تشخیص رفتار غیرعادی سعی می‌کند که تعیین کند آیا می‌توان رفتار غیرعادی ایجاد شده را به عنوان یک نفوذ دانست یا خیر. در حالی که در روش تشخیص مبتنی بر امضا از الگوهای حملات انجام شده یا نقاط ضعف سیستم برای شناسایی نفوذ استفاده می‌شود [۲].

۱-۳ خلاصه

در این فصل توضیحات اولیه و مختصری برای آشنایی با سیستم‌های تشخیص نفوذ، به اختصار IDS، داده شد و در مورد اهمیت وجود آن‌ها نکاتی ذکر شد. در فصل‌های دوم و سوم به معرفی و بررسی الگوریتم‌های تشخیص نفوذ مبتنی بر یادگیری ماشین می‌پردازیم که بحث اصلی این گزارش است تا بتوانیم یک آشنایی اولیه با هر یک از الگوریتم‌های مطرح در این حوزه داشته باشیم و از مزایا، معایب و ویژگی‌های هرکدام اطلاعات کافی داشته باشیم.

فصل دوم

رویکردهای مبتنی بر یادگیری ماشین (الگوریتم‌های تحت نظارت)

۲-۱ رویکردهای تشخیص نفوذ مبتنی بر یادگیری ماشین

در این بخش قصد داریم به معرفی الگوریتم‌های معروف یادگیری ماشین که در زمینه‌ی تشخیص نفوذ مورد استفاده قرار می‌گیرند، بپردازیم. دو نوع اصلی از یادگیری ماشین وجود دارد: یادگیری تحت نظارت و یادگیری نظارت نشده.

یادگیری تحت نظارت به اطلاعات مفید موجود در *داده‌های دارای برچسب* متکی است. طبقه‌بندی، رایج‌ترین کار در یادگیری تحت نظارت است (و همچنین اغلب در IDS استفاده می‌شود). با این حال، برچسب‌گذاری داده‌ها به صورت دستی گران و وقت گیر است. درنتیجه، عدم وجود اطلاعات کافی دارای برچسب، مشکل اصلی یادگیری تحت نظارت است. در این فصل به بررسی الگوریتم‌های یادگیری ماشین تحت نظارت خواهیم پرداخت.

در مقابل، یادگیری بدون نظارت که در فصل بعدی انواع آن توضیح داده شده، اطلاعات ارزشمندی را از داده‌های بدون برچسب استخراج می‌کند. الگوریتم‌های رایج یادگیری ماشین مورد استفاده در IDS در شکل ۲ نشان داده شده است [۳].

۲-۲ مدل‌های کم‌عمق تحت نظارت

الگوریتم‌هایی در این بخش قصد داریم به معرفی آن‌ها بپردازیم، مدل‌های کم‌عمق تحت نظارت[۱] هستند. الگوریتم‌های تحت نظارت از نظر کم‌عمق یا عمیق بودن به دو دسته تقسیم می‌شوند. مدل‌های کم‌عمق، مدل‌هایی هستند که چندین دهه مورد مطالعه قرار گرفته‌اند و روش آن‌ها بالغ است. آن‌ها نه تنها بر روی اثر ردیابی بلکه بر روی مشکلات عملی، مانند بازده ردیابی و مدیریت داده‌ها نیز تمرکز دارند [۳].

۲-۲-۱ k نزدیک‌ترین همسایه

روش k نزدیک‌ترین همسایه[۲] (k-NN) یکی از ساده‌ترین و سنتی‌ترین تکنیک‌های غیرپارامتری برای طبقه‌بندی نمونه‌ها است. در این روش فاصله‌ی تقریبی بین نقاط مختلف بردارهای ورودی محاسبه می‌شود و سپس نقطه‌ی بدون برچسب به کلاس k-NN آن‌ها اضافه می‌شود.

در فرآیند ایجاد این طبقه بندی، k یک پارامتر مهم است و مقادیر مختلف آن باعث عملکردهای مختلف می شود. اگر k به طور قابل ملاحظه‌ای بزرگ باشد، همسایگانی که برای پیش‌بینی استفاده می‌کنند، زمان طبقه‌بندی زیادی دارند و بر دقت پیش‌بینی تأثیر می‌گذارند.

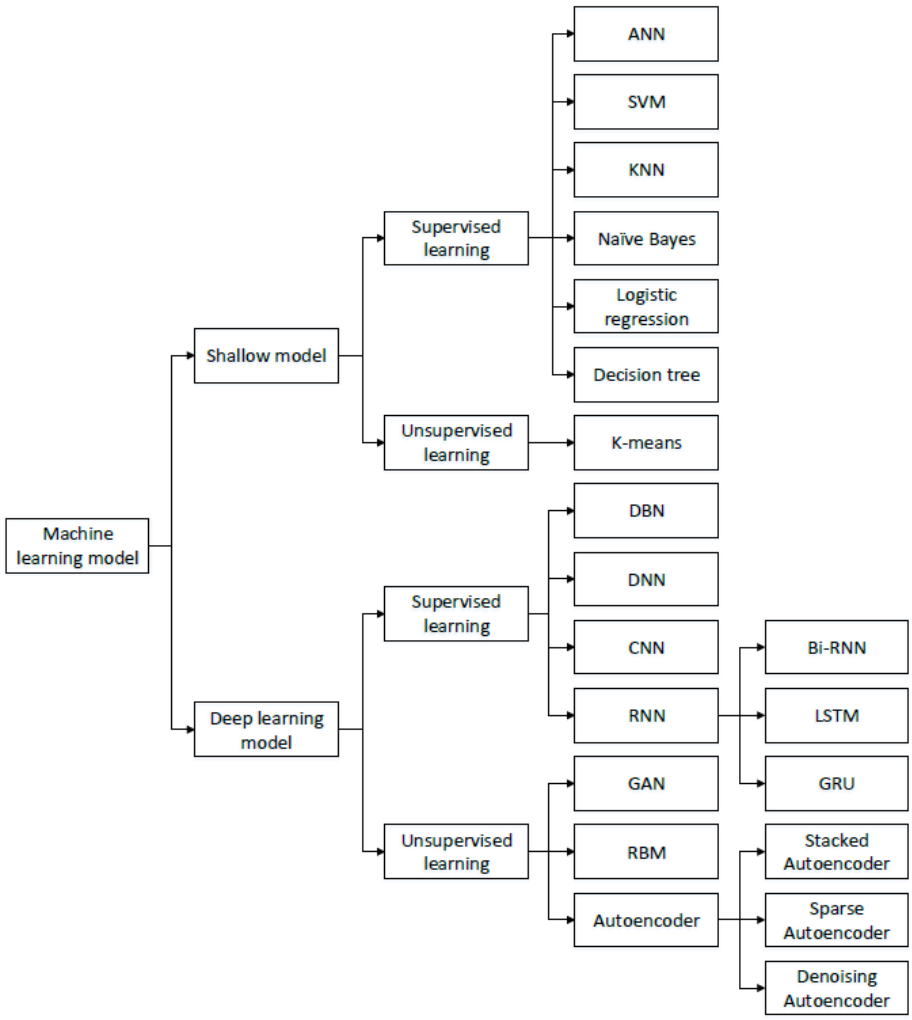
مدل k نزدیک‌ترین همسایه، یادگیری *مبتنی بر نمونه*[۳] نامیده می‌شود و با رویکرد یادگیری استقرایی متفاوت است [۲].

---

۱ Supervised Shallow Models

۲ K-Nearest Neighbor (KNN)

۳ instance based learning



شکل ۲ - طبقه‌بندی الگوریتم‌های یادگیری ماشین [۳]

۲-۲-۲ ماشین بردار پشتیبان

ماشین بردار پشتیبان[۱] توسط Vapnik در سال ۱۹۹۸ ارایه شده است. SVM ابتدا بردار ورودی را در یک فضای با بُعد بالاتر ترسیم می‌کند و سپس بخش بهینه‌ای از آن را به دست می‌آورد. علاوه بر این، یک مرز تصمیم‌گیری، مانند همان محدوده‌ای که از فضای اصلی جدا شده، به جای کل نمونه‌های آموزشی توسط بردارهای پشتیبان تعیین می‌شود و بنابراین نسبت به نقاط دور از آن محدوده بسیار قوی است.

به طور خاص، یک طبقه‌بندی SVM برای طبقه‌بندی به صورت باینری طراحی شده است. منظور از باینری این است که این روش، مجموعه‌ای از بردارهای آموزشی را که به دو کلاس مختلف تعلق دارند، جدا می‌کند. توجه داشته باشید که بردارهای پشتیبانی، نمونه‌های آموزشی نزدیک به مرز تصمیم‌گیری هستند.

[۱] Support Vector Machines (SVM)

SVM همچنین یک پارامتر مشخص شده توسط کاربر به نام ضریب مجازات[۱] را فراهم می‌کند. این پارامتر به کاربران این امکان را می‌دهد تا بین تعداد نمونه‌های طبقه بندی اشتباه و پهنای مرز تصمیم‌گیری معامله کنند [۲].

۲-۲-۳ شبکه‌های عصبی مصنوعی

شبکه‌ی عصبی مصنوعی[۲] یک واحد پردازش برای اطلاعات است که به تقلید از نورون‌های مغز انسان توسط Haykin در سال ۱۹۹۹ ابداع شده است.

پرسپترون چند لایه[۳] یکی از معماری‌های شبکه‌ی عصبی‌ست که به طور گسترده‌ای در بسیاری از مسائل تشخیص الگو استفاده می‌شود. یک شبکه‌ی MLP از یک لایه‌ی ورودی شامل مجموعه‌ای از گره‌های حسی به عنوان گره‌های ورودی، یک یا چند لایه‌ی مخفی از گره‌های محاسباتی و یک لایه‌ی خروجی از گره‌های محاسباتی تشکیل شده است. هر اتصال داخلی با یک عدد به عنوان وزن آن اتصال همراه است که در مرحله یادگیری تنظیم می‌شود.

برای آموزش MLP، معمولاً از الگوریتم یادگیری تولید متناوب استفاده می‌شود؛ به این شبکه‌ها، شبکه‌های عصبی انتشار مجدد[۴] نیز گفته می‌شود. در این شبکه‌ها، ابتدا وزن‌های تصادفی آموزش داده می‌شوند. سپس، الگوریتم وزن‌ها را تنظیم می‌کند تا برای هر چیز، یک واحد تعریف کند. این کار در به حداقل رساندن خطای طبقه بندی‌های غلط موثر است [۲].

۲-۲-۴ نقشه‌های خود سازمان‌دهی شده

نقشه خود سازمان‌دهی شده[۵] توسط الگوریتم *یادگیری رقابتی بدون نظارت*، آموزش داده می‌شود. هدف SOM کاهش بُعد تجسم داده‌ها است. به این معنی که SOM بردارهای ورودی با ابعاد بالا را بر روی یک نقشه تصویری با ابعاد کم تجسم می‌کند که معمولاً این تصویر برای سادگی دو بُعدی است.

این الگوریتم معمولاً از یک لایه ورودی و لایه کوهونن[۶] تشکیل شده که به صورت آرایش دو بعدی نورون‌ها طراحی شده است و ورودی های n بعدی را در دو بُعد ترسیم می کند. لایه کوهونن وظیفه‌ی ایجاد ارتباط بین هر یک از بردارهای ورودی با یک خروجی نماینده را دارد.

شبکه، نزدیکترین گره به هر مورد آموزشی را پیدا می‌کند و سپس گره برنده را که نزدیک‌ترین نورون (یعنی نورون با حداقل فاصله) است، به عنوان گره آموزشی انتخاب می‌کند. یعنی SOM بردارهای ورودی مشابه را روی واحدهای خروجی

---

[۱] penalty factor

[۲] Artificial Neural Networks (ANN)

[۳] Multilayer perceptron (MLP)

[۴] backpropagation neural networks

[۵] Self-Organizing Maps (SOM)

[۶] Kohonen layer

یکسان یا مشابه روی یک نقشه دو بعدی ترسیم می‌کند. بنابراین، واحدهای خروجی خود را با یک نقشه‌ی مرتب سازمان‌دهی می‌کند و هم‌چنین واحدهای خروجی با وزن مشابه نیز پس از آموزش در همان نزدیکی قرار می‌گیرند [۲].

۲-۲-۵ درخت تصمیم

درخت تصمیم[۱] یک نمونه را از طریق دنباله‌ای از تصمیمات طبقه‌بندی می‌کند که در آن تصمیم فعلی به تصمیم‌گیری بعدی کمک می‌کند. چنین توالی تصمیماتی در یک ساختار درختی نشان داده می‌شود. طبقه‌بندی یک نمونه از گره ریشه به گره(ها)ی مناسب برگ منتهی می‌شود، جایی که هر گره برگ (گره انتهایی) نشان‌دهنده‌ی یک دسته‌بندی طبقه‌بندی شده است. ویژگی‌های نمونه‌ها به هر گره اختصاص می‌یابد و مقدار هر شاخه متناسب با صفات است.

یک برنامه‌ی شناخته شده برای ساخت درختان تصمیم، *طبقه‌بندی و بازگشت درخت*[۲]است. اگر به درخت تصمیم برچسب‌های گسسته یا نمادین کلاس‌بندی را بیفزاییم، درخت طبقه‌بندی نامیده می‌شود؛ در حالی که درخت تصمیم با دامنه‌ی مقادیر پیوسته یا عددی، درخت رگرسیون[۳] نامیده می‌شود [۲].

۲-۲-۶ شبکه‌های خلیج ساده

موارد بسیاری وجود دارد که ما وابستگی‌های آماری یا روابط علت و معلولی بین متغیرهای سیستم را می‌دانیم. با این وجود، بیان دقیق روابط احتمالی میان این متغیرها ممکن است دشوار باشد. برای بهره‌برداری از این وابستگی‌های گاه به گاه بین متغیرهای تصادفی یک مسئله، می‌توان از یک مدل نمودار احتمالی به نام شبکه‌های خلیج ساده[۴] استفاده کرد. این مدل به سوالاتی مانند "با توجه به برخی از وقایع مشاهده شده در سیستم، احتمال این که نوع خاصی از حمله باشد، چیست؟" با استفاده از فرمول احتمال شرطی (فرمول ۱) پاسخ می‌دهد.

$$P(X = x \mid Y = c_k) = \prod_{i=1}^{n} P\left(X^{(i)} = x^{(i)} \mid Y = c_k\right) \qquad (1)$$

ساختار یک NB به طور معمول توسط یک گراف بدون دور جهت‌دار نشان داده می‌شود، جایی که هر گره یکی از متغیرهای سیستم را نشان می‌دهد و هر لینک تأثیر یک گره بر دیگری را مشخص می‌کند. بنابراین، اگر پیوندی از گره A به گره B وجود داشته باشد، A مستقیماً بر B تأثیر می‌گذارد [۲].

---

۱ Decision Tree

۲ Classification and Regressing Tree (CART)

۳ Regression Tree

۴ Naïve Bayes Networks (NBN)

۲-۲-۷ الگوریتم‌های ژنتیک

الگوریتم‌های ژنتیکی[۱] از کامپیوتر برای اجرای نظریه‌ی انتخاب طبیعی و تکامل استفاده می‌کنند. این الگوریتم توسط کوزا در سال ۱۹۹۲ پیشنهاد شده است.

الگوریتم با تولید تصادفی تعداد زیادی از برنامه‌های کاندید آغاز می‌شود. سپس از نوعی اندازه‌گیری تناسب اندام برای ارزیابی عملکرد هر فرد در یک جمعیت استفاده می‌شود. آن‌گاه تعداد زیادی تکرار انجام می‌شود تا برنامه‌های کم عملکرد با ترکیبات ژنتیکی برنامه‌های با عملکرد بالا جایگزین شوند. یعنی برنامه‌ای با اندازه‌گیری تناسب اندام کم حذف شده و برای تکرار بعدی کامپیوتر زنده نمی‌ماند [۲].

۲-۲-۸ منطق فازی

منطق فازی[۲](یا نظریه مجموعه‌های فازی) مبتنی بر مفهوم پدیده‌ی فازی است که اغلب در دنیای واقعی رخ می‌دهد. نظریه مجموعه‌های فازی برای استدلال کردن، برخلاف *مفهوم عضویت در مجموعه* پیش می‌رود و مقادیر بین ۰ و ۱ را به گزاره‌ها نسبت می‌دهد. یعنی در منطق فازی درجه حقیقت یک گزاره می‌تواند بین ۰ و ۱ باشد و محدود به دو مقدار نیست (یعنی فقط درست و غلط). به عنوان مثال، "باران" یک پدیده طبیعی است و ممکن است بتواند شرایط محیطی را از عادی به اوضاع بحرانی تبدیل کند [۲].

۲-۲-۹ رگرسیون لجستیک

رگرسیون لجستیک[۳]نوعی مدل خطی لگاریتم است. الگوریتم LR همانطور که در زیر نشان داده شده است احتمال کلاس‌های مختلف را از طریق توزیع لجستیک پارامتریک محاسبه می‌کند. در این فرمول نمونه‌ی x در کلاسی با حداکثر احتمال قرار می‌گیرد.

$$P(Y = k|x) = \frac{e^{w_k * x}}{1 + \sum_{k}^{K-1} e^{w_k * x}} \quad .k = 1.2 \dots K - 1 \qquad (2)$$

ساخت یک مدل LR آسان است و آموزش مدل نیز کارآمد است. با این حال، LR نمی‌تواند به خوبی با داده‌های غیرخطی برخورد کند، که کاربرد آن‌ها را محدود می‌کند [۳].

---

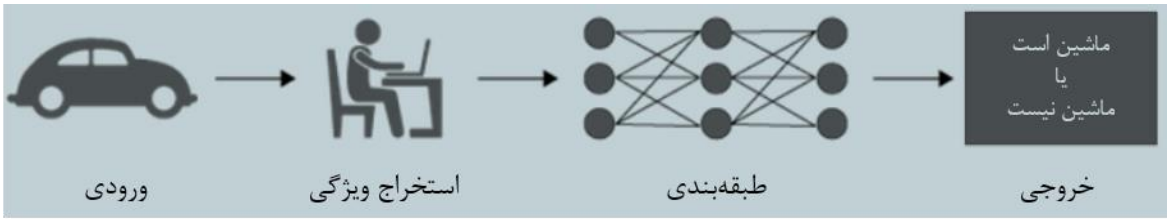
۱ Genetic Algorithms (GA)

۲ Fuzzy Logic

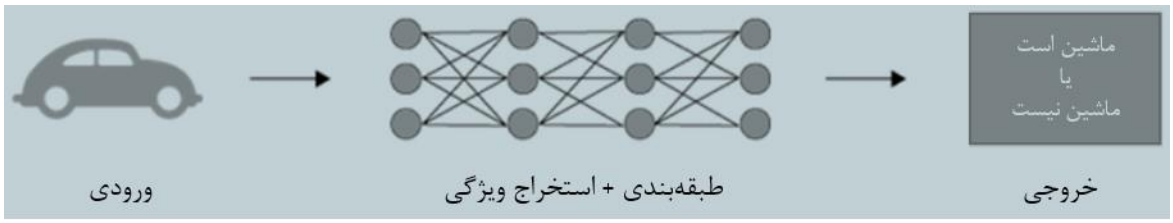
۳ Logistic Regression (LR)

۲-۳ مدل‌های عمیق تحت نظارت

حال می‌خواهیم انواع دیگری از الگوریتم‌های تشخیص نفوذ تحت نظات را بررسی کنیم که در دسته‌ی مدل‌های عمیق تحت نظارت[۱]قرار می‌گیرند. با مشاهده‌ی شکل‌های ۳ و ۴ می‌توانید تفاوت عملکرد الگوریتم‌های یادگیری ماشین عادی و الگوریتم‌های یادگیری عمیق را ببینید.



شکل ۳ – یادگیری ماشین



شکل ۴ – یادگیری عمیق

تعداد مطالعات IDS مبتنی بر یادگیری عمیق از سال ۲۰۱۵ تاکنون به سرعت افزایش یافته است. مدل‌های یادگیری عمیق بدون نیاز به مهندسی به شیوه‌ی دستی، مستقیماً بازنمایی ویژگی‌ها را از داده‌های اصلی مانند تصاویر و متون یاد می‌گیرند. بنابراین، روش‌های یادگیری عمیق می‌توانند به صورت انتها-به-انتها[۲] اجرا شوند. برای مجموعه داده‌های بزرگ، روش‌های یادگیری عمیق نسبت به مدل‌های کم عمق مزیت قابل توجهی دارند. در مطالعه‌ی یادگیری عمیق، تأکیدات اصلی بر معماری شبکه، انتخاب آَبر پارامتر و استراتژی بهینه‌سازی است.

۲-۳-۱ شبکه‌ی کوتاه عمیق

شبکه‌ی کوتاه عمیق[۳] یا DBN از چندین لایه ماشین بولتزمن محدود[۴] و یک لایه طبقه‌بندی سافت‌مکس[۵] تشکیل شده است، همانطور که در شکل ۵ نشان داده شده‌است.

---

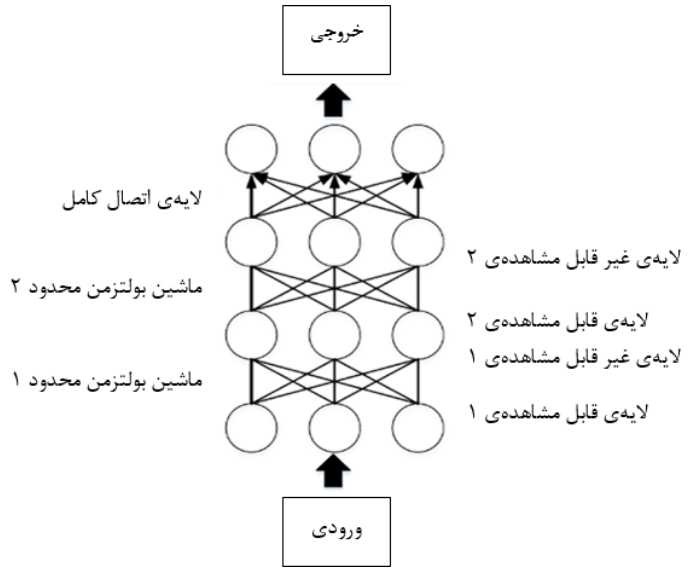
۱ Unsupervised Shallow Models

۲ End-to-end

۳ Deep Brief Network (DBN)

۴ Restricted Boltzmann Machine (RBM)؛ رجوع به بخش ۳-۳-۲

۵ Softmax

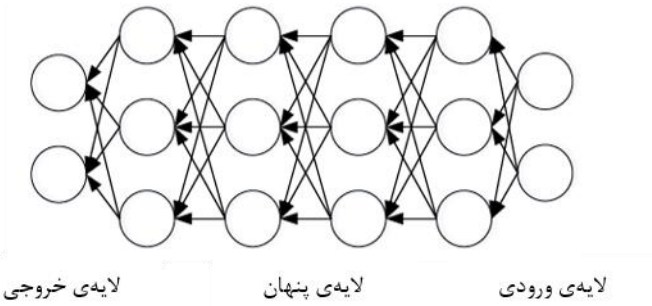


شکل ۵ - ساختار شبکه‌ی کوتاه عمیق [۳]

آموزش DBN شامل دو مرحله است: ابتدا یک مرحله آموزش بدون نظارت که در آن ماشین بولتزمن محدود به عملکردی حریصانه و لایه‌ای آموزش می‌یابد. سپس، وزن لایه‌ی سافت‌مکس با داده‌های دارای برچسب به روز رسانی می‌شود. در شناسایی حمله، DBNها هم برای استخراج ویژگی و هم برای طبقه‌بندی استفاده می‌شوند [۳].

۲-۳-۲ شبکه‌ی عصبی عمیق

همانطور که در شکل ۶ می‌بینید، ساخت یک شبکه‌ی عصبی عمیق[۱] با استفاده از استراتژی پیش‌یادگیری و با تنظیم دقیق چندین لایه امکان‌پذیر است. هنگام آموزش یک DNN، پارامترها ابتدا با استفاده از داده‌های بدون برچسب، که یک مرحله یادگیری بدون نظارت است، آموزش می‌بینند. سپس، شبکه از طریق داده‌های دارای برچسب، که یک مرحله یادگیری تحت نظارت است، تنظیم می‌شود. دستاوردهای حیرت انگیز DNNها عمدتاً به دلیل مرحله‌ی یادگیری به طور بدون نظارت است [۳].

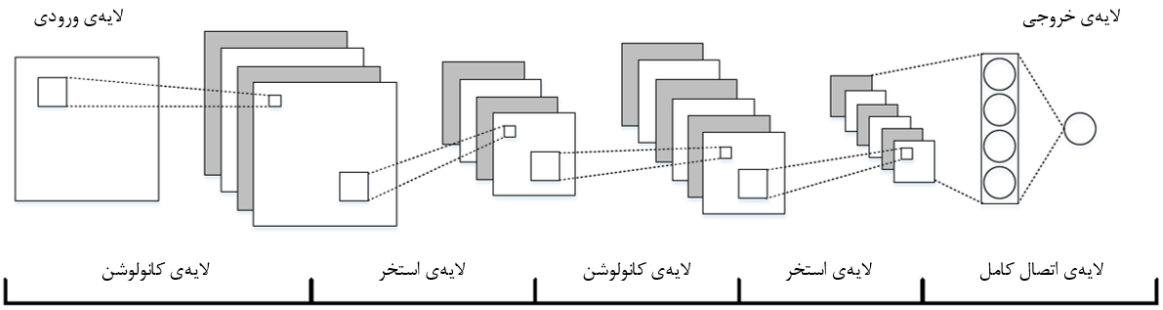


شکل ۶ - ساختار شبکه‌ی عصبی عمیق [۳]

[۱] Deep Neural Network (DNN)

۲-۳-۳ شبکه‌ی عصبی کانولوشن

شبکه‌های عصبی کانولوشن[۱] به تقلید از سیستم بینایی انسان طراحی شده‌اند. در نتیجه، در زمینه‌ی بینایی کامپیوتر دستاوردهای بزرگی داشته‌اند. همانطور که در شکل ۷ هم نشان داده شده است، یک CNN با لایه‌های کانولوشن و استخر جایگزین انباشته[۲] شده است. CNNها بر روی داده‌های دو بُعدی کار می‌کنند، بنابراین داده‌های ورودی باید برای شناسایی حمله به ماتریس ترجمه شوند [۳].

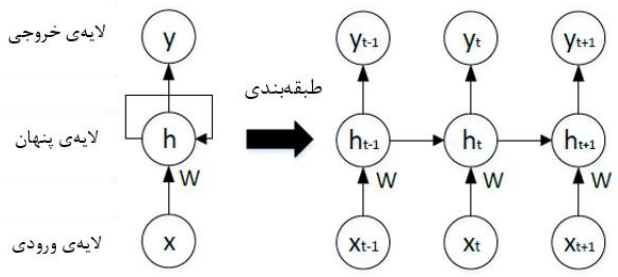


شکل ۷ - ساختار شبکه‌ی عصبی کانولوشن [۳]

۲-۳-۴ شبکه‌ی عصبی راجعه

شبکه‌های عصبی راجعه[۳] شبکه‌هایی هستند که برای داده‌های پی‌درپی و متوالی طراحی شده‌اند و به طور گسترده‌ای در پردازش زبان طبیعی[۴] استفاده می‌شوند. ویژگی‌های داده‌های متوالی زمینه‌ای است. تجزیه و تحلیل داده‌های جدا شده از توالی معنی ندارد. برای به دست آوردن اطلاعات متنی، هر واحد در RNN نه تنها وضعیت فعلی بلکه حالت‌های قبلی را نیز دریافت می‌کند [۳].

ساختار RNN در شکل ۸ نشان داده شده است. RNNهای استاندارد فقط با توالی‌هایی با طول محدود سروکار دارند.



شکل ۸ - ساختار شبکه‌ی عصبی راجعه [۳]

---

۱ Convolutional Neural Network (CNN)

۲ stacked

۳ Recurrent Neural Network (RNN)

۴ natural language processing (NLP)

۲-۴ خلاصه

تا اینجا الگوریتم‌های زیادی را در زمینه‌ی تشخیص نفوذ به کمک یادگیری ماشین بررسی کردیم که همگی در دسته‌ی الگوریتم‌های تحت نظارت بودند. در فصل بعدی الگوریتم‌های بیش‌تری معرفی می‌شوند که بر پایه‌ی عدم نظارت بنا شده‌اند و تفاوت‌هایی با الگوریتم‌های ذکر شده دارند.

فصل سوم

رویکردهای مبتنی بر یادگیری ماشین (الگوریتم‌های نظارت نشده)

۳-۱ رویکردهای تشخیص نفوذ مبتنی بر یادگیری ماشین

در فصل قبل انواعی از الگوریتم‌های یادگیری ماشین تحت نظارت که در زمینه‌ی تشخیص نفوذ مورد استفاده قرار می‌گیرند، معرفی شدند. در این فصل قصد داریم تا با الگوریتم‌های یادگیری بدون نظارت نیز آشنا شویم و نکات مربوط به آن‌ها را نیز بررسی کنیم.

همانطور که گفته شد، این الگوریتم‌ها بر خلاف الگوریتم‌های تحت نظارت اطلاعات ارزشمندی را از *داده‌های بدون برچسب* استخراج می‌کند و در نتیجه به دست آوردن داده‌های آموزشی با آن بسیار آسان‌تر خوهد بود. با این حال، عملکرد تشخیص روش‌های یادگیری بدون نظارت معمولاً در مقایسه با روش‌های یادگیری تحت نظارت پایین است [۳].

در ادامه انواع این الگوریتم‌ها را می‌خوانید.

۳-۲ مدل‌های کم‌عمق نظارت نشده

در این بخش الگوریتمی که از مدل کم‌عمق نظارت نشده[۱] بررسی خواهیم کرد، الگوریتم k-میانگین است.

مدل k-میانگین[۲]

مدل k-میانگین یک الگوریتم خوشه‌بندی معمولی است، به طوری که k تعداد خوشه‌ها و منظور از میانگین، میانگین صفات است. الگوریتم k-میانگین از "فاصله" به عنوان معیار اندازه گیری شباهت استفاده می‌کند. هرچه فاصله‌ی بین دو شی داده کمتر باشد، احتمال قرار گرفتن آن‌ها در یک خوشه بیش‌تر است.

الگوریتم k-میانگین به خوبی با "داده‌های خطی" سازگار است، اما نتایج آن در داده‌های غیر محدب ایده‌آل نیست.

علاوه بر این، الگوریتم k-میانگین به شرایط مقداردهی اولیه و پارامتر k حساس است. در نتیجه، برای تنظیم مقدار پارامتر مناسب باید آزمایش‌های مکرر زیادی انجام شود [۳].

۳-۳ مدل‌های عمیق نظارت نشده[۳]

مدل های یادگیری عمیق از شبکه‌های عمیق متنوع تشکیل شده‌است. در میان آن‌ها ، شبکه‌های کوتاه عمیق (DBNها)، شبکه‌های عصبی عمیق (DNNها)، شبکه‌های عصبی کانولوشن (CNNها) و شبکه‌های عصبی مکرر (RNNها) مدل‌های یادگیری تحت نظارت هستند که در فصل گذشته به آن‌ها پرداخته شد. در اینجا روش‌های نظارت نشده توضیح داده خواهد

[۱] Supervised Deep Learning Models

[۲] K-means

[۳] Unsupervised Deep Learning Models

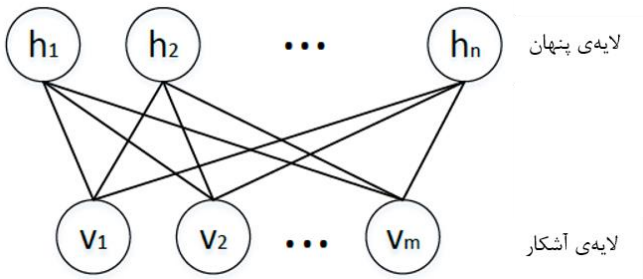
شد که مدل‌های خود رمزگذاران، ماشین‌های بولتزمن محدود (RBMها) و شبکه‌های خصمانه تولیدی (GAN) در این دسته قرار می‌گیرند.

۳-۳-۱ شبکه‌های خصمانه تولیدی

شبکه‌ی خصمانه‌ی تولیدی[۱]شامل دو شبکه‌ی فرعی است، به عنوان مثال، یک مولد و یک تفکیک‌کننده. هدف این مولد تولید داده‌های مصنوعی مشابه داده‌های واقعی است و تفکیک‌کننده قصد دارد داده‌های مصنوعی را از داده‌های واقعی تشخیص دهد. بنابراین، مولد و تفکیک‌کننده یکدیگر را بهبود می‌بخشند. GANها در حال حاضر یک موضوع داغ تحقیقاتی هستند که برای افزایش داده‌ها در شناسایی حمله مورد استفاده قرار می‌گیرند، که تا حدی مشکل کمبود مجموعه داده‌های تشخیص نفوذ را کاهش می‌دهد. در همین حال، GANها به رویکردهای یادگیری خصمانه تعلق دارند که می‌توانند با افزودن نمونه‌های خصمانه به مجموعه‌ی آموزش، دقت تشخیص مدل‌ها را افزایش دهند.

۳-۳-۲ ماشین بولتزمن محدود

ماشین بولتزمن محدود[۲] یک شبکه‌ی عصبی تصادفی است که در آن واحدها از توزیع بولتزمن پیروی می‌کنند. RBM از یک لایه‌ی قابل مشاهده و یک لایه‌ی مخفی تشکیل شده‌است. واحدهای موجود در همان لایه به هم متصل نیستند. اما، همانطور که در شکل ۹ نشان داده شده است، واحدها در لایه‌های مختلف کاملاً به هم متصل شده‌اند. RBM بین مسیرهای جلو و عقب تفاوت قائل نیستند. بنابراین، وزن در هر دو جهت یکسان است. RBMها مدل‌های یادگیری بدون نظارت هستند که توسط الگوریتم واگرایی انقباضی آموزش دیده‌اند، و آن‌ها معمولاً برای استخراج ویژگی یا خنثی‌سازی استفاده می‌شوند.



شکل ۹ - ساختار ماشین محدود بولتزمن [۳]

۳-۳-۳ خود رمزگذار

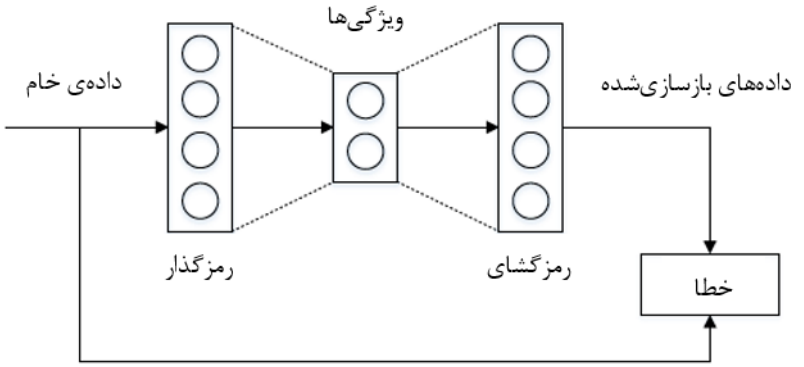
همانطور که در شکل ۱۰ نشان داده شده‌است، یک رمزگذار خودکار[۳]شامل دو جزء متقارن است، یک رمزگذار و یک رمزگشای. رمزگذار ویژگی‌هایی را از داده‌های خام استخراج می‌کند و رمزگشای داده‌ها را از ویژگی‌های استخراج شده

[۱] Generative Adversarial Network (GAN)

[۲] Restricted Boltzmann Machine (RBM)

[۳] Autoencoder

بازسازی می‌کند. در حین آموزش، به تدریج از اختلاف بین ورودی رمزگذار و خروجی رمزگشای کاسته می‌شود. وقتی رمزگشای موفق شد داده‌ها را از طریق ویژگی‌های استخراج شده بازسازی کند، به این معنی است که ویژگی‌های استخراج شده توسط رمزگذار نمایانگر ماهیت داده‌ها هستند. توجه به این نکته ضروری است که کل این فرایند به هیچ اطلاعات تحت نظارت احتیاج ندارد. بسیاری از انواع معروف رمزگذارهای خودکار وجود دارد؛ همانند خود رمزگذارهای بی‌صدا و خود رمزگذارهای پراکنده [۳].



شکل ۱۰ – ساختار مدل خود رمزگذار [۳]

۳-۴ خلاصه

در فصل سوم به بررسی برخی الگوریتم‌های نظارت نشده‌ی یادگیری ماشین پرداختیم که در زمینه‌ی تشخیص نفوذ می‌توانند کمک‌های فراوانی به سیستم بکنند. الگوریتم‌های بررسی شده همانند فصل گذشته در دو دسته‌ی کم‌عمق و عمیق جای گرفتند تا از نظر شیوه‌ی برخورد با داده‌ها، از یکدیگر متمایز شوند.

در فصل بعدی می‌خواهیم به مقایسه‌ی تمامی الگوریتم‌هایی که تا به حال در مورد آن‌ها در فصل‌های دوم و سوم صحبت کردیم بپردازیم و ضمن آشنا شدن بیش‌تر با مزایا و معایب هر یک از آن‌ها، نسبت به زمینه‌های استفاده و توان الگوریتم‌ها در مقایسه‌ی با یکدیگر مطالبی را بخوانیم.

فصل چهارم

مقایسه و بررسی الگوریتم‌ها

۴-۱ مقایسه و بررسی مزایا و معایب الگوریتم‌ها

تا اینجا الگوریتم‌های مطرح در زمینه‌ی تشخیص نفوذ معرفی شدند. در اینجا قصد داریم که این الگوریتم‌ها را با هم مقایسه کنیم. پیش از هر چیز باید در مورد مزایا و معایب هر کدام از این الگوریتم‌ها مطلع شویم تا بتوانیم موارد استفاده‌ی هر کدامشان را تا حدودی از یکدیگر تمیز دهیم. همانطور که می‌بینید در جدول ۱ این اطلاعات، به همراه برخی اقدامات در جهت بهبود الگوریتم‌ها آمده‌است.

جدول ۱ - مزایا و معایب مدل‌های مختلف

الگوریتم (بخش مربوطه)	مزایا	معایب
k نزدیک‌ترین همسایه (KNN)	درخواست برای داده‌های عظیم؛ مناسب برای داده‌های غیرخطی؛ آموزش سریع؛	دقت کم در کلاس اقلیت؛ زمان آزمون طولانی؛ حساس به متغیر K
ماشین بردار پشتیبان (SVM)	توانایی آموختن اطلاعات مفید از مجموعه قطارهای کوچک؛ قابلیت تولید قوی	نبود عملکرد خوب در داده‌های بزرگ یا چندین کار طبقه‌بندی شده؛ حساس به متغیرهای عملکرد هسته
شبکه‌های عصبی مصنوعی (ANN)	قادر به مقابله با داده‌های غیر خطی؛ توانایی اتصالات قوی	مناسب برای نصب بیش از حد؛ مستعد گیر افتادن در یک بهینه‌ی محلی؛ وقت‌گیر بودن آموزش مدل
درخت تصمیم (Decision tree)	انتخاب خودکار ویژگی‌ها؛ تعبیر قوی	متمایل بودن نتایج طبقه بندی به کلاس اکثریت؛ نادیده گرفتن همبستگی داده‌ها
شبکه‌های خلیج ساده (Naïve Bayes)	قادر به یادگیری افزایشی	نداشتن عملکرد خوب در داده‌های مرتبط به یک ویژگی
رگرسیون لجستیک (LR)	ساده است؛ می‌تواند به سرعت آموزش داده شود؛ ویژگی‌ها را به صورت خودکار مقیاس بندی می‌کند	نبود عملکرد خوب در داده‌های غیر خطی؛ مناسب برای نصب بیش از حد
مدل k-میانگین (K-means)	ساده است، می‌تواند به سرعت آموزش داده شود؛ مقیاس پذیری قوی؛ توانایی تناسب با داده‌های بزرگ	نبود عملکرد خوب در داده‌های غیر محدب؛ حساس به مقداردهی اولیه؛ حساس به پارامتر K

۴-۲ انواع داده‌های پردازشی توسط الگوریتم‌ها

یکی از ویژگی‌های مهم هر کدام از الگوریتم‌ها که تا کنون به آن‌ها پرداخته نشده، نوع داده‌هایی‌ست که هر یک بررسی می‌کنند. برخی از نوع‌های مختلف داده‌ها به همراه الگوریتم‌هایی که قابلیت پردازش آن‌ها را دارند، در ادامه آمده است.

- بسته (Packet)

الگوریتم‌های ماشین بردار پشتیبان و مدل k-میانگین قادر به تجزیه‌ی بسته‌ها هستند. به علاوه، الگوریتم‌های شبکه‌ی عصبی کانولوشن، خود رمزگذار و شبکه‌ی خصمانه تولیدی می‌توانند بار[۱]بسته را تجزیه و تحلیل نمایند. [۳]

- جریان (Flow)

در این بخش ما ما می‌توانیم سه دسته‌ی مختلف از الگوریتم‌ها را شاهد باشیم. دسته‌ی اول، الگوریتم‌هایی مانند ماشین بردار پشتیبان، درخت تصمیم، شبکه‌ی خلیج ساده، k-نزدیک‌ترین همسایه و مدل k-میانگین هستند که به ویژگی‌های آماری جریان‌ها علاقه‌مندند و این دسته از داده‌ها را پردازش می‌کنند. دسته‌ی دوم که شامل الگوریتم‌هایی مثل شبکه‌ی عصبی کانولوشن، خود رمزگذار و شبکه‌ی خصمانه تولیدی است، به بررسی جریان‌ها بر پایه‌ی یادگیری عمیق می‌پردازند. اما دسته‌ی سوم که کار گروه‌بندی ترافیک را از روی داده‌های ورودی دارد، شامل الگوریتم‌های شبکه‌ی عصبی عمیق و ماشین بردار پشتیبان می‌باشد. [۳]

- نشست (Session)

برای بررسی اطلاعات به دست آمده از هر نشست، می‌توانیم بر ویژگی‌های آماری این اطاعات تکیه کنیم و یا داده‌هایی را از توالی نشست‌هایی که رخ داده، استخراج نماییم. دو الگوریتم درخت تصمیم و مدل k-میانگین بیش‌تر با اطلاعات آماری نشست‌ها کار می‌کنند. درحالی که پردازش توالی نشست‌ها از جمله توانایی‌های الگوریتم شبکه‌ی عصبی کانولوشن است. [۳]

- وقایع ثبت شده (Log)

وقایع ثبت شده در سیستم، مانند اطلاعات مربوط به ورود به سیستم، استفاده از منابع مختلف و غیره، نکات بسیار مهمی را به سیستم‌های تشخیص نفوذ می‌دهند. برخی از این سیستم‌ها بر پایه‌ی الگوریتم‌هایی برنامه‌ریزی شده‌اند که بتوانند اقدامات بر خلاف قوانین را شناسایی کنند و خبر دهند. الگوریتم k نزدیک‌ترین همسایه و الگوریتم شبکه‌ی عصبی عمیق از این دسته مدل‌ها هستند.

از مدل‌های دیگر می‌توان به الگوریتم‌های شبکه‌ی عصبی کانولوشن، شبکه‌ی عصبی عمیق، شبکه‌ی عصبی راجعه و مدل k-میانگین اشاره نمود که با کمک پنجره‌ی کشویی[۲]اطلاعاتی را از وقایع ثبت شده استخراج می‌کنند.

در آخر الگوریتم ماشین بردار پشتیبان را نیز می‌توانیم از جمله الگوریتم‌هایی که با وقایع ثبت شده کار می‌کنند، شناخت. این الگوریتم با اطلاعات این وقایع به تحلیل متن می‌پردازد.

۳-۴ برخی اقدامات برای بهبود الگوریتم‌ها

حال که مزایا و معایب و برخی خصوصیات هر یک از الگوریتم‌ها بررسی گشت، لازم است که به برخی اقداماتی که می‌توان انجام داد، تا الگوریتم‌ها بهبود یابند را بررسی نمود.

[۱] payload

[۲] sliding window

جدول۲ - اقدامات برای بهبود الگوریتم‌ها

الگوریتم	اقدامات برای بهبود الگوریتم
k نزدیک‌ترین همسایه	کاهش زمان مقایسه با نابرابری مثلثاتی؛ پارامترهای بهینه شده توسط بهینه‌سازی تراکم ذرات[۱] مجموعه داده‌های متعادل با استفاده از روش نمونه‌برداری از اقلیت مصنوعی (SMOTE) [٤]
ماشین بردار پشتیبان	پارامترهای بهینه شده توسط بهینه سازی تراکم ذرات (PSO) [٥]
شبکه‌های عصبی مصنوعی	به دست آوردن بهینه‌سازها، توابع فعال‌سازی و توابع از دست رفته [3]
درخت تصمیم	مجموعه داده‌های متعادل با SMOTE؛ معرفی متغیرهای نهان
شبکه‌های خلیج ساده	وارد کردن متغیرهای نهفته برای فرض مستقل
رگرسیون لجستیک	نظم وارداتی برای جلوگیری از نصب بیش از حد
مدل k-میانگین	روش مقدماتی بهبود یافته

۴-۴ خلاصه

در فصل چهارم در کنار بررسی دقیق‌تر هر یک از مدل‌های ارائه شده در فصول گذشته، مزایا و معایب و همچنین نوع داده‌های پردازشی آن‌ها بررسی شدند. در انتها نیز برخی اقدامات به سوی بهبود طبق پژوهش‌های انجام شده در این زمینه، ارائه شد. در فصل آینده، جمع‌بندی و نتیجه‌گیری از مطالب فصل‌های گذشته به همراه پیشنهاداتی در زمینه‌ی مطالعتی سیستم تشخیص نفوذ برای خوانندگان فراهم گردیده است.

[۱] particle swarm optimization (PSO)

فصل پنجم

جمع‌بندی و نتیجه‌گیری و پیشنهادات

۵-۱ جمع‌بندی و نتیجه‌گیری

امروزه تحقیقات زیادی در زمینه‌ی امنیت شبکه‌های کامپیوتری و به دنبال آن مباحثی چون تشخیص نفوذ به سیستم‌های کامپیوتری در جریان است. ما نیز سعی کردیم در این مقاله الگوریتم‌های مهم در این زمینه را معرفی کنیم و مزایا و معایب هر کدام را در مقایسه‌ی با یکدیگر بررسی کنیم که نتایج آن در فصل قبل قابل مشاهده است. این نتایج جمع‌آوری شده از تحقیقات و منابع مختلف به ما کمک می‌کند تا بتوانیم با داشتن یک دید جامع در مورد راهکارهای مختلف در این زمینه، با توجه به نیاز سازمان و موارد مورد استفاده‌ی خود، الگوریتم بهینه‌تر و کاراتری را انتخاب کنیم که حداکثر هم‌خوانی را با منابع در دسترسی و هزینه و نیازهای ما داشته باشد. به علاوه الگوریتم انتخابی باید توانایی کار با داده‌های مدنظر ما را داشته باشد که به این مهم نیز در فصل گذشته پرداخته شد.

۵-۲ پیشنهادات

در انتها پیشنهاد می‌دهیم در زمینه‌ی افزایش کارآیی و بهبود مشکلاتی که برای الگوریتم‌ها در فصل چهارم اشاره شده منابع آن‌ها را با دقت بیشتری مطالعه کنید. این بهبود می‌تواند با ترکیب چندین الگوریتم که مکمل یکدیگر هستند به دست آید؛ همانطور که برخی از الگوریتم‌های بیان شده هم به تنهایی ترکیبی از چند الگوریتم بودند. همچنین می‌توان علاوه بر ملاک‌های بررسی شده در مقاله برای هر الگوریتم، ویژگی‌های بیش‌تری در نظر گرفت و از جهات دیگر نیز آن‌ها را مورد بررسی و مقایسه قرار داد.

مراجع و منابع

[1] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin, “Intrusion detection by machine learning: A review,” *Expert Syst. Appl.*, vol. 36, no. 10, pp. 11994–12000, Dec. 2009, doi: 10.1016/j.eswa.2009.05.029.

[2] J. A. Anderson, *An Introduction to Neural Networks*. MIT Press, 1995.

[3] H. Liu and B. Lang, “Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey,” *Appl. Sci.*, vol. 9, no. 20, Art. no. 20, Jan. 2019, doi: 10.3390/app9204396.

[4] H. H. Pajouh, G. Dastghaibyfard, and S. Hashemi, “Two-tier network anomaly detection model: a machine learning approach,” *J. Intell. Inf. Syst.*, vol. 48, no. 1, pp. 61–74, Feb. 2017, doi: 10.1007/s10844-015-0388-x.

[5] F. Kuang, S. Zhang, Z. Jin, and W. Xu, “A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection,” *Soft Comput.*, vol. 19, no. 5, pp. 1187–1199, May 2015, doi: 10.1007/s00500-014-1332-7.