
بسم الله الرحمن الرحيم

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد مهندسی کامپیوتر- هوش مصنوعی

تشخیص هوشمند دامنه‌های مشکوک از داده‌های DNS

نگارنده: فهیمه باقری

استاد راهنما

دکتر محسن رضوانی

اساتید مشاور

دکتر منصورفاتح
دکتر اسماعیل طحانیان

پایان نامه ارشد جهت اخذ درجه کارشناسی ارشد

مهرماه ۱۳۹۹

شماره: ۷۰۷ ر ت
تاریخ: ۹۹/۸/۱۹

باسمه‌تعالی

دانشگاه صنعتی شاهرود
مدیریت تحصیلات تکمیلی

فرم شماره (۳) صورتجلسه نهایی دفاع از پایان نامه دوره کارشناسی ارشد

با نام و یاد خداوند متعال، ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد خانم / آقای ..... با شماره دانشجویی ..... رشته ..... گرایش ..... تحت عنوان .....
..... که در تاریخ .....
..... با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می‌گردد:

الف) درجه عالی: نمره ۲۰-۱۹ ☐ ب) درجه خیلی خوب: نمره ۱۸/۹۹ – ۱۸ ☑
ج) درجه خوب: نمره۱۷/۹۹– ۱۶ ☐ د) درجه متوسط: نمره ۱۵/۹۹-۱۴ ☐
ه) کمتر از ۱۴ غیر قابل قبول و نیاز به دفاع مجدد دارد ☐
نوع تحقیق: **نظری** ☐ **عملی** ☐

عضو هیأت داوران	نام ونام خانوادگی	مرتبهٔ علمی	امضاء
۱ـ استادراهنمای اول	محسن زمانی	استادیار	
۲- استادراهنمای دوم			
۳- استاد مشاور	اسماعیل طحانی	استادیار	
۴- نماینده تحصیلات تکمیلی	محسن زاهر	مربی	
۵- استاد ممتحن اول	حسین سرسنگلو	استادیار	
۶ـ استاد ممتحن دوم	فاطمه حضری نژاد	استادیار	

نام و نام خانوادگی رئیس دانشکده:

تاریخ و امضاء و مهر دانشکده:

۴

تقدیم اثر

خدایی که آفرید

جهان را، انسان را، عقل را، معرفت را، عشق را

و به کسانی که عشقشان را در وجودم دمید

گاهی بیایم و احوالشان را بپرسیم

با سپاس از سه وجود مقدس:

آنان که ناتوان شدند تا ما به توانایی برسیم ...

موهایشان سپید شد تا ما روسفید شویم ...

و عاشقانه سوختند تا گرما بخش وجود ما و روشنگر راهمان باشند...

پدرانمان

مادرانمان

استادانمان

محضر ارزشمند پدر و مادر عزیزم به خاطر همه ی تلاشهای محبت آمیزی که در دوران مختلف زندگی ام انجام داده اند و با مهربانی چگونه

زیستن را به من آموخته اند.

تشکر و قدردانی

از استاد گرامیم جناب آقای دکتر محسن رضوانی بسیار سپاسگزارم چرا که بدون راهنمایی‌های ایشان تامین این پایان نامه بسیار مشکل می نمود.

از جناب آقایان دکتر منصور فاتح و دکتر اسماعیل طحانیان که اساتید مشاور این پایان‌نامه بوده‌اند نیز قدردانی می نمایم.

در پایان از کلیه دوستان خوبم که مطالب زیادی را به من آموختند تشکر کرده و برایشان آرزوی موفقیت و سربلندی می کنم.

فهیمه باقری

شهریور۹۹

تعهدنامه

- اینجانب **فهیمه باقری** دانشجوی دوره کارشناسی ارشد (دکتری) رشته **مهندسی کامپیوتر** دانشکده مهندسی کامپیوتر دانشگاه صنعتی شاهرود نویسنده پایان‌نامه **"تشخیص هوشمند دامنه‌های مشکوک از داده‌های DNS** تحت راهنمائی دکتر محسن رضوانی، متعهد می شوم.
- تحقیقات در این پایان‌نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان‌نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می‌باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « Shahrood University of Technology » به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان‌نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان‌نامه ، در مواردی که از موجود زنده (یا بافتهای آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است.

تاریخ

امضای دانشجو

چکیده

با پیشـرفت فناوری در شـبکه اینترنت یکی از مهم‌ترین چالش‌های امنیتی و ریسـک‌های موجود حملات فیشـینگ و کلاه‌برداری توسط فیشرها است. فیشینگ (Phishing) نوعی حمله سایبری است، که همواره در تلاش برای بدسـت آوردن اطلاعاتی مانند نام کاربری، گذرواژه، اطلاعات حسـاب بانکی و مانند آنها از طریق جعل یک وب‌سایت، آدرس ایمیل و متقاعد کردن کاربر به منظور واردکردن این اطلاعات می‌باشد. ازآن‌جایی که فیشـینگ شـامل فریب‌های روانی اسـت و به جای اشـتباهات نرم‌افزاری یا سـخت‌افزاری به اشتباهات انسانی متکی می‌باشد، کاربران سهم زیادی در جلوگیری از به دام افتادن در این نوع حملات را دارند. از طرفی تشـخیص دامنه‌های مشـکوک با مشـاهده توسـط کاربران مقدور نمی‌باشد و سیستم‌های تشـخیص فیشـینگ فعلی اغلب نمی‌توانند خود را با حملات جدید تطبیق دهند و دارای دقت پایین در شـناسایی و نرخ مثبت کاذب بالا هستند. روش‌های مبتنی بر گراف یکی از روش‌های شناسایی دامنه‌های مشـکوک اسـت. چالش اصـلی در این پایان‌نامه این اسـت که چگونه می‌توان با روش مبتنی برگراف و یادگیری عمیق دقت تشـخیص این دامنه‌ها را افزایش داد. از این رو در این پایان‌نامه سـیسـتم تشخیص فیشـینگ مبتنی برگراف با استفاده از یادگیری عمیق ارائه شده است. ایده کلیدی این روش استخراج IP از دامنـه، تعریف ارتبـاط بین دامنـه‌هـا، وزن آن‌هـا وهمچنین تبـدیـل داده‌ها به بردار توسـط الگوریتم Node2vec می‌باشـد. در ادامه با استفاده از مدل‌های یادگیری عمیق CNN و DENSE عمل طبقه‌بندی و شـناسـایی انجام می‌شـود. نتایج نشان دادند که روش مبتنی برگراف و استفاده از یادگیری عمیق تاثیر مطلوبی در شـناسایی این دامنه‌ها دارد. به طوری که روش پیشنهادی دارای دقت ۹۹ درصد در مجموعه دادگان اول است که در مقایسه با بهترین روش قبل یعنی BP افزایش یک درصدی داشته است.

کلیدواژه: تشخیص دامنه مشکوک، داده‌های DNS، فیشینگ، یادگیری عمیق

فهرست مطالب

فهرست جداول

فهرست شکل‌ها

فصل۱ : مقدمه

۱-۱ کلیات پژوهش

جامعه جهانی در عصری به سر می‌برد که در آن اطلاعات و دانش را سرمایه اصلی و عامل اساسی رشد و توسعه یک جامعه می‌شناسد تا آنجا که « دانایی، توانایی است. » شعار اصلی این عصر است، که نوید بخش جهانی نو با شیوه‌های نوین بکارگیری اطلاعات و دانش است [۱]. حرف اول را دراین دوران نوین، فناوری اطلاعات می‌زند. به این معنی که اقتدار اقتصادی، سیاسی، فرهنگی و علمی هر کشور متناسب با میزان تسلط و بهره‌گیری آن کشور از این فناوری برای انجام فعالیت‌های خود در سطح جامعه خواهد بود. فناوری اطلاعات به عنوان یکی از دستاوردهای جدید بشری، نه تنها دست‌خوش تغییرات شگرفی شده بلکه به سرعت درحال تاثیرگذاری برالگوهای زندگی وکاری افراد و جامعه، شیوه تحقیق وآموزش، امنیت، بهداشت و مهم‌تر از همه تجارت شده است [۲].

یکی از چالش‌های اصلی در فناوری اطلاعات حوزه امنیت آن است و هر روزه سارقان اینترنتی راه‌های جدیدی را برای بدست آوردن هویت شخصی افراد و دست‌یابی به اطلاعات شخصی و اطلاعات مالی آنها به کار می‌برند. یکی از روش‌هایی که اخیرا بسیار مورد توجه آنها قرار گرفته و البته کمی هم پیچیده می‌باشد، فیشینگ[۱] نام دارد [۳]. حملات موسوم به فیشینگ به آن دسته از حملات اینترنتی گفته می‌شود که معمولا طراحان آن‌ها به روش‌های مختلف تلاش می‌کنند به اطلاعات بانکی شما از طریق روش‌های متنوع مهندسی اجتماعی مانند ایمیل، تماس تلفنی، صفحات جعلی پرداخت، پیامک، انواع مدل‌های ربات‌های تلگرام و انواع روش‌های جدیدی که انتظار آن نمی‌رود دست یابد، در این‌گونه حملات، فیشر با ارسال یک ایمیل، خود را به جای فرد یا شرکت معتبر و حتی بانک‌های معتبر جا می‌زند و با تکنیک‌های گول‌زننده سعی می‌کند تا اطلاعات حساس را از قربانی بگیرد.

برای ارسال این قبیل ایمیل‌ها از یک‌سری سرورهای ارسال ایمیل‌های جعلی که به صورت رایگان می‌باشد، استفاده می‌کنند. این عمل مجرمانه جدید، امروزه در شبکه اینترنت در دسترس به یک

[۱] Phishing

پدیده و معضـل گسترده، تبدیل شده است [۳]. این نوع از حملات بیشتر از آنکه دانش و آگاهی فنی بخواهد، به شـیطنت و شـخصـیت نادرسـت نیاز دارد. این روش‌ها وابسته به هیچ‌کدام از انواع مختلف سـیسـتم‌های عامل، مرورگرهای اینترنت و یا سیستم پستی نیستند [۴]. نسل جدید این نوع حملات که در قالب برنامه‌های مختص گوشی‌های هوشمند و صفحات فیشینگ شکل گرفته‌اند به راحتی قابل تشـخیص نمی‌باشـند و باید راه‌کارهایی را ارائه داد که با رعایت آن توسـط کاربر و سرویس دهندگان، جرائم فیشینگ را تا حد زیادی کاهش داد.

۱-۲ تعریف مسئله

اینترنت زندگی انسان را به‌طور معنی‌داری تغییر داده و بسیاری از زمینه‌های تجارت، بهداشت، درمان و غیره را تحت‌الشعاع خود قرار داده است. از همین‌رو برقراری امنیت در این فضا نیز به‌طور چشم‌گیری مورد توجه قرار گرفته است [۵]. دسترسی تعداد زیادی از مردم جهان به شبکه اینترنت و گسترش ارتباطات الکترونیکی بین افراد و سازمان‌های مختلف از طریق دنیای مجازی بستری مناسب را برای برقراری مراودات تجاری و اقتصادی فراهم کرده است [۲]. در حملات فیشینگ آنچه که باعث شده همیشه جزء قربانیان باشیم سهل‌انگاری و نداشتن آگاهی است [۵ و ۶].

برای مقابله با این فریب‌کاری که به عنوان یک جرم مهندسـی اجتماعی تعبیر می‌شود تدابیر ضد فیشـینگ (Anti Phishing) متعددی اندیشـیده شده است. همان‌طور که قبل‌تر اشاره شد، عمده‌ترین علت گسـترش فیشـینگ ناآگاهی کاربران از وجود چنین کلاه‌برداری است [۷]. روش‌های متنوعی در شـناسـایی فیشـینگ توسط الگوریتم بهینه‌سازی در سه طبقه فیشینگ، قانونی و مشکوک پیشنهاد و پیاده‌سـازی گردیده اسـت [۶ و ۸]. اما توافقی در مورد نرم‌افزارهای ضد فیشینگ برای رسیدن به یک استاندارد مشترک وجود ندارد از طرفی با وجود نکات امنیتی بازهم راه‌ها و روش‌هایی برای جلوگیری از تشخیص فیشینگ وجود دارد.

فرآینـد فیشـــینگ وبســـایت را می‌توان یکی از چالش‌های امنیتی مهم برای ارتباطات آنلاین دانســـت، چرا که در چنین جوامعی، روزانه با حجم زیادی از تراکنش‌ها روبرو هســـتیم که به صـــورت آنلاین صورت می‌گیرند. در این پایان‌نامه سناریوی حمله فیشینگ تحت DNS[۲] و شناسایی دامنه‌های فیشـــینـگ با روش مبتنی بر گراف با تکیه بر مدل‌های یادگیری عمیق و الگوریتم طبقه‌بند یادگیری ماشین نشان داده شده است؛ در این رویکرد به شناخت، تحلیل و معرفی شناسایی دامنه‌های مشکوک از ســالم پرداخته شده است که با استفاده از مجموعه دادگانی متشکل از دامنه‌های مورد اعتماد عمل شناسایی انجام می‌شود. در این راستا پرسش‌های زیر مطرح می‌شود:

- آیا سیستم پیشنهادی در تشخیص فیشینگ به صورت دقیق و با اطمینان بالا سایت فیشینگ را شناسایی می‌کند؟
- یادگیری عمیق تا چه حد جوابگوی مسئله ما با دقت بالا می‌باشد؟
- استفاده از روش مبتنی بر گراف تا چه حد قادر به دسته‌بندی دامنه‌های فیشینگ است؟
- آیا الگوریتم مورد استفاده در روش پیشنهادی نسبت به سایر روش‌ها و الگوریتم‌های استفاده شده دارای دقت بالاتر و خطای کمتری است؟

۱-۳ سابقه موضوع

امروزه مهم‌ترین ریسک و چالش مورد توجه در تجارت و مراودات اجتماعی، خطرکلاه‌برداری آنلاین و حملات فیشرها است. اختلافات زیادی در زمینه بهترین روش برای پیشگیری از خطرات سایت‌های فیشینگ وجود دارد. در مرجع [۹]، جهت شناسایی و پیش‌بینی وب‌سایت‌های فیشینگ از الگوریتم‌های طبقه‌بندی براساس مشخصه‌های صفحات وب استفاده می‌شود. این الگوریتم‌ها نرخ خطای کمتری نسبت به سایر تکنیک‌های مقابله با حملات فیشینگ دارند.

[۲] Domain Name System

روش‌های اســتنباطی، یکی از روش‌های عمده برای تحلیل داده‌ها و شــناســایی دامنه‌های مخرب می‌باشــد. این روش به تجزیه و تحلیل آدرس‌های IP اختصاصی و عمومی متکی است. در این روش‌ها نه تنها پوشــش دامنه‌ها بلکه صــحت تشــخیص نیز نســبت به ارتباط‌های بین دامنه‌های موجود بهبود می‌یابد [۱۰ و ۱۱]. در رویکرد موجود در مرجع [۱۰]، از الگوریتم‌های مبتنی بر مســیر و انتشــار باور BP[۳] برای تحلیل داده‌های DNS و شــناســایی دامنه‌های مشــکوک در جهت بهبود صحت تشخیص و پوشش دامنه‌ها، استفاده شده است.

علی‌رغم راه‌کارهای زیادی که برای ردیابی حملات فیشینگ، انجام شده است [۱۲- ۱۴]، هنوز هم فقدان دقت برای سیستم‌های تشخیص فیشینگ در حالت آنلاین وجود دارد. در مرجع [۱۲]، برای اولین بار یک شبکه عصبی با یادگیری تقویتی به منظور تشخیص حملات فیشینگ در حالت آنلاین ترکیب شده است. مدل پیشنهادی از یک شبکه عصبی، یادگیری تقویتی، تکنیک‌های طبقه‌بندی، داده‌کاوی و مجموعه‌ای از الگوریتم‌های یادگیری ماشین برای ردیابی حملات فیشینگ استفاده می‌کند.

راه‌حل‌هایی نظیر نشانی‌های اینترنتی لیست سیاه تاحدی موثر بوده‌اند. از این‌رو در مرجع [۱۳]، یک الگوریتم تطابق تقریبی ارائه شده است که در آن ابتدا یک نشانی اینترنتی به چند مولفه تقسیم می‌شود. سپس هر مولفه به‌طور جداگانه با ورودی‌های موجود در لیست سیاه مقایسه می‌شود. هم‌چنین در مرجع [۱۴]، برای شناسایی دامنه‌های مخرب از طریق تحلیل داده‌های DNS، برای طبقه‌بندی گره‌ها براساس ویژگی‌های محلی دامنه‌های مرتبط با DNS بهره برده است.

در مرجع [۱۵]، با ایجاد یک افزونه برای مرورگر کروم که به عنوان یک میان‌افزار بین کاربران و وب‌سایت‌های مخرب عمل می‌کند، از ورود کاربران به چنین وب‌سایت‌هایی جلوگیری کرده و ورود کاربر

[۳] Belief Propagation

را مسدود می‌کند. این افزونه با استفاده از الگوریتم‌های یادگیری ماشین به شناسایی وب‌سایت‌های مخرب می‌پردازد و به مرورگر کروم اضافه می‌شود.

در مرجع [۱۶]، به فعالیت‌های مخربی که از طریق DNS انجام می‌شود، پرداخته شده است. دامنه‌های مخرب یکی از منابع اصلی مورد نیاز هکرها برای انجام حملات، از طریق اینترنت است. در نتیجه تامین امنیت سرور DNS می‌تواند بسیار مهم باشد.

یک روش هوشمندانه در مرجع [۱۷]، که بر مبنای داده‌کاوی می‌باشد، دسته‌بندی انجمنی (AC[۴]) نام دارد که می‌تواند به صورت کارآمد، فیشینگ وب‌سایت را با میزان صحت بالایی تشخیص دهد. بر اساس مطالعات تجربی، AC می‌تواند دسته‌بندهایی که شامل قوانین "اگر ... آنگاه ..." هستند را استخراج و با میزان صحت بالایی آنها را پیش‌بینی سازد. این مسئله با استفاده از متد AC که دسته‌بند چند برچسبه بر مبنای دسته‌بندی مشارکتی یا انجمنی (MCAC[۵]) نام دارد ارائه می‌شود. همچین ویژگی‌هایی که بین وب‌سایت‌های فیشینگ با وب‌سایت‌های قانونی تمایز قائل می‌شود، نیز بدست می‌آید. نتایج آزمایشی با استفاده از داده های واقعی جمع‌آوری شده از منابع مختلف نشان داده است که AC و به طور خاص MCAC می‌تواند وب‌سایت‌های فیشینگ را با میزان صحت بالایی نسبت به سایر الگوریتم‌های هوشمند تشخیص دهد. علاوه بر این، MCAC می‌تواند دانش مخفی جدیدی را ایجاد کند که سایر الگوریتم‌ها قادر به یافتن آن نیستند و این باعث بهبود کارائی پیش‌بینی دسته‌بندها شده است.

با این حال، علی‌رغم تمام راه‌کارهای استفاده شده کاربران اینترنتی ممکن است همچنان در معرض تهدیدهای عمده‌ی آنلاینی قرار گیرند که ممکن است آسیب‌های مالی، سرقت هویت و از دست‌دادن اطلاعات را برای آنها به همراه داشته باشد. بنابراین، متناسب بودن اینترنت به عنوان یک کانال برای مبادلات تجاری، امری سؤال برانگیز و چالش برانگیز است. امروزه، اینترنت نه تنها برای کاربران خاص،

[۴] Association Classification

[۵] Multi Classifier Associative Classification

بلکه برای سازمان‌هایی که پروسه‌های شغلی‌شان را به صورت آنلاین انجام می‌دهند ضرورت و اهمیت پیدا کرده است. لذا با توجه به رشد صعودی حملات فیشینگ و افزایش کاربردی اینترنت برای کاربران مقابله با این حملات بسیار مهم بوده و ارائه راه‌کارهای امنیتی در جهت کاهش و شناسایی هرچه بیشتر این نوع حملات احساس می‌شود. با توجه به پیچیده‌ترشدن شیوه‌های این حملات بهبود روش‌های شناسایی یک الزام است.

۱-۴ ضرورت انجام تحقیق

یکی از مشکلات اصلی در تحقیقات انجام شده در حوزه تشخیص فیشینگ، دقت پایین این روش‌ها است [۱۰ و ۱۱]. برای نمونه در مرجع [۱۰]، برای رسیدن نرخ مثبت واقعی[۶] به ۹۳/۱۶ درصد در داده‌های DNS ، نرخ مثبت کاذب[۷] به ۲/۳۶ درصد نیز افزایش یافته است، در نتیجه صحت تشخیص کاهش می‌یابد. از این‌رو، بسیاری از اتصالات معنی‌دار بین دامنه‌ها توسط رویکردهای موجود حذف می‌شوند.

امروزه مهاجمان سایبری نسبت به قبل آگاه‌تر و از تکنیک‌های جدید و پیچیده‌ای برای گریز از شناسایی و مسدود شدن توسط لیست سیاه استفاده می‌کنند [۱۲]. با توجه به این که استفاده از صفحات فیشینگ و حملات اینترنتی افزایش یافته است لزوم تشخیص اینگونه صفحات و شناسایی صفحات فیشینگ و صفحات اصلی احساس می‌شود. در زمینه حملات اینترنتی امنیت مطلق وجود ندارد می‌توان با تسلط بر مباحث امنیتی، بصورت قابل‌توجهی از مشکلات امنیتی جلوگیری کرد. فیشینگ نیز یکی از روش‌های مخرب برای دستیابی به اطلاعات است که روش‌ها و تکنیک‌های مختلفی دارد. اما خوشبختانه با کمی تحقیق و مطالعه می‌توان از گرفتارشدن در دام حملات فیشینگ جلوگیری کرد. از این‌رو با توجه به ضرورت این موضوع و با هدف بررسی ابعاد آن و ارائه راه‌کارهای عملی به منظور کاهش

[۶] True Positive Rate

[۷] False Positive Rate

حجم جرائم و بالابردن دقت تشخیص حملات نیاز به ارائه راه‌حلی است که نه‌تنها شناسایی دامنه‌ها را در مقایسه با رویکردهای موجود بهبود داده، بلکه صحت تشخیص بالاتر و نرخ خطا تا حد امکان کاهش یابد.

۱-۵ روش تحقیق

فیشینگ معضلی است که در تمام دنیا رواج دارد و هنوز هیچ کشوری نتوانسته آن را به صفر برساند؛ از این‌رو در این پایان‌نامه سعی بر آن است که از یک روش مبتنی بر تحلیل گراف برای تشخیص دامنه‌های مشکوک استفاده شود. هدف این است در این شرایط در سریع‌ترین زمان ممکن راه‌کارهای جایگزین سریع را برای مقابله با شیوه‌های جدید فیشینگ ارائه دهیم. در همین راستا سیستم پیشنهادی با تکنیک مبتنی بر تحلیل گراف حاصله از ارتباط بین دامنه‌ها قادر به شناسایی و تشخیص دامنه‌های مشکوک با دقت بالاتری نسبت به روش‌های قبل خواهد بود.

۱-۶ هدف تحقیق

با توجه به رشد جهشی فناوری اطلاعات که بر همگان مبرهن است، شیوه‌های این نوع حملات نیز تغییرات گسترده‌ایی داشته و نیاز به بهبود روش‌ها و نوآوری در مقابله با آن می‌باشد. از این‌رو در این پایان‌نامه به شناخت، تحلیل و معرفی روشی نوین برای شناسایی دامنه‌های مشکوک می‌پردازیم. در واقع طرح پیشنهادی در شناسایی دامنه‌های فیشینگ در مقایسه با روش‌های موجود به طور قابل‌توجهی دقت تشخیص را بهبود بخشیده است. بنابراین صرفه‌جویی در زمان و نیروی انسانی می‌تواند از اهداف مهم در این زمینه نیز باشد.

۱-۷ نوآوری تحقیق

در این طرح، یک الگوریتم مبتنی بر گراف را پیشنهاد می‌کنیم. به طور خاص، ابتدا گراف دامنه-IP را

با رابطه دامنه‌ها و آدرس‌های IP می‌سازیم و گراف وزن‌دار را از روی گراف دامنه-IP بدست می‌آوریم. سپس با استفاده از الگوریتم Node2vec داده‌ها برای ورود به طبقه‌بند به بردار تبدیل می‌شوند. در نهایت از مدل‌های یادگیری عمیق[۸] برای شناسایی اعتبار آن دامنه‌ها جهت دسته‌بندی به فیشینگ و سالم، استفاده می‌شود. در طول این مراحل، از IPهای کلاینت، نام‌های دامنه، و نتیجه تفکیک دامنه به عنوان استخراج اعتبار گره‌ها با استفاده از یادگیری عمیق استفاده شده است.

۱-۸ دادگان تحقیق

مجموعه دادگان مورد استفاده در این پایان‌نامه، نیز مجموعه رکوردهای DNS است. از آنجایی که هیچ‌گونه مجموعه داده استاندارد در سطح جهانی وجود نداردکه بتوان از آن استفاده کرد و اغلب مجموعه دادگانی که موجود می‌باشد توسط خود محققان جمع‌آوری شده است؛ لذا در این پایان‌نامه علاوه بر استفاده از مجموعه دادگان سایر محققان سعی شده است؛ مجموعه دادگان سایت‌های داخلی به منظور ارزیابی عملکرد دقیق‌تر شناسایی دامنه‌های فیشنگ نیز مورد استفاده قرار گیرد. مجموعه دادگان داخلی مورد استفاده شامل سایت‌های فیشینگی است که توسط بانک مرکزی[۹] جمع‌آوری شده است و دامنه‌های سالم و غیر فیشنگ نیز از سازمان فناوری اطلاعات ایران که متولی این مباحث است گرفته شده است. بنابراین از سه مجموعه دادگان در این پایان نامه استفاده می‌شود که مجموعه دادگان اول از بانک مرکزی و سازمان فناوری اطلاعات [۱۹]، مجموعه دادگان دوم از مرجع [۱۸] و مجموعه دادگان سوم ترکیبی از دو مرجع [۱۸ و ۲۰]، گردآوری شده است. این مجموعه داده شامل رکوردهای DNS فعلی سایت‌هایی است که در اینترنت فعال و در دسترس می‌باشند. به طور کلی مجموعه داده‌های DNS را می‌توان به انواع زیر دسته‌بندی نمود [۱۰].

- ✓ Active DNS data

[۸] Deep learning

[۹] مجموعه دادگان مورد استفاده از بانک مرکزی کاملا محرمانه است و امکان انتشار آن وجود ندارد.

- ✓ Passive DNS data
- ✓ Logs of DNS data

که مجموعه داده‌های Active DNS به علت قابل دسترس بودن استفاده می‌شود. سایر داده‌های DNS به دلیل مشکلات امنیتی غیر قابل دسترس است.

۱-۹ ساختار پایان‌نامه

در ادامه پایان‌نامه، در فصل دوم ادبیات تحقیق، مروری بر روش‌های پیشین و معرفی یادگیری عمیق خواهد شد. در فصل سوم الگوریتم‌های پیشنهادی برای آماده‌سازی داده‌ها و استفاده از الگوریتم پیشنهادی، ارائه شده است. در فصل چهارم، نتایج آزمایش‌ها مورد بررسی قرار گرفته است. در فصل پایانی این پایان‌نامه، نتیجه‌گیری و جمع‌بندی کلی از بحث ارائه شده است.

فصل ۲ : ادبیات تحقیق و روش‌های پیشین

۲-۱ مقدمه

در این فصل، ابتدا دو الگوریتم BP و SVM که با روش پیشنهادی مقایسه می‌شود مورد بررسی قرار می‌گیرد. در مرحله بعد الگوریتم Node2vec به منظور تبدیل داده‌های گراف به بردار توضیح داده می‌شود. سپس مهم‌ترین ساختارهای مورد استفاده یادگیری عمیق در روش پیشنهادی مطرح می‌شود. در قسمت آخر به بررسی جدیدترین کارهای انجام شده در زمینه تحلیل و شناسایی حملات پرداخته می‌شود.

۲-۲ الگوریتم انتشار باور BP

الگوریتم انتشار باور [۲۱]، که به عنوان مجموع حاصل‌ضرب انتقال پیام نیز شناخته می‌شود. یک الگوریتم انتقال پیام برای انجام استنتاج در مدل‌های گرافی احتمالی همچون شبکه‌های بیزی و میدان تصادفی مارکوف است. به طور خلاصه این روش توزیع احتمال حاشیه‌ای مربوط به هر گره مشاهده نشده را مشروط بر هر گره مشاهده شده، محاسبه می‌کند.

تمرکز اصلی برروی متغیرهای تصادفی Xi می‌باشد که به صورت زیر در توزیع احتمال p({x}) وارد می‌شوند.

$$p(\{x\}) = \frac{1}{z}\prod_{(ij)} \omega_{(ij)}(xi, xj) \prod_i \emptyset_i(xi)$$ معادله ۲-۱

در معادله ۲-۱ $\omega_{(ij)}(xi, xj)$ و $\emptyset_i(xi)$ توابعی هستند که تابع احتمال کل را فاکتوربندی می‌کنند. که z یک ثابت نرمالیزاسیون است و حاصل‌ضرب روی (i,j)، شامل تمام نقاط همسایه می‌باشد.

در الگوریتم BP [۶۶، ۶۷] متغیرهایی مانند $m_{ij}(x_j)$ معرفی می‌کنیم، این متغیرهای جدید به طور ضمنی می‌تواند به عنوان یک "پیام" از متغیر تصادفی i به متغیر تصادفی j معرفی شود. متغیر mij(xj) یک بردار خواهد بود که ابعاد آن همان ابعاد xj (حالتهای که xj می‌تواند بگیرد) می‌باشد، اگر متغیر iام فکر می‌کند متغیر jام با احتمال p در حالت kام خودش است آنگاه مولفه‌های kام بردار

mij(xj) متناسب با احتمال p است.

در الگوریتم انتشار باور، بیلیف یا باور در نقطه i متناسب است حاصل‌ضرب تابع محلی و همه پیام‌های ورودی به نقطه i که در معادله ۲-۲ مشاهده می‌کنید.

معادله ۲-۲ $bi(xi)= \emptyset_i(xi) \prod_{j\in N(i)} mij(xi)$

که k یک ثابت نرمالیزاسیون است (جمع بیلیف‌ها باید یک باشد) و N(i) نمایش‌دهنده نقاط همسایه i است. پیام‌ها به طور خودسازگار و توسط معادله ۲-۳ آپدیت می‌شوند.

معادله ۲-۳ $mij(xj)=\sum_{x_i} \emptyset_i(x_i)\omega_{ij}(x_i,x_j) \prod_{\frac{k\epsilon N(i)}{j}} m_i(x_i)$

در این معادله N(i)/j مجموعه همه همسایه های i و j است. احتمال مارجینال دو نقطه‌ای i,j را معرفی می‌کنیم که i و j نقاط همسایه هستند. این احتمال مارجینال با مارجنالیزاسیون روی تابع توزیع احتمال کل برروی کل نقاط به جز i وj از معادله ۲-۴ بدست می‌آید.

معادله ۲-۴ $p_{ij}(x_i,x_j)= \sum_{z_i z_{ij}=(x_i x_j} p(\{z\})$

۲-۳ الگوریتم SVM

یکی از الگوریتم‌ها و روش‌های بسیار رایج در حوزه دسته‌بندی داده‌ها، الگوریتم SVM یا ماشین بردار پشتیبان است. الگوریتم‌های یادگیری ماشین هر یک کاربرد خاص خود را دارند و باید در جایگاه خاصشان مورد استفاده قرار گیرند. الگوریتم‌های یادگیری ماشین نیز بسیار متنوع هستند و هر یک برای حل نوع خاصی از مسائل قابل استفاده به حساب می‌آیند، که توانایی کار کردن با داده‌های دارای پیچیدگی بالا را ندارد.

ماشین بردار پشتیبان، یک الگوریتم یادگیری ماشین با ناظر است که هم برای مسائل طبقه‌بندی و هم مسائل رگرسیون قابل استفاده است. در ساخت مدل‌ها، از برخی الگوریتم‌های دیگر بسیار

قدرتمندتر عمل می‌کند. با این حال از آن بیشتر در مسائل طبقه‌بندی استفاده می‌شود. در الگوریتم SVM، هر نمونه داده را به عنوان یک نقطه در فضای n بعدی روی نمودار پراکندگی داده‌ها ترسیم کرده (n تعداد ویژگی‌هایی است که یک نمونه داده دارد.) و مقدار هر ویژگی مربوط به داده‌ها، یکی از مؤلفه‌های مختصات نقطه روی نمودار را مشخص می‌کند. سپس، با ترسیم یک ابر صفحه، داده‌های مختلف و متمایز از یکدیگر را دسته‌بندی می‌کند. به بیان ساده، بردارهای پشتیبان در واقع مختصات یک مشاهده منفرد هستند. ماشین بردار پشتیبان مرزی است که به بهترین شکل دسته‌های داده‌ها را از یکدیگر جدا می‌کند [۲۲].

یک قانون کلیدی وجود دارد که برای تعیین خط راست صحیح باید آن را همواره به یاد داشت. "خط راستی که دو دسته را به طور بهتری از یکدیگر جدا می‌کند، خطی است که باید انتخاب شود." محاسبه فاصله نزدیک‌ترین نقطه داده (که از هر دسته‌ای می‌تواند باشد.) از خط راست می‌تواند به انتخاب خط راست صحیح کمک کند. به این فاصله حاشیه گفته می‌شود. اگر خط راست حاشیه کمی داشته باشد، احتمال طبقه‌بندی نشدن برخی داده‌ها (Miss classification) وجود دارد.

یکی از ویژگی‌های ماشین بردار پشتیبان آن است که دورافتادگی‌ها را نادیده گرفته و تنها خط راستی را که بیشترین حاشیه را با نقاط داده دسته‌ها دارد انتخاب می‌کند. ماشین بردار پشتیبان از روشی که به آن ترفند هسته (کرنل) گفته می‌شود، استفاده می‌کند. در این روش در واقع توابعی وجود دارند که فضای ورودی بُعد پایین را دریافت کرده و آن را به فضای بُعد بالاتر تبدیل می‌کنند. این تبدیل، یک مسئله غیر قابل جداسازی را به مسئله قابل جداسازی مبدل می‌کند. به این توابع، تابع‌های هسته (کرنل) گفته می‌شود. تنظیم پارامترها برای الگوریتم ماشین بردار پشتیبان؛ یعنی «کرنل»، «گاما» و «C» به طور مؤثر باعث بهبود کارایی مدل می‌شود. گزینه‌های گوناگونی شامل «liner»، «rbf»، «poly» برای کرنل وجود دارند و در حالت پیش‌فرض کرنل روی پارامتر rbf قرار دارد، rbf و poly برای خط جداساز غیر راست مفید هستند. پیشنهاد می‌شود در صورت زیاد بودن تعداد ویژگی‌ها (۱۰۰۰<) از کرنل خطی استفاده شود زیرا داده‌ها در فضای بُعد بالا به صورت خطی قابل جداسازی

هستند. همچنین می‌توان از کرنل rbf استفاده کرد [۲۳].

گاما: ضـریب کرنل برای rbf، poly و sigmoid اسـت. هرچه مقدار گاما بیشـتر باشـد، الگوریتم تلاش می‌کند برازش را دقیقاً بر اسـاس مجموعه داده‌های تمرینی انجام دهد و این امر موجب تعمیم یافتن خطا و وقوع مشکل بیش برازش می‌شود.

C: پـارامتر جریمه C، برای جمله خطا اسـت. این پارامتر همچنین برقراری تعادل بین مرزهای تصمیم‌گیری هموار و طبقه‌بندی نقاط داده تمرینی را کنترل می‌کند.

برای داشتن ترکیبی مؤثر از این پارامترها و پیشگیری از بیش برازش پارامتر کرنل می‌تواند روی یکی از گزینه‌های linear، poly یا rbf تنظیم شود. مقدار گاما و C هم تنظیم می‌شود.

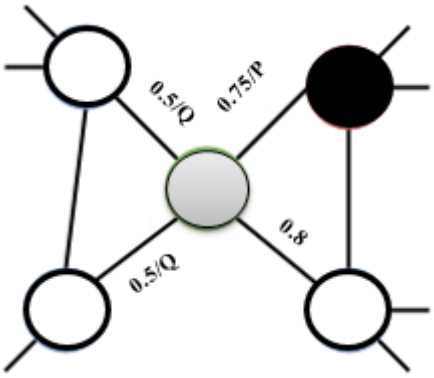
- مزایا
  - حاشیه جداسازی برای دسته‌های مختلف کاملاً واضح است.
  - در فضاهای با ابعاد بالاتر کارایی بیشتری دارد.
  - در شرایطی که تعداد ابعاد بیش از تعداد نمونه‌ها باشد نیز کار می‌کند.
  - یک زیر مجموعه از نقاط تمرینی را در تابع تصمیم‌گیری استفاده می‌کند که به آن‌ها بردارهای پشتیبان گفته می‌شود، بنابراین در مصرف حافظه نیز به صورت بهینه عمل می‌کند.
- معایب
  - هنگامی که مجموعه داده‌ها بسـیار بزرگ باشـد، عملکرد خوبی ندارد، زیرا نیازمند زمان آموزش بسیار زیاد است.
  - هنگامی که مجموعه داده نویز زیادی داشـته باشد، عملکرد خوبی ندارد و کلاس‌های هدف دچار همپوشانی می‌شوند.
  - ماشین بردار پشتیبان به طور مستقیم تخمین‌های احتمالاتی را فراهم نمی‌کند [۲۲-۲۵].

۲-۴ الگوریتم Node2vec

مشـــکل بســـیار بزرگ در گراف این اســـت که گراف در فضـــا و ابعاد زندگی نمی‌کند به عبارت دیگر نمی‌توان گره‌ها و یال‌ها را با عدد معرفی نموده و عملیات ریاضی را روی آنها اجرا نمود. به همین دلیل انجام بسـیاری از محاسـبات کلاسـیک بر روی گراف غیر ممکن می‌شود. لذا در بسیاری از کاربردهای یادگیری ماشـــین بر روی یک گراف، ابتدا گره‌ها و یال‌های گراف را تبدیل به ماتریس یا بردار نموده و سپس محاسبات لازم در الگوریتم رو اجرا می‌کنند [۲۶ و ۲۷].

در این پایان‌نامه از الگوریتم Node2vec وزن‌دار اسـتفاده شده است، که هر گره را باید به عددی بر روی یک فضـا تبدیل کند که بتوان بر روی آن محاسبات انجام داد. بنابراین ضرورت ایجاب می‌کند تا اطلاعات بدست آمده در گراف وزن‌دار برای استفاده در زمینه‌های بعدی به بردار تبدیل شود.

روش Node2vec [۲۷]، که توسط Aditya Grover و Jure Leskovec ارائه شده است، از دو پارامتر p و q استفاده می‌کند. این دو پارامتر تعین‌کننده رفتار الگوریتم هستند که مسیر قدم‌زنی تصادفی در محدوده نود اولیه باشد یا به سمت نودهای بیرونی حرکت بکند و در هر قدم دورتر بشود. در شکل ۲-۱ پیمایش از دایره سیاه شروع شده و به سمت نود طوسی حرکت کرده (حرکت اول به صورت تصادفی هست و نظمی ندارد.) بعد از رسیدن به نود طوسی حال می‌خواهیم به نود بعدی حرکت بکنیم و اینجاست که p,q خود را نشان می‌دهند. اگر احتمال رفتن به هر نود معادل وزن هر یال باشد، متغیر p برای یال‌هایی استفاده می‌شود که به نود قبلی برمی‌گردند و متغیر q برای یال‌هایی استفاده می‌شود که به نودهایی جدید و بدون تغییر می‌رسند. در تصویر دقت کنید که نود پایین سمت راست به دلیل همسایگی با نود سیاه بدون تغییر می‌ماند و بر عددی تقسیم نمی‌شود و همان وزن روی یال است. بعد از به دست آوردن اعداد یعنی تقسیم عدد هر یال بر متغیر موردنظر خود اعداد را به وسیله تابع Softmax نرمال کرده و عدد تصادفی انتخاب می‌شود و این پروسه برای تمام نودها و تعداد پیمایش‌های موردنظر انجام می‌شود [۲۷- ۲۹].



شکل ۲-۱.الگوریتم Node2Vec

به وسیله این الگوریتم اکنون می‌توان دیتاهای گراف را به وسیله بردار نشان داد و هر نود که دارای نشانی عددی هست می‌توان عملیات‌های کلاسیک یادگیری ماشین را بسیار راحت‌تر انجام داد. داده‌های تولید شده در گراف وزن‌دار که شامل گره مبدا و مقصد و وزن بین گره‌ها بود توسط این الگوریتم به بردار تبدیل می‌شود. در واقع به ازای هر نود الگوریتم Node2vec یک بردار در خروجی ایجاد می‌کند.

۲-۵ مقدمه‌ایی بر یادگیری عمیق

یادگیری عمیق شاخه‌ای از بحث یادگیری ماشین و هوش مصنوعی[۱۰] و مجموعه‌ای از الگوریتم‌هایی است که تلاش می‌کنند، مفاهیم انتزاعی سطح بالا را با استفاده از یادگیری در سطوح و لایه‌های مختلف مدل کنند. مطالعات بالینی نشان می‌دهند که ساختار مغز پستانداران از معماری شبکه‌های عصبی عمیق [۱۱]بهره می‌برد که در آن، مفاهیم انتزاعی در لایه‌های مختلف، به ترتیب از مفاهیم و ویژگی‌های ساده تا مفاهیم سطح بالا، در نواحی مختلف قشر مغز، پردازش می‌شوند. ایده یادگیری عمیق با الهام از ساختار طبیعی مغز انسان و به کمک امکانات و فن‌آوری‌های جدید، توانسته است در بسیاری از حوزه‌های مربوط به هوش مصنوعی و یادگیری ماشین، موفقیت‌های چشم‌گیری را کسب

[۱۰] Artificial intelligence

[۱۱] Deep neural networks

کند [۳۰ و ۳۱].

مهم‌ترین مزایای یادگیری عمیق عبارت‌اند از:

- یادگیری خودکار ویژگی‌ها
- یادگیری چند لایه ویژگی‌ها
- دقت بالا در نتایج
- قدرت تعمیم بالا و شناسایی داده های جدید
- پشتیبانی گسترده سخت افزاری و نرم افزاری
- پتانسیل ایجاد قابلیت ها و کاربردهای بیشتر در آینده

در سال‌های اخیر، یادگیری عمیق، تحول بزرگی را در یادگیری ماشین و هوش مصنوعی ایجاد کرده است. از سال ۲۰۱۲ تاکنون، تمامی رتبه‌های برتر چالش شناسایی بصری ImageNet، که به جام جهانی بینایی ماشین معروف است، از شبکه های عصبی عمیق استفاده کرده‌اند. همچنین، تمام روش‌های برتر در رقابت‌های دسته‌بندی تصاویر اعداد دست نویس MNIST با ۲۱ خطا در ۱۰،۰۰۰ تصویرو تصاویر طبیعی CLFAR با خطای کمتر از ۵۰٪ نیز به مدل‌های شبکه عصبی عمیق تعلق دارد. از سال ۲۰۱۲ به بعد، شرکت‌های بزرگ نرم‌افزاری و سخت‌افزاری مانند Google، NVIDIA و Microsoft نیز بخش مهمی از فعالیت‌های پژوهشی و تجاری خود را به یادگیری عمیق اختصاص داده‌اند.

با این که یادگیری عمیق در سال‌های ابتدایی توسعه خود قرار دارد، اما روند تحقیقات، مقالات و سرمایه‌گذاری‌های شرکت‌های بزرگ در این حوزه، نشان‌دهنده گسترش روز افزون کاربردهای یادگیری عمیق است. یادگیری عمیق تاکنون در کاربردهای گوناگون داده‌کاوی، پردازش تصویر و صدا، رباتیک و پزشکی مورد استفاده قرار گرفته است. طبق پیش‌بینی‌های مراکز علمی، در سال‌های آینده، بسیاری از تحقیقات، کاربردها و مشاغل موفق، به طور مستقیم یا غیرمستقیم از یادگیری عمیق بهره خواهند برد. دلیل اصلی نهفته در پس یادگیری عمیق این ایده است که هوش مصنوعی باید از مغز انسان الهام

بگیرد. این چشم‌انداز موجب سربرآوردن مفاهیم شبکه عصبی شده است. مغز انسان دارای میلیاردها نورون با ده‌ها هزار اتصال میان آن‌ها است. الگوریتم یادگیری عمیق، مغز انسان را در بسیاری از شرایط شبیه‌سازی می‌کند، از همین رو هم مغز و هم مدل‌های یادگیری عمیق دارای گستره وسیعی از واحدهای محاسباتی نورون‌ها هستند که به صورت منزوی فوق‌العاده هوشمند نیستند، اما هنگامی که با یکدیگر تعامل دارند هوشمند می‌شوند [۳۲- ۳۴].

مبنای اساسی شبکه‌های عصبی، نورون‌های مصنوعی هستند که از نورون‌های مغز انسان تقلید می‌کنند. این نورون‌ها، واحدهای محاسباتی ساده و قدرتمند دارای سیگنال‌های ورودی وزن‌داری محسوب می‌شوند که سیگنال خروجی را با استفاده از یک تابع فعال‌سازی تولید می‌کنند. نورون‌ها در چندین لایه در شبکه عصبی منتشر می‌شوند.

یادگیری عمیق دربرگیرنده شبکه‌های عصبی مصنوعی است که روی شبکه‌هایی مشابه با آنچه در مغز انسان وجود دارد مدل شده‌اند. با جابه‌جایی داده در این مِش مصنوعی[۱۲] هر لایه یک جنبه از داده‌ها را پردازش، دورافتادگی‌ها را فیلتر، موجودیت‌های مشابه را علامت‌گذاری و خروجی نهایی را تولید می‌کند [۳۳ و ۳۴]. در بخش بعد دو تا از مهم‌ترین و پرکاربردترین معماری یادگیری عمیق Dense و CNN[۱۳] معرفی شده است.

۲-۵-۱ لایه‌های Dense

برای حل مسئله دسته‌بندی در یادگیری عمیق از معماری شبکه عصبی تماماً متصل استفاده می‌شود. به همین منظور برای اضافه کردن لایه به مدل ترتیبی، از لایه‌های استاندارد Keras Dense استفاده می‌شود. به این لایه‌ها Dense می‌گوییم چون تمامی گره‌ها در یک لایه به تمام گره‌ها در لایه بعدی متصل می‌شوند [۳۵].

[۱۲] Artificial Mesh

[۱۳] Convolution neural network

وقتی از لایه‌های Dense استفاده می‌کنیم ابتدا باید تعداد نورون‌ها هر لایه را مشخص کنیم. تعداد این نورون‌ها یک ابر پارامتر است که مقدار مناسب برای آن را می‌توان با سعی و خطا به دست آورد. البته معمولاً هر چه تعداد نورون‌ها بیشتر شود صحت مدل نیز بیشتر می‌شود ولی از طرفی دیگر افزایش بیش از حد آن هم منجر به بیش برازش می‌شود [۳۶].

به علاوه، باید توجه کرد که پیش از این که مقادیر خروجی شبکه از یک لایه به لایه دیگر منتقل شوند باید از یک تابع فعال‌سازی غیرخطی عبور کنند چون در غیر این صورت شبکه قادر به یادگیری الگوهای غیرخطی موجود در داده‌ها نخواهد بود. یکی از توابع فعال‌سازی مشهور که از آن استفاده می‌شود، تابع فعال‌سازی Rlue است که اگر مقدار ورودی آن کوچکتر از، صفر باشد (منفی باشد) عدد صفر را بازمی‌گرداند و اگر بزرگتر از صفر باشد خود آن مقدار را باز می‌گرداند [۳۵ و ۳۶].

در نهایت، در لایه اول هم باید ابعاد داده‌های ورودی‌ها را مشخص کنیم. لایه آخر هم معمولاً بسته به نوع مسئله تعداد نورون‌های متفاوتی دارد. فراتر از توابع فعال‌سازی، تعداد نورون‌ها و ابعاد ورودی، کراس از اعمال تغییرات بیشتر دیگری مانند تعریف کردن تابع برای وزن گره‌ها و توابع تنظیم‌سازی برای وزن‌های گره در هر لایه پشتیبانی می‌کند. اما یک اصل نانوشته در کراس وجود دارد که بهتر است از همان گزینه‌های پیش‌فرض استفاده کرد چون این پیش‌فرض‌ها بر اساس نتایج بهترین رویه‌های ممکن تعیین شده‌اند. برای این که شبکه بر روی دیتاست آموزش ببیند باید سه چیز مشخص شود.

تابع زیان: تابع زیان عملکرد شبکه را بر روی داده‌های آموزش اندازه‌گیری می‌کند تا مشخص شود که آیا جهتی که طی می‌شود جهت درستی است یا خیر.

الگوریتم بهینه‌ساز: الگوریتم بهینه‌سازی بر اساس تابع زیان و داده‌ها جهتی که وزن‌های شبکه باید به روزرسانی شوند تا شبکه به بهینگی برسد را مشخص می‌کند. از الگوریتم بهینه‌ساز Adam

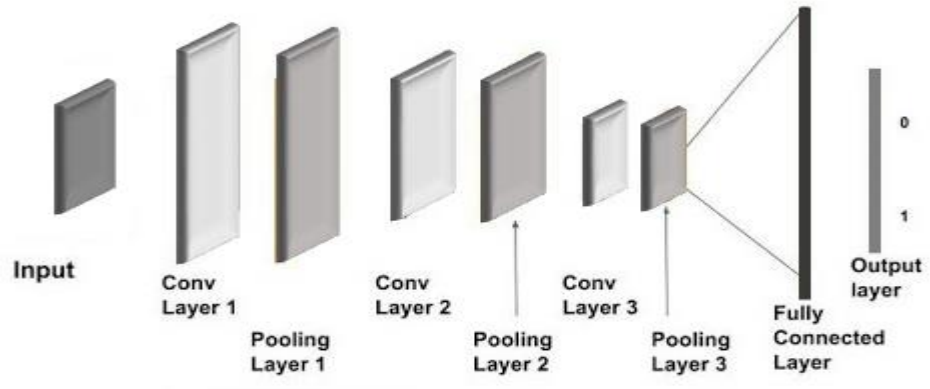
(گرادیان نزولی تصادفی)[۱۴] استفاده می‌شود.

معیار ارزیابی: معمولاً برای مسائل دسته‌بندی از معیار صحت استفاده می‌شود که نسبت مشاهدات درست پیش‌بینی شده به کل داده‌ها است.

گام بعدی اضافه‌کردن لایه‌هاست. اولین چیزی که باید مشخص شود ابعاد داده‌های ورودی و تعداد نورون‌های لایه‌ها است. همچنین در هر لایه تابع فعال‌سازی مشخص می‌شود و برای لایه آخر شبکه از تابع فعال‌سازی Softmax استفاده شده است.

۲-۵-۲ شبکه عصبی کانولوشن CNN

الگوریتم‌های یادگیری عمیق، زیر مجموعه‌ای از الگوریتم‌های یادگیری ماشین هستند که هدف آنها کشف چندین سطح از بازنمودهای (نمایش)های توزیع شده، از داده ورودی است. اخیرا الگوریتم‌های یادگیری عمیق زیادی برای حل مسائل هوش مصنوعی سنتی ارائه شده‌اند.



شکل ۲-۲. معماری کانولوشن

شبکه‌های عصبی کانولوشن (CNN) یکی از مهم‌ترین روش‌های یادگیری عمیق هستند که در آنها چندین لایه با روشی قدرتمند آموزش می‌بینند، این روش بسیار کارآمد بوده و یکی از رایج‌ترین

[۱۴] Random Descending Gradient

روش‌ها در کاربردهای مختلف بینایی کامپیوتر است. بطور کلی یک شبکه کانولوشنی از سه لایه اصلی تشکیل می‌شود که عبارتند از لایه کانولوشن، لایه pooling و لایه تماما متصل، لایه‌های مختلف وظایف مختلفی را انجام می‌دهند. در شکل ۲-۲ یک معماری کلی از شبکه عصبی کانولوشن برای دسته‌بندی بصورت لایه به لایه نمایش داده شده است. در هر شبکه عصبی کانولوشن دو مرحله برای آموزش وجود دارد. مرحله feed-forward و مرحله backpropagation یا پس‌انتشار. در مرحله اول ورودی به شبکه تغذیه می‌شود و این عمل چیزی جز ضرب نقطه‌ای بین ورودی و پارامترهای هر نورون و نهایتا اعمال عملیات کانولوشن در هر لایه نیست. سپس خروجی شبکه محاسبه می‌شود. در این جا به منظور تنظیم پارامترهای شبکه و یا به عبارت دیگر همان آموزش شبکه، از نتیجه خروجی جهت محاسبه میزان خطای شبکه استفاده می‌شود. برای اینکار خروجی شبکه را با استفاده از یک تابع خطا loss function با پاسخ صحیح مقایسه کرده و این‌طور میزان خطا محاسبه می‌شود. در مرحله بعدی بر اساس میزان خطای محاسبه شده مرحله backpropagation آغاز می‌شود. در این مرحله گرادیان هر پارامتر محاسبه می‌شود و تمامی پارامترها با توجه به تاثیری که بر خطای ایجاد شده در شبکه دارند تغییر پیدا می‌کنند. بعد از به‌روزرسانی پارامترها مرحله بعدی feed-forward شروع می‌شود. بعد از تکرار تعداد مناسبی از این مراحل آموزش شبکه پایان می‌آید [۳۷- ۳۹]. در ادامه لایه‌های کانولوشنی معرفی می‌شود [۴۰ و ۴۱].

۲-۵-۲-۱ لایه کانولوشن

لایه کانولوشن، این لایه خروجی نورون‌هایی که به نواحی محلی در ورودی متصل هستند را محاسبه می‌کند. عمل محاسبه هم از طریق ضرب نقطه‌ای بین وزن‌های هر نورون و ناحیه‌ای که آنها به آن (توده فعال‌سازی ورودی) متصل هستند صورت می‌گیرد.

۲-۵-۲-۲ لایه pooling

قرار دادن یک لایه Pooling بین چندین لایه کانولوشنی پشت سر هم در یک معماری کانولوشن امری

رایج است. کارکرد این لایه کاهش اندازه مکانی ورودی به جهت کاهش تعداد پارامترها و محاسبات در داخل شبکه و بنابراین کنترل overfiting است. لایه Pooling بصورت مستقل بر روی هر برش عمقی از توده ورودی عمـل کرده و آن‌را بـا اســـتفاده از عملیات Max از لحاظ مکانی تغییر اندازه می‌دهد. رایج‌ترین شکل استفاده از این لایه به صورت استفاده این لایه با فیلترهایی با اندازه گام ۲ است.

۲-۵-۲-۳ لایه تماما متصل (Fully Connected Layer)

نورون‌هایی که در یک لایه تماما متصـل قرار دارند با تمام نورون‌های موجود در لایه قبلی ارتباط دارد دقیقا همانند همان چیزی که در شـبکه های عصـبی معمولی دیده می‌شود. بنابراین می‌توان از ضرب ماتریسـی و سـپس جمع نتیجه حاصله با بایاس جهت محاسبه خروجی تمامی نورون‌ها (فعالسازی‌ها) در یک لحظه اسـتفاده کرد. بطور خلاصه تمامی قوانین مطرحی در شبکه های عصبی معمولی در این بخش صادق است.

۲-۶ مروری برکارهای پیشین

روش‌های زیادی برای دســته‌بندی دامنه‌های فیشـینگ از غیر فیشــنگ وجود دارد که بســته به نوع کاربرد، متنوع هسـتند. به طور کلی روش‌های پیشـین به دو دسته سنتی و مبتنی بر یادگیری عمیق تقســیم می‌شــوند. در بخش ۲-۷ به معرفی روش‌های ســنتی و در بخش ۲-۸ روش‌های مبتنی بر یادگیری عمیق معرفی می‌شــود. اما اســتفاده از این روش‌ها در تعدادی از مقالات، دقت پایینی را به همراه داشته است. تعدادی از این مقالات را شرح می‌دهیم.

۲-۷ روش‌های سنتی

در این قسـمت روش‌هایی معرفی می‌شـود که در آن از طبقه‌بندی‌کننده‌های یادگیری ماشین و سایر الگوریتم‌ها جهت پیش‌پردازش و استخراج ویژگی‌های مهم برای طبقه‌بندی استفاده شده است.

مطالعات موجود در مورد شناسایی دامنه مخرب را می‌توان به طور گسترده‌ای بر اساس تکنیک‌های اساسی به سه دسته طبقه‌بندی کرد:

- ✓ راه‌حل‌های مبتنی بر طبقه‌بندی
- ✓ راه‌حل‌های مبتنی بر خوشه‌بندی
- ✓ تکنیک‌های مبتنی بر نمودار

در ادامه به تفصیل هریک از روش‌ها معرفی شده است.

۲-۷-۱ راه‌حل‌های مبتنی بر طبقه‌بندی

راه‌حل‌های مبتنی بر طبقه‌بندی، ابتدا به دانش دامنه از کارشناسان شبکه و امنیت برای استخراج ویژگی‌های آماری مربوط به دامنه‌های خوش‌خیم و مخرب متکی هستند. بر اساس چنین ویژگی‌هایی و مجموعه داده‌های دارای برچسب با دامنه‌های مخرب و خوش‌خیم شناخته شده، این راه‌حل‌ها از الگوریتم‌های طبقه‌بندی نظارت شده مانند ماشین‌های بردار پشتیبان و درخت تصمیم‌گیری برای ساخت طبقه‌بندی باینری برای تمایز دامنه‌های خوش‌خیم و بدخیم استفاده می‌کنند [۴۲- ۴۴]. از آنجا که DNS به شدت توسط مهاجمان مورد سوء استفاده قرار گرفته است، در مرجع [۴۳] کار قابل توجهی انجام شده است که به دنبال توسعه مکانیسم‌های شناسایی بدافزار مبتنی بر DNS است. این سیستم پیشنهادی EXPOSURE نام دارد.

هدف EXPOSURE شناسایی دامنه‌های مخربی است که به عنوان بخشی از عملیات مخرب در اینترنت استفاده می‌شود. سیستم EXPOSURE با استفاده از ۱۵ ویژگی منحصر به فرد که در ۴ دسته، گروه‌بندی شده‌اند، برای شناسایی چنین دامنه‌هایی در زمان واقعی طراحی شده است. سیستم EXPOSURE برای انجام آنالیز Passive DNS به منظور شناسایی دامنه‌هایی که در فعالیت‌های

مخرب درگیر هستند؛ از جمله میزبانی صفحات وب فیشینگ، سرورهای C&C[۱۵]، باتنت، dropzones و غیره طراحی شده است. این سیستم از تکنیک‌های یادگیری ماشین برای ساختن قوانینی برای شناسایی استفاده می‌کند؛ که به طور موثر قادر به تشخیص رفتار DNS سرویس‌های مخرب از سرویس خوش‌خیم است. به طور خاص، از چهار مجموعه ویژگی‌هایی که از رکوردهای DNS استخراج می‌شوند استفاده می‌کند. برای ارزیابی سیستم یک تحلیل منفعل از ترافیک DNS که توسط کاربران واقعی ایجاد شده است را انجام می‌دهد. در واقع یک آزمایش کنترل شده با یک مجموعه داده بزرگ و واقعی در دنیای واقعی متشکل از میلیاردها درخواست DNS انجام می‌شود.

پایگاه داده به طور خاص، از دامنه‌های مخرب و آدرس‌های مشکوک استخراج شده است. همچنین از لیست دامنه‌هایی که توسط الگوریتم تولید دامنه تولید می‌شود و مجموعه‌های دامنه خوش‌خیم از اطلاعات به دست آمده از سایت Alexa و تعدادی از سرورها تشکیل شده است [۴۵-۵۱].

برای ارزیابی دقت دسته‌بند درخت تصمیم j48، مجموعه آموزشی با 10-Fold Cross Validation که در آن ۶۶٪ از مجموعه آموزشی برای آموزش استفاده می‌شود و بقیه برای تست، با استفاده از سیستم EXPOSURE، ۱۷،۶۸۶ از ۳۰۰۰۰۰ دامنه، حدود ۵,۹٪ مخرب بودند. درصد تفکیک و ارزیابی اعتبارسنجی بر روی مجموعه آموزشی نشان می‌دهد که میزان تشخیص طبقه‌بندی کننده در حدود ۹۸٪ است. به منظور ارزیابی عملکرد سیستم که آیا قادر به شناسایی دامنه‌های مخرب قبل از منابع دیگر است، زمان تشخیص EXPOSURE با ۱۱ لیست سیاه دیگر برای یک دوره شش ماهه مقایسه شد (ژوئیه ۲۰۱۱ - دسامبر ۲۰۱۱). تعداد دامنه‌های شناسایی شده توسط EXPOSURE بیش از ۵۰٪ می‌باشد.

[۱۵] Control &Command

یک راه حل امیدوارکننده و هوشمندانه ضد فیشینگ، سیستم تشخیص مبتنی بر یادگیری ماشین به نام مجموعه انتخاب ترکیبی هیبرید (HEFS)[۱۶] ارائه شده است [۵۲]. این تکنیک از مقدار زیادی از شاخص‌ها که به عنوان ویژگی شناخته می‌شود و به آن قابلیت تشخیص وب‌سایت‌های جدید فیشینگ را می‌دهد تشکیل یافته است. در مرحله اول HEFS، یک الگوریتم گرادیان تابع توزیع تجمعی برای تولید زیر مجموعه‌های ویژگی اصلی استفاده می‌کند، سپس در یک مجموعه اغتشاش داده‌ها تخصیص داده می‌شوند تا زیر مجموعه‌های ویژگی ثانویه راتولید کنند. مرحله دوم مجموعه‌ایی از ویژگی‌های پایه را از زیر مجموعه‌های ویژگی ثانویه با استفاده از یک مجموعه اختلال تابع استخراج می‌کند. برای استخراج ویژگی از الگوریتم‌های طبقه‌بندی SVM، Naïve Bayes، C4.5، Random Forest، JRip و PART استفاده شده است.

به منظور جمع‌آوری ویژگی‌های مورد نیاز، وب‌سایت‌ها در دو جلسه جداگانه یعنی بین ژانویه تا مه ۲۰۱۵ و بین ماه مه تا ژوئن ۲۰۱۷ جمع‌آوری شد. به طور خاص، ۵۰۰۰ صفحه فیشینگ بر اساس آدرس‌های PhishTank 2 و OpenPhis 3، ۵۰۰۰ صفحه وب قانونی دیگر بر اساس آدرس‌های Alexa 4 و Common Crawl 5 انتخاب شده است.

در اجرای آزمایشات، از یک روش طبقه‌بندی مشابه تقسیم Test-Train استفاده شده است. برای هر بخش، ۷۰٪ از داده‌ها برای آموزش استفاده می‌شود، در حالی که ۳۰٪ برای تست در نظر گرفته می‌شود. دقت در تمام بخش‌ها به طور متوسط برای تولید یک امتیاز واحد انجام می‌شود. براساس نتایج بدست آمده، در میان طبقه‌بندی‌کننده‌ها، به نظر می‌رسد که جنگل تصادفی با امتیازدهی بالاتر از ۹۵٪ از همه طبقه‌بندی‌های باقیمانده بهتر است. نتایج ارزیابی نشان می‌دهد که ویژگی‌های پایه با استفاده از جنگل تصادفی از کلیه ویژگی‌های کامل که از C4.5، JRip، PART، SVM، Naïve Bayes

[۱۶] Hybird Ensemble Feature Selection

استفاده می‌کنند، بهتر است. این ثابت می‌کند که ویژگی‌های بدست‌آمده، هنگامی که با طبقه‌بندی RF همراه هستند، در تمایز بین وب‌سایت‌های فیشینگ و قانونی بسیار مؤثر است.

در مقاله [۵۳]، روش‌های شناسایی فیشینگ از جمله ویژگیهای ترکیبی، طبقه‌بندی‌کننده‌های یادگیری ماشین و روش‌های انتخاب ویژگی تحقیقی برجسته را بررسی می‌کند، محدودیت‌های آن‌ها را مشخص می‌کند و تاکید می‌کند که چگونه می‌توان آن‌ها را بهبود داد تا عملکرد تشخیص تشدید شود. این بررسی ویژگی‌های اضافی را برای ارتقا جنبه‌های خاص روند تحقیق حاضر با امید کمک به محققان در مورد چگونگی توسعه روش‌های تشخیص و دستیابی به بهترین نتایج کیفیت انتخاب ویژگی فراهم می‌کند.

هدف از این تحقیق برجسته کردن وضعیت فعلی انتخاب ویژگی در تحقیقات مربوطه و آشکارکردن علیت بین محدودیت‌های برجسته و اجرای تشخیص فیشینگ می‌باشد. برای انجام این کار، مروری گسترده بر تحقیقات و ارزیابی انتقادی در این مقاله انجام و معرفی شده است. بر اساس بینش‌های به دست آمده از بررسی انجام شده، این مطالعه تأکید می‌کند که انتخاب بهترین ویژگی‌ها یا بهترین زیرمجموعه ویژگی برای تشخیص فیشینگ می‌تواند از ویژگی‌های خاص به طور قابل توجهی الهام بگیرد. بنابراین ویژگی‌های اضافی را توصیه می‌کند تا به عنوان معیارهای بهینه برای بهبود انتخاب ویژگی و صلاحیت نتیجه انتخاب ویژگی از چندین جنبه استفاده شود.

در مرجع [۵۴]، براساس تحقیقات عمیق از دستاوردهای تشخیص دامنه مخرب، یک چارچوب جدید که مزایای سه روش تشخیص دامنه مخرب مختلف را ترکیب می‌کند، به نام OMDD[۱۷] پیشنهاد شده است. دامنه مخرب فعال، غیر فعال و مبتنی بر Domain-IP است. در روش موجود در حوزه شناسایی دامنه مخرب، چارچوب OMDD یک معماری موتور چندگانه است که مزایای سه رویکرد تشخیص دامنه مخرب را ترکیب می‌کند. در نهایت، امتیاز هر موتور ردیابی، ادغام شده تا امتیاز اعتبار

[۱۷] Online Malicious Domain Detection

نهـایی نـام دامنه را ارایه دهد که می‌تواند به طور عینی درجه مخرب نام دامنه را توصـــیف کند. این چارچوب برای استقرار همگرایی در یک شبکه شرکت بزرگ تجاری مناسب است.

اطلاعات دامنه مخرب برای موتور تشـخیص طبقه‌بندی مبتنی بر ویژگی، اسـتخراج می‌شـود. به طور تصـادفی سـوابق ۱۴ روز DNS برای آموزش مدل طبقه‌بند انتخاب می‌شـود که حاوی داده‌های DNS مربوط به نام‌های دامنه مخرب شـناخته شـده اسـت که برچسـب‌گذاری شـده‌اند. برای موتور تشـخیص مبتنی بر تجزیه و تحلیل رابطه Domain-IP، از Active DNS استفاده می‌شود و به ترتیب داده های DNS مربوط به لیست سفید و سیاه جستجو می‌شود.

سـپس یک سیستم براساس چارچوب OMDD اجرا می‌شود. ترافیک DNS تصادفی انتخاب و به ترتیب ۷ روز برای تشخیص آنلاین مورد بررسی قرار می‌گیرد. نتایج تجربی سه موتور تشخیص ارزیابی و در نهایت نتایج آن‌ها با نتایج نهایی OMDD مقایسه می‌شود. نتایج آزمایشی سه موتور شناسایی، از جمله موتور تشـــخیص مبتنی بر طبقه‌بندی، موتور تشـــخیص مبتنی بر قانون و موتور شـــناســـایی Domain-IP با هم مقایسه می‌شود. با توجه به نتایج بدست آمده، چارچوب OMDD از سایر روش‌های تشخیص عملکرد بهتری در شناسایی دامنه‌های مخرب داشته است. همچنین با توجه به نتایج تجربی، OMDD کارایی و دقت بالاتری نسـبت به روش‌های موجود انجام می‌دهد، چرا که میزان دقت بدست آمده در OMDD حدود ۹۲,۹۰٪ است.

در مقاله [۵۵]، یک روش مبتنی بر یادگیری ماشـــین برای شـــناســـایی نام‌های دامنه مخرب با اسـتفاده از ELM[۱۸] پیشـنهاد می‌شـود. سـیسـتم ELM یک شبکه عصبی مدرن با دقت بالا و سرعت یادگیری سـریع اسـت. از ELM برای طبقه‌بندی نام دامنه بر اساس ویژگی‌های استخراج‌شده از منابع متعدد اسـتفاده می‌شـود. ارتباطات C&C به DNS که اصـلی‌ترین بخش در پیاده‌سـازی بدافزارها و استخراج اطلاعات است، در بیشتر موارد نیاز دارد. بنابراین، تجزیه و تحلیل DNS یک روش تشخیص

[۱۸] Extreme Learning Machine

مهم در شـــناســـایی اســـت. همچنین از چهار طبقه‌بندی‌کننده متفاوت LR (رگرســـیون لجســـتیک)، CART، BPNN (شـبکه عصـبی برگشتی) و SVM استفاده شده است. در ELM پیشنهادی n تعداد نرون‌هـا در یک مجموعه آموزشـــی با k طبقه متمایز اســـت که در آن هر نرون با m ویژگی متفاوت مشخص می‌شود.

در مرحله بعد برای ارزیابی عملکرد روش تشخیص، دامنه‌های مخرب به عنوان نمونه‌های مثبت و دامنه‌های غیرمخرب به عنوان منفی قرار داده می‌شود. سپس نرخ تشخیص و دقت به عنوان معیارهای اصـلی تعریف می‌شـود. میانگین نتایج در همه اجراها به عنوان نتیجه دریافت می‌شود. برای زمانی که تعداد گره‌های مخفی به ترتیب ۵۰ و ۵۰۰ است، نرخ تشخیص ۹۵,۹۰٪ و دقت ۹۶,۲۹٪ بدست آمده است. زمان آموزش و تست با تعداد گره‌های مخفی رشد می‌کند، در حالی که نرخ تشخیص و دقت به طور پایدار بالا باقی می‌ماند. این نشـــان می‌دهد که می‌توان تعداد نرون‌ها را برای ســـرعت یادگیری سـریع‌تر، کاهش داد. اما نرخ تشخیص و دقت همچنان بالا باقی خواهد ماند. اگرچه کارایی محاسباتی LR و CART عالی است، ولی دقت قابل‌قبول نیست. این درحالی است که BPNN و SVM می‌توانند به دقت بالا دســـت یابند. با توجه به تجزیه و تحلیل بیشـــتر آزمایش تطبیقی، ELM نه تنها عملکرد خوبی در تشخیص دامنه مخرب دارد بلکه یک مزیت واضح از سرعت یادگیری سریع را نشان می‌دهد. همچنین ELM بهترین عملکرد را زمانی ایجاد می‌کند که تمام ویژگی‌ها با هم ترکیب شوند.

داده‌های استفاده شده در این مقاله از پرس‌وجو DNS دریافت‌شده [۵۶]، توسط پنج سرور DNS در شـبکه و اطلاعات دانشـگاه Jiaotong شـانگ‌های جمع‌آوری شـد که به طور متوسـط در هر ثانیه ۳۰۰۰ پرس‌وجو پردازش می‌شود، در مجموع ۹۳۳۵۲۷۰ پرس‌وجو با نظارت بر ترافیک برای یک هفته به دست آمده است. برای کاهش مقیاس داده‌های ترافیک، یک فرآیند فیلترکردن برای ایجاد داده‌های قابل اطمینان و قابل مدیریت برای دامنه‌های بی‌خطر و مخرب ایجاد می‌شود.

۲-۷-۲ راه‌حل‌های مبتنی برخوشه‌بندی

راه‌حل‌های مبتنی بر خوشــه‌بندی [۵۷- ۵۹]، شــباهت ویژگی را بین دامنه‌ها اســتخراج می‌کنند و از مقیاس شــباهت برای گروه‌بندی دامنه‌ها در خوشــه‌های مشــخص استفاده می‌کنند. در [۵۸]، شباهت بین دامنه‌های غیر موجود را در ســطح معتبر محاســبه می‌کند و ســپس خوشه‌بندی سلسله مراتبی را برای شناسایی گروه‌های دامنه تولید شده توسط الگوریتم‌های تولید دامنه اتخاذ می‌کند. به طور مشابه درمرجع [۵۷]، همبستگی زمانی بین کوئری داده‌های DNS را بررسی می‌کند و سپس از الگوریتم‌های خوشــه‌بندی XMeans با چند دامنه مخرب برای شــناســایی تعداد زیادی از گروه‌های دامنه مخرب استفاده می‌کند.

در مرجع [۶۰]، یک ســیســتم شــناسایی دامنه هوشمند به نام HinDom[۱۹] پیشنهاد شده است. HinDom به جای تکیه بر ویژگی‌های انتخاب‌شــده دســتی، DNS scene را به عنوان یک شــبکه اطلاعات ناهمگن ایجاد می‌کند که شامل مشتری، دامنه، آدرس IP و روابط مختلف آنها است. علاوه بر این، روش طبقه‌بندی انتقال مبتنی بر مســیر HinDom را قادر می‌ســازد، دامنه‌های مخرب را تنها با بخش کوچکی از نمونه برچســب تشخیص دهد. سپس از الگوریتم KMeans برای خوشه‌بندی بردارها به دسته k و تبدیل نتیجه خوشه‌بندی به ماتریس s استفاده می‌کند.

سه منبع داده اصلی که جمع‌آوری شده است عبارتند از رویدادنامه‌های DNS[۲۰]، ترافیک DNS و مجموعــه داده‌هــای Passive DNS [۶۱- ۶۳]. در مرحلــه بعــد، آزمایش‌های جامعی برای ارزیابی HinDom انجام می‌شــود که در دو شــبکه CERENT 2 و TUNET در دنیای واقعی ارزیابی می‌شود. نتایج آزمایش نشـــان می‌دهد که کارایی HinDom دارای ACC با مقدار ۰,۹۹۶۵و F1-score برابر ۰,۹۹۰۲ با ۹۰٪ نمونه از پیش برچســب دار اســت و هنوز هم می تواند دامنه های مخرب را با ACC ۰,۹۶۲۶ و F1-score ۰,۹۱۱۶ تشخیص دهد.

[۱۹] Heterogeneous Information Netword

[۲۰] Logs

۲-۷-۳ راه‌حل‌های مبتنی بر نمودار

راه‌حل‌های مبتنی بر نمودار [۶۴ و ۱۴]، ابتدا رفتار دامنه‌ها را از طریق گراف‌های مختلف مدل می‌کنند، سپس از روش‌های استخراج نمودار برای شناسایی دامنه‌های مخرب استفاده می‌کنند. در [۶۴]، نمودارهایی برای شکست DNS ایجاد می‌کند تا از تعامل بین میزبان نهایی و نام دامنه استفاده کند و با موفقیت گروه‌های منسجمی را از نمودارهای DNS با یک روش فاکتورسازی ماتریس سه گانه غیرمنفی آماری استخراج کند، در حالی که در [۱۴]، از الگوریتم استنباط انتشار باور (BP) [۶۵ و ۶۶] نمودار ارتباط دامنه میزبان را برای شناسایی دامنه‌های مخرب می‌سازد. انتشار باور که به عنوان مجموع حاصل‌ضرب انتقال پیام نیز شناخته می‌شود. یک الگوریتم انتقال پیام برای انجام استنتاج در مدل‌های گرافی احتمالی همچون شبکه‌های بیزی و میدان تصادفی مارکوف است. به طور خلاصه این روش توزیع احتمال حاشیه‌ای مربوط به هر گره مشاهده نشده را مشروط بر هر گره مشاهده شده، محاسبه می‌کند.

در مقاله [۶۷]، نمودارهای دو بخشی را برای ثبت تعامل بین میزبان‌های نهایی و دامنه‌ها، شناسایی آدرس‌های IP مربوط به دامنه‌ها و توصیف الگوهای سری زمانی Query DNS برای دامنه‌ها، معرفی کرده و پیش‌بینی‌های یک حالته از این نمودارهای دو طرفه را برای مدل‌سازی شباهت‌های رفتاری، IP ساختاری و زمانی بین دامنه‌ها بررسی می‌کند، همچنین از تکنیک تعبیه نمودار استفاده می‌شود تا به طور خودکار نمایش ویژگی پویا را برای دامنه‌های دارای برچسب یاد بگیرید و یک الگوریتم طبقه‌بندی مبتنی بر SVM را برای پیش‌بینی دامنه‌های مخرب یا خوش‌خیم توسعه می‌دهد.

به منظور صحت، الگوریتم پیشنهادی را با بردار ویژگی ترکیبی و همچنین هر بردار ویژگی جداگانه برای تشخیص مخرب با K-Fold Cross Validation اندازه‌گیری می‌کند. برای هر یک از این گروه‌های k، آن را به عنوان یک گروه نگه‌دارنده به منظور آزمایش مجموعه داده‌های تست در نظر می‌گیرد و از گروه‌های k-1 باقیمانده به عنوان مجموعه داده آموزشی برای توسعه مدل طبقه‌بندی SVM استفاده می‌کند. درنهایت، کیفیت مدل آموزش دیده را با گروه احتمالی، ارزیابی می‌کند. سطح

زیر منحنی، ۰,۹۴ است که نشان می‌دهد سیستم پیشنهادی قادر به شناسایی موثر و دقیق دامنه‌های مخرب است. به طور خاص، سه طبقه‌بندی‌کننده باینری مختلف مبتنی بر SVM بر اساس سه بردار ویژگی آموزش می‌دهد. نتایج نشان می دهد، سطح زیر منحنی برای طبقه‌بندی‌کننده SVM، به ترتیب ۰,۸۹، ۰,۸۳ و ۰,۶۵ می‌باشد.

مجموعه داده‌های مورد استفاده در این مقاله، یک لیست سیاه دامنه‌های مخرب و لیست سفید دامنه‌های خوش‌خیم از یک شرکت بزرگ امنیتی اینترنتی است. مجموعه داده‌های دارای برچسب نهایی شامل بیش از ۱۰،۰۰۰ دامنه است. در بین این دامنه‌ها ، ۳۰٪ دامنه‌های مخرب تأیید شده و ۷۰٪ باقیمانده دامنه خوش‌خیم هستند.

در مرجع [۶۸]، برای مقابله با حملات سایبری با استفاده DNS، سیستمی را برای کشف نام دامنه‌های مخرب پیشنهاد می‌کند، ایده اصلی این سیستم استخراج الگوهای تغییرات زمان (TVPs)[۲۱] نام دامنه است. سیستم DOMAIN PROFILER الگوهای تغییرات زمانی نام دامنه‌ها را شناسایی کرده و نام دامنه‌های مخرب را پیش‌بینی می‌کند. در حالت ایده‌آل، برای رسیدگی کامل به مسئله اصلی با لیست‌های سیاه نام دامنه، باید هر نام دامنه تازه ثبت شده و بروز شده را در زمان واقعی مشاهده و پیگیری کند و قضاوت شود که آیا نام دامنه در زیرساخت‌های هر مهاجمی دخیل است یا خیر. بر اساس ایده فوق، سیستم ایجاد شده به طور فعال رویدادنامه‌های DNS را جمع‌آوری می‌کند، الگوهای تغییرات زمانی آنها را تجزیه و تحلیل می‌کند و پیش‌بینی می‌شود که آیا از یک نام دامنه خاص برای یک هدف مخرب استفاده خواهد شد یا خیر. این سیستم با استفاده از مجموعه داده‌های واقعی از جمله تعداد گسترده‌ای از نام دامنه مورد ارزیابی قرار می‌گیرد. اولین مجموعه داده نام دامنه که از مجموعه‌های آموزشی و آزمایشی تشکیل شده است، مجموعه آموزش داده‌ها برای ایجاد یک مدل یادگیری در یک جنگل تصادفی است. مجموعه داده Sandbox-Malware، Legitimate-Ale،

[۲۱] Temporal Variation Patterns

Pro-C & C و ProPhishing اســت. مجموعه داده دوم یک پایگاه داده، لیست نام دامنه بود که برای شناسایی الگوهای تغییرات زمان استفاده می‌شود. سایت‌های برتر Alexa را به عنوان یک لیست قانونی و HpHostها را به عنوان یک لیســت مخرب انتخاب کرده اســت. مجموعه داده ســوم، رویدادنامه‌های مربوط به Historical DNS بود که شامل مجموعه‌های سری زمانی نام دامنه و نگاشت‌های آدرس IP بود. برای ارزیابی مدل پیشنهادی از معیارهای ارزیابی TP، FP، FN، TN، TPR و TNR استفاده شده است. نتایج نشان می‌دهد که سیستم پیشنهادی، نام دامنه مخرب را ۲۲۰ روز قبل با نرخ مثبت واقعی ۰,۹۸۵ در بهترین حالت پیش‌بینی کرده اســت. همچنین Recall، TNR به ترتیب مســاوی ۰,۹۷۵ و ۰,۹۹۱، دقت ۰,۹۹۰ و مقدار F-score برابر ۰,۹۸۳ می‌باشــد. از نتایج بدســت می‌آید که ایده اصــلی مبتنی بر اســتفاده از الگوهای تغییرات زمانی برای بهبود TPR و TNR درســت در نظر گرفته شــده است.

یک چالش کلیدی تعیین راه‌حل دفاع بهینه برای هر نام دامنه مخرب اســت. مقاله [۶۹]، قصــد دارد چگونگی اتخاذ اقدامات متقابل در برابر چنین نام‌های دامنه بدخواهانه را نشــان دهد. که بر روی روابط بین مقوله‌های نام دامنه و راه‌حل‌های دفاعی عملی تمرکز کرده تا مشــخص کند که چگونه می‌توان از اطلاعات نام دامنه مخرب شناسایی‌شده استفاده کرد. درواقع، یک خط لوله تجزیه و تحلیل یکپارچه و عینی یا قابل مشــاهده یا علمی را با ترکیب راه‌حل‌های دفاعی موجود برای تحقق دفاع‌های عملی و بهینه در برابر نام دامنه مخرب امروز، طراحی و پیاده‌سازی می‌کند. در نهایت، خط لوله تحلیل و اطلاعات دفاعی خروجی را با استفاده از یک مجموعه داده بزرگ و واقعی ارزیابی می‌کند تا کارایی و اعتبار رویکرد پیشــنهادی را نشان دهد. مفهوم اساسی رویکرد تحلیلی، کروماتوگرافی نام دامنه مخرب است. خط لوله آنالیز به نام DOMAINCHROMA طراحی شده است.

یک مجموعه داده ورودی از نام دامنه‌های مخرب ارائه شــده، تهیه شده است. ورودی مورد انتظار DOMAINCHROMA مجموعه‌ای از نام دامنه‌های مخرب که توســط ســیســتم‌های شناسایی دامنه

ارائه می‌شـود [۷۰- ۷۲]، یا شـامل لیسـت‌های سـیاه نام دامنه است. همچنین از شش نوع نام دامنه مخرب اسـتفاده می‌کند. این نام‌های دامنه مخرب از نام‌های دامنه واقعاً مخرب تأیید شـده توسط یک honeypot، سیستم sandbox و سرویس‌های commercial and professional تجاری و حرفه‌ای ارائه شـده توسـط یک سرویس امنیتی تشکیل شده‌اند. این نمونه‌ها بطور تصادفی از VirusTotal بارگیری می‌شوند و شامل فایل‌های اجرایی اخیر (در عرض ۲۴ ساعت) و فایل‌های اجرایی مخرب مورد استفاده در ویندوز مایکروسافت هستند.

نتایج نشـان می‌دهد که هنگام اسـتفاده از خروجی DOMAINCHROMA، بیش از ۷۰٪ از نام دامنه درگیر، در انواع مختلف حملات سایبری می‌توانند فقط در نقاط سطح DNS بدون هیچ خسارت جانبی از دسترسی‌های قانونی دفاع شوند.

با توجه به محدودیت اطلاعات ارائه شده توسط داده‌های DNS، طراحی یک طرح ارتباط با دقت بالا و پوشـش خوب به چالشی تبدیل شده است. روش‌های مبتنی بر استنباط[۱۰]، یکی از روش‌های عمده برای تحلیل داده‌های DNS و شـناسـایی دامنه‌های مخرب می‌باشـد. ایده کلیدی تکنیک‌های اسـتنباط، تعریف ارتباط بین دامنه‌ها براسـاس ویژگی‌های اسـتخراج‌شـده از داده‌های DNS اسـت. از الگوریتم استنباطی مبتنی بر مسیر، به‌طور خاص برای تحلیل داده‌های DNS استفاده شده است. برای نشـان دادن بیشـتر قدرت طرح ارتباط دامنه‌ها و همچنین بهبود کارایی استنتاج، الگوریتم انتشار باور مورد بررسـی قرار می‌گیرد. برای سـاخـت گراف، دو نوع ارتبـاط بین دامنـه‌هـا براسـاس نتـایج طبقه‌بندی‌کننده IP تعریف می‌شود. گراف حاصل از اولین نوع ارتباط‌ها G-New و گراف حاصل از نوع دوم ارتباط G-Relaxed نامیده می‌شـود. دو ارتباط تعریف شـده با توجه به گراف اولیه G-Baseline شکل می‌گیرد.

به منظور ارزیابی طرح پیشـنهادی آزمایشاتی برای بررسی تأثیر انواع مختلف ارتباط‌ها بر پوشش دامنه و دقت تشخیص انجام شده است. گراف G-New تقریباً ۱۰ برابر اندازه G-Baseline در مجموعه

داده‌ها است. همچنین نتایج نشان می‌دهد نوع دوم ارتباط‌ها حتی در مقایسه با نوع اول، پوشش دامنه بیشـــتری را در اختیـار قرار می‌دهـد. الگوریتم G-Relaxed تقریبـاً ۲,۷ برابر اندازه G-New اســـت. همچنین مجموعه گســترده‌ای از آزمایشــات برای اســتنباط دامنه‌های مخرب با اســتفاده از الگوریتم مبتنی بر مسیر بر روی سه گراف ذکرشده انجام می‌شود. هنگام محاسبه نرخ مثبت واقعی TPR و نرخ مثبت کاذب FPR، از ten-fold cross validation اســتفاده شده است. در هر دور، TPR و FPR برای مقادیر آستانه‌های مختلف محاسبه می‌شود. اولین نوع ارتباط منجر به دقت تشخیص بهتری نسبت به G-Baseline شــده اســت. بنابراین، اولین نوع ارتباطات نه تنها به طور قابل‌توجهی پوشــش دامنه‌ها را گسترش می‌دهد، بلکه ثابت می‌کند که این ارتباط قوی‌تر و معنادارتر است.

در رویکرد مبتنی بر مســیر و الگوریتم BP با Gnew و BP ناشــی از گراف دوبخشــی در اولین مشاهدات حاکی از این است که هر سه روش به دقت تشخیص بالا دست می‌یابند. در هر دو مجموعه داده، آنها به ۹۸٪ نرخ TP با کمتر از ۱۰٪ نرخ FP دســت یافتند. طرح پیشــنهادی به تجزیه و تحلیل IPهای اختصــاصــی و عمومی متکی اســت که نه تنها پوشش دامنه، بلکه دقت تشخیص نیز نسبت به ارتباط دامنه‌های موجود بهبود می‌یابد. همچنین طرح پیشـــنهادی می‌تواند با الگوریتم اســـتنباطی مبتنی بر مسیر و الگوریتم استنباط عمومی انتشار باور ادغام شود. ضمنا مجموعه دادگان مورد استفاده در این تحقیق، روی مجموعه داده‌های Active DNS انجام شــده اســت که در دو هفته متوالی مورد استفاده قرار گرفت [۷۳- ۷۷].

۲-۸ روش‌های مبتنی بر یادگیری عمیق

روش‌های قبلی همگی روش‌های معمولی و ســنتی شــناســایی بودند، در این بخش روش طبقه‌بندی مبتنی بر یادگیری عمیق بررسی می‌شود. همچنین یادگیری عمیق در بخش ۲-۵ معرفی شده است.

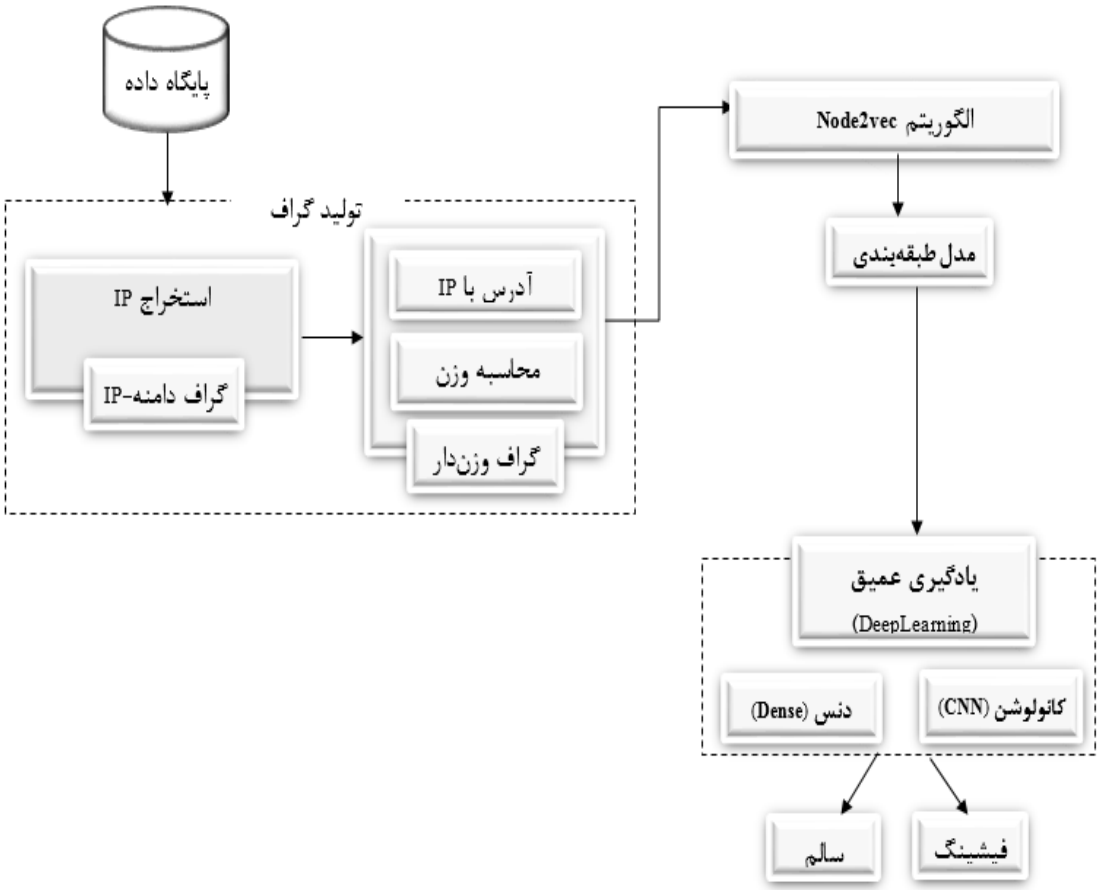
در مقاله[۱۸]، رویدادنامه‌های DNS را از دســتگاه‌های مشــتری در شــبکه محلی (LAN) جمع می‌کننــد و آن را در یک ســـرور ذخیره می‌کنند. برای یافتن نام دامنه به عنوان خوش‌خیم یا مخرب،

یک رویکرد مبتنی بر یادگیری عمیق پیشنهاد می‌شود. برای مقایسه، اثربخشی روش‌های مختلف یادگیری عمیق مانند شبکه عصبی RNN، حافظه کوتاه مدت بلند LSTM و سایر طبقه‌بندی‌کننده‌های یادگیری سنتی را ارزیابی کرده است. رویکردهای مبتنی بر یادگیری عمیق نسبت به سایر طبقه‌بندی‌کننده‌های یادگیری ماشین کلاسیک عملکرد خوبی داشته‌اند. دلیل اصلی این است که الگوریتم‌های یادگیری عمیق توانایی به دست آوردن ویژگی‌های مناسب را بطور ضمنی دارند. علاوه بر این، LSTM در مقایسه با سایر روش‌های یادگیری عمیق بالاترین میزان تشخیص مخرب را در همه آزمایشات بدست آورده‌اند. مجموعه داده‌های مورد استفاده در این مقاله از داده‌های جمع‌آوری شده توسط سایت Alexa [۷۸]، الگوریتم تولید دامنه [۷۹] و OSNIT [۸۰] بوده است.

فصل ۳ : روش پیشنهادی

۳-۱ مقدمه

در این فصل جزئیات مربوط به روش پیشنهادی برای تشخیص دامنه‌های مشکوک شرح داده می‌شود. روش پیشنهادی مورد استفاده در این پایان‌نامه یک روش مبتنی بر تحلیل گراف با استفاده از یادگیری عمیق می‌باشد. به طور کلی کار ما در این پایان‌نامه از چهار بخش آماده‌سازی داده، تولید گراف، الگوریتم تبدیل داده به بردار و دسته‌بندی تشکیل یافته است. همان‌طور که در معماری روش پیشنهادی شکل ۳-۱ نشان داده شده است، ابتدا به آماده‌سازی داده جهت تشخیص می‌پردازیم. آماده‌سازی داده دراین مرحله شامل گرفتن داده از مجموعه دادگان و استخراج IP است. در مرحله بعد نوبت به تولید گراف دو بخشی دامنه-IP با استفاده از دامنه و IPهای استخراج شده از هریک از دامنه‌های مربوطه است؛ سپس از روی گراف دو بخشی، گراف وزن‌دار دامنه ساخته می‌شود که در این مرحله ارتباط بین دامنه‌ها و وزن بین آن‌ها محاسبه می‌شود. در گام بعد با استفاده از الگوریتم Node2vec که یک الگوریتم تبدیل داده به بردار است، داده‌های تولید شده در گراف وزن‌دار جهت ورود به طبقه‌بند با الگوریتم Node2vec به بردار تبدیل می‌شود. در گام نهایی بردارهای ایجاد شده به مدل طبقه‌بندی یادگیری عمیق داده می‌شود تا داده‌ها به دو دسته سالم و فیشینگ دسته‌بندی شوند. با کمک این الگوریتم‌ها می‌توان دامنه‌های مشکوک از سالم را بادقت بالا و به طور موثرتری شناسایی کرد. لازم به ذکر است در روش پیشنهادی مدل‌های طبقه‌بندی مبتنی بر یادگیری عمیق با الگوریتم SVM و انتشار باور BP در بخش ۲-۲ و ۲-۳ به آن اشاره شده است مورد مقایسه قرار می‌گیرد؛ تا دقت عملکرد الگوریتم‌های مطرح شده با هم سنجیده شود و الگوریتمی که دسته‌بندی بهتری با دقت بالاتر در شناسایی دامنه‌های فیشینگ از غیرفیشینگ را داشته است، مشخص شود. در ادامه به تفکیک، به توضیح هر قسمت پرداخته شده است.



شکل ۳-۱.معماری روش پیشنهادی

۳-۲ آماده‌سازی داده برای تحلیل در گام‌های بعدی

آمـاده‌ســـازی بر روی داده‌هـا همیشـــه یک بحث چالش برانگیز بوده اســـت. دراین پایان‌نامه جهت آماده‌ســازی داده ابتدا داده‌ها (آدرس دامنه) همراه با برچسب سالم و فیشینگ، به IP تبدیل می‌شوند. علت اســتفاده از IP وجود ارتباطاتی اســت که بین دامنه و IP وجود دارد؛ همچنین فیشرها همیشه از IPهـای عمومی برای حملات فیشـــینـگ اســـتفاده می‌کنند، بنابراین از IP به عنوان یک ویژگی در تشـخیص دامنه‌های سـالم از فیشـنگ همراه با دامنه اسـتفاده می‌شود. سپس از IPهای بدست آمده

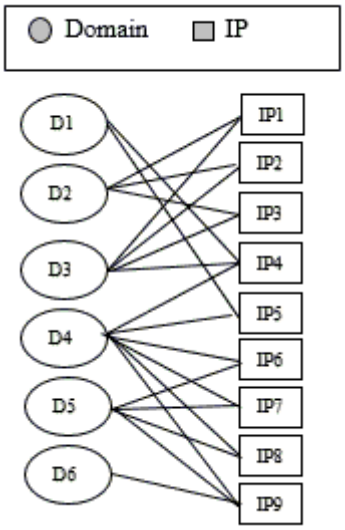
همراه با دامنه‌های مربوط به هر IP جهت شـکل‌گیری گراف دو بخشـی دامنه-IP استفاده می‌شود. در بخش بعدی نحوه ساخت گراف دو بخشی دامنه-IP شرح داده شده است.

۳-۳ ساخت گراف دو بخشی دامنه-IP

از گراف‌ها برای حل مسایل زیادی در ریاضیات و علوم کامپیوتر استفاده می‌شود. ساختارهای زیادی را می‌توان به کمک گراف‌ها به نمایش درآورد [۸۱]. یک گراف از مجموعه‌ای گره تشکیل شده، که آن را با V نشـان می‌دهیم و مجموعه‌ای شـامل یال‌ها، که گره‌ها را به هم وصل می‌کنند و با E نمایش داده می‌شـود. یک چنین گرافی را با $G = (V, E)$ نشان می‌دهیم. بعد از مرحله آماده‌سازی داده در مرحله قبل، اکنون با اسـتفاده از دامنه‌ها و IPهای ترجمه شـده مربوط به دامنه وارد مرحله ساخت گراف دو بخشی دامنه-IP می‌شویم.

گراف دو بخشـــی دامنه-IP، $G(D, I, E)$ اســـت که در آن D مجموعه‌ای از دامنه‌ها اســـت، I مجموعه‌ای از IPها و E یک یال بین این دو اسـت یعنی $\{d,i\} \in E$ است اگر دامنه d به IP(i) ترجمه شـود. IP(d) یعنی مجموعه IPهایی که Domain(d) به آن IPها ترجمه شـده اسـت. به طور مشـابه، Domain(i) مجموعه دامنه‌های ترجمه شـــده به IP(i) را مشـــخص می‌کند. در این گراف دامنه‌ها به IPهای ترجمه شده متصل هستند، هر دامنه ممکن است به بیش از یک IP، ترجمه شود.

درگراف دو بخشی دامنه-IP یک گره شامل IP و گره دیگر شامل دامنه است هر کدام از دامنه‌ها به IPهایی که ترجمه شـده اسـت اشـاره دارند، مدل سـاخت گراف دو بخشی دامنه-IP در شکل ۳-۲ نمایش داده شده است. در بخش بعدی به نحوه ساخت گراف وزن‌دار پرداخته شده است.



شکل ۳-۲.گراف دو بخشی دامنه-IP

شبه کد مربوط به گراف دو بخشی دامنه-IP را در زیر مشاهده می‌کنید:

Input: Domain and label

Output: Domain and IP Graph

For All Domain Implementation Graph Domain- IP

    Extract IP

    Draw the edge between the domain and IP

Exit

۳-۴ ساخت گراف وزن‌دار

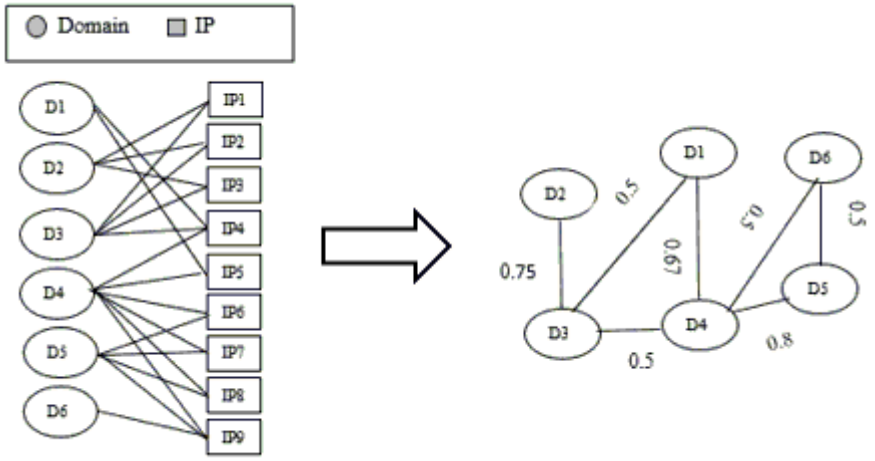
در این بخش برای تبدیل گراف دو بخشی دامنه-IP به گراف وزن‌دار، اطلاعات موجود در گراف از بین می‌رود. در گراف دو بخشی ارتباط بین دامنه و IP مشخص است اما در گراف وزن‌دار ارتباطات یک تعداد از دامنه‌ها با IP از دست می‌رود، اطلاعات حذف شده تنها با ایجاد یال بین دامنه‌ها بدست نمی‌آید. لذا از پارامتر وزن به منظور مشخص شدن تعداد یال استفاده می‌شود. در ادامه ساخت گراف وزن‌دار توضیح داده شده است.

برای تبدیل گراف دو بخشـــی دامنه-IP به گراف وزن‌دار که در بخش قبل توضـــیح داده شـــد، دامنه‌هایی در نظر گرفته می‌شـــود که حداقل یک IP مشـــترک بین آن‌ها وجود دارد. در واقع برای نمـایش چگونگی رابطه دامنه‌ها به یکدیگر می‌توان هر دامنه را به یک گره در گراف تبدیل کرده و در صـورتی‌که در این دامنه ارتباطی با دامنه دیگر وجود داشت، یک یال از این دامنه به گره‌ایی که دامنه دیگر را نمایش می‌دهد وصـل می‌شود. به عبارت دیگر می‌توان گفت بین دامنه‌هایی یال رسم می‌شود که IP مشـترک بین دو آدرس وجود دارد. بعد از اینکه گراف ساخته شد، گام بعدی محاسبه وزن بین دامنه‌ها اســـت. هر چه تعداد IP بین دامنه‌ها بیشـــتر باشـــد وزن بین آن‌ها نیز بیشـــتر اســـت. برای پیاده‌سازی این مفهوم از معادله ۳-۱ که در مرجع [۱۰] برای محاسبه وزن بین دامنه‌ها استفاده کرده است، استفاده می‌شود. گراف وزن‌دار در شکل ۳-۳ نمایش داده شده است.

معادله ۳-۱

$$\mathrm{w(d1,d2)} = \begin{cases} 1 - \dfrac{1}{1 + ip(d1) \cap ip(d2)} & \text{if d1} \neq \text{d2} \\ 1 & \text{otherwise} \end{cases}$$

در معادله ۳-۱ بخش $ip(d1) \cap ip(d2)$ تعداد ipهای مشترک دامنه d1 و دامنه d2 را نشان می‌دهد. همچنین بین دامنه با خودش ارتباطی وجود دارد و وزن طبق فرمول یک در نظر گرفته می‌شـود. بعد از تشـکیل گراف و بدسـت آوردن وزن بین دامنه‌ها نوبت به طبقه‌بندی داده‌ها اسـت. در این مرحله با اسـتفاده از مدل طبقه‌بند یادگیری عمیق به سـنجش میزان تشـخیص دامنه‌های مشـکوک از سـالم پرداخته شده است.



شکل ۳-۳.ساخت گراف وزن‌دار از گراف دو بخشی دامنه–IP

شبه کد مربوط به گراف وزن‌دار را در زیر مشاهده می‌کنید:

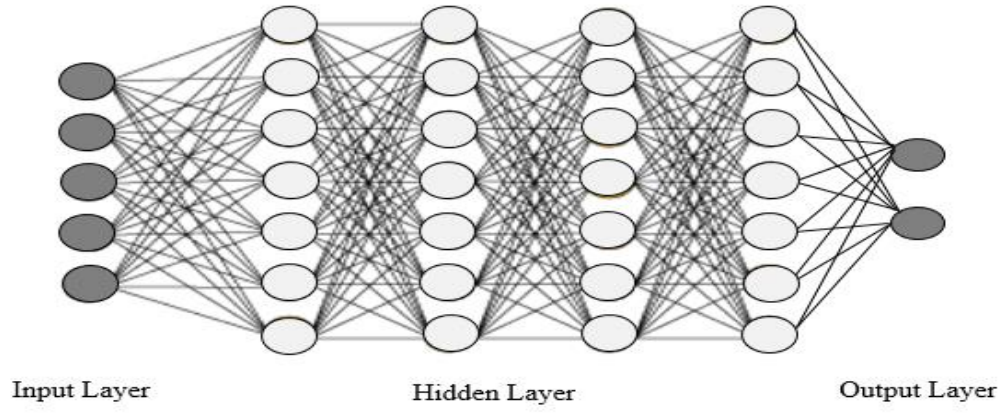
```
Input: Domain- IP Graph
Output: Weight Graph
  For All Domain&Ip Implementation Graph Weight
        Find Domains With Common Ip
        Draw Edge Between Domains
        Calculat Weight Of The Domains
  Exit
```

بعد از محاسبه گراف وزن‌دار، به منظور تبدیل داده‌های گراف به بردار، جهت استفاده از مدل‌های طبقه‌بند یادیگری عمیق از الگوریتم Node2vec وزن‌دار که در بخش ۲-۴ به طور کامل توضیح داده شده است، استفاده می‌شود. گره‌ها و ارتباطات بین گره‌ها به عنوان ورودی به الگوریتم Node2vec داده می‌شود، الگوریتم بعد از پیمایش گراف به ازای هر نود در خروجی یک بردار ایجاد می‌کند. در نتیجه گره‌های شبیه به هم بردارهای نزدیک به هم دارند که با یک طبقه‌بند می‌توان عمل دسته‌بندی را روی گره‌های گراف انجام داد.

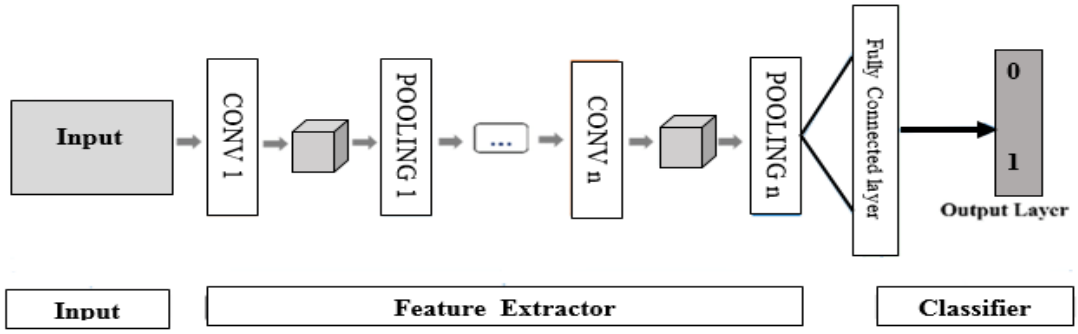
۳-۵ دسته‌بندی گره‌های گراف

بعد از اجرای مراحل قبل، ورودی الگوریتم برای اجرای عملیات دسته‌بندی آماده شده است. در این مرحله تکنیک یادگیری عمیق شبکه DENSE و شبکه CNN برای دسته‌بندی گره‌های گراف اجرا می‌گردد. شبکه DENSE و شبکه CNN در بخش ۲-۵-۱ و ۲-۵-۲ به طور مفصل توضیح داده شده است. لازم به ذکر است معماری DENSE و CNN استفاده شده در این پایان‌نامه را در شکل ۳-۴ و شکل ۳-۵ مشاهده می‌کنید.

به منظور استفاده از Dense در ابتدا یک مدل Sequential ایجاد کرده و لایه‌ها را اضافه می‌کنیم. غالبا بهترین نوع ساختار از نظر تعداد لایه‌ها و انواع آن، از طریق سعی و خطا مشخص می‌شود. در این پایان‌نامه، از یک ساختار شبکه کاملا متصل با چهار لایه استفاده شده است. لایه های کاملا متصل با استفاده از Dense تعریف می‌شوند. تعداد نورون‌ها یا گره‌ها را در لایه به عنوان اولین آرگومان مشخص کرده و با استفاده از تابع فعال‌سازی، عملکرد فعال‌سازی تعیین می‌شود. از تابع فعال‌سازی واحد خطی اصلاح شده Relu در هر چهار لایه استفاده شده است. پیش از این توابع فعال‌سازی Sigmoid و Tanh برای همه لایه‌ها ارجحیت داشت؛ اما امروزه با استفاده از تابع فعال‌سازی Relu عملکرد بهتری حاصل می‌شود. در لایه خروجی از تابع Softmax استفاده می‌شود.



شکل ۳-۴.معماری Dense



شکل ۳-۵.معماری CNN

شبکه های عصبی کانولوشن تا حد بسیار زیادی شبیه شبکه‌های عصبی مصنوعی هستند که در بخش ۲-۵-۲ در مورد آنها توضیح داده شد. این نوع شبکه‌ها متشکل از نورون‌هایی با وزن‌ها و بایاس‌های قابل یادگیری (تنظیم) هستند. در حالت کلی، یک شبکه عصبی کانولوشن یک شبکه عصبی سلسله مراتبی است که لایه‌های کانولوشنی آن به صورت یک در میان با لایه‌های pooling بوده و بعد از آن‌ها تعدادی لایه تماماً متصل وجود دارد. شبکه CNN روش پیشنهادی در شکل ۳-۵ از سه لایه‌ی کانولوشنی و pooling بعد از هر کدام، یک لایه متراکم تمام متصل و یک لایه متراکم برای لایه خروجی که دارای دو نورون است استفاده می‌کند. تابع فعال‌سازی لایه آخر Softmax برای خروجی دو کلاس می‌باشد. همچنین به منظور ارزیابی دقیق عملکرد مدل طبقه‌بند یادگیری عمیق با دو الگوریتم BP و SVM مورد مقایسه قرار می‌گیرد.

۳-۶ مدل استقرار سامانه

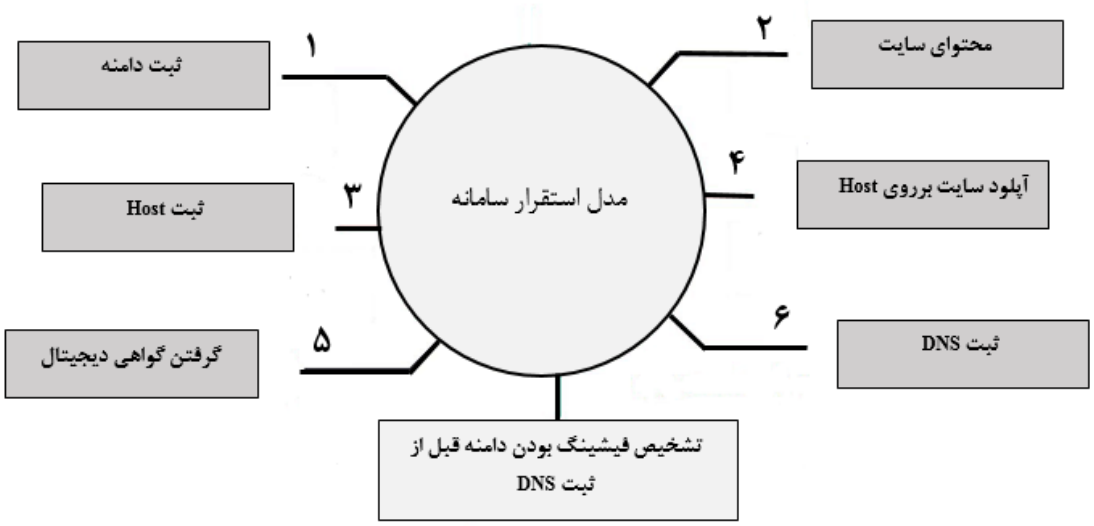
مدل پیشنهادی در این پایان‌نامه ساخت دسته‌بندی است که با دریافت رکورد موردنظر، کلاس رکورد را تشخیص می‌دهد. به منظور ارزیابی دسته‌بند در تشخیص با بخشی ازداده‌ها که دارای برچسب هستند مدل آموزش داده می‌شود و با بخشی دیگر داده برای تست و صحت دسته‌بند در تشخیص مورد استفاده قرار می‌گیرد؛ هرچه دقت تشخیص بیشتر باشد، سامانه دقیق‌تر و با اطمینان بیشتری

عمل طبقه‌بندی را انجام می‌دهد. از آنجایی که این دسته‌بند با مجموعه دادگان تست می‌شود این سوال مطرح می‌شود اگر یک دامنه فیشینگ جدید بیاید که در مجموعه دادگان مورد نظر وجود ندارد آیا سیستم قادر به تشخیص می‌باشد و یا زمان تشخیص این دامنه چقدر طول می‌کشد.

در پاسخ به این سوالات، می‌توان گفت؛ فیشرها برای راه‌اندازی سایت فیشینگ و Https نیاز به ثبت دامنه و گرفتن گواهی دیجیتال[۲۲] دارند. برای راه‌اندازی دامنه‌های Https همان‌طور که در شکل ۳-۶ نشان داده شده است، فیشر ابتدا یک نام دامنه برای خود ثبت می‌کند. در مرحله بعد نوبت به نوشتن محتوای سایت و دریافت Host مربوط به این دامنه است. بعد از گرفتن Host در مرحله بعد فیشر باید محتوای سایت نوشته شده را روی Host مورد نظر نصب کند؛ سپس به منظور راه‌اندازی این دامنه نیاز به گرفتن گواهی دیجیتال دارد تا بتواند دامنه‌های بانکی یا دامنه‌های خاصی را فیشینگ کند، از آنجایی که سایت‌های Https برای گرفتن گواهی دیجیتال از تعداد مراکز محدودی می‌توانند درخواست گواهی دیجیتال دهند قوانین سختگیرانه‌ی بین‌المللی نیز توسط این مراکز اعمال می‌شود. بعد از گرفتن ثبت این گواهی از مراکز موجود، رکوردی از ثبت این گواهی در اینترنت قرار می‌گیرد و در نهایت با ثبت DNS، سایت فیشینگ برای کاربر قابل رویت و استفاده می‌شود. از طرفی یک سری سایت‌ها رکوردهای ثبت شده را نشان می‌دهند. مرکز آپا دانشگاه صنعتی شاهرود رکوردهای ثبت شده در این سایت‌ها را بررسی می‌کند و هر زمان دامنه‌ایی ثبت شود و Https هم باشد رکورد موردنظر را دریافت می‌کند. این دامنه نیازمندی سیتم پیشنهادی در این پایان‌نامه می‌باشد. در واقع کاری که در این مرحله انجام می‌شود تشخیص فیشینگ بودن دامنه قبل از ثبت DNS است، یعنی قبل از ثبت DNS، دامنه مورد نظر به سامانه دسته‌بندی داده می‌شود که در این پایان‌نامه به آن پرداخته شده است و دامنه موردنظر توسط دسته‌بند ساخته شده، احتمال فیشینگ یا سالم بودن آن بررسی می‌شود و در صورت تشخیص فیشنگ با دقت بالا و جمع‌آوری شواهد با دریافت حکم فیلترینگ توسط مراجع ذیربط درکمتر از چند دقیقه عمل فیلترینگ برای سایت موردنظر اجرا می‌شود. در واقع می‌توان گفت

[۲۲] Certificate

سامانه موردنظر تشخیص جرم قبل از وقوع خطر را پیش‌بینی کرده است.



شکل ۳-۶.نحوه استقرار

۳-۷ جمع‌بندی

در روش پیشنهادی هدف تولید سامانه‌ای است که بتوان تشخیص داد چه دامنه‌هایی فیشینگ یا غیر فیشینگ است. این کار با توجه به الگوریتم‌ها و تکنیک‌های پیشرفته هوش مصنوعی به صورت خودکار قابل اجرا است. هدف در این روش پوشش گسترده‌تری از شناسایی دامنه‌های جدیدی است که فیشینگ یا سالم بودن آن مشخص نمی‌باشد.

فصل ۴ : نتایج و آزمایشات

۴-۱ معیارهای ارزیابی مدل

یکی از مهم‌ترین مسائل پس از طراحی و ساخت یک مدل یا یک تکنیک مبتنی بر دسته‌بندی، ارزیابی کارایی آن است. انتخاب یک معیار برای ارزیابی به مسئله‌ایی که سعی در حل آن داریم وابسته است. برای مسائل دسته‌بندی از معیار صحت استفاده می‌شود. این معیار صحت کل دسته‌بند را محاسبه می‌نماید. این معیار نشان‌دهنده این حقیقت است که دسته‌بند طراحی شده چند درصد از کل آزمایش را به درستی دسته‌بندی کرده است. ماتریس درهم‌ریختگی[۲۳] چگونگی عملکرد الگوریتم دسته‌بندی را با توجه به مجموعه داده ورودی به تفکیک انواع دسته‌های مسئله دسته‌بندی نشان می‌دهد. در این مبحث هدف از معیار ارزیابی برای سنجش مدل، صحت می‌باشد [۸۲].

۴-۱-۱ دسته‌بندی داده‌ها

۱- مثبت صحیح (True Positive): داده‌ها درست شناسایی شده است.

۲- مثبت کاذب (False Positive): داده‌ها اشتباه شناسایی شده است.

۳- منفی صحیح (True Negative): داده‌ها به درستی رد شده است.

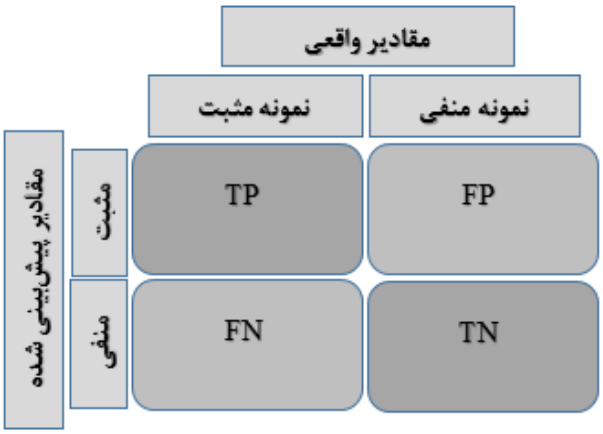
۴- منفی کاذب (False Negative): داده‌ها به اشتباه رد شده است.

۴-۱-۲ ماتریس درهم‌ریختگی

در حوزه هوش مصنوعی، ماتریس درهم‌ریختگی به ماتریسی گفته می‌شود که در آن عملکرد الگوریتم‌های مربوطه را نشان می‌دهند. معمولاً چنین نمایشی برای الگوریتم‌های یادگیری با ناظر استفاده می‌شود، اگرچه در یادگیری بدون ناظر نیز کاربرد دارد. معمولاً به کاربرد این ماتریس در

[۲۳] Confusion Matrix

الگوریتم‌های بدون ناظر ماتریس تطابق می‌گویند. هر سطر از ماتریس، نمونه‌ای از مقدار پیش‌بینی‌شده را نشـــان می‌دهد. درصـــورتی‌که هر ســـتون نمونه‌ای واقعی (درســـت) را در بر دارد. در خارج از هوش مصـــنوعی این ماتریس معمولاً ماتریس پیشـــایندی[۲۴] یا ماتریس خطا[۲۵] نامیده می‌شـــود. ماتریس در هم‌ریختگی را در شکل ۴-۱ مشاهده می‌کنید.



شکل ۴-۱.ماتریس درهم‌ریختگی

برای ارزیابی روش پیشـــنهادی آزمایشـــاتی برای موارد مختلف تفســـیر و انجام شـــد. در ارزیابی معمولا پارامترهای زیر متصور است [۸۳ و ۸۴].

۱. **تشکیل ماتریس اختلاط یا درهم‌ریختگی**
۲. **صحت (Accuracy)**
۳. **دقت (Precision)**
۴. **بازخوانی (Recall)**
۵. **پارامتر F1-score**

[۲۴] Matrix Contingency
[۲۵] Error Matrix

همچنین علاوه بر معیارهای ارزیابی ذکر شده، نمودارهای ROC[۲۶] مربوط به همه مدل‌ها محاسبه شده است. منحنی ROC را به عنوان نمودار مشخصه نسبی عملکرد می‌شناسند زیرا مقایسه‌ای بین دو نحوه عملکرد TPR و FPR ارائه می‌کند. به طور کلی با شـرط مشـخص بودن توزیع احتمالی برای هر دو بخش TPR و FPR منحنی ROC در صـورتی حاصل خواهد شد، که تابع توزیع تجمعی[۲۷] یا سطح زیر منحنی توزیع احتمال تشخیص درست در محور عمودی و تابع توزیع تجمعی تشخیص نادرست در محور افقی در نظر بگیریم. به این ترتیب TPR مشـخص می‌کند که به چه نسـبتی پیش‌بینی صحیح صـورت گرفته اسـت. یعنی تعداد پیش‌بینی‌های صحیح بر تعداد نتایج مثبت واقعی تقسیم شده و نرخ پیش‌بینی صـحیح مثبت محاسـبه می‌شـود. از طرف دیگر FPR نشـانگر تعداد شناسایی‌های مثبت از میان مشـاهدات منفی است. این نسبت نیز به عنوان نرخ مثبت کاذب در نمودار ROC به کار می‌رود. بنابراین فضـای ROC بوسیله این دو شاخص یعنی FPR روی محور افقی و TPR روی محور عمودی نشان داده می‌شود. در نتیجه یک توازن بین TP و FP روی نمودار شکل می‌گیرد.

در این نمودار قسـمت افقی FPR نرخ تولید خطا و قسـمت عمودی نشان‌دهنده TPR نرخ تولید داده‌های درست است. در این نمودار نقاطی که بالای خط نیمساز قرار گرفته‌اند مقدار حساسیت آن‌ها نسـبت به FPR بیشـتر اسـت. به معنای آن که در این بخش، نرخ مثبت صـحیح بیشتر از نرخ مثبت کاذب است. قرار گرفتن نقاط در این محیط مطلوب خواهد بود. در مقابل نقاطی که روی خط نیمساز قرار گرفته‌اند، مقدار عددی نرخ مثبت صحیح و نرخ مثبت کاذب با یکدیگر برابر است. به عبارتی از هر ۱۰۰ نمونه، آزمایش در ۵۰ نمونه به درسـتی و در ۵۰ نمونه دیگر نیز به اشـتباه دامنه‌ها را شـناسایی کرده اسـت. این نقاط چندان مطلوب نیستند. مقدار عددی AUC[۲۸] (سطح زیر منحنی نمودار ROC) به وضـوح عددی بین صـفر تا یک اسـت و نشـان می‌دهد قدرت تشخیص یا درستی نتایج یک آزمون چقدر می‌باشـد. درسـتی نتایج آزمون به این بستگی دارد که روش آزمون چقدر توانایی تفاوت درست

[۲۶] Relative Operating Characteristic

[۲۷] Cumulative Distribution Function

[۲۸] Area Under The Curve

نشـان دادن نتایج مثبت صـحیح TP و منفی صحیح TF را دارد. اگر این عدد به یک نزدیک باشد، به معنای آن اسـت که داده‌ها عموماً در بالای خط نیمسـاز قرار گرفته‌اند و میزان نرخ مثبت صـحیح بالا اسـت و روش آزمون از قدرت تشخیص یا درستی مناسبی برخوردار است. اعداد AUC نزدیک به ۰,۵ همان برابری نرخ مثبت صـحیح و نرخ مثبت کاذب را نشـان می‌دهد و اعداد کمتر از ۰,۵ بیانگر بالاتر بودن نرخ مثبت کاذب در مقایسـه با نرخ مثبت صـحیح اسـت. در این بخش نقاطی قرار گرفته‌اند که مقدار حسـاسـیت آن‌ها نسبت به FPR کمتر است. اگر خط منحنی زیر خط نیمساز قرار گرفته باشد، نرخ مثبت صـحیح کمتر از نرخ مثبت کاذب است. قرار گرفتن نقاط در این بخش اصلاً مطلوب نیست. در واقع هر چه نمودار به یک نزدیکتر باشد، دقت بالای مدل را در تشخیص نشان می‌دهد [۸۵ و ۸۶].

۴-۲ مجموعه دادگان

در این بخش قصد داریم مجموعه دادگان مورد استفاده در این پایان‌نامه را معرفی کنیم. مجموعه دادگان شامل دامنه‌های غیرفیشینگ و فیشینگ می‌باشد که از سه مجموعه دادگان با رکوردهای DNS استفاده شده است. هر مجموعه داده از تعدادی رکورد به صورت ماتریس با دو ویژگی تشکیل یافته است. این رکوردها شامل آدرس‌های دامنه و برچسب مربوط به هر دامنه می‌باشد. همان‌طور که پیش‌تر به آن اشاره شد لازم بود به دلیل ارتباطاتی که بین دامنه و IP وجود دارد ابتدا دامنه‌ها به IPهای مربوطه ترجمه شوند. همچنین هر دامنه ممکن است از دو یا تعداد بیشتری IP تشکیل شده باشد. از اطلاعات بدست آمده در این قسمت، دامنه و IP برای بدست‌آوردن ارتباط بین دامنه‌ها با IP مشترک و محاسبه وزن استفاده می‌شود. اطلاعات هریک از سه مجموعه دادگان مورد استفاده در جدول ۴-۱، نمایش داده شده است. در این جدول تعداد دامنه‌های مشکوک، دامنه‌های سالم و تعداد کل دامنه‌ها در هریک از مجموعه دادگان بیان شده است. مجموعه دادگان اول از دامنه‌های فیشینگ ثبت شده توسط بانک مرکزی و دامنه‌های غیر فیشینگ از سازمان فناوری اطلاعات ایران [۱۹] گردآوری و در اختیار قرارگرفته است. ضمنا دامنه‌های فیشینگ و غیر فیشینگ در مجموعه دادگان اول شامل سایت‌های داخلی است.

مجموعه دادگان دوم از مرجع [۱۸] بدست آمده است. در نهایت مجموعه دادگان سوم شامل ترکیب مجموعه دادگان استفاده شده در مرجع [۱۸ و ۲۰] می‌باشد.

جدول ۴-۱.مجموعه دادگان

مجموعه دادگان	سالم	مشکوک	کل داده‌ها
مجموعه دادگان اول	۲۰۰۱۲	۸۱۲۱	۲۸۱۳۳
مجموعه دادگان دوم	۱۵۵۹۷۲	۱۰۸۰۲۸	۲۶۴۰۰۰
مجموعه دادگان سوم	۵۶۶۶۲۸	۳۳۳۳۷۲	۹۰۰۰۰۰

۴-۳ محیط ارزیابی و پیاده‌سازی

در این بخش، برای پیاده‌سازی مدل، نحوه تولید داده‌های آموزشی برای آموزش مدل شرح داده خواهد شد. همچنین، نحوه کدنویسی فرایند آموزش و تست، زبان برنامه نویسی مورد استفاده و سخت‌افزار سیستم استفاده شده نیز شرح داده می‌شود. لازم به ذکر است که برای عملیات ارزیابی، مجموعه‌ای متشکل از داده‌های تصادفی تولید و از آن‌ها برای آموزش و تست مدل یادگیری استفاده شده است.

۴-۳-۱ تولید داده آموزشی و تست برای مدل

نکته مهم در مورد داده‌های آموزشی این است که از این داده‌ها فقط برای آموزش مدل استفاده می‌شود و نباید از آن‌ها برای تست مدل نیز استفاده کرد. به عبارت دیگر، داده‌های مرحله تست باید برای سیستم آموزش داده شده، دیده نشده[۲۹]باشند؛ یعنی پیش از این برای آموزش مدل استفاده نشده باشند. به منظور تقسیم داده‌های آموزشی و تست برای ارزیابی مدل در اجرای آزمایشات از تابع Train_Test_Split استفاده می‌شود. در واقع از یک روش طبقه‌بندی مشابه تقسیم Test- Train استفاده شده است. برای هر بخش، ۸۵٪ از داده‌ها برای آموزش استفاده می‌شود، در حالی که ۱۵٪

[۲۹] Unseen

برای تست در نظر گرفته می‌شود. صحت در تمام بخش‌ها به طور متوسط برای تولید یک امتیاز واحد انجام می‌شود.

۴-۳-۲ معرفی نرم‌افزار

آموزش مدل به زبان Python به منظور یادگیری مرحله عملی و اجرایی سیستم پیشنهادی به زبان برنامه‌نویسی متن‌باز پایتون ارائه شده است. عموما استفاده از زبان‌های برنامه‌نویسی در محیط‌های اجرایی باعث سختی و پیچیدگی کار می‌شود، این درحالی است که توزیع‌های مختلف که دارای محیط‌های گرافیکی و ویرایشگر هستند، در این زمینه توصیه می‌شود. محیط پیاده‌سازی Spyder می‌باشد. همچنین با توجه به حجم بالای مجموعه دادگان استفاده شده از سرور دانشگاه صنعتی شاهرود، با مشخصات سخت‌افزاری زیر استفاده شده است.

Windows 10: 64 bit
RAM: 32GB
CPU: Intel(R) COREi7 6700K 4.00GHz
NVIDIA: GFORCE 1060
HARD: 2TB

۴-۴ نتایج و آزمایشات

برای ارزیابی عملکرد روش تشخیص خود، دامنه‌های مشکوک را به عنوان نمونه‌های منفی و دامنه‌های سالم به عنوان مثبت قرار داده شد. صحت به عنوان معیار اصلی تعریف شد، در ادامه به بررسی صحت مدل پیشنهادی به کمک ماتریس درهم‌ریختگی پرداخته شده است. این ارزیابی با سه مجموعه دادگان که در بخش ۴-۳ به آن اشاره شد؛ انجام می‌شود. آزمایشات برروی هر مجموعه دادگان به‌طور جداگانه انجام گرفت و معیارهای ارزیابی برای هر یک از الگوریتم‌های مورد بررسی محاسبه شد، همچنین نمودار ROC برای هرکدام از الگوریتم‌ها بدست آمد. مقدار AUC مربوط به هر کدام از مدل‌ها به صورت جداگانه در هر جدول نشان داده شده است. نتایج کامل مربوط به هر کدام از مجوعه دادگان در

جدول ۴-۲، جدول ۴-۳ و جدول ۴-۴ نمایش داده شـده است. نمودارهای مربوط به ROC هر یک از مجموعه دادگان در شکل ۴-۲، شکل ۴-۳ و شکل ۴-۴ نمایش داده شده است.

جدول ۴-۲.معیارهای ارزیابی و AUC برای مجموعه دادگان اول

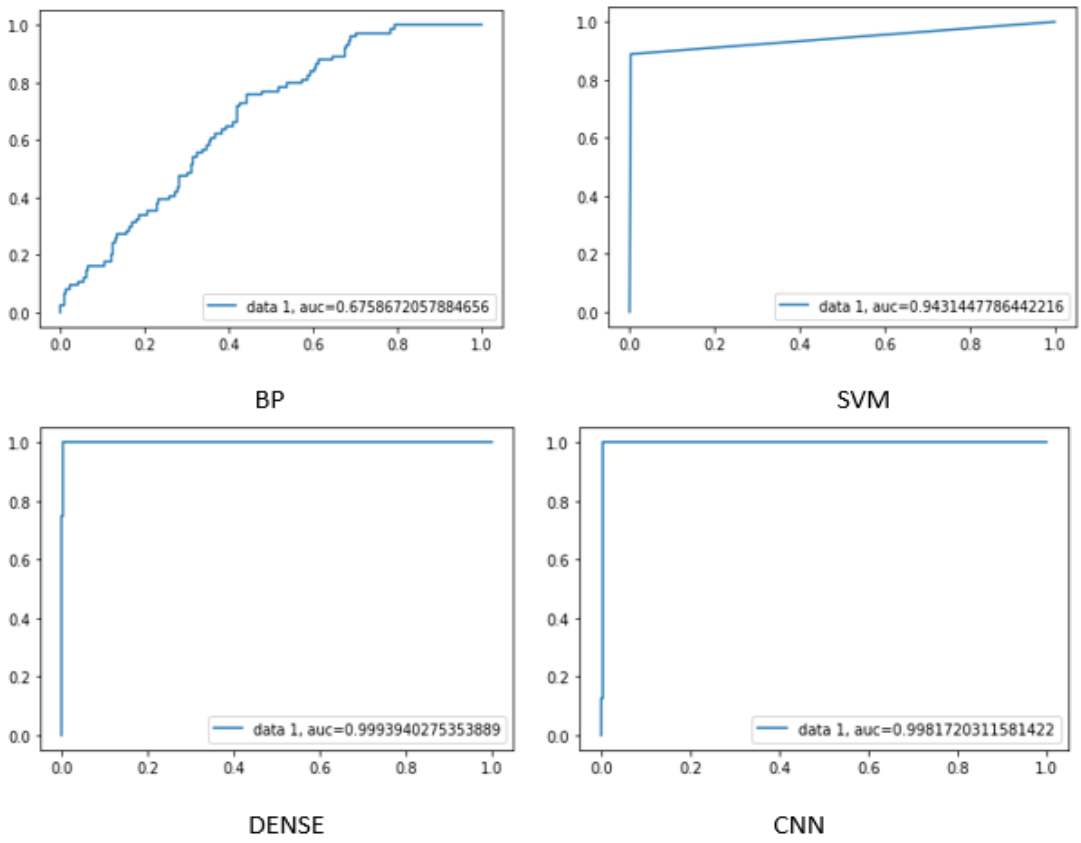
	Acc	Precision	Recall	F1-score	AUC
BP	۰,۶۵	۰,۶۳	۰,۶۴	۰,۶۳	۰,۶۷
SVM	۰,۹۷	۰,۹۷	۰,۹۶	۰,۹۶	۰,۹۴
DENSE	۰,۹۹	۰,۹۹	۰,۹۹	۰,۹۹	۰,۹۹
CNN	۰,۹۹	۰,۹۹	۰,۹۹	۰,۹۹	۰,۹۹

جدول ۴-۳.معیارهای ارزیابی و AUC برای مجموعه دادگان دوم

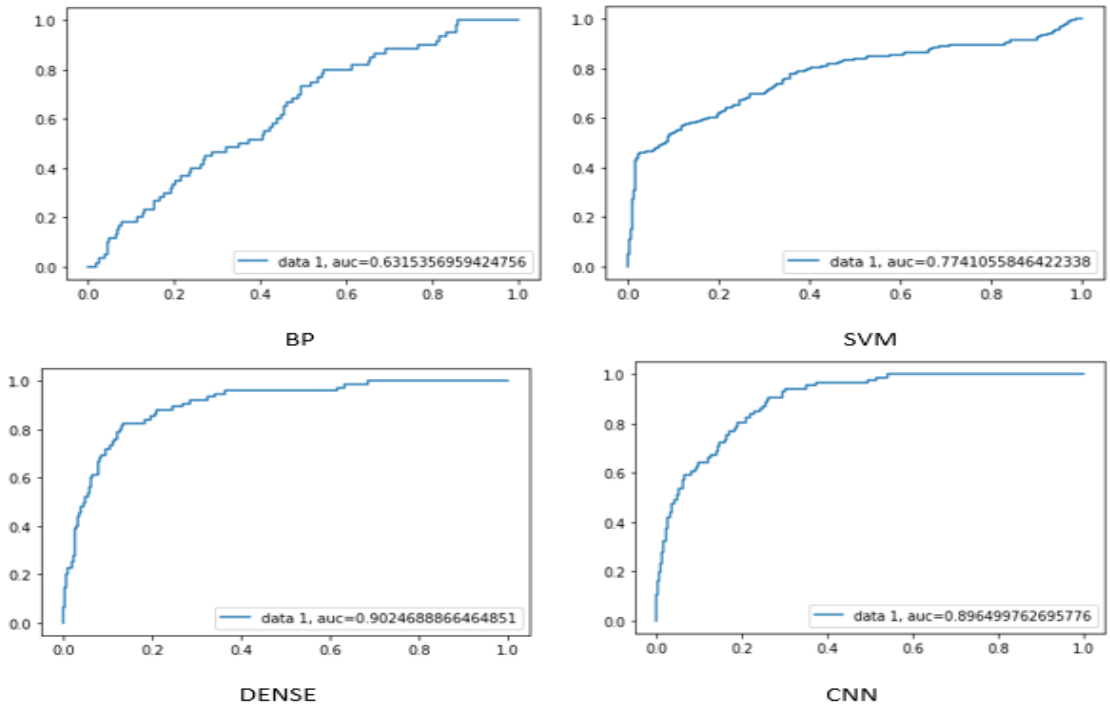
	Acc	Precision	Recall	F1-score	AUC
BP	۰,۶۱	۰,۶۶	۰,۵۹	۰,۶۳	۰,۶۳
SVM	۰,۸۰	۰,۸۹	۰,۸۱	۰۸۴,	۰,۷۷
DENSE	۰,۹۲	۰,۹۶	۰,۹۵	۰,۹۵	۰,۹۰
CNN	۰,۹۲	۰,۹۷	۰,۹۳	۰,۹۴	۰,۸۹

جدول ۴-۴.معیارهای ارزیابی و AUC برای مجموعه دادگان سوم

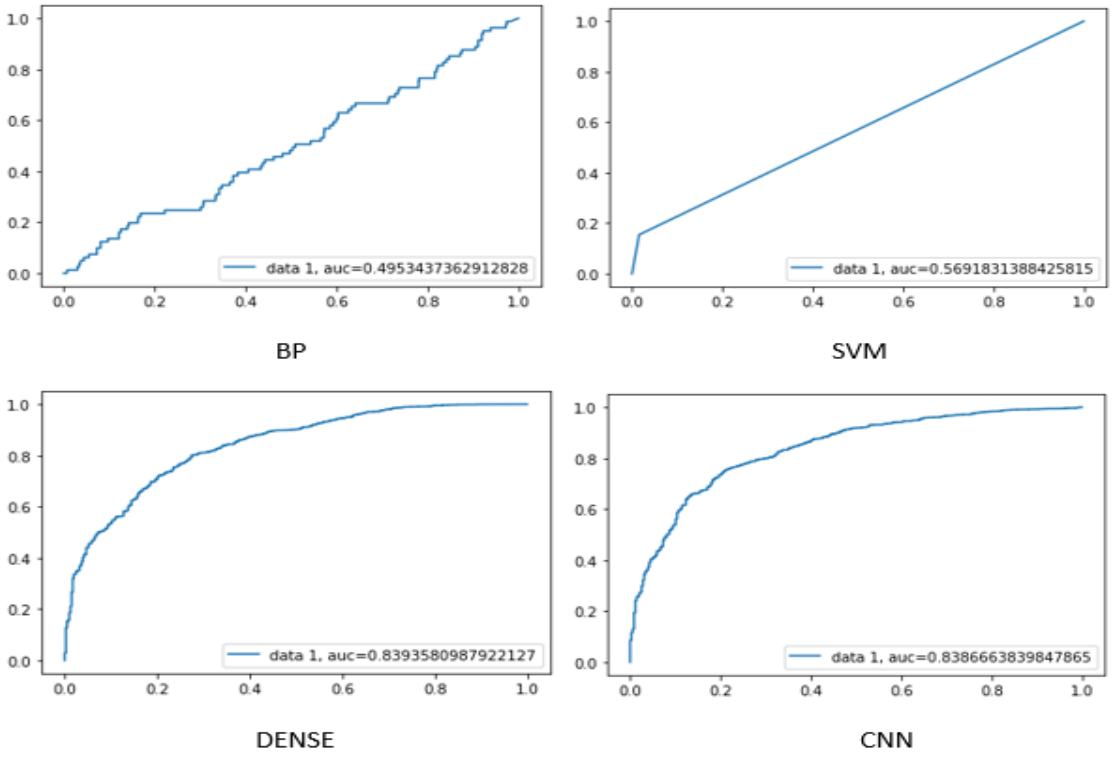
	Acc	Precision	Recall	F1-score	AUC
BP	۰,۳۴	۰,۱۳	۰,۱۲	۰,۱۲	۰,۴۹
SVM	۰,۴۶	۰,۵۱	۰,۴۶	۰,۴۸	۰,۵۶
DENSE	۰,۸۶	۰,۳۶	۰,۵۹	۰,۴۴	۰,۸۳
CNN	۰,۸۴	۰,۴۷	۰,۵۰	۰,۴۹	۰,۸۳



شکل ۴-۲.نمودار ROC مربوط به مجموعه داده اول



شکل ۴-۳.نمودار ROC مربوط به مجموعه داده دوم



شکل ۴-۴.نمودار ROC مربوط به مجموعه داده سوم

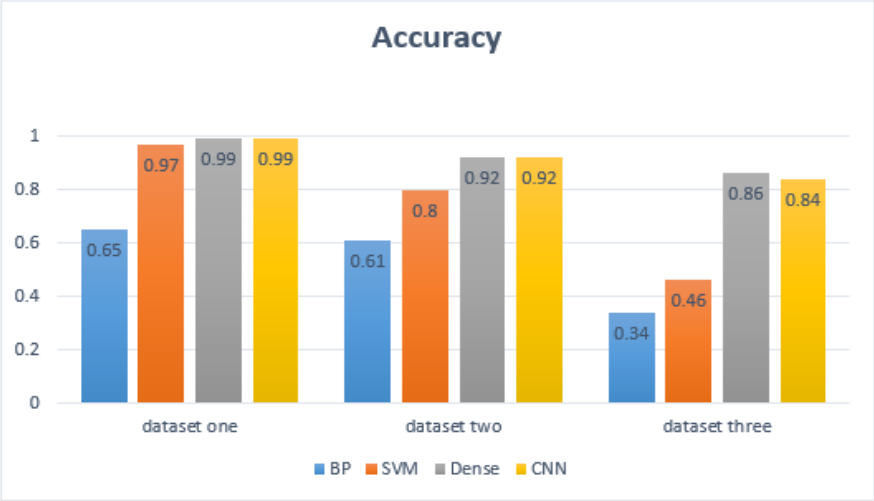
با توجه به نتایج بدست آمده در سه جدول ۴-۲، جدول ۴-۳ و جدول ۴-۴ و نمودارهای ROC هریک از روش‌های استفاده شده این چنین بر می‌آید که الگوریتم BP در تشخیص و طبقه‌بندی دامنه‌های مشکوک از سالم در هر سه مجموعه دادگان به صحتی پایین‌تر از سایر مدل‌ها دست یافته است. در مقابل SVM در مقایسه با BP بهتر عمل کرده است. اما مدل‌های یادگیری عمیق در مقایسه با هر دو روش ذکر شده در هر سه مجموعه دادگان به صحت قابل توجهی دست یافته‌اند.

نمودارهای ROC برای طبقه‌بندی کننده‌های، BP برابر ۰,۶۷، ۰,۶۳، ۰,۴۹ والگوریتم SVM به ترتیب برابر ۰,۹۴، ۰,۷۷، ۰,۵۶، مدل DENSE برابر ۰,۹۹، ۰,۹۰، ۰,۸۳و CNN به ترتیب ۰,۹۹، ۰,۸۹، ۰,۸۳ می‌باشد. طبقه‌بندی کننده SVM که بر روی بردار ویژگی ساخته شده است، از طریق یادگیری نمودار سطح زیر منحنی دامنه پرس و جو ساخته شده است، SVM بالاترین دقت تشخیص را در مقایسه با BP بدست آورده است. چرا که در هر سه مجموعه داده نمودار بالاتر از خط نیمساز قرار گرفته است، بنابراین عمل دسته‌بندی به درستی انجام شده است. این در حالی است که

طبقه‌بندی کننده BP بر اساس نمودار سطح زیر منحنی دامنه دارای کمترین دقت است. با این حال، همان‌طور که در نمودارها و جداول نشان داده شده است، مدل پیشنهادی مبتنی بر یادگیری عمیق در مقایسه با دو الگوریتم قبلی به طور قابل توجهی صحت و دقت کلی تشخیص را بهبود بخشیده است. با توجه به نتایج و نمودار سطح زیر منحنی سیستم پیشنهادی قادر به شناسایی موثر و دقیق دامنه‌های مشکوک از سالم است.

لازم به ذکر است مجموعه دادگان اول نسبت دو مجموعه دادگان دیگر بهتر عمل کرده است و این طور استنباط می‌شود که هر چه مجموعه داده گسترده شده است پارامترها مقدار کمتری نسبت به قبل پیدا کرده‌اند و دقت رو به کاهش است. توجه به این نکته ضروری است؛ داده زیاد در یادگیری عمیق همیشه باعث دقت بالا نمی‌شود چرا که داده زیاد باعث به وجود آمدن مشکلات زیادتری از جمله آموزش نامناسب، می‌شود. دلیل دیگر بد عمل‌کردن دو مجموعه دادگان دوم و سوم این است که با افزایش داده وجه تمایز داده‌ها کم و تشابه هر کدام از آن‌ها زیاد می‌شود که این به نوبه خود باعث افت شدید دقت و سایر پارامترهای ارزیابی شده است و مدل، قدرت تمایز و شناسایی صحیح دامنه‌ها را نداشته است. همچنین در توجیه کاهش صحت می‌توان گفت؛ شناسایی ما مبتنی بر اطلاعات IP است، پس هر چقدر دسته‌بند توانسته است دسته‌بندی را صحیح انجام دهد؛ بر اساس تمایز IPها در مجموعه دادگان می‌باشد که با کاهش تمایز IP، صحت کاهش یافته است.

به طور کلی تمام معیارها حاکی از کارا بودن عملکرد روش پیشنهادی با داشتن صحت بالا در مجموعه دادگان اول را دارد. مجموعه دادگان اول با صحت ۹۹ درصد بالاترین صحت را در مقایسه با سایر روش‌های طبقه‌بندی بدست آورده است. در میان سه مدل یادگیری عمیق نیز مدل DENSE در مقایسه با CNN عملکرد بهتری داشته است. نمودار مقایسه صحت مربوط به سه مجموعه دادگان مورد استفاده در شکل ۴-۵ آورده شده است. هریک از نمودارها در بازه زمانی مشخص بهترین صحت را برای هر الگوریتم نشان می‌دهد.



شکل ۴-۵.مقایسه صحت سه مجموعه دادگان

۴-۵ نتیجه‌گیری

در این بخش یک سیستم تشخیص دامنه مشکوک هوشمند با داده‌های DNS که با شناسایی دامنه‌های مشکوک از سالم و ایده جدید در شناسایی این نوع ازحملات با استفاده از سه مجموعه داده توسط زبان برنامه نویسی پایتون ارائه شد. این سیستم از داده‌های برچسب‌دار، IP و وزن بین گره‌ها برای شناسایی استفاده کرده است. همان‌طور که از نتایج نشان داده شده است، صحت تشخیص و کارایی محاسباتی سیستم‌های تشخیص فیشینگ مبتنی بر یادگیری عمیق در مقایسه با الگوریتم SVM و BP با افزایش خوبی همراه بوده است. به این معنی که بعد شناسایی این نوع حملات فیشینگ توسط سیستم پیشنهادی آدرس موردنظر فیلترشده و کاربران بیشتری در دام این نوع حملات فیشینگ قرار نمی‌گیرند. همچنین الگوریتم SVM در مقایسه با مرجع [۴۷] با افزایش ۳٪ صحت طبقه‌بندی همراه بوده است. در مقابل الگوریتم BP حتی در مقایسه با صحت بدست آمده در مرجع [۱۰] بسیار بدتر عمل شناسایی را انجام داده است و می‌توان گفت الگوریتم BP در شناسایی عملکرد ضعیفی داشته است.

فصل ۵ : نتیجه گیری و جمع بندی

۵-۱ نتیجه‌گیری و جمع‌بندی

هدف این پایان‌نامه شناسایی حملات فیشینگ و ارائه راه‌کارهای مقابله می‌باشد. به این منظور و با توجه به اینکه حملات فیشینگ و امکان شناسایی آن توسط کاربران در صورت داشتن آموزش‌های لازم در خصوص شناسایی این نوع حملات وجود دارد، نقاط قوت و ضعف و پیشنهادهایی برای توسعه این پژوهش در این قسمت ارائه می‌شود.

در این پایان‌نامه یک سیستم تشخیص هوشمند دامنه‌های مشکوک از سالم مبتنی بر یادگیری عمیق ارائه شد. درابتدا تمرکز برروی بدست‌آوردن IP دامنه‌ها و سپس ساختن گراف دو بخشی دامنه-IP و تبدیل آن به گراف وزن‌دار و بدست‌آوردن ارتباط بین دامنه‌ها و وزن بین آنها و تبدیل داده‌ها به بردار با استفاده از الگوریتم Node2vec و سپس عمل طبقه‌بندی دامنه‌ها با مدل‌های یادگیری عمیق می‌باشد. مدل‌های استفاده شده DENSE و CNN بودند. طبقه‌بند استفاده شده در لایه آخر Softmax بود. مدل پیشنهادی با یادگیری عمیق به خوبی توانست دامنه‌های سالم و مشکوک را با صحت بالایی ازهم تفکیک کند.

روش پیشنهادی جهت شناسایی باعث افزایش صحت تشخیص و کارایی محاسباتی در مقایسه با الگوریتم‌های طبقه‌بند دیگر شده است. نتایج نشان دادند مدل‌های یادگیری عمیق یک روش شناسایی قابل اعتماد هستند. در توجیه نتایج بدست آمده می‌توان گفت که دو الگوریتم BP و SVM در تشخیص به مشکل برخورده و در نتیجه صحت شناسایی پایین می‌آید؛ لذا استفاده از یادگیری عمیق به بالا بردن صحت کمک بسزایی می‌کند. به طور کلی در میان الگوریتم‌های طبقه‌بندی بهترین وجود ندارد اما با توجه به داده‌های استفاده شده و عملکرد،کارایی بدست آمده در هر یک از مدل‌ها باید الگوریتم موردنظر را انتخاب کرد. ضعف عمده روش‌های شناسایی دامنه مشکوک معمولی مبتنی بر چنین روش‌هایی در عدم توانایی آنها در استخراج و ساماندهی ویژگی‌های متمایزکننده و پویا از داده‌ها

است. در بخش بعدی به عنوان کارهایی که در آینده می‌توان این نقص را برطرف کرد پیشنهاداتی ارائه شده است.

۵-۲ پیشنهاد برای آینده

ایجاد یک چارچوب انتخاب ویژگی کاملاً اتوماتیک، انعطاف پذیر و قوی می‌تواند از طریق داده‌های مختلف به صورت عمومی مورد استفاده قرار گیرد تا زیر مجموعه‌های بهینه کاملاً مؤثر از داده‌ها تولید شود. همچنین استفاده از الگوریتم‌های پردازش زبان طبیعی و متن کاوی می‌تواند در استخراج ویژگی موثرتر، توانمند عمل نماید. انتخاب داده‌های مناسب کمک بسیار زیادی به بالارفتن صحت شناسایی خواهد کرد. از این رو به عنوان کار آینده می‌توان با استفاده از استخراج ویژگی اتوماتیک بر روی داده‌ها و استفاده از پردازش زبان طبیعی و یافتن پارامترهای موثرتر برروی شبکه عمیق و ترکیب یادگیری عمیق با سایر الگوریتم‌های هوشمند صحت را تا حد قابل توجهی بهبود بخشید.

مراجع

[۱] قیومیان، منیره، " اعتبارسنجی گواهینامه دیجیتالی با استفاده از روش‌های داده‌کاوی." پایان‌نامه کارشناسی ارشد، دانشگاه فردوسی گیلان، ۱۳۹۴.

[۲] سهولیان، سهیل، "تجارت الکترونیک در صنعت نفت و گاز." رساله دوره کارشناسی ارشد، اراک، واحد محلات، دانشگاه آزاد اسلامی، ۱۳۹۴.

[۵] شهاب، شمسی. " تکنیکهای فیشینگ و آنتی‌فیشینگ." آکادمی ایران ۱۲۳ http://www.Secureity Man.org

[۷] غروی، عرفانه و ابوالفضل اسکندرپور، ۱۳۸۹، فیشینگ، اولین کنفرانس دانشجویی فناوری اطلاعات ایران، سنندج، دانشگاه کردستان، ۱۳۹۴ (https://www.civilica.com/Paper-ISCIT01-ISCIT01_018.html)

[۸] نفیسه لنگری، مجید عبدالرزاق."شناسایی وبگاه فیشینگ در بانکداری اینترنتی با استفاده از الگوریتم بهینه‌سازی صفحات شیب‌دار." مجله علمی- پژوهشی پدافند الکترونیکی و سایبری، بهار ۱۳۹۴، سال سوم، شماره ۱.

[۹] مهدی دادخواه، محمد داورپناه‌جزی، مجید سعیدی‌مبارکه. "ارائه رویکردی به منظور شناسایی و پیش‌بینی وب‌سایت‌های فیشینگ به وسیله الگوریتم‌های کلاس‌بندی براساس مشخصه‌های صفحات وب." مجله مدل سازی در مهندسی، زمستان ۱۳۹۵، سال چهاردهم، شماره ۴۷.

[۲۱] محمد علی جعفری زاده، فرزاد غفاری جوقانی، محمد بابازاده، رقیه بایرام زاده. "الگوریتم انتشار باور کوانتومی." مقاله نامه کنفرانس فیزیک ایران، دانشگاه تبریز، ۱۳۹۴.

[3] Longley, Dennis, and Michael Shain. *Dictionary of information technology*. Macmillan International Higher Education, 1988.

[4] Li, Haifei, Jun-Jang Jeng, and Jen-Yao Chung. "Commitment-based approach to categorizing, organizing and executing negotiation processes." In *EEE International Conference on E-Commerce, 2003. CEC 2003.*, pp. 12-15. IEEE, 2003.

[6] Marchal, Samuel, Kalle Saari, Nidhi Singh, and N. Asokan. "Know your phish: Novel techniques for detecting phishing sites and their targets." In *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, pp. 323-333. IEEE, 2016.

[10] Khalil, Issa M., Bei Guan, Mohamed Nabeel, and Ting Yu. "A domain is only as good as its buddies: Detecting stealthy malicious domains via graph inference." In *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*, pp. 330-341. 2018.

[11] Khalil, Issa, Ting Yu, and Bei Guan. "Discovering malicious domains through passive DNS data graph analysis." In Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, pp. 663-674. 2016.

[12] Smadi, Sami, Nauman Aslam, and Li Zhang. "Detection of online phishing email using dynamic evolving neural network based on reinforcement learning." *Decision Support Systems* 107 (2018): 88-102.

[13] Prakash, Pawan, Manish Kumar, Ramana Rao Kompella, and Minaxi Gupta. "Phishnet: predictive blacklisting to detect phishing attacks." In *2010 Proceedings IEEE INFOCOM*, pp. 1-5. IEEE, 2010.

[14] Manadhata, Pratyusa K., Sandeep Yadav, Prasad Rao, and William Horne. "Detecting malicious domains via graph inference." In European Symposium on Research in Computer Security, pp. 1-18. Springer, Cham, 2014.

[15] Desai, Anand, Janvi Jatakia, Rohit Naik, and Nataasha Raul. "Malicious web content detection using machine leaning." In *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 1432-1436. IEEE, 2017.

[16] Zhauniarovich, Yury, Issa Khalil, Ting Yu, and Marc Dacier. "A survey on malicious domains detection through DNS data analysis." *ACM Computing Surveys (CSUR)* 51, no. 4 (2018): 1-36.

[17] Abdelhamid, Neda, Aladdin Ayesh, and Fadi Thabtah. "Phishing detection based associative classification data mining." *Expert Systems with Applications* 41, no. 13 (2014): 5948-5959.

[18] Vinayakumar, R., K. P. Soman, and Prabaharan Poornachandran. "Detecting malicious domain names using deep learning approaches at scale." *Journal of Intelligent & Fuzzy Systems* 34, no. 3 (2018): 1355-1367.

[19] https://g2b.ito.gov.ir/index.php/site/page/view/list_ip

[20] https://www.domcop.com/top-10-million-domains

[22] Statnikov, Alexander. *A gentle introduction to support vector machines in biomedicine: Theory and methods*. Vol. 1. world scientific, 2011.

[23] Cao, Hui, Takashi Naito, and Yoshiki Ninomiya. "Approximate RBF kernel SVM and its applications in pedestrian classification." 2008

[24] Support Vector Machines scikit-learn 0.20.2 documentation". Archived from the original on 2017-11-08. Retrieved 2017-11-08.

[25] Hsu, Chih-Wei, and Chih-Jen Lin. "A comparison of methods for multiclass support vector machines." *IEEE transactions on Neural Networks* 13, no. 2 (2002): 415-425.

[26] Chen, Haochen, Syed Fahad Sultan, Yingtao Tian, Muhao Chen, and Steven Skiena. "Fast and Accurate Network Embeddings via Very Sparse Random Projection." In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 399-408. 2019.

[27] Grover, Aditya, and Jure Leskovec. "node2vec: Scalable feature learning for networks." In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855-864. 2016.

[28] Grohe, Martin. "word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data." In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pp. 1-16. 2020.

[29] Cai, Hongyun, Vincent W. Zheng, and Kevin Chen-Chuan Chang. "A comprehensive survey of graph embedding: Problems, techniques, and applications." *IEEE Transactions on Knowledge and Data Engineering* 30, no. 9 (2018): 1616-1637.

[30] Collobert, Ronan. "Deep learning for efficient discriminative parsing." In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 224-232. 2011.

[31] Gomes, Lee. "Machine-learning maestro michael jordan on the delusions of big data and other huge engineering efforts." *IEEE spectrum* 20 (2014).

[32] Farshadfar, Zahra, and Marcel Prokopczuk. "Improving Stock Return Forecasting by Deep Learning Algorithm." *Advances in Mathematical Finance and Applications* 4, no. 3 (2019): 1-13.

[33] Deng, Li, and Dong Yu. "Deep learning: methods and applications." *Foundations and trends in signal processing* 7, no. 3–4 (2014): 197-387.

[34] Bengio, Yoshua. Learning deep architectures for AI. Now Publishers Inc, 2009. Now Publishers. Archived from the original (PDF) on 21 March 2014. Retrieved 17 February 2013.

[35] Zhou, Yibo, Zubair Md Fadlullah, Bomin Mao, and Nei Kato. "A deep-learning-based radio resource assignment technique for 5G ultra dense networks." *IEEE Network* 32, no. 6 (2018): 28-34.

[36] Brun, Olivier, Yonghua Yin, Erol Gelenbe, Y. Murat Kadioglu, Javier Augusto-Gonzalez, and Manuel Ramos. "Deep learning with dense random neural networks for detecting attacks against iot-connected home environments." In *International ISCIS Security Workshop*, pp. 79-89. Springer, Cham, 2018.

[37] Zafar Iffat, Giounona Tzanidou, Richard Burton, Nimesh Patel, and Leonardo Araujo. Hands-on Convolutional Neural Networks with TensorFlow: Solve Computer Vision Problems with Modeling in TensorFlow and Python. Packt Publishing Ltd, 2018.

[38] Boominathan, Lokesh, Srinivas SS Kruthiventi, and R. Venkatesh Babu. "Crowdnet: A deep convolutional network for dense crowd counting." In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 640-644. 2016.

[39] Zafar Iffat, Giounona Tzanidou, Richard Burton, Nimesh Patel, and Leonardo Araujo. *Hands-on Convolutional Neural Networks with TensorFlow: Solve Computer Vision Problems with Modeling in TensorFlow and Python*. Packt Publishing Ltd, 2018.

[40] Zeiler, Matthew D., and Rob Fergus. "Stochastic pooling for regularization of deep convolutional neural networks." *arXiv preprint arXiv:1301.3557* (2013).

[41] Romanuke, V. V. "Appropriate number and allocation of ReLUs in convolutional neural networks." *Наукові вісті Національного технічного університету України Київський політехнічний інститут* 1 (2017): 69-78.

[42] Rahbarinia, Babak, Roberto Perdisci, and Manos Antonakakis. "Efficient and accurate behavior-based tracking of malware-control domains in large ISP networks." *ACM Transactions on Privacy and Security (TOPS)* 19, no. 2 (2016): 1-31.

[43] Bilge, Leyla, Sevil Sen, Davide Balzarotti, Engin Kirda, and Christopher Kruegel. "Exposure: A passive DNS analysis service to detect and report malicious domains." *ACM Transactions on Information and System Security (TISSEC)* 16, no. 4 (2014): 1-28.

[44] Chiba, Daiki, Takeshi Yagi, Mitsuaki Akiyama, Toshiki Shibahara, Takeshi Yada, Tatsuya Mori, and Shigeki Goto. "DomainProfiler: Discovering domain names abused in future." In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 491-502. IEEE, 2016.

[45] Domain, M. 2009. Malware Domain Block List. http://www.malwaredomain.com/.

[46] M. Wepawet. http://wepawet.iseclab.org/.

[47] http://www.malwaredomainlist.com/.http://www.malwaredomainlsit.com/mdl.php

[48] LIST, Z. B. 2009b. Zeus domain blosklist http://zeustraker.abuse.ch/blocklist.php

[49] PHISHTANK. 2009. Phishtank. http://www.phishtank.com/.

[50] 2009. Alexa Web Information Company. http://www.alexa.com/topsites/.

[51] 2010. DNSBL - Spam Database Lookup. http://www.dnsbl.info/.

[52] Chiew, Kang Leng, Choon Lin Tan, KokSheik Wong, Kelvin SC Yong, and Wei King Tiong. "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system." *Information Sciences* 484 (2019): 153-166.

[53] Zuhair, Hiba, Ali Selamat, and Mazleena Salleh. "Feature selection for phishing detection: a review of research." *International Journal of Intelligent Systems Technologies and Applications* 15, no. 2 (2016): 147-162.

[54] Cui, Jia, Li Zhang, Zhaohui Liu, Juan Li, and Lei Shi. "An Efficient Framework for Online Malicious Domain Detection." In *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1-6. IEEE, 2018.

[55] Shi, Yong, Gong Chen, and Juntao Li. "Malicious domain name detection based on extreme machine learning." Neural Processing Letters 48, no. 3 (2018): 1347-1357.

[56] Does Alexa have a list of its top-ranked websites? https:// support.alexa.com/, Accessed: 2017-04-02.

[57] Gao, Hongyu, Vinod Yegneswaran, Jian Jiang, Yan Chen, Phillip Porras, Shalini Ghosh, and Haixin Duan. "Reexamining DNS from a global recursive resolver perspective." *IEEE/ACM Transactions on Networking* 24, no. 1 (2014): 43-57.

[58] Thomas, Matthew, and Aziz Mohaisen. "Kindred domains: detecting and clustering botnet domains using DNS traffic." In *Proceedings of the 23rd International Conference on World Wide Web*, pp. 707-712. 2014.

[59] Wang, Tzy-Shiah, Hui-Tang Lin, Wei-Tsung Cheng, and Chang-Yu Chen. "DBod: Clustering and detecting DGA-based botnets using DNS traffic analysis." *Computers & Security* 64 (2017): 1-15.

[60] Sun, Xiaoqing, Mingkai Tong, Jiahai Yang, Liu Xinran, and Liu Heng. "Hindom: A robust malicious domain detection system based on heterogeneous information network with transductive classification." In *22nd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2019)*, pp. 399-412. 2019.

[61] Passivedns.cn. Sign In-passiveDNS. https:// passivedns.cn, 2019. [Online].

[62] Malwaredomainlist.com. MDL. https://www. malwaredomainlist.com, 2019. [Online].

[63] Malwaredomains.com. DNS-BH – Malware Domain Blocklist by RiskAnalytics. http://www. malwaredomains.com, 2019.

[64] Jiang, Nan, Jin Cao, Yu Jin, Li Erran Li, and Zhi-Li Zhang. "Identifying suspicious activities through dns failure graph analysis." In *The 18th IEEE International Conference on Network Protocols*, pp. 144-153. IEEE, 2010.

[65] Yedidia, Jonathan S., William T. Freeman, and Yair Weiss. "Understanding belief propagation and its generalizations." *Exploring artificial intelligence in the new millennium* 8 (2003): 236-239.

[66] Leifer, Matthew S., and David Poulin. "Quantum graphical models and belief propagation." *Annals of Physics* 323, no. 8 (2008): 1899-1946.

[67] Lei, Kai, Qiuai Fu, Jiake Ni, Feiyang Wang, Min Yang, and Kuai Xu. "Detecting Malicious Domains with Behavioral Modeling and Graph Embedding." In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 601-611. IEEE, 2019.

[68] Chiba, Daiki, Takeshi Yagi, Mitsuaki Akiyama, Toshiki Shibahara, Takeshi Yada, Tatsuya Mori, and Shigeki Goto. "DomainProfiler: Discovering domain names abused in future." In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 491-502. IEEE, 2016.

[69] Zuhair, Hiba, Ali Selamat, and Mazleena Salleh. "Feature selection for phishing detection: a review of research." *International Journal of Intelligent Systems Technologies and Applications* 15, no. 2 (2016): 147-162.

[70] Antonakakis, Manos, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster. "Building a dynamic reputation system for dns." In *USENIX security symposium*, pp. 273-290. 2010.

[71] Bilge, Leyla, Engin Kirda, Christopher Kruegel, and Marco Balduzzi. "EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis." In *Ndss*, pp. 1-17. 2011.

[72] Khalil, Issa M., Bei Guan, Mohamed Nabeel, and Ting Yu. "A domain is only as good as its buddies: Detecting stealthy malicious domains via graph inference." In *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*, pp. 330-341. 2018.

[73] Active DNS Project. https://activednsproject.org/. Accessed: 17-04- 2017.

[74] Black Hole DNS. Black hole dns list. http://www.malwaredomains.com / bhdns.html/. Accessed: 17-05-2017

[75] The DNS-BH project. DNS-BH – Malware Domain Blocklist. http: //www.malwaredomains.com/. Accessed: 16-05-2016.

[76] Zeus Tracker. Zeus domain blocklist. https://zeustracker.abuse.ch/.Accessed: 17-05-2017.

[77] Shi, Yong, Gong Chen, and Juntao Li. "Malicious domain name detection based on extreme machine learning." *Neural Processing Letters* 48, no. 3 (2018): 1347-1357.

[78] Does Alexa have a list of its top-ranked websites? https:// support.alexa.com/, Accessed: 2017-04-02.

[79] https://github.com/baderj/domain-generation-algorithms, Accessed: 2017-04-05.

[80] Bambenek consulting – master feeds, http://osint.bambene kconsulting.com/. Accessed: 2016-04-05

[81] Bollobás, Béla. Modern graph theory. Vol. 184. Springer Science & Business Media, 2013.

[82] Townsend, James T. "Theoretical analysis of an alphabetic confusion matrix." *Perception & Psychophysics* 9, no. 1 (1971): 40-50.

[83] Westgard, James O., R. Neill Carey, and Svante Wold. "Criteria for judging precision and accuracy in method development and evaluation." *Clinical chemistry* 20, no. 7 (1974): 825-833.

[84] Chu, Kevin. "An introduction to sensitivity, specificity, predictive values and likelihood ratios." *Emergency Medicine* 11, no. 3 (1999): 175-181.

[85] Altman, Douglas G., and J. Martin Bland. "Diagnostic tests 3: receiver operating characteristic plots." BMJ: British Medical Journal 309, no. 6948 (1994): 188.

[86] Bowers, Alex J., and Xiaoliang Zhou. "Receiver operating characteristic (ROC) area under the curve (AUC): A diagnostic measure for evaluating the accuracy of predictors of education outcomes." Journal of Education for Students Placed at Risk (JESPAR) 24, no. 1 (2019): 20-46.

Abstract

With the advancement of technology in the Internet, one of the most important security challenges and risks is phishing attacks and fraud by Fishers. Phishing is a type of cyber attack, which always tries to get information such as username, password, bank account information and the like by forging a website, email address and convincing the user to enter this information.
Although phishing involves psychological deception and relies on human error instead of software or hardware errors, users play a major role in preventing them from being caught in this type of attack. On the other hand, it is not possible to detect suspicious domains by users and current phishing detection systems often can not adapt to new attacks and have low detection accuracy and high false positive rates.
Graph-based methods are one of the methods for identifying malicious domains. The main challenge in this dissertation is how to increase the accuracy of recognizing these domains with Graph-based method and deep learning. Therefore, in this dissertation, Graph-based phishing detection system using deep learning is presented.
The key idea of this method is to extract IP from the domain, define the relationship between the domains, their weight and also convert the data to vector by Node2vec algorithm. Then, using CNN and DENSE deep learning models, the classification and identification operation is performed. The results showed that the Graph-based method and the use of deep learning have a favorable effect on identifying these domains. Thus that the proposed method has 99% accuracy in the first data set, which has increased by 1% compared to the previous best method, BP.

Keywords: Benign and Malicious Domain Detection, DNS Data, Phishing

Shahrood University of Technology

Faculty of Computer Engineering

M.Sc. Thesis in Artificial Intelligence

Malicious Domain Detection Using DNS Records

By: Fahime Bagheri

Supervisor:
Dr. Mohsen Rezvani

Advisor:
Dr. Mansoor Fateh

Dr. Esmaeel Tahanian

October 2020