
بسم الله الرحمن الرحيم

دانشگاه صنعتی شریف

دانشکده مهندسی برق

پایان‌نامه کارشناسی

گرایش بایوالکتریک

عنوان:

حملات مقابله‌ای و خصمانه و مقایسه راه‌های مقابله با آن

نگارش:

ثنا امین ناجی

استاد راهنما:

دکتر محمدباقر شمس الهی

زمستان ۱۴۰۱

چکیده

با گسترش سریع و بی سابقه تکنیک‌های هوش مصنوعی و یادگیری عمیق و کـاربرد گسـترده آنهـا در زمینـه‌های مختلف اعم از طبقه‌بندی تصاویر، تشخیص صدا، بازسازی مدارهای مغز، بررسی جهش‌های ژنتیکـی و مـوارد بیشـمار دیگـر، اطمینان از امنیت و صحت خروجی‌های تولید شده توسط این سیستم‌ها بسیار مهم و حائز اهمیت است. اخیـرا آسـیب‌پذیری سیستم‌ها توسط حملات متخاصم مشاهده و شناسایی شده است. نمونه‌های مذکور میتوانند منجـر بـه رفتارهـای نادرسـت و متعاقبا عملکرد نامطلوب سیستم یادگیری عمیق شده در صورتیکه توسط ناظر زنده قابل تمایز نبوده‌اند. پیاده‌سازی موفقیـت آمیز این حملات در کاربردهای واقعی و روزمره اهمیت بررسی و مقابله با این مشکل را دوچندان میکند [۱] . در این پژوهش ابتدا به شناخت و معرفی سیستم‌های یادگیری، حملات و مقابله‌های آن‌ها پرداخته، سپس چند نمونه از تولید حمله و مقابلـه مناسب آن‌ها را به تفصیل بررسی میکنیم. در انتها به مقایسه مختصر روش هـای ذکـر شـده پرداختـه و جمع‌بنـدی مقـالات بررسی شده را ارائه خواهیم داد.

واژه‌های کلیدی

سیستم یادگیری عمیق، شبکه عصبی، حمله خصمانه، حمله مقابله‌ای، مقابله، دفاع، دیتاست، آموزش شبکه

فهرست

فصل اول: مقدمه

۱-۱ پیش درآمد

یادگیری عمیق شاخه‌ای از یادگیری ماشین است که مدل‌های محاسباتی متشکل از تعداد زیادی لایه پردازشی با سطح بالایی از مشاهدات را برای شبیه‌سازی تجربه‌های زیستی و روزمره در اختیار میگذارد. این نوع یادگیری از الگوریتم backpropagation برای کشف جزئیات پیچیده در داده‌های بزرگ به منظور محاسبه میزان اهمیت و دخالت داده لایه قبل در لایه فعلی استفاده میکند. با تکامل مدل‌های شبکه‌های عمیق و دسترسی به سخت‌افزارهای پیشرفته با قدرت پردازشی بالا، این شبکه‌ها پیشرفت به‌سزایی در مدل‌های سنتی طبقه‌بندی تصاویر، تشخیص صدا، ترجمه و همچنین در زمینه های پیچیده و تازه‌تر از جمله بازسازی مدارات مغزی، تحلیل و آنالیز داروها، جهش‌های ژنتیکی در DNA را رقم زدند. یکی از کابردهای گسترده شبکه‌های یادگیری عمیق در محیط‌های امنیتی از جمله اتومبیل‌های خودران، شناسایی انواع بدافزارها و هواپیماهای بدون سرنشین است. از هنگامی که این شبکه‌ها و کاربردهای آن‌ها وارد زندگی واقعی شده‌اند امنیت این برنامه‌ها به یک دغدغه اساسی تبدیل شده است [۲]. می توان هر سیستم یادگیری ماشینی را به طبقه بندی اشتباه تصاویر با اطمینان بالا با اضافه کردن برخی اختلالات نامحسوس بر روی آنها سوق داد. این تغییر برای یک ناظر انسانی تقریباً غیرقابل تشخیص است. بنابراین، تصاویر متخاصم به دلیل این واقعیت که سیستم‌های یادگیری ماشینی می‌توانند در معرض حمله انجام شده توسط این تصاویر قرار بگیرند، یک تهدید جدی به شمار می‌آیند. طبیعتاً، اهمیت امنیت و حریم شخصی در برنامه‌های یادگیری ماشینی در چند سال گذشته افزایش یافته است [۳]. مقاومت شبکه‌های یادگیری عمیق در مقابل حملات خصمانه به منظور مقاوم‌سازی ای شبکه‌ها در برابر طیف وسیعی از حملات مورد مطالعه جامع قرار گرفته است. در این پژوهش تلاش شده است که علاوه بر تعریف حمله و انواع آن، بررسی انواع مقابله، مقایسه‌ای جامع بین انواع روش‌های دفاع انجام شده و نتایج به شیوه کامل و قابل فهمی ارائه شوند.

۱-۲ چالش‌ها

در بررسی و شناخت شبکه‌های عمیق و همچنین حملات تعریف شده برای هر یک از شبکه‌ها یکی از اصلی‌ترین چالش‌ها اختصاصی بودن هر حمله و به طور متعاقب مقابله متناسب برای هر کاربرد و روش خاص است. این ویژگی مقایسه کارآرایی روش‌های مقابله را به شدت سخت کرده و الزام یافتن متغیرهای ثابت و قابل استخراج برای اکثر انواع حملات را بیش از پیش مشخص می‌کند. همچنین افزایش روزافزون طراحی و تولید شبکه‌های یادگیری عمیق و کثرت مقالات و پژوهش‌های موجود در این زمینه انتخاب روش‌های مناسب به منظور بررسی و مقایسه را به امری چالش برانگیز تبدیل می‌کند.

۱-۳ اهداف

اهداف در نظر گرفته شده در این پژوهش به چهار بخش ذکر شده در زیر تقسیم میشوند:

- شناخت انواع حمله
- شناخت انواع مقابله
- انتخاب و بررسی مثال‌های تولید حمله و مقابله
- تعریف معیاری مناسب برای مقایسه‌ای کاربردی و قابل استناد بین روش‌های مختلف دفاع و حمله

۱-۴ ساختار گزارش

در این گزارش ابتدا به معرفی و توصیف مفاهیم اصلی مورد نیاز در پژوهش (شبکه عصبی، یادگیری عمیق، حمله، مقابله) می‌پردازیم (فصل دوم)۰ در ادامه تلاش بر بررسی مثال‌های مختلف حملات و دفاع‌های متناظر داریم؛ در این بخش به بررسی نوع دیتاست مورد بحث، شبکه عصبی استفاده شده برای طبقه‌بندی، حمله تولید و اعمال شده، استراتژی مقابله و

در نهایت نتایج اعمال حمله و مقابله متناظر می‌پردازیم (فصل سوم). بخش آخر (فصل چهارم) به جمع‌بندی مقالات ذکر شده، نتیجه گیری مباحث بررسی شده و در نهایت پیشنهادات قابل استفاده برای ادامه پژوهش اختصاص یافته است.

فصل دوم: مفاهیم کلی

۲-۱ مقدمه

قبل از اینکه در مورد مدل‌های حمله و اقدامات متقابل آنها صحبت کنیم، در این بخش یک طبقه‌بندی کیفی در مورد اصطلاحات مختلف و کلمات کلیدی مرتبط با حملات خصمانه ارائه می‌کنیم و مدل‌های تهدید را دسته‌بندی می‌کنیم. در این بخش، ما رویکردهای عمدتاً مورد استفاده را با تأکید بر شبکه‌های عصبی برای حل مشکلات یادگیری ماشین و کاربرد مربوطه آنها خلاصه می‌کنیم [۲].

۲-۲ مفاهیم کلی

۲-۲-۱ شبکه عصبی

شبکه‌های عصبی مصنوعی (ANN) که از شبکه‌های عصبی بیولوژیکی الهام گرفته شده‌اند، ساختاری بر اساس مجموعه‌ای از پرسپترون‌ها به نام نورون‌ها دارند. هر نورون مجموعه ای از ورودی ها را با استفاده از یک تابع فعال سازی به خروجی نگاشت می کند. یادگیری بر وزن ها و عملکرد فعال سازی تعریف میشود تا بتوان خروجی را به درستی تعیین کرد. وزن‌ها در یک فیدبک چند لایه توسط الگوریتم backpropagation به‌روز می‌شوند. نورون اولین بار توسط مک کالوخ پیتس معرفی شد و به دنبال آن قانون یادگیری هب، در نهایت منجر به ایجاد پرسپترون فیدبک چند لایه و الگوریتم backpropagation شد [۲].

شبکه‌های عصبی به دو گروه عمده یادگیری همراه ناظر (supervised) و بدون ناظر (unsupervised) تقسیم می‌شوند. پارامتر موثر در این تقسیم بندی وجود یا عدم وجود خروجی مشخص به ازای هر ورودی می‌باشد.

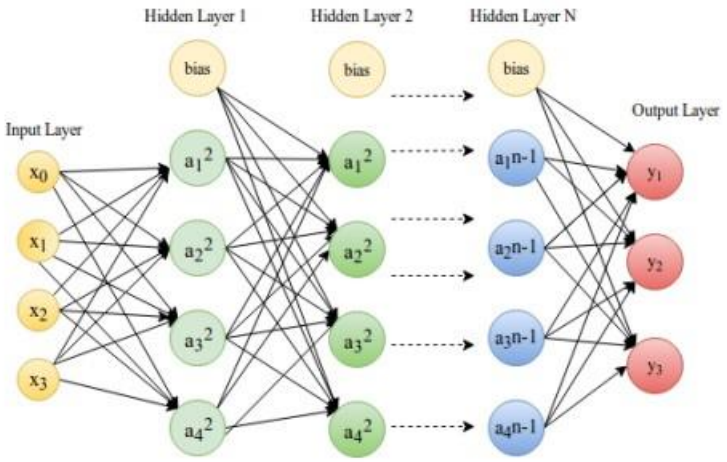
۲-۲-۲ شبکه عمیق

شبکه‌های عمیق شاخه‌ای از شبکه‌های عصبی با تعداد لایه‌های بیشتر هستند. به دو گروه مرجع تقسیم میشوند:

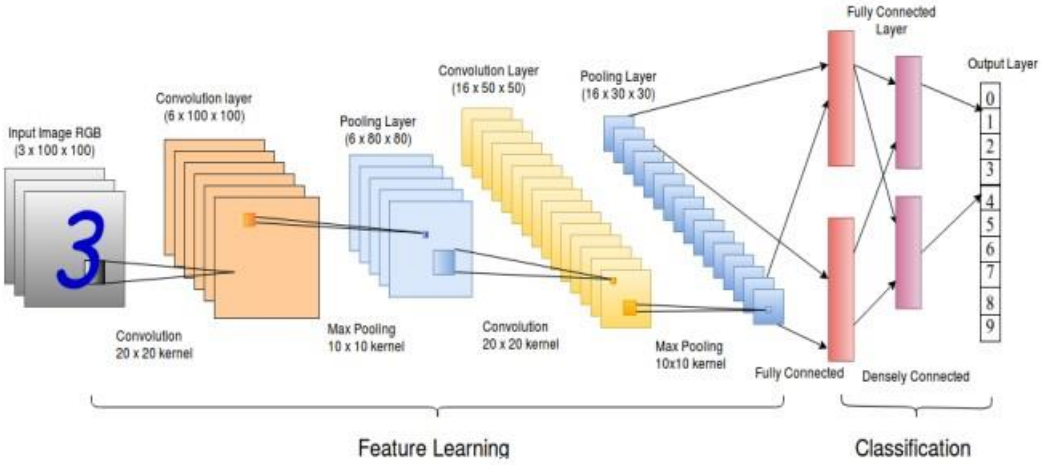
- DNN: در حالی که شبکه عصبی تک لایه یا پرسپترون یک رویکرد مهندسی ویژگی است، شبکه عصبی عمیق یادگیری ویژگی را با استفاده از داده های خام به عنوان ورودی امکان پذیر می کند. لایه های پنهان چندگانه و

اتصالات متقابل آن ویژگی ها را از ورودی پردازش نشده استخراج می کنند و بنابراین با یافتن ساختارهای نهفته در داده های بدون برچسب و بدون ساختار، عملکرد را افزایش می دهند. یک معماری معمولی DNN که به صورت گرافیکی در شکل۱-۱ نشان داده شده است، از چندین لایه متوالی (حداقل دو لایه پنهان) نورون تشکیل شده است. هر لایه پردازشی را می توان به عنوان یادگیری یک نمایش متفاوت و انتزاعی تر از توزیع ورودی چند بعدی اصلی مشاهده کرد. به طور کلی، یک DNN را می توان به عنوان یک تابع بسیار پیچیده در نظر گرفت که قادر است به صورت غیرخطی نقاط داده با ابعاد بالا اصلی را به یک فضای با ابعاد پایین تر نگاشت کند.

- CNN: یک شبکه عصبی کانولوشن از یک یا چند لایه کانولوشن یا نمونه فرعی تشکیل شده است که به دنبال آن یک یا چند لایه کاملاً به هم متصل می‌شوند تا وزن‌ها را به اشتراک بگذارند و تعداد پارامترها را کاهش دهند. معماری CNN، در شکل۱-۲ نشان داده شده است، برای استفاده از ساختار ورودی دوبعدی (به عنوان مثال تصویر ورودی) طراحی شده است. لایه پیچیدگی یک نقشه ویژگی ایجاد می‌کند. ادغام (که نمونه‌برداری فرعی یا پایین‌نمونه‌سازی نیز نامیده می‌شود) ابعاد هر نقشه ویژگی را کاهش می‌دهد، اما مهم‌ترین اطلاعات را برای داشتن مدلی مقاوم در برابر اعوجاج‌های کوچک حفظ می‌کند. به عنوان مثال، برای توصیف یک تصویر بزرگ، مقادیر ویژگی در ماتریس اصلی را می‌توان در مکان‌های مختلف (مثلاً حداکثر ادغام) جمع کرد تا ماتریسی با ابعاد پایین‌تر تشکیل شود. آخرین لایه کاملاً متصل از ماتریس ویژگی تشکیل شده از لایه‌های قبلی برای طبقه بندی داده ها استفاده می‌کند. CNN عمدتاً برای استخراج ویژگی استفاده می شود، بنابراین در پیش پردازش داده ها نیز کاربرد دارد که معمولاً در تشخیص تصویر استفاده می شود [۲].



شکل (۱-۱) : ساختار یک شبکه DNN [۲]



شکل (۱- ۲) : ساختار یک شبکه CNN [۲]

۲-۲-۳ آموزش شبکه

به طور کلی یک شبکه عصبی را مشابه زیر معرفی میکنیم [۱]:

- دیتاست: $\{x_i, y_i\}_{i=1}^{N}$
  - ورودی: x_i
  - لیبل: y_i
  - اندازه دیتاست: N
- تابع هزینه: $J(\theta, x, y)$
  - متغیر وزن‌ها: θ
- معرف شبکه: $f(x)$

آموزش شبکه‌های عصبی به روش‌های متفاوتی انجام میگیرد، عمده‌ترین این روش‌ها الگوریتم backpropagation به شمار میرود.

الگوریتم backpropagation به شبکه های عصبی فیدبک‌دار چندلایه اجازه می دهد تا نگاشت ورودی/ خروجی را از نمونه‌های آموزشی یاد بگیرند. این شبکه‌ها خود را برای یادگیری رابطه بین مجموعه الگوهای نمونه تطبیق می‌دهند و می‌توانند همان رابطه را برای الگوهای ورودی جدید اعمال کنند. شبکه باید بتواند روی ویژگی های یک ورودی دلخواه

تمرکز کند. تابع فعال سازی برای تبدیل سطح فعال سازی یک واحد (نورون) به سیگنال خروجی استفاده می شود [٤]. در این شیوه وزن‌ها به صورت بازگشتی و از لایه آخر به اول به منظور کمینه کردن تابع هزینه و به تبع خطا اصلاح می‌شوند.

۲-۳ حملات

به طور کلی یک حمله خصمانه به آشفتگی‌هایی گفته می‌شود که در ورودی یک سیستم یادگیری عمیق ایجاد شده و باعث فریب سیستم و ایجاد خروجی نامعتبر میشود؛ این آشفتگی‌ها که برای بینایی/شنوایی انسان نامحسوس هستند، برای ترغیب مدل به پیش‌بینی اشتباه با اطمینان بالا کافی هستند:

$$x' : D(x, x') < \eta, f(x') \neq y \qquad (۱-۲)$$

در این معادله D تابع فاصله است.

تقسیم بندی کلی ارائه شده برای حملات از روی شیوه تولید آن‌ها به شرح زیر است:

- جعبه سیاه (black-box): یک دشمن ساختار شبکه هدف یا پارامترها را نمی‌داند، اما می‌تواند با الگوریتم DL تعامل کند تا پیش‌بینی‌های ورودی‌های خاص را جستجو کند. دشمنان همیشه نمونه‌های متخاصم را روی یک طبقه‌بندی‌کننده جایگزین آموزش داده شده توسط جفت‌های داده و پیش‌بینی به‌دست‌آمده و سایر نمونه‌های متضاد می‌سازند. با توجه به قابلیت انتقال نمونه‌های متخاصم، حملات جعبه سیاه همیشه می توانند یک مدل غیردفاعی آموزش دیده طبیعی را به خطر بیندازند.
- جعبه خاکستری (gray-box): فرض بر این است که یک دشمن معماری مدل هدف را می داند، اما به وزن‌های موجود در شبکه دسترسی ندارد. حریف همچنین می تواند با الگوریتم DL تعامل داشته باشد. در این مدل تهدید، از دشمن انتظار می‌رود نمونه‌های متخاصم را بر روی یک طبقه‌بندی جایگزین از همان معماری بسازد. با توجه به اطلاعات ساختار اضافی، یک دشمن جعبه خاکستری همیشه عملکرد حمله بهتری را در مقایسه با دشمن جعبه سیاه نشان می‌دهد.
- جعبه سفید (white-box): قوی‌ترین نوع حمله به مدل هدف، به تمام پارامترها دسترسی کامل دارد، به این معنی که دشمن می‌تواند حملات را تطبیق دهد و مستقیماً نمونه‌های متخاصم را روی مدل هدف ایجاد کند.

دلایل معمول برای اشتباه در طبقه‌بندی ورودی‌ها (موفقیت یک حمله تولید شده) [۵]:

- حمله از مرز گزینه‌های موجود در طبقه‌بندی بسیاردور است. به طور مثال در طبقه‌بندی اعداد دست‌نویس یک ورودی تحت عنوان تصویر یک حیوان به سیستم می‌دهیم و شبکه توان عدم طبقه‌بندی نداشته و مجبور به نسبت دادن یک برچسب به ورودی می‌شود.
- حمله به مرز یک طبقه اشتباه نزدیک می‌باشد، در این صورت شبکه طبقه‌بندی را انجام می‌دهد اما به صورت کاملا اشتباه.

۲-۴ رویکردهای کلی مقابله با حملات

مشکل یادگیری متخاصم به طور طبیعی شبیه بازی بین سیستم یادگیری و حریف است. در چنین بازی‌ای، هر دو بازیکن سعی می‌کنند بهترین استراتژی‌های خود را در مقابل یکدیگر انجام دهند و در عین حال سود خود را به حداکثر برسانند [۶].

در نتیجه تولید دفاع متناسب باید با توجه به نوع حمله تنظیم شود و دفاع‌ها تا حدودی کارکرد اختصاصی دارند.

به طور کلی راه‌های مقابله یک حمله در سه رویکرد جامع خلاصه می‌شوند [۵]:

- آموزش شبکه اصلی با نمونه‌های متخاصم.
- آموزش یک شبکه ثانویه برای تشخیص و تمایز نمونه سالم از متخاصم.
- مقاوم سازی شبکه در مقابل نمونه‌های متخاصم که منجر به برچسب‌زنی اشتباه نشود.

۲-۵ جمع‌بندی

در این بخش به تعریف مفاهیم کلی پرداخته و با این مفاهیم آشنا شدیم.

- شبکه‌های عصبی مصنوعی، ساختاری بر اساس مجموعه‌ای از پرسپترون‌ها به نام نورون‌ها دارند. هر نورون مجموعه ای از ورودی ها را با استفاده از یک تابع فعال سازی به خروجی نگاشت می کند.

⊂ شبکه‌های عمیق شاخه‌ای از شبکه‌های عصبی با تعداد لایه‌های بیشتر هستند. به دو گروه مرجع DNN, CNN تقسیم میشوند.

- آموزش شبکه‌های عصبی به روش‌های متفاوتی انجام میگیرد، عمده‌ترین این روش‌ها الگوریتم backpropagation به شمار میرود.
- یک حمله خصمانه به آشفتگی‌هایی گفته میشود که در ورودی یک سیستم یادگیری عمیق ایجاد شده و باعث فریب سیستم و ایجاد خروجی نامعتبر میشود.

فصل سوم: روش‌های مختلف تولید حمله و دفاع متناظر

۳-۱ مقدمه

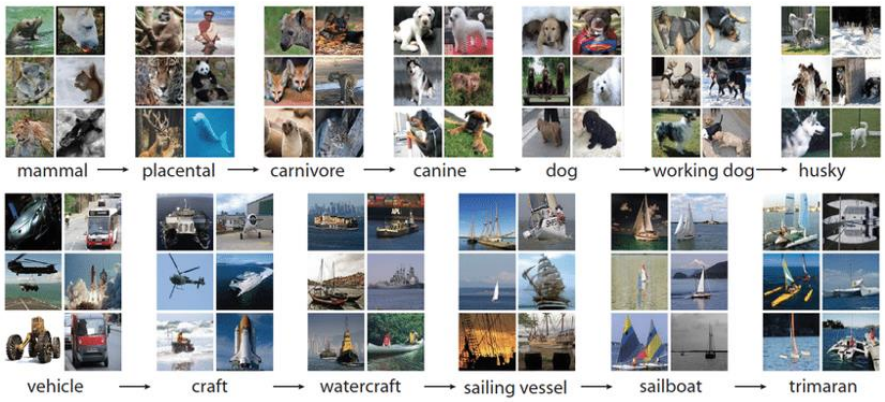
در این فصل قصد داریم تا تعدادی از مثال‌های تولید حمله روی یک شبکه و پس از آن راهکار مقاوم سازی شبکه در برابر آن حمله را مورد بررسی قرار داده تا بتوانیم در انتها به مقایسه‌ای جامع و کاربردی در این خصوص دستیابیم.

۳-۲ روش اول: Denoiser

از آنجا که عمده حملات از طریق اضافه کردن یک نویز ناچیز به ورودی‌های اولیه تولید میشوند در نتیجه یک راه مقابله با آنها استفاده از denoiser است [۷].

۳-۲-۱ معرفی دیتاست

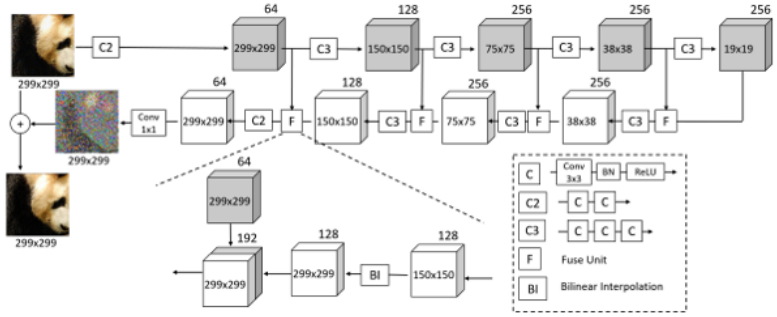
برای تولید دیتاست در این روش از دیتاست آماده ImageNet استفاده میشود. یک پایگاه داده تصویری است که بر اساس سلسله مراتب WordNet سازماندهی شده است، که در آن هر گره از سلسله مراتب توسط صدها و هزاران تصویر به تصویر کشیده می شود. این پروژه در پیشبرد بینایی کامپیوتر و تحقیقات یادگیری عمیق بسیار مفید بوده است. داده ها برای استفاده غیرتجاری به صورت رایگان در دسترس محققان است (شکل۳-۱).



شکل (۳-۱) : ImageNet [۸]

۳-۲-۲ معرفی شبکه

برای تولید شبکه از شبکه آماده DUNET استفاده شده است ساختار کلی شبکه در شکل۳-۲ نشان داده شده است.



شکل (۳-۲) : ساختار شبکه یادگیری DUNET [۷]

۳-۲-۳ معرفی حمله

حمله در این پژوهش در دو نوع جعبه سیاه و جعبه سفید تولید شده است در هر دو این بخش‌ها از روش FGSM(Fast Gradient Sign Method) استفاده شده است [۹].

- FGSM: هدف اصلی این روش ساخت نمونه مخرب با اضافه کردن مقدار کوچکی نویز به یک نمونه معتبر است:

$$w^T x' = w^T x + w^T \eta \qquad (۱-۳)$$

به طوریکه:

$$\eta = \varepsilon * sign\left(\nabla_x J(\theta, x, y)\right) \qquad (۲-۳)$$

۳-۲-۴ معرفی مقابله

در این بخش به معرفی دو روش متفاوت برای تولید یک denoiser مناسب میپردازیم و در بخش بعد نتیجه اعمال این روش‌های مقابله را بر روی شبکه بررسی میکنیم.

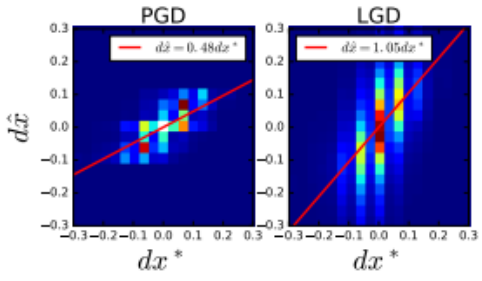
- PGD: در این روش یک تابع برای این denoiser تعریف میکنیم به نام D که دامنه آن ورودی‌های مخرب و خروجی آن $\hat{\mathbf{x}}$ است. این خروجی را با استفاده از کمینه کردن تابع هزینه معادله (۳-۳) بدست می‌آوریم.

$$L = \left|\left|x - \hat{\mathbf{x}}\right|\right| \qquad (۳-۳)$$

- LGD: این روش مانند حالت بالاست با این تفاوت که به جای استفاده از ورودی‌های ابتدایی در تابع هزینه از ورودی‌های لایه‌های بالاتر استفاده می‌کنیم. این تغییر به این منظور ایجاد می‌شود که حتی تفاوت‌های کم با ورودی سالم با پیشرفت در شبکه تقویت شده و گاها منجر به نتایج نامطلوب می‌شوند. در نتیجه استفاده از ورودی‌های لایه‌های بالاتر باعث تقویت کمتر این اختلاف می‌شود.

۳-۲-۵ نتایج

نتایج استفاده از این روش در شکل‌های۳-۳ و ۳-۴ آمده است و همانطور که مشاهده میشود روش LGD بهتر از روش PGD عمل میکند.



شکل (۳-۳) : نتایج-فاصله نویز حذف شده و فاصله نویز سوار بر نمونه متخاصم [۷]

The transferability of HGD to different model. Resnet is used as the target model.

Denoiser for Resnet	Clean	WhiteTestSet		BlackTestSet	
		$\epsilon = 4$	$\epsilon = 16$	$\epsilon = 4$	$\epsilon = 16$
NA	**78.5%**	63.3%	38.4%	67.8%	48.6%
IncV3 guided LGD	77.4%	75.8%	71.7%	76.1%	72.7%
Resnet guided LGD	78.4%	**76.1%**	**72.9%**	**76.5%**	**74.6%**

The transferability of HGD to different classes. The 1000 ImageNet classes are separated in training and test test.

Defense	Clean	WhiteTestSet		BlackTestSet	
		$\epsilon = 4$	$\epsilon = 16$	$\epsilon = 4$	$\epsilon = 16$
NA	**76.6%**	15.4%	15.3%	61.5%	41.7%
LGD	76.3%	**73.9%**	**65.7%**	**74.8%**	**72.2%**

شکل (۳-٤) : جدول نتایج-افزایش دقت با استفاده از راهکار مقابله‌ای مذکور [۷]

۳-۳ روش دوم: detector subnetwork

هدف کلی این روش این است که شبکه‌های عصبی عمیق را با یک زیرشبکه کوچک «آشکارساز» تقویت کند که بر روی وظیفه طبقه‌بندی دودویی برای تشخیص داده‌های واقعی از داده‌های حاوی آشفتگی‌های متخاصم آموزش دیده است [۱۰].

۳-۳-۱ معرفی دیتاست

در این مثال از دو نوع دیتاست مختلف استفاده شده است.

- ImageNet: این دیتاست در بخش قبل معرفی شده است (شکل۳-۱). در این بخش از ۱۰ کلاس این مجموعه استفاده شده است.
- CIRFAR10: کیفیت این دیتاست کمتر بوده و آموزش و آزمایش ابتدایی روی این دیتاست انجام شده است.

شکل (۳-۵) : CIRFAR10 [۱۱]

۳-۳-۲ معرفی شبکه

برای تولید شبکه طبقه بندی دو شبکه مختلف هریک برای یکی از دیتاست‌ها استفاده میکینم.

- ImageNet: برای این دیتاست از شبکه طبقه بندی کننده Pretrained VGG16 استفاده میکنیم.
- CIRFAR10: برای این دیتاست از شبکه طبقه بندی کننده ۳۲ لایه‌ای Residual Network استفاده میکنیم.

۳-۳-۳ معرفی حمله

به منظور تولید حمله باید به یک نکته توجه داشت؛ دشمن میتواند از وجود ساختار آشکارساز بیخبر باشد که در این صورت تولید حمله به صورت ساکن یا Static انجام میگیرد، در صورت اطلاع دشمن از حضور آشکار ساز تولید حمله از ویژگی‌های آشکارساز نیز تاثیر میپذیرد و ورودی خصمانه پویا یا Dynamic تولید میشود.

- متد سریع: این حالت همان FGSM است. که در بخش قبل توضیح داده شد.
- متد برای حالت پویا:

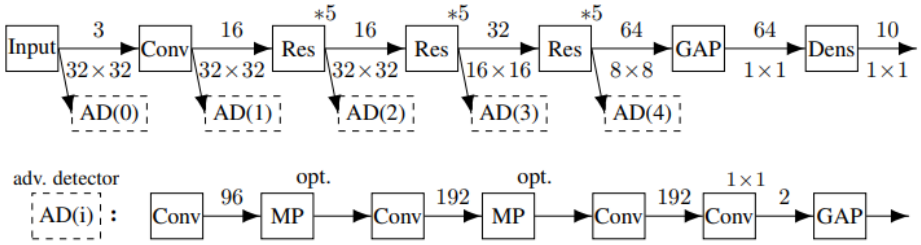
(۳-٤)

$$x_0^{\text{adv}} = x;\ \ x_{n+1}^{\text{adv}} = \text{Clip}_x^{\varepsilon}\left\{x_n^{\text{adv}} + \alpha\left[(1-\sigma)\,\text{sgn}(\nabla_x \text{J}_{\text{cls}}(x_n^{\text{adv}}, y_{\text{true}}(x))) + \sigma\,\text{sgn}(\nabla_x \text{J}_{\text{det}}(x_n^{\text{adv}}, 1))\right]\right\}$$

۳-۳-۴ معرفی مقابله

ما شبکه‌های طبقه‌بندی را با زیرشبکه‌های (نسبتاً کوچک) تقویت می‌کنیم که از شبکه اصلی در برخی از لایه‌ها منشعب می‌شوند و یک خروجی $p^{adv} \in [۰.۱]$ تولید می‌کنند که به عنوان احتمال مخرب بودن ورودی تفسیر می‌شود.

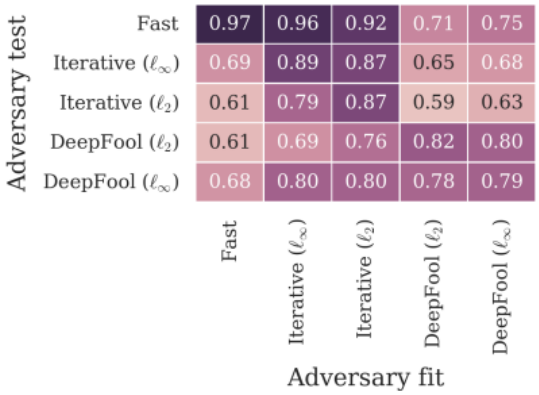
بنابراین ما یک مجموعه داده طبقه بندی باینری متعادل و دو برابر اندازه مجموعه داده اصلی را به دست می آوریم که از داده های اصلی (برچسب صفر) و نمونه های متخاصم مربوطه (برچسب یک) تشکیل شده است. از آن برای آموزش شبکه آشکارساز شکل۳-۶ استفاده میکنیم.



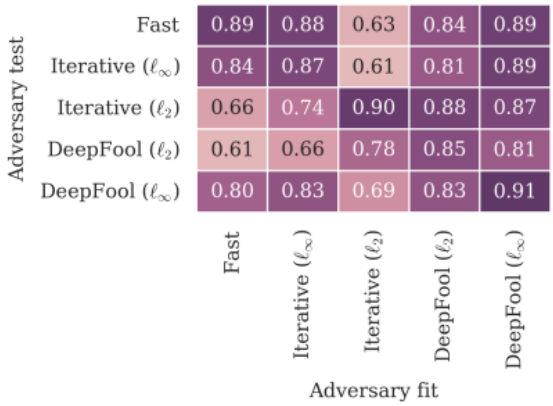
شکل (۳-۶) : detector structure [۱۰]

۳-۳-۵ نتایج

نتایج برای انجام آزمایش روی هر دو دیتاست برحسب دقت پاسخ های تولید شده (در تشخیص نمونه مخرب از سالم) با انجام آزمایش برای دادههای مخرب مختلف در شکل‌های۳-۷ و ۳-۸ آمده است.



شکل (۳-۷) : میزان دقت در حالت‌های مختلف تولید حمله و آشکارساز برای دیتاست CIFAR10 [۱۰]



شکل (۳-۸) : میزان دقت در حالت‌های مختلف تولید حمله و آشکارساز برای دیتاست ImageNet [۱۰]

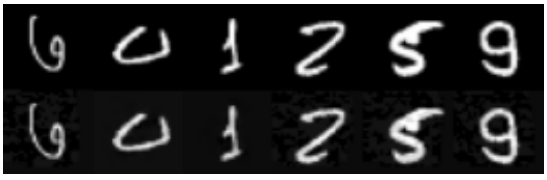
۳-۴ روش سوم: Robust Optimization

هدف این روش ارائه چارچوبی است که درک نظری کاملی از آموزش خصمانه و همچنین طرح‌های بهینه‌سازی جدید بر اساس بهینه‌سازی قوی ارائه دهد. اساساً، این الگوریتم پایداری شبکه‌های عصبی مصنوعی را با توجه به اغتشاشات در داده‌های ورودی، از طریق یک روش کمینه‌سازی-بیشینه‌سازی تکراری، که در آن پارامترهای شبکه با توجه به بدترین داده‌ها به‌جای داده‌های آموزشی اصلی به‌روزرسانی می‌شوند، افزایش می‌دهد [۱۲].

۳-۴-۱ معرفی دیتاست

دو دیتاست استفاده شده در این روش به شرح زیر است:

- دیتاست اعداد دست‌نویس یا همان MNIST است که در شکل۳-۹ نشان داده شده.



شکل (۳-۹) : MINST [۱۲]

- دیتاست CIFAR10 که در بخش قبل معرفی شد (شکل۳-۵).

۳-۴-۲ معرفی شبکه

برای تولید شبکه طبقه بندی دو شبکه مختلف هریک برای یکی از دیتاست‌ها استفاده میکینم.

- MNIST: برای این دیتاست از شبکه طبقه بندی کننده convent با دو لایه ۳۲ و ۶۴ تـایی کانولوشـنال و یـک مپ بعد از هر لایه استفاده می‌کنیم.
- CIRFAR10: برای این دیتاست از شبکه طبقه بندی کننده VGG net استفاده می‌کنیم.

۳-۴-۳ معرفی حمله

برای تولید نمونه‌های مخرب در این آزمایش از معادله۶ استفاده کرده و دیتاست جدید $A_{MINST}, A_{CRAFT10}$ را تولید میکنیم. مشابه حالات قبل و در حقیقت در فضای ورودی‌ها همسایه‌های نزدیک به هر ورودی را انتخاب میکنیم.

(۵-۳)

$$\hat{\Delta}_{x_i} \in \arg \max_{\Delta: x_i+\Delta \in \mathcal{U}_i} J_{\theta,y_i}(x_i) + \langle \nabla J_{\theta,y_i}(x), \Delta \rangle.$$

۳-۴-۴ معرفی مقابله

مقابله معرفی شده در این بخش به نوعی باز تولید شبکه اصلی است به طوری که به صورت پیوسـته‌تری عمـل کرده و به همسایگی‌های ورودی اصلی نیز لیبل مشابه ورودی معتبر را نسبت دهد. در این صورت ورودی مخرب بـه ایـن سیستم نفوذ منفی کمتری خواهد داشت.

برای این منظور به بازتولید تابع هزینه میپردازیم:

$$\min_{\theta} \tilde{J}(\theta, x, y) \quad = \min_{\theta} \sum_{i=1}^{m} \max_{\tilde{x}_i \in U_i} J(\theta, \tilde{x}_i, y_i) \qquad (۶-۳)$$

در این معادله U_i یک فضای عدم قطعیت است که همان همسایگی مذکور بوده و به دلخواه تعیین می‌شود.

۳-۴-۵ نتایج

نتایج آزمایش برای هر دو دیتاست بدست آمده (شکل ۳-۱۰و ۳-۱۱) در این جدول دقت پاسخ به دو نمونه سالم و سالم به همراه مخرب برای شبکه های بدون مقابله و با مقابله‌هایی با فضاهای عدم قطعیت متفاوت آورده شده است.

Net	MNIST test set	$\mathcal{A}_{\text{mnist}}$
Baseline	99.09%	0%
Robust ℓ_1	99.16%	33.83%
Robust ℓ_2	99.28%	76.55%
Robust ℓ_∞	99.33%	79.96%

شکل (۳-۱۰) : نتایج استفاده از مقابله برای همسایگی‌های مختلفMNIST [۱۲]

Net	CIFAR-10 test set	$\mathcal{A}_{\text{cifar10}}$
Baseline	90.79%	0%
Robust ℓ_1	91.11%	56.31%
Robust ℓ_2	91.04%	59.92%
Robust ℓ_∞	91.36%	65.01%

شکل (۳-۱۱) : نتایج استفاده از مقابله برای همسایگی‌های مختلف CRAFT10 [۱۲]

۳-۵ روش چهارم: Distillation

تقطیر: به ظور کلی نوعی روش برای آموزش شبکه است. هدف آن طراحی یک شبگه ثانویه با تعداد کمتری پارامتر و بار محاسباتی پایین تر با استفاده از اطلاعات سیستم اولیه است [۱۳].

۳-۵-۱ معرفی دیتاست

دو دیتاست استفاده شده در این روش به شرح زیر است

- دیتاست اعداد دست‌نویس یا همان MNIST است (شکل۳-۹).
- دیتاست CIFAR10 که در بخش قبل معرفی شد (شکل۳-۵).

۳-۵-۲ معرفی شبکه

برای هریک از دیتاست‌های معرفی شده یک دیتاست استفاده شده است که اطلاعات آنها در جدول‌های ۱و ۲ آمده است.

جدول (۳-۱) [۱۳]

Layer Type	MNIST Architecture	CIFAR10 Architecture
Relu Convolutional	32 filters (3x3)	64 filters (3x3)
Relu Convolutional	32 filters (3x3)	64 filters (3x3)
Max Pooling	2x2	2x2
Relu Convolutional	64 filters (3x3)	128 filters (3x3)
Relu Convolutional	64 filters (3x3)	128 filters (3x3)
Max Pooling	2x2	2x2
Relu Fully Connect.	200 units	256 units
Relu Fully Connect.	200 units	256 units
Softmax	10 units	10 units

جدول (۳-۲) [۱۳]

Parameter	MNIST Architecture	CIFAR10 Architecture
Learning Rate	0.1	0.01 (decay 0.5)
Momentum	0.5	0.9 (decay 0.5)
Decay Delay	-	10 epochs
Dropout Rate (Fully Connected Layers)	0.5	0.5
Batch Size	128	128
Epochs	50	50

۳-۵-۳ معرفی حمله

برای تولید حمله در این بخش مانند روش اول از روش FGSM استفاده میشود و داده‌های مخرب ساخته میشوند.

۳-۵-۴ معرفی مقابله

شهود کلی در پشت این تکنیک استخراج بردارهای احتمال کلاس تولید شده توسط شبکه اولیه یادگیری عمیق برای آموزش شبکه عمیق ثانویه با ابعاد کاهش یافته بدون از دست دادن دقت است. مزیت استفاده از احتمالات کلاس به جای برچسب های سخت مشهود است زیرا احتمال ها علاوه بر ارائه کلاس صحیح نمونه، اطلاعات اضافی را در مورد هر کلاس رمزگذاری می کنند.

پس از آخرین لایه پنهان شبکه اولیه خروجی‌های آن را به لایه SOFTMAX با تابع مشخص شده در معادله۸ داده و در نهایت یک بردار احتمالی خواهیم داشت که آن را به شبکه ثانویه داده و آن را آموزش میدهیم:

$$F(X) = \left[\frac{e^{z_i(x)/T}}{\sum_{j=0}^{N-1} e^{z_j(x)/T}} \right] \qquad (۷-۳)$$

۳-۵-۵ نتایج

نتایج حاصل از آموزش شبکه ثانویه برای هر دو دیتاست با ارائه دقت حاصله در جدول۳-۳ آمده است. (در این جدول چندین شبکه ثانویه با T های مختلف تابع SOFTMAX آورده شده است)

جدول (۳-۳) نتایج تغیرات دقت با استفاده از روش مقابله مذکور [۱۳]

Distillation Temperature	MNIST Adversarial Samples Success Rate (%)	CIFAR10 Adversarial Samples Success Rate (%)
1	91	92.78
2	82.23	87.67
5	24.67	67
10	6.78	47.56
20	1.34	18.23
30	1.44	13.23
40	0.45	9.34
50	1.45	6.23
100	0.45	5.11
No distillation	95.89	87.89

۳-۶ روش پنجم: Feature Squeezing

رویکرد کلی این روش از توجه به این حقیقت منجر می‌شود که فضای ویژگی‌های ورودی یک شبکه یادگیری عموما بزرگتر از نیاز است و این تعدد ویژگی‌ها باعث سهولت تولید حمله و فریب خوردن سیستم می‌شود [۱۴].

۳-۶-۱ معرفی دیتاست

در این روش از هر سه دیتاست معرفی شده در این پژوهش استفاده می‌کنیم:

- دیتاست اعداد دست‌نویس یا همان MNIST است (شکل۳-۹).

- دیتاست CIFAR10 (شکل۳-۵).
- دیتاست ImageNet (شکل۳-۱).

۳-۶-۲ معرفی شبکه

در این بخش در جدول۳-۴ به معرفی مدل‌های استفاده شده برای طبقه‌بندی هر دیتاست میپردازیم:

جدول (۳-٤) شبکه‌های استفاده شده برای طبقه‌بندی [١٤]

Dataset	Model
MNIST	7-Layer CNN [4]
CIFAR-10	DenseNet [17, 22]
ImageNet	MobileNet [16, 23]

۳-۶-۳ معرفی حمله

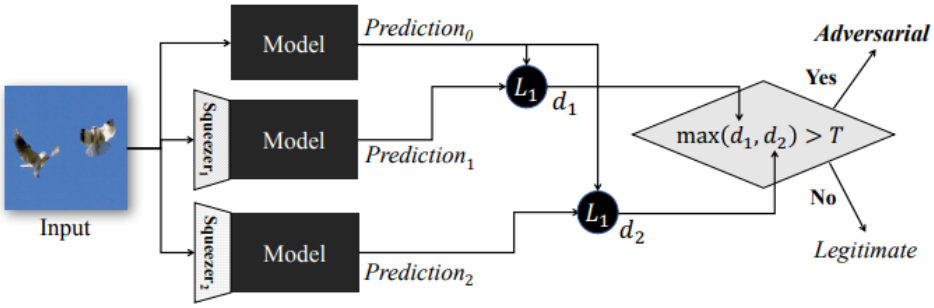
برای تولید حمله در این روش از پنج روش مختلف استفاده میکنیم این پنج روش به اختصار در زیر توضیح داده می‌شوند:

- Fast Gradient Sign Method (FGSM): این روش پیشتر در بخش۳-۲-۳ توضیح داده شده.
- Basic Interval Method (BIM): این روش از گسترش روش FGSM بدست می‌آید به طوری که در این حالت با انجام گام‌های کوچک متعدد FGSM حمله را تولید می‌کند.
- Deep Fool: این روش به دنبال اختلالی با کمترین اندازه می‌گردد که بتواند سیستم را فریب دهد. بسته به این که شبکه خطی یا غیرخطی است از روش‌های مختلف استفاده می‌کند. (اگر خطی باشد در یک همسایگی چندضلعی از داده صحیح به دنبال حمله بهینه میگردد و اگر غیر خطی باشد سعی بر یافتن پاسخ بهینه با استفاده از تکرار و کیمنه کردن در هر گام خواهد داشت)
- Jacobian Saliency Map Approach (JSMA): به طور مکرر پیکسل‌هایی از تصویر را که دارای ارزش خصمانه بالایی دارند یافته و به آن خطا را وارد می‌کنند. ارزش خصمانه هر پیکسل از روی ماتریس ژاکوبین تعیین میشود.
- Carlini/Wagner: این روش هم از منطقی مانند Deep Fool با درجه آزادی‌های متفاوت استفاده می‌کند.

۳-۶-۴ معرفی مقابله

رویکرد کلی این روش کاهش اطلاعات ورودی به منظور کاهش امکان برای حمله‌گر بوده به طوری که قدرت عمل و آزادی برای نفوذ در شبکه کاهش یابد.

در این روش به تولید یک آشکارساز می‌پردازیم این آشکارسازها متناسب با ساختار نشان داده شده در شکل۳-۱۲ تولید می‌شوند که پیش‌بینی مناسبی درباره خصمانه بودن یا نبودن ورودی ارائه می‌دهد.



شکل (۳-۱۲) : ساختار کلی آشکارساز استفاده شده در این روش [۱۴]

در این مدل اگر فاصله پیش‌بینی مدل اصلی و هر یک از مدل‌های مختصر شده بیشتر از یک مقدار آستانه باشد آنگاه مدل جامع ورودی را خصمانه گزارش خواهد داد.

روش‌های تولید مدل‌های مختصر شده دو رویکرد کلی هستند:

- کاهش اندازه اطلاعات رنگی: در این روش اطلاعات را از هشت بیت به تعداد بیت‌های دلخواه کمتر کاهش میدهیم.
- هموارسازی فضایی: در این روش یک پنجره با اندازه دلخواه انتخاب کرده و سعی بر جایگزین کردن عدد پیکسل میانه با میانگین اعداد پنجره داریم. این روش داده های دور از هم را از بین می‌برد.

۳-۶-۵ نتایج

نتایج بدست آمده برای این روش در جدول۳-۵ آمده است:

جدول (۳-۵) : اطلاعات نتایج استفاده از آشکارسازها [۱٤]

Dataset	Squeezer		L_∞ Attacks				L_2 Attacks			L_0 Attacks				All Attacks	Legitimate
	Name	Parameters	FGSM	BIM	CW_∞		Deep-Fool	CW_2		CW_0		JSMA			
					Next	LL		Next	LL	Next	LL	Next	LL		
MNIST	None		54%	9%	0%	0%	-	0%	0%	0%	0%	27%	40%	13.00%	99.43%
	Bit Depth	1-bit	**92%**	**87%**	**100%**	**100%**	-	**83%**	**66%**	0%	0%	50%	49%	**62.70%**	99.33%
	Median Smoothing	2x2	61%	16%	70%	55%	-	51%	35%	39%	36%	62%	56%	48.10%	99.28%
		3x3	59%	14%	43%	46%	-	51%	53%	**67%**	**59%**	**82%**	**79%**	55.30%	98.95%
CIFAR-10	None		15%	8%	0%	0%	2%	0%	0%	0%	0%	0%	0%	2.27%	94.84%
	Bit Depth	5-bit	17%	13%	12%	19%	40%	40%	47%	0%	0%	21%	17%	20.55%	94.55%
		4-bit	21%	29%	69%	74%	72%	84%	84%	7%	10%	23%	20%	44.82%	93.11%
	Median Smoothing	2x2	**38%**	**56%**	**84%**	**86%**	**83%**	**87%**	83%	**88%**	**85%**	**84%**	**76%**	**77.27%**	89.29%
	Non-local Means	11-3-4	27%	46%	80%	84%	76%	84%	**88%**	11%	11%	44%	32%	53.00%	91.18%
ImageNet	None		1%	0%	0%	0%	11%	10%	3%	0%	0%	-	-	2.78%	69.70%
	Bit Depth	4-bit	5%	4%	66%	79%	44%	84%	82%	38%	67%	-	-	52.11%	68.00%
		5-bit	2%	0%	33%	60%	21%	68%	66%	7%	18%	-	-	30.56%	69.40%
	Median Smoothing	2x2	22%	28%	75%	81%	**72%**	81%	84%	**85%**	**85%**	-	-	**68.11%**	65.40%
		3x3	**33%**	**41%**	73%	76%	66%	77%	79%	81%	79%	-	-	67.22%	62.10%
	Non-local Means	11-3-4	10%	25%	**77%**	**82%**	57%	**87%**	**86%**	43%	47%	-	-	57.11%	65.40%

۳-۷ روش ششم: آشکارسازی حمله در داده‌های سری زمانی

در این روش به معرفی ساختاری برای آشکارسازی حملات ساخته شده بر روی داده‌های سری زمانی می‌پردازیم [۱۵].

۳-۷-۱ معرفی دیتاست

در این روش از آنجا که داده‌های موزد بررسی داده‌های سری زمانی هستند در نتیجه از دیتاست‌های قبلی استفاده نمی‌شود بلکه از دیتاست‌های دیگر که در جدول۳-۶ آورده شده در نتایج به تفصیل آمده است استفاده می‌کنیم (در مجموع ۷۲ دسته داده).

۳-۷-۲ معرفی شبکه

برای طبقه‌بندی داده‌ها در این روش از UCR Time Series Classification (TSC) استفاده شده است که دارای ۸۵ دیتاست مختلف است.

۳-۷-۳ معرفی حمله

برای تولید حملات در این روش از دو رویکرد مختلف استفاده کرده و سه دسته مختلف ورودی برای آزمایش آشکارساز طراحی شده تولید میکنیم. روش‌های تولید حمله به صورت زیر اند:

- Fast Gradient Sign Method (FGSM): این روش پیشتر در بخش۳-۲-۳ توضیح داده شده.
- Basic Interval Method (BIM): این روش پیشتر در بخش۳-۶-۳ توضیح داده شده.

سه دسته داده ورودی تولید شده برای تست کردن مدل پیش‌نهادی به شرح زیر اند:

- دسته اول: ترکیبی از داده‌های اصلی و حملات تولید شده با استفاده از FGSM.
- دسته دوم: ترکیبی از داده‌های اصلی و حملات تولید شده با استفاده از BIM.
- دسته سوم: ترکیبی از داده‌های اصلی و حملات تولید شده با استفاده از FGSM و همچنین BIM.

۳-۷-۴ معرفی مقابله

سه پیش‌فرض درباره حملات متخاصم وجود دارد که به شرح زیر اند:

- از آنجا که اغتشاش‌ها به اندازه کافی کوچک هستند، تاثیر آن‌ها در سیگنال‌های تفاضلی فاحش‌تر از سیگنال‌های اصلی بوده است.
- با این‌که اغتشاش‌ها عمدتا کوچک هستند اما تاثیر زیادی روی آنتروپی و پیچیدگی سیگنال‌ها می‌گذارند.
- این اغتشاش‌ها ظاهرا آشفته هستند و در نتیجه گرچه قابل مشاهده نبستند اما می‌توان آن‌ها را با اندازه‌گیری میزان آشوب شناسایی کرد.

بنابراین، ما از یک توصیفگر سه بعدی ساده (شامل آنتروپی نمونه، DFA و نسبت آنتروپی نمونه به DFA) برای توصیف نمونه‌های متعلق به برخی از مجموعه داده‌ها استفاده می‌کنیم.

۳-۷-۵ نتایج

در این بخش نتایج صحت تشخیص نمونه مخرب را برای دیتاست‌های مختلف با استفاده از آشکارساز گفته شده بیان میکنیم (در جدول۳-۶).

جدول (۳-۶) : نتایج [۱۵]

Dataset	FGSM	BIM	FGSM+BIM
50words	95.16	95.16	96.78
Adiac	91.69	92.84	93.95
ArrowHead	85.71	86.29	88.95
Beef	90.00	90.00	93.33
BeetleFly	95.00	92.50	95.00
BirdChicken	75.00	80.00	80.00
CBF	50.00	50.00	39.15
Car	96.67	96.67	97.78
ChlorineConcentration	61.47	58.35	50.39
CinC_ECG_torso	84.09	82.79	88.53
Coffee	91.07	91.07	94.05
Computers	44.20	45.20	31.87
Cricket_X	67.05	64.10	58.63
Cricket_Y	69.10	66.41	59.15
Cricket_Z	65.51	65.26	57.52
DiatomSizeReduction	90.52	90.36	93.25
DistalPhalanxOutlineAgeGroup	93.50	93.00	95.33
DistalPhalanxTW	90.88	87.50	88.42
ECG200	51.00	49.50	38.00
ECG5000	54.70	53.23	42.56
ECGFiveDays	92.86	85.25	88.23
Earthquakes	52.33	53.57	39.65
ElectricDevices	48.74	48.69	36.96
FISH	93.14	93.14	95.43
FaceAll	49.26	48.99	39.88
FaceFour	50.57	50.00	42.05
FacesUCR	50.37	49.90	37.17
FordA	95.47	95.49	96.98
FordB	91.41	91.41	94.27
Gun_Point	93.67	93.67	95.78
Ham	86.67	90.48	89.52
HandOutlines	91.65	91.65	94.43
Haptics	94.97	94.97	96.65
Herring	95.31	95.31	96.88
InlineSkate	87.82	73.64	80.42
InsectWingbeatSound	88.66	95.00	92.44
LargeKitchenAppliances	50.27	53.20	40.00
Lighting2	47.54	49.18	35.52
Lighting7	52.74	54.79	43.84
MALLAT	95.16	95.16	96.77
Meat	96.67	96.67	97.78
MedicalImages	62.04	71.97	59.08
MoteStrain	53.23	54.55	46.57
NonInvasiveFatalECG_Thorax1	94.27	94.27	96.18
NonInvasiveFatalECG_Thorax2	94.71	94.71	96.47
OSULeaf	94.21	94.01	96.01
OliveOil	53.33	53.33	68.89
Phoneme	50.92	51.27	37.62
Plane	89.52	90.00	91.43
RefrigerationDevices	48.67	48.27	37.24
ScreenType	46.80	46.53	31.38
ShapeletSim	44.44	45.28	36.85
ShapesAll	76.42	90.83	81.11
SmallKitchenAppliances	62.93	87.47	69.16
StarLightCurves	94.01	94.01	96.01
Strawberry	94.29	94.29	96.19
SwedishLeaf	65.52	64.08	58.08
Symbols	63.72	82.31	66.16
ToeSegmentation1	57.68	54.17	47.22
ToeSegmentation2	49.23	47.31	36.67
Trace	44.50	45.00	30.67
TwoLeadECG	56.63	58.38	48.52
Two_Patterns	76.88	71.79	69.43
UWaveGestureLibraryAll	95.27	95.28	96.85
Wine	87.04	87.04	91.36
WordsSynonyms	97.02	97.02	98.01
Worms	61.88	56.08	50.64
WormsTwoClass	62.71	59.94	53.96
uWaveGestureLibrary_X	95.53	95.53	97.02
uWaveGestureLibrary_Y	94.85	94.85	96.57
wafer	95.19	93.64	95.71
yoga	94.98	94.98	96.66

نتایج نشان می‌دهد که روش فوق به صورت میانگین توانایی شناسایی ۹۰٪ حملات را داراست که مقدار قابل قبولی است.

۳-۸ روش هفتم: feature scattering-based adversarial training

در این روش به معرفی یک رویکرد آموزشی خصمانه مبتنی بر پراکندگی ویژگی را برای بهبود استحکام مدل در برابر حملات خصمانه می‌پردازیم [۱۶].

۳-۸-۱ معرفی دیتاست

در این روش از سه دیتاست مختلف استفاده می‌شود:

- دیتاست CIFAR10 (شکل۳-۵).
- دیتاست SVHN: دیتاست مربوط به طبقه‌بندی اعداد پلاک خانه (شکل۳-۱۳).
- دیتاست CIFAR100: همان قبلی فقط با ۱۰۰ دسته مختلف.

شکل (۳-۱۳) : دیتاست SVHN [۱۷]

۳-۸-۲ معرفی شبکه

برای طبقه‌بندی در این بخش از شبکه‌هایی استفاده میشود که در پبوست مقالـه [۱۶] آورده شـده اسـت اسـتفاده می‌شود.

۳-۸-۳ معرفی حمله

حملات به خالات مختلفی در این روش تولید می‌شوند:

- Fast Gradient Sign Method (FGSM): این روش پیشتر در بخش۳-۲-۳ توضیح داده شده.
- Projected Gradient Descent (PGD): در این روش ابتدا نویز تصادفی‌ای را روی یک تصویر اصلی سوار می‌کنیم سپس در گام‌های متوالی سعی بر بهینه کردن حمله می‌کنیم.

۳-۸-۴ معرفی مقابله

روش مقابله در این بخش به صورت آموزش سیستم با استفاده از نمونه‌های مخرب تولید شـده بـا الگـوریتم خـاص توضیح داده شده در مقاله [۱۶] به منظور افزایش مقاومت سیستم در برابر سایر حملات است.

ویژگی متمایز این روش تولید حمله تفاوت جزئی تابع هزینـه بـا روش‌هـای مرسـوم اسـت. ایـن روش بـه سـاختار منظم‌تری در آموزش شبکه منجر شده و از تفاضل بردار ویژگی‌ها با نمونه اصلی استفاده می‌کند.

۳-۸-۵ نتایج

نتایج مربوط به انجام آزمایش روی هر سه دیتاست معرفی شده در جدول‌های ۳-۷ و ۳-۸ و ۳-۹ آورده شده است.

جدول (۳-۷) : نتایج مربوط به دیناست CIFAR10 [۱۶]

Models	Clean	Accuracy under White-box Attack ($\epsilon = 8$)								
		FGSM	PGD10	PGD20	PGD40	PGD100	CW10	CW20	CW40	CW100
Standard	**95.6**	36.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Madry	85.7	54.9	45.1	44.9	44.8	44.8	45.9	45.7	45.6	45.4
Bilateral	91.2	70.7	–	57.5	–	55.2	–	56.2	–	53.8
Proposed	90.0	**78.4**	**70.9**	**70.5**	**70.3**	**68.6**	**62.6**	**62.4**	**62.1**	**60.6**

جدول (۳-۸) : نتایج مربوط به دیناست SVHN [۱۶]

Models	Clean	White-box Attack ($\epsilon = 8$)				
		FGSM	PGD20	PGD100	CW20	CW100
Standard	**97.2**	53.0	0.3	0.1	0.3	0.1
Madry	93.9	68.4	47.9	46.0	48.7	47.3
Bilateral	94.1	69.8	53.9	50.3	–	48.9
Proposed	96.2	**83.5**	**62.9**	**52.0**	**61.3**	**50.8**

جدول (۳-۹) : نتایج مربوط به دیناست CIFAR100 [۱۶]

Models	Clean	White-box Attack ($\epsilon = 8$)				
		FGSM	PGD20	PGD100	CW20	CW100
Standard	**79.0**	10.0	0.0	0.0	0.0	0.0
Madry	59.9	28.5	22.6	22.3	23.2	23.0
Bilateral	68.2	60.8	26.7	25.3	–	22.1
Proposed	73.9	**61.0**	**47.2**	**46.2**	**34.6**	**30.6**

۳-۹ جمع‌بندی

در این فصل به بررسی هفت نمونه از مقابله با حملات پرداختیم که به تفصیل توضیح داده شد.

- استفاده از Denoiser
- تعریف آشکارساز
- بهینه سازی شبکه اصلی
- تولید شبکه کمکی با استفاده از ایده تقطیر
- استفاده از مختصر کردن ویژگی‌ها
- آشکارسازی حمله در داده‌های سری زمانی
- آموزش سیستم مبتنی بر پراکندگی ویژگی

فصل چهارم: جمع‌بندی، نتیجه‌گیری، پیشنهادات

۴-۱ جمع‌بندی

علیرغم دقت و عملکرد بالا، الگوریتم‌های یادگیری ماشینی در برابر آشفتگی‌های ظریف آسیب‌پذیر هستند که می‌توانند پیامدهای فاجعه‌باری در محیط‌های مرتبط با امنیت داشته باشند. تهدید زمانی که برنامه ها در محیط متخاصم کار می کنند جدی تر می شود. بنابراین، ابداع تکنیک‌های یادگیری قوی که در برابر حملات متخاصم انعطاف‌پذیر باشند، به یک ضرورت فوری تبدیل شده است. از زمانی که Szegedy [۱۸] آسیب پذیری الگوریتم های یادگیری ماشین را نشان داد، تعدادی از مقالات تحقیقاتی در مورد حملات خصمانه و همچنین اقدامات متقابل آنها منتشر شد. در این پژوهش سعی شده است برخی از حملات شناخته شده و راهبردهای دفاعی پیشنهادی بررسی شود [۲].

۴-۲ نتیجه‌گیری

یادگیری خصمانه یک تهدید واقعی برای کاربرد یادگیری ماشین در دنیای فیزیکی است. اگرچه اقدامات متقابل خاصی وجود دارد، اما هیچ یک از آنها نمی تواند به عنوان نوشدارویی برای همه چالش ها عمل کند. این به عنوان یک مشکل باز برای جامعه یادگیری ماشینی باقی می ماند که یک طراحی قوی در برابر این حملات خصمانه ارائه دهد [۲].

۴-۳ پیشنهادات

در مطالعات آینده اگر بتوان روش‌های مقابله ذکر شده در این پژوهش را به روش مناسبی یکسان سازی کرد به طوریکه برای آزمودن تمام روشها بتوان از دیتاستی واحد استفاده کرد میتوان به مقایسه معتبرتری دست یافت.

مراجع

[1] T. Z. Z. Q. X. L. Kui Ren, "Adversarial Attacks and Defenses in Deep Learning," *elsevier,* p. 15, 2020.

[2] M. A. V. D. A. C. D. M. ANIRBAN CHAKRABORTY, "Adversarial Attacks and Defences: A Survey," p. 31, 2018.

[3] M. Ozdag, "Adversarial Attacks and Defenses Against Deep Neural Networks: aSurvey," *ELsevier,* p. 10, 2018.

[4] S. J. P. P.SIBI, "ANALYSIS OF DIFFERENT ACTIVATION FUNCTIONS," *Journal of Theoretical and Applied Information Technology,* p. 5, 2013.

[5] H. C. Dongyu Meng, "MagNet: a Two-Pronged Defense against Adversarial Examples," p. 13, 2017.

[6] M. K. B. X. Yan Zhou, "A survey of game theoretic approach for adversarial machine learning," *wiley,* p. 9, 2018.

[7] M. L. Y. D. T. P. X. H. Fangzhou Liao, "Defense against Adversarial Attacks Using High-Level Representation Guided Denoiser," p. 10, 2018.

[8] "https://devopedia.org/imagenet," [Online].

[9] J. S. a. C. S. Ian J. Goodfellow, "Explaining and harnessing adversarial examples," 2014.

[10] T. G. V. F. B. B. Jan Hendrik Metzen, "ON DETECTING ADVERSARIAL PERTURBATIONS," in *ICLR*, 2017.

[11] "https://www.cs.toronto.edu/~kriz/cifar.html," [Online].

[12] Y. Y. S. N. Uri Shaham, "Understanding Adversarial Training: Increasing Local Stability of Neural Nets through Robust Optimization," p. 12, 2016.

[13] P. M. X. W. S. J. a. A. S. Nicolas Papernot, "Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks," *IEEE,* p. 16, 2016.

[14] D. E. Y. Q. Weilin Xu, "Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks," p. 15, 2018.

[15] W. G. Y. M. Y. Y. Mubarak G. Abdu-Aguye, "DETECTING ADVERSARIAL ATTACKS IN TIME-SERIES DATA," *IEEE,* p. 5, 2020.

[16] J. W. Haichao Zhang, "Defense Against Adversarial Attacks Using Feature Scattering-based

Adversarial Training," in *NeurIPS*, Vancouver, Canada, 2019.

[17] "https://www.researchgate.net," [Online].

[18] W. Z. I. S. J. B. D. E. I. J. G. a. R. F. Christian Szegedy, "Intriguing properties of neural networks.,"2013.