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

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده‌ی مهندسی کامپیوتر

گزارش نوشتاری موضوع

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 اســتفاده‌ی روزانه از اینترنت، به دنبال خود مشــکلات امنیتی فراوانی را به وجود آورده که به تنهایی یکی از مســایل پیچیده و بســیار مهم حوزه‌ی شــبکه‌های کامپیوتری اســت. با گســترش 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[1] 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 ســیســتم‌های تشــخیص نفوذ[2]، 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 اســتفاده از یادگیری ماشــین این اســت که دیگر برای تشــخیص نفوذ شــبکه 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 ســعی دارد تا معرفی کوتاهی از الگوریتم‌های تشــخیص نفوذ مبتنی بر یادگیری ماشــین به عمل آورد و هم‌چنین به کمک آمار و نمودارهای جمع‌آوری شده آن‌ها را از جهت‌های مختلف با یکدیگر مقایسه کند.

واژه‌های کلیدی:

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

[1] Firewalls

[2] Intrusion Detection System (IDS)

فهرست مطالب

عنوان صفحه

فهرست اشکال

عنوان صفحه

فهرست جداول

عنوان صفحه

فصل اول

مقدمه

۱-۱ مقدمه

حتما تا کنون 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[1] ترافیک‌های ورودی، استفاده از احراز هویت برای کنترل کردن اطلاعات و داده‌ها، استفاده از ضد ویروس برای جلوگیری کردن از انتشار کرم 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[2] (IPS) می‌توانند به تلاش‌های نفوذ در شبکه و همچنین جلوگیری از آن‌ها کمک کنند. در ادامه توضیحات بیش‌تری در مورد این سیستم‌ها داده شده است.

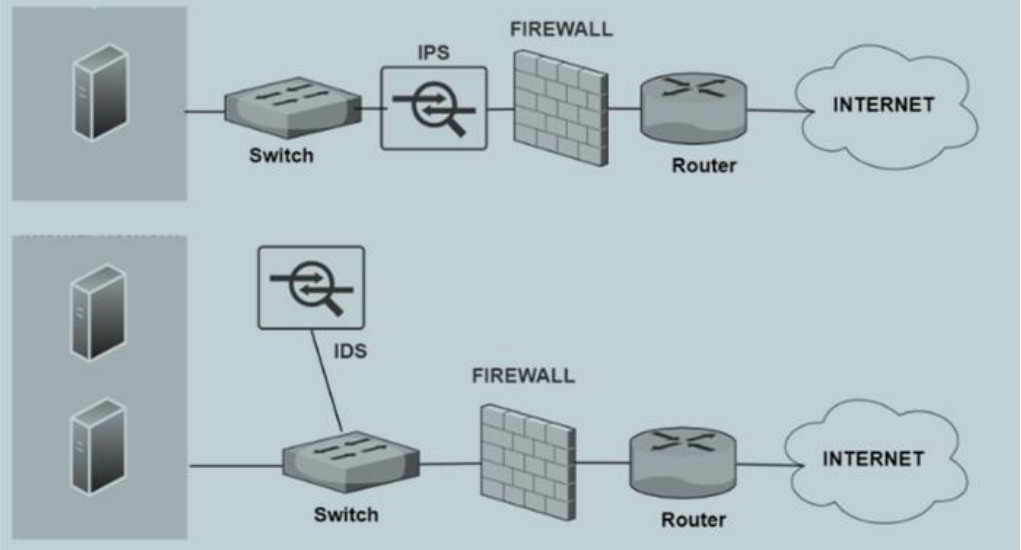
۱-۲ سیستم تشخیص نفوذ

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.

[1] filter

[2] 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[1] و روش تشخیص مبتنی بر امضا[2] [1].

[1] Anomaly-based

[2] 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 [2].

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 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 خوهیم 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 [3].

۲-۲ مدل‌های کم‌عمق تحت نظارت

در ادامه توضیح مختصری از انواع الگوریتم‌های تحت نظارت در بخش‌های ۳-۱ و ۳-۲ داده شده است. این الگوریتم‌ها 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 [3].

۲-۲-۱ k نزدیک‌ترین همسایه[1]

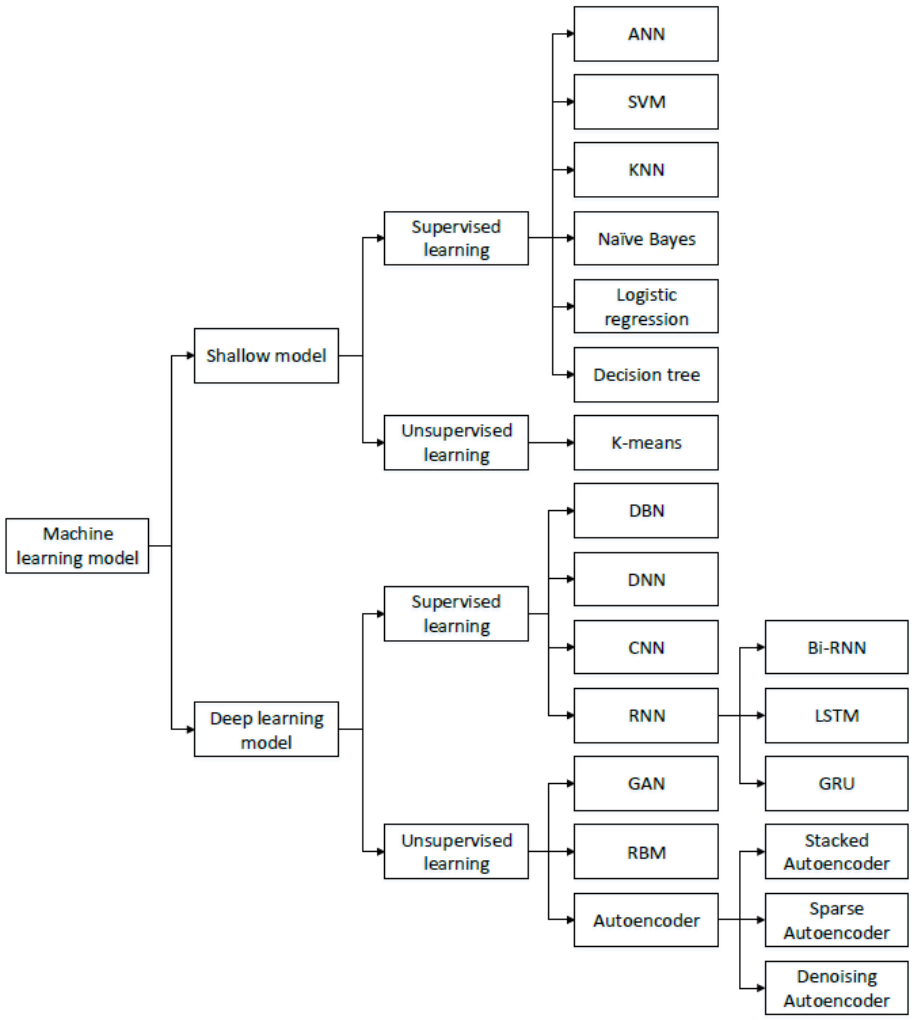
روش k نزدیک‌ترین همسایه (k-NN) یکی از ساده‌ترین و سنتی‌ترین تکنیک‌های غیرپارامتری برای طبقه‌بندی نمونه‌ها است. در این روش فاصله‌ی تقریبی بین نقاط مختلف بردارهای ورودی محاسبه می‌شود و سپس نقطه‌ی بدون برچسب به کلاس k-NN آن‌ها اضافه می‌شود.

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.

k-NN یادگیری مبتنی بر نمونه[2]نامیده می‌شود و با رویکرد یادگیری استقرایی متفاوت است [2].

[1] K-Nearest Neighbor (KNN)

[2] instance based learning



شکل ۲ - طبقه‌بندی الگوریتم‌های یادگیری ماشین

۲-۲-۲ ماشین بردار پشتیبان[1]

ماشین بردار پشتیبان 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.

[1] Support Vector Machines (SVM)

SVM همچنین یک پارامتر مشخص شده توسط کاربر به نام ضریب مجازات[1]را فراهم می‌کند. این پارامتر به کاربران این امکان را می‌دهد تا بین تعداد نمونه‌های طبقه بندی اشتباه و پهنای مرز تصمیم‌گیری معامله کنند [2].

۲-۲-۳ شبکه‌های عصبی مصنوعی[2]

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

پرسپترون چند لایه[3] 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 آموزش تنظیم می‌شود.

برای آموزش MLP، معمولاً از الگوریتم یادگیری تولید متناوب استفاده می‌شود؛ به این شبکه‌ها، شبکه‌های عصبی انتشار مجدد[4]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 [2].

۲-۲-۴ نقشه‌های خود سازمان دهی شده[5]

نقشه خود سازمان دهی شده 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 تصویربرای سادگی دو بُعدی است.

این الگوریتم معمولاً از یک لایه ورودی و لایه کوهونن[6]تشکیل شده که به صورت آرایش دو بعدی نورون‌ها طراحی شده است و ورودی های n بعدی را در دو بعد 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

[1] penalty factor
[2] Artificial Neural Networks (ANN)
[3] Multilayer perceptron (MLP)
[4] backpropagation neural networks
[5] Self-Organizing Maps
[6] Kohonen layer

یکسان یا مشابه روی یک نقشه دو بعدی ترسیم می‌کند. بنابراین، واحدهای خروجی خود را با یک نقشه‌ی مرتب سازماندهی می کند که و هم‌چنین واحدهای خروجی با وزن مشابه نیز پس از آموزش در همان نزدیکی قرار می‌گیرند [2].

۲-۲-۵ درخت تصمیم[1]

درخت تصمیم 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 انتهایی نشان‌دهنده‌ی یک دسته‌بندی طبقه‌بندی شده است. ویژگی‌های نمونه‌ها به هر گره اختصاص می‌یابد و مقدار هر شاخه متناسب با صفات است.

یک برنامه شناخته شده برای ساخت درختان تصمیم "طبقه‌بندی و بازگشت درخت[2]" است. اگر به درخت تصمیم برچسب های گسسته یا نمادین کلاس‌بندی را بیفزاییم، درخت طبقه‌بندی نامیده می‌شود، در حالی که درخت تصمیم با دامنه‌ی مقادیر پیوسته یا عددی، درخت رگرسیون نامیده می‌شود [2].

۲-۲-۶ شبکه‌های خلیج ساده[3]

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 بهره برداری از این وابستگی‌های گاه به گاه بین متغیرهای تصادفی یک مسئله، می‌توان از یک مدل نمودار احتمالی به نام شبکه‌های خلیج ساده 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 می شود، 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 [2].

۲-۲-۷ الگوریتم‌های ژنتیک[4]

[1] Decision Tree
[2] Classification and Regressing Tree (CART)
[3] Naïve Bayes Networks (NBN)
[4] Genetic Algorithms (GA)

الگوریتم‌های ژنتیکی 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 [2].

۲-۲-۸ منطق فازی[1]

منطق فازی (یا نظریه مجموعه‌های فازی) مبتنی بر مفهوم پدیده فازی است که اغلب در دنیای واقعی رخ می‌دهد. نظریه مجموعه‌های فازی برای استدلال کردن، به مفهوم عضویت در مجموعه، مقادیر بین ۰ و ۱ را 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 [2].

۲-۲-۹ رگرسیون لجستیک[2]

LR 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 [3].

۲-۳ مدل‌های عمیق تحت نظارت

۲-۳-۱ شبکه کوتاه عمیق[3]

شبکه‌ی کوتاه عمیق یا DBN از چندین لایه ماشین بولتزمن محدود[4] و یک لایه طبقه‌بندی سافت‌مکس[5] تشکیل شده است، همانطور که در شکل ۳ نشان داده شده‌است.

---

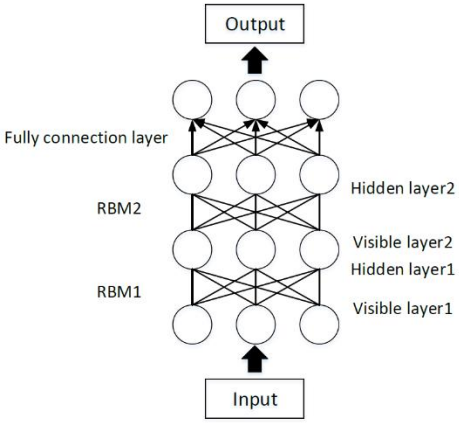
[1] Fuzzy Logic

[2] Logistic Regression (LR)

[3] Deep Brief Network (DBN)

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

[5] softmax

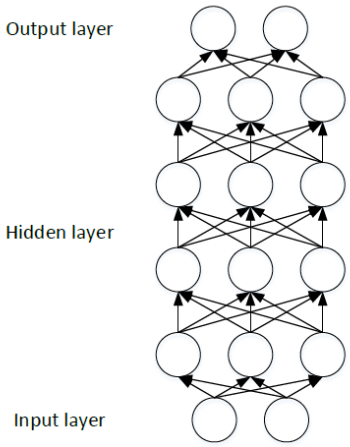


شکل ۳ - ساختار شبکه‌ی کوتاه عمیق

آموزش DBN شامل دو مرحله است: ابتدا یک مرحله آموزش بدون نظارت که در آن RBM به عملکردی حریصانه و لایه‌ای آموزش می‌یابد. سپس، وزن لایه سافت‌مکس با داده‌های دارای برچسب به روز رسانی می‌شود. در شناسایی حمله، DBNها هم برای استخراج ویژگی و هم برای طبقه‌بندی استفاده می‌شوند [3].

۲-۳-۲ شبکه عصبی عمیق[1]

همانطور که در شکل ۴ می‌بینید، ساخت یک DNN با استفاده از استراتژی پیش‌یادگیری و با تنظیم دقیق چندین لایه امکان پذیر است. هنگام آموزش یک DNN، پارامترها ابتدا با استفاده از داده‌های بدون برچسب، که یک مرحله یادگیری ویژگی بدون نظارت است، آموزش می‌بینند. سپس، شبکه از طریق داده‌های دارای برچسب، که یک مرحله یادگیری تحت نظارت است، تنظیم می‌شود. دستاوردهای حیرت انگیز DNN ها عمدتا به دلیل مرحله یادگیری ویژگی بدون نظارت است.

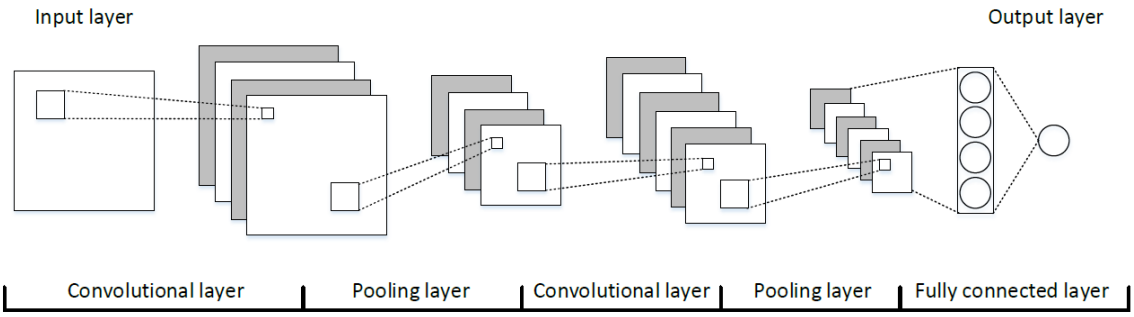


شکل ۴ - ساختار شبکه عصبی عمیق

[1] Deep Neural Network (DNN)

۲-۳-۳ شبکه عصبی کانولوشن[1]

CNNها به تقلید از سیستم بینایی انسان طراحی شده‌اند. در نتیجه، در زمینه‌ی بینایی کامپیوتر دستاوردهای بزرگی داشته اند. همانطور که در شکل ۵ هم نشان داده شده است، یک CNN با لایه‌های کانولوشن و استخر جایگزین انباشته[2] شده است. CNNها بر روی داده‌های ۲ بُعدی کار می‌کنند، بنابراین داده‌های ورودی باید برای شناسایی حمله به ماتریس ترجمه شوند.



شکل ۵ - ساختار شبکه عصبی کانولوشن

۲-۳-۴ شبکه عصبی راجعه[3]

RNNها شبکه‌هایی هستند که برای داده‌های پی‌درپی طراحی شده‌اند و به طور گسترده‌ای در پردازش زبان طبیعی[4] 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.

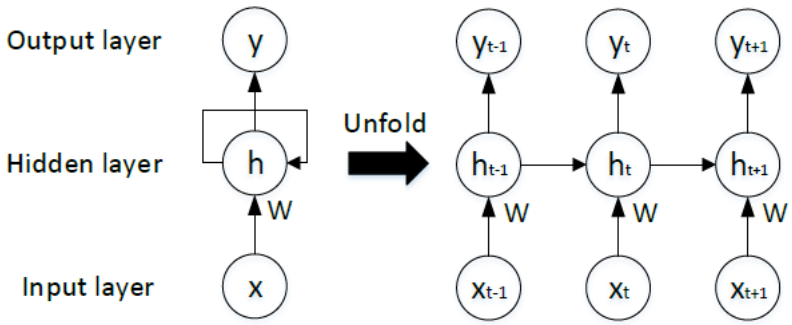
ساختار RNN در شکل ۸ نشان داده شده است. RNNهای استاندارد فقط با توالی‌هایی با طول محدود سروکار دارند.

[1] Convolutional Neural Network (CNN)

[2] stacked

[3] Recurrent Neural Network (RNN)

[4] natural language processing (NLP)



شکل ۶ - ساختار شبکه عصبی راجعه

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 هستند. 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 انواع الگوریتم‌های یادگیری ماشین تحت نظارت که در زمینه‌ی تشخیص نفوذ مورد استفاده قرار می‌گیرند، تا حدودی 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.

گفتیم که 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 [3].

در ادامه انواع این الگوریتم‌ها را می‌خوانید.

۳-۲ مدل‌های کم‌عمق نظارت نشده

.

۳-۲-۱ مدل k-معنی[1]

.

۳-۳ مدل‌های عمیق نظارت نشده

۳-۳-۱ شبکه‌های خصمانه تولیدی

.

۳-۳-۲ ماشین بولتزمن محدود

.

۳-۳-۳ خود رمزگذار

.

۳-۴ خلاصه

.

[1] K-means

فصل چهارم

مقایسه و بررسی الگوریتم‌ها

۴-۱ مقایسه و بررسی الگوریتم‌ها

تا اینجا الگوریتم‌های مطرح در زمینه‌ی تشخیص نفوذ معرفی شدند. در اینجا قصد داریم که این الگوریتم‌ها را با هم مقایسه کنیم.

جدول ۱ - مزایا و معایب مدل‌های مختلف کم عمق

اقدامات برای بهبود الگوریتم	معایب	مزایا	الگوریتم (بخش مربوطه)
بهینه سازها ، توابع فعال سازی و توابع از دست رفته بهبود یافته	مناسب برای نصب بیش از حد؛ مستعد گیر افتادن در یک بهینه محلی؛ آموزش مدل وقت گیر است	قادر به مقابله با داده‌های غیر خطی؛ توانایی اتصالات قوی	ANN شبکه‌های عصبی مصنوعی (۳-۱-۲)

۴-۲ خلاصه

.

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 بود. 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.

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 کرد و از جهات دیگر نیز آن‌ها را مورد بررسی و مقایسه قرار داد.

مراجع و منابع

[1] J. Anderson, An introduction to neural networks, London: Cambridge: MIT Press, 1995.

[2] T. Chih-Fong, H. Yu-Feng, L. Chia-Ying and L. Wei-Yang, "Intrusion detection by machine learning: A review," *Expert Systems with Applications,* vol. 36, no. 10, p. 11994–12000, December 2009.

[3] L. Hongyu and L. Bo, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *applied sciences*, vol. 9, no. 20, pp. 43-96, 2019.