دانشکده مهندسی کامپیوتر

مقاوم کردن الگوریتم شبکه عصبی پیچشی در برابر فریب خوردن

گزارش پروژه پایانی کارشناسی

در رشته مهندسی کامپیوتر

نام دانشجو:

امیرحسین امینی مهر

استاد راهنما:

دکتر ناصر مزینی

خرداد ماه ۱۳۹۹


بسم الله الرحمن الرحيم

تأییدیه‌ی هیأت داوران جلسه‌ی دفاع از پایان‌نامه/رساله

نام دانشکده:

نام دانشجو:

عنوان پایان‌نامه یا رساله:

تاریخ دفاع:

رشته:

گرایش:

ردیف	سمت	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضا
۱	استاد راهنما				
۲	استاد راهنما				
۳	استاد مشاور				
۴	استاد مشاور				
۵	استاد مدعو خارجی				
۶	استاد مدعو خارجی				
۷	استاد مدعو داخلی				
۸	استاد مدعو داخلی				

تأییدیه‌ی صحت و اصالت نتایج

باسمه تعالی

اینجانب امیرحسین امینی مهر به شماره دانشجویی ۹۵۵۲۲۰۶۷ دانشجوی رشته مهندسی کامپیوتر مقطع تحصیلی کارشناسی تأیید می‌نمایم که کلیه‌ی نتایج این پایان‌نامه/رساله حاصل کار اینجانب و بدون هرگونه دخل و تصرف است و موارد نسخه‌برداری‌شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. درصورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مؤلفان و مصنفان و قانون ترجمه و تکثیر کتب و نشریات و آثار صوتی، ضوابط و مقررات آموزشی، پژوهشی و انضباطی ...) با اینجانب رفتار خواهد شد و حق هرگونه اعتراض درخصوص احقاق حقوق مکتسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم. در ضمن، مسؤولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی‌صلاح (اعم از اداری و قضایی) به عهده‌ی اینجانب خواهد بود و دانشگاه هیچ‌گونه مسؤولیتی در این خصوص نخواهد داشت.

نام و نام خانوادگی: امیرحسین امینی مهر

امضا و تاریخ:

مجوز بهره‌برداری از پایان‌نامه

بهره‌برداری از این پایان‌نامه در چهارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما به شرح زیر تعیین می‌شود، بلامانع است:

☐ بهره‌برداری از این پایان‌نامه/ رساله برای همگان بلامانع است.

☐ بهره‌برداری از این پایان‌نامه/ رساله با اخذ مجوز از استاد راهنما، بلامانع است.

☐ بهره‌برداری از این پایان‌نامه/ رساله تا تاریخ ..... ممنوع است.

نام استاد یا اساتید راهنما:

تاریخ:

امضا:

تشکر و قدردانی:

با تشکر و سپاس فراوان از دکتر ناصر مزینی برای رهنمود ها و کمک های مفیدشان.

چکیده

در این گزارش در ابتدا سعی شده است که به الگوریتم شبکه عصبی پیچشی، کاربرد های آن در زندگی روزمره و اهمیتی که برای انسان ها دارد اشاره‌ی کوتاهی شود و در ادامه در مورد داده های Mnist توضیحاتی داده شده است.

در ادامه در این گزارش به طور کلی به حمله های شبکه‌ی تقابلی برای فریب دادن الگوریتم شبکه عصبی پیچشی و به صورت جزئی به فریب دادن الگوریتم شبکه عصبی پیچشی و به روش نشانه‌ی گرادیان سریع پرداخته شده است. در ادامه راه های جلوگیری از فریب خوردن الگوریتم شبکه عصبی پیچشی و دفاع در برابر آن ها بررسی شده است و پیاده سازی هایی که با زبان برنامه نویسی پایتون برای فریب دادن این الگوریتم و جلوگیری از این حملات انجام شده است توضیح داده شده است. نتیجه های به دست آمده نشان می دهد که به میزان خوبی جلوی این حملات گرفته شده است. نتایج به دست آمده در این گزارش از اعمال الگوریتم های حمله‌ای و دفاعی بر روی داده های Mnist به وجود آمده است که این روش ها شامل آموزش شبکه‌ی تقابلی و روش دفاع تقطیری می باشند. در ادامه نتیجه‌ای که از این گزارش می توان گرفت این است که به میزان خوبی می توان جلوی حمله از روش نشانه‌ی گرادیان سریع را گرفت. در واقع نتیجه‌ی این گزارش شامل نتیجه‌ی اعمال دو دفاع آموزش شبکه‌ی تقابلی و روش دفاع تقطیری برای جلوگیری از فریب خورد الگوریتم شبکه عصبی پیچشی می باشد ولی روش های جدیدی نیز برای حمله به الگوریتم های یادگیری عمیق وجود دارند که راه های دیگری باید برای مقابله با اینگونه حملات در نظر گرفت.

واژه‌های کلیدی: فریب دادن الگوریتم شبکه عصبی پیچشی - حمله‌ی شبکه‌ی تقابلی - روش نشانه‌ی گرادیان سریع - آموزش شبکه‌ی تقابلی - دفاع تقطیری

فهرست مطالب

فهرست اشکال

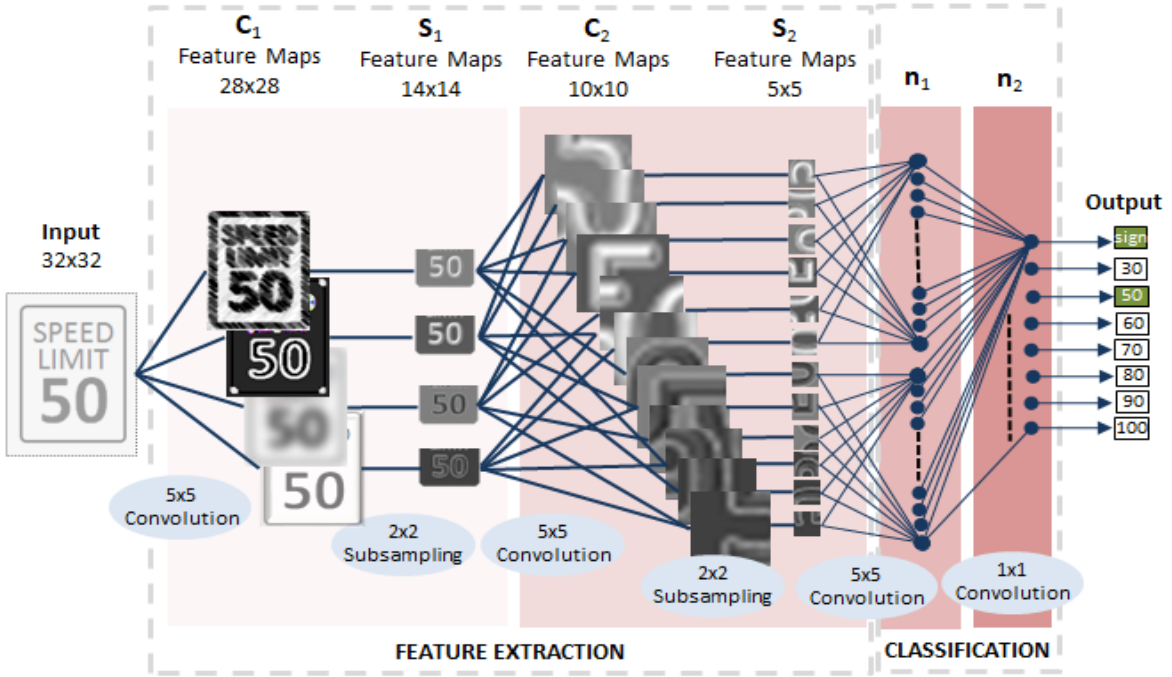
فهرست جداول

فصل ۱:

مقدمه

۱-۱- مقدمه

هیچ شکی وجود ندارد که در حال حاضر وابستگی ما به الگوریتم های یادگیری عمیق روز به روز در حال زیاد تر شدن هستند. که یکی از مهم ترین الگوریتم های یادگیری عمیق، الگوریتم شبکه عصبی پیچشی می باشد. در دوره‌ای که ما از برنامه های کاربردی[۱] مبتنی بر الگوریتم شبکه عصبی پیچشی برای کاربرد های مختلفی مانند شناسایی محتوای نامناسب در رسانه های اجتماعی تا تفسیر محیط اطراف برای جهت یابی در وسایل نقلیه خود مختار[۲] استفاده می کنیم، قطعا عملکرد درست این الگوریتم برای ما خیلی مهم خواهد بود. به عنوان مثال فرض کنید که یک خودرو هوشمند یک تابلو "حداکثر سرعت مجاز ۳۰ کیلومتر بر ساعت" را به صورت "حداکثر سرعت مجاز ۱۲۰ کیلومتر بر ساعت" دریافت کند، آنگاه قطعا فجایع زیادی به بار خواهد آمد. یا به عنوان مثال یکی دیگر از کاربرد های دیگر الگوریتم شبکه عصبی پیچشی در تحلیل تصاویر پزشکی می باشد، که اگر بر اثر حمله هایی باعث بشود که الگوریتم شبکه عصبی پیچشی به اشتباه بیافتد باز هم فجایع زیادی به بار خواهد آمد. این دو مثال، مثال هایی بودند که به طور مستقیم با جان انسان ها در ارتباط بودند. کاربرد های دیگری نیز از این الگوریتم وجود دارد که با حمله کردن به این الگوریتم برای فریب دادن آن ممکن است مشکلات دیگری به وجود بیاورد و در بسیاری از کار های روزمره-ی انسان ها اختلال به وجود بیاورد.



شکل ۱: چگونگی عملکرد الگوریتم شبکه عصبی پیچشی برای طبقه بندی علائم راهنمایی و رانندگی

[۱] Applications

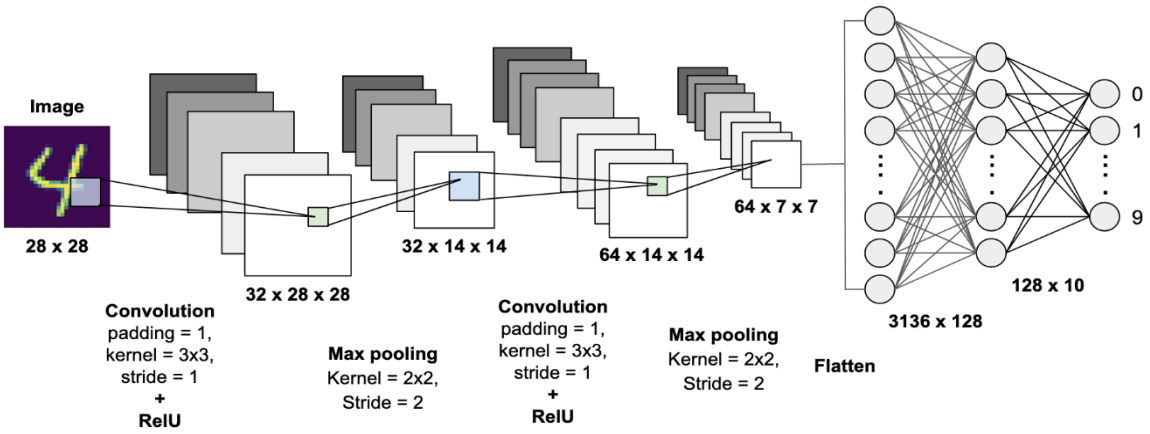
[۲] Autonomous vehicles

فصل ۲:

شرح مسئله

۱-۲- الگوریتم شبکه عصبی پیچشی(کانولوشن)

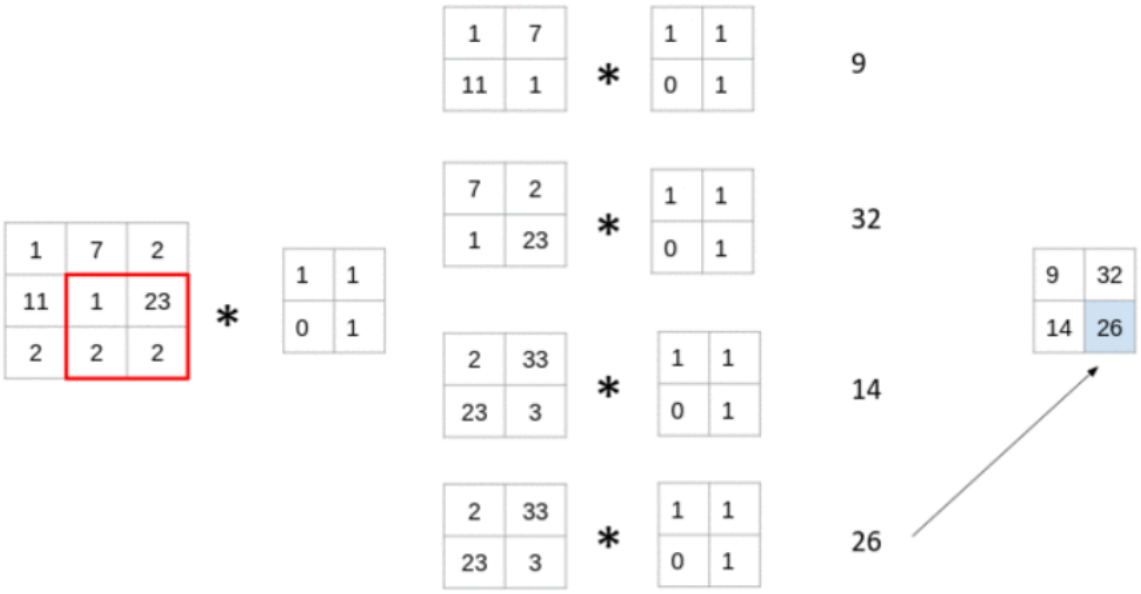
الگوریتم شبکه عصبی پیچشی[۱] یکی از انواع الگوریتم های یادگیری عمیق[۲] می باشد.
از کاربرد های مختلف این الگوریتم می توان به طبقه بندی و شناسایی کردن اشاره کرد که در این پروژه، هدف بررسی کاربرد طبقه بندی از الگوریتم شبکه عصبی پیچشی و فریب آن و راه های دفاع در مقابل آن می باشد. یکی از معروف ترین کاربرد های شبکه عصبی پیچشی در طبقه بندی داده ها، طبقه بندی کردن اعداد صفر تا نه می باشد که در این طبقه بندی تصاویر اعداد صفر تا نه که با دست خط انسان نوشته شده است به مدل داده می شود و مدل باید تشخیص دهد که تصویر کدام عدد به مدل داده شده است. به این طبقه بندی طبقه بندی کردن Mnist گفته می شود.



شکل ۲: چگونگی عملکرد الگوریتم شبکه عصبی پیچشی برای طبقه بندی داده های Mnist

الگوریتم شبکه عصبی پیچشی از یک ورودی و خروجی و از چندین لایه تشکیل شده است که به لایه های میانی لایه های پنهان[۳] نیز می گویند. هر لایه در این الگوریتم از تعدادی نورون تشکیل شده است.
در الگوریتم شبکه عصبی پیچشی اول از همه یک ماتریس n*n داریم که نام آن نگاشت ویژگی می باشد که باید ماتریس تصویر ورودی را که m*m می باشد و m از n بزرگتر می باشد از اول شروع کرده (از بالا سمت چپ ماتریس تصویر) و به صورت n*n جدا کرده و در ماتریس نگاشت ویژگی ضرب کرده و همین روال را ادامه داده تا مطابق شکل زیر یک ماتریس n*n جدید ساخته شود. این کاری است که در لایه‌ی اول الگوریتم صورت می گیرد و در بقیه لایه ها نیز ادامه پیدا می کند.

[۱] convolutional neural network
[۲] Deep Learning
[۳] hidden layers



شکل ۳: چگونگی ضرب ماتریس نگاشت ویژگی در ماتریس های جدا شده از تصویر

خروجی این الگوریتم نیز تعدادی احتمال می باشد که تعداد آن ها برابر با تعداد دسته تصاویری که الگوریتم قرار است داده های آنها را تمرین کند می باشد که به آنها برچسب[۱] می گویند. (مثلا در داده های Mnist تعداد برچسب ها برابر با ده عدد می باشد که این برچسب ها اعداد صفر تا نه می باشند) در واقع خروجی نشان دهنده احتمال هایی است که مدل از یک تصویر ورودی تشخیص داده است و تشخیص داده که به چه احتمالی تصویر ورودی کدام یک از برچسب ها می باشد.[۱]

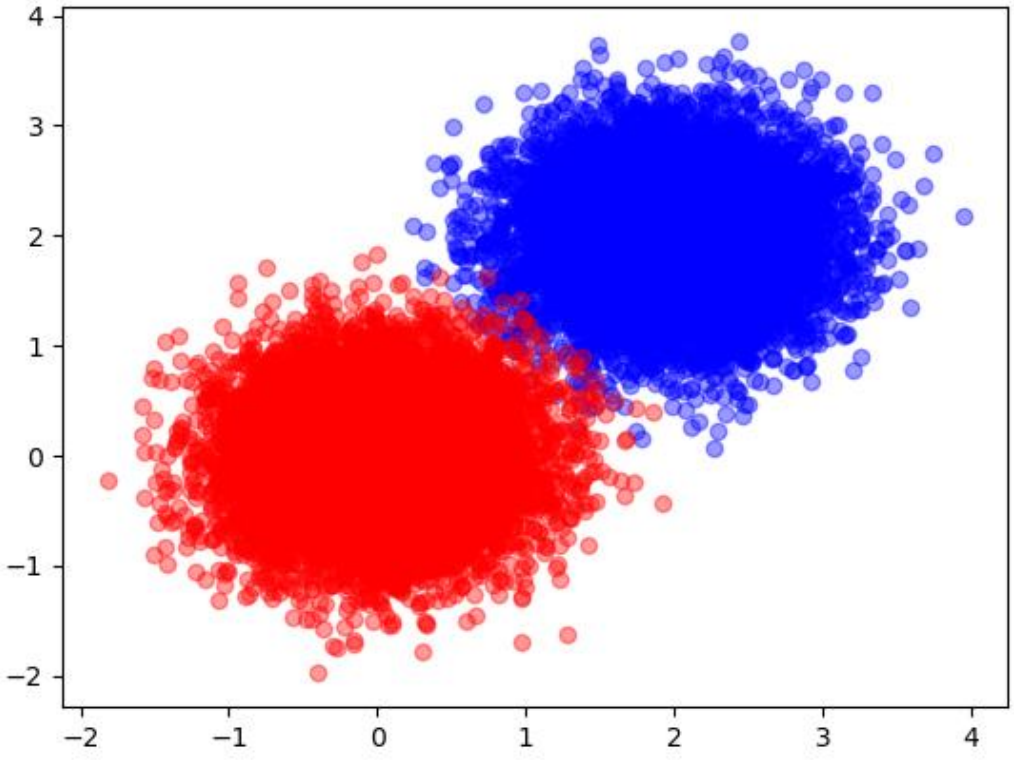
[۱] Label

۲-۲- فریب دادن الگوریتم شبکه عصبی پیچشی

فرض کنید که یک تصویر از عدد دو داریم که وقتی آن را به مدل خود که توسط الگوریتم شبکه عصبی پیچشی از یادگیری عمیق ساخته شده می دهیم به درستی عدد دو را تشخیص می دهد. زمانی که روی آن نویزی برنامه ریزی شده به وجود می آوریم (این نویز ها یک نویز تصادفی نیستند و با الگوریتم های خاصی به وجود آمده‌اند و هدف آن ها این است که مدل به جای عدد دو یک عدد به خصوص دیگری مثل عدد پنج را تشخیص دهد) هم چنان از دید چشم انسان این تصویر نشان دهنده‌ی عدد دو می باشد و فقط کمی نویز در آن وجود دارد ولی اگر همین عکس را برای تشخیص به مدل خود بدهیم ناگهان خواهیم دید که عدد دیگری را برای آن در نظر می گیرد. (به عنوان مثال فکر می کند که عکس مورد نظر عدد پنج را نشان می دهد.) این اتفاق ممکن است در مثال های روزمره تبعات بسیار زیادی برای ما داشته باشد. برای همین باید راه حلی برای آن بیابیم که احتمال به وجود آمدن چنین خطا هایی بسیار کم شود. در ادامه به بررسی حمله هایی که ممکن است حمله کننده ها برای ایجاد اشتباه در تشخیص های مدل به وجود بیاورند و همچنین به راه های جلوگیری از این حملات خواهیم پرداخت.

۳-۲- حمله‌ی شبکه‌ی تقابلی

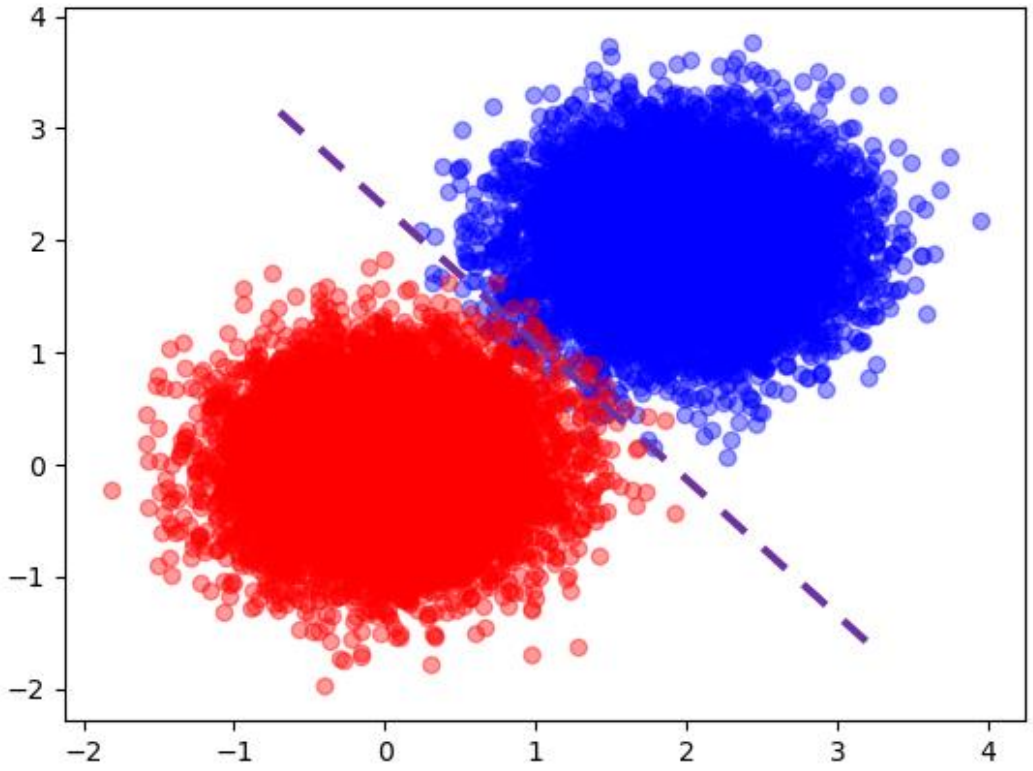
در حمله‌ی شبکه‌ی تقابلی [۱] فرض کنید که ما میخواهیم یک طبقه بندی میان داده های آبی و قرمز در شکل پایین انجام دهیم و اگر داده‌ی جدیدی دریافت شد به درستی تشخیص دهیم که جزو داده های دسته‌ی آبی می باشد یا جزو داده های دسته‌ی قرمز. همانطور که مشخص می باشد مرز تصمیم گیری [۲] باید دقیقا میان این دو منطقه باشد که به صورت یک خطچین در شکل مشخص شده است. با وجود این مرز هر داده‌ی جدیدی که در سمت راست و بالای این مرز باشد جزو داده های آبی محسوب می شود و هر داده‌ای که در سمت چپ و پایین این مرز باشد به جزو داده های قرمز محسوب خواهد شد.



شکل ۴ داده های آبی و قرمز قبل از مرزبندی

[۱] Adversarial Attack

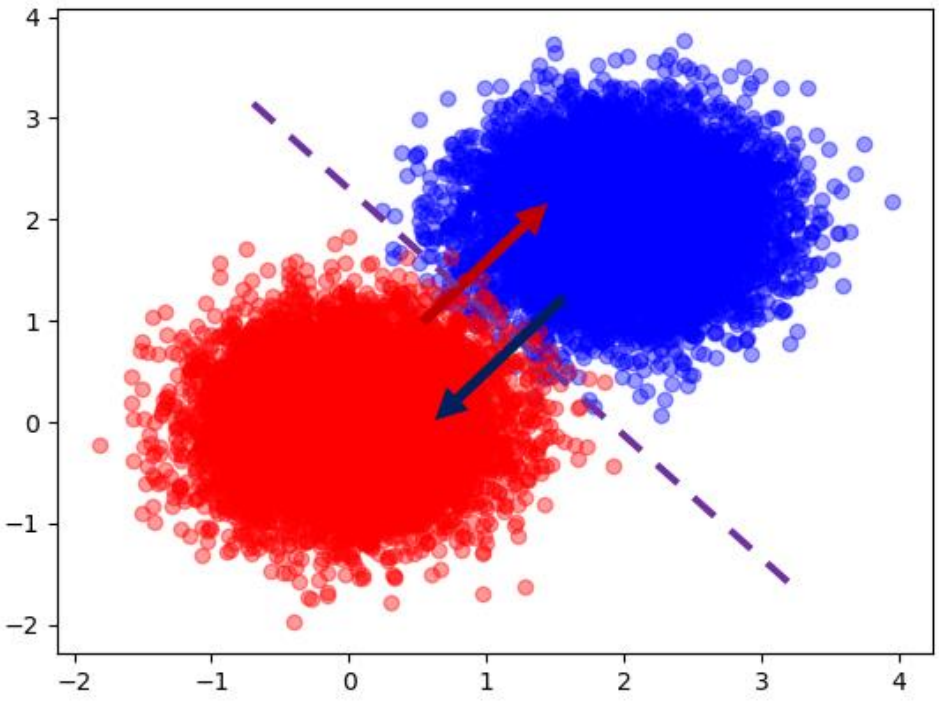
[۲] decision boundary



شکل ۵: چگونگی قرار گیری مرز تصمیم گیری

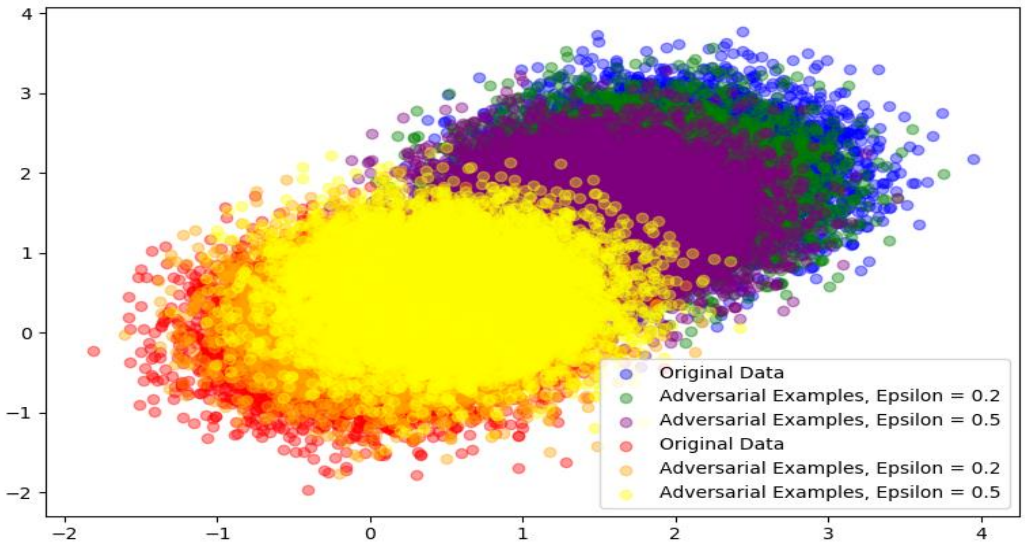
نمونه های مخالف[۱] در واقع نمونه هایی هستند که به خوبی برای فریب دادن مدل ما برای اینکه به اشتباه بیافتند طراحی شده‌اند. حال باید ببینیم که چگونه می توانیم مدل خود را فریب بدهیم که در این مثال خاص باید کاری کنیم که یک نمونه که در اصل قرمز می باشد به عنوان یک نمونه‌ی آبی شناسایی شود یا برعکس یک نمونه‌ای که در اصل آبی می باشد به عنوان یک نمونه‌ی قرمز توسط مدل ما شناسایی شود. همانطور که مشخص است ما نمی توانیم برای تحقق این کار خط چین مشخص کننده‌ی مرز تصمیم گیری را اجابجا کنیم و به یک مکان دیگر ببریم تا مدل به اشتباه بیافتد. پس برای اینکه مدل به اشتباه بیافتد باید از راهکار دیگری استفاده کنیم. که اگر بخواهیم به زبان ساده این راهکار را از روی شکل توصیف کنیم باید کاری کنیم که نقاط آبی به سمت محل نقاط قرمز بروند و نقاط قرمز به سمت محل نقاط آبی بروند و اینگونه مدل را فریب بدهیم.

[۱] Adversarial Examples



شکل ۶: فریب مدل با جابجایی محل نقاط آبی و قرمز

اپسیلون[۱] در واقع یک عدد برای کنترل کردن شدت حمله با استفاده از نمونه های مخالف می باشد. هر چه اپسیلون بیشتر باشد شدت حمله بیشتر خواهد بود ولی بدلیل اینکه در اپسیلون های زیاد تر تغییرات عکس بسیار زیاد خواهد شد و به طور کلی تغییر شکل خواهد داد و چشم انسان نیز خواهد توانست آن را تشخیص دهد پس نمی توان اپسیلون را تا مقدار دلخواهی زیاد کرد.[۲]



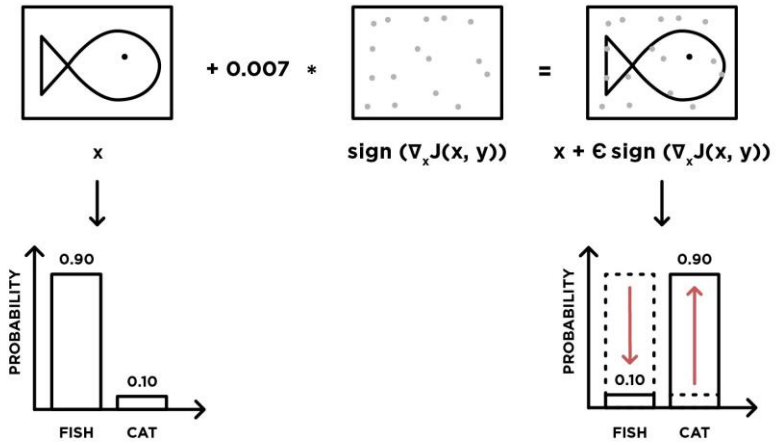
شکل ۷: فریب مدل با اپسیلون های مختلف

[۱] Epsilon

۴-۲- روش نشانه‌ی گرادیان سریع

هدف روش نشانه‌ی گرادیان سریع[۱] این است که یک نویز که رندوم نیست و در راستای شیب تابع زیان در تصویر اولیه می باشد را به تصویر اولیه اضافه کند. مقیاس این نویز یک متغیر به نام اپسیلون می باشد که همانطور که پیش از این به آن اشاره شد معیاری برای شدت حمله می باشد. و اپسیلون معمولا یک عدد بسیار کوچک می باشد. در فرمول روش نشانه‌ی گرادیان سریع در واقع مقدار شیب اهمیتی ندارد ولی منفی یا مثبت بودن آن اهمیت دارد.

اگر بخواهیم به صورت دقیق تر در مورد این روش صحبت کنیم باید بگوییم که در این روش حمله کننده از همان روشی که باعث شکل گرفتن مدل و به وجود آمدن یک مرز دقیق برای طبقه بندی تصاویر مختلف می شود برای حمله‌ی خود استفاده می کند. در واقع در این جا الگوریتم شبکه عصبی پیچشی برای اینکه بتواند تصاویر را تشخیص دهد و آنها را طبقه بندی کند باید تعدادی تصویر را گرفته و به اصطلاح آنها را تمرین[۲] کند. در این الگوریتم وقتی یک تصویر به عنوان داده‌ی تمرینی به الگوریتم داده می شود، یک پیشبینی از آن انجام می شود و بعد از آن الگوریتم با توجه به برچسب صحیح تصویر مورد نظر و پیشبینی-ای که توسط مدل انجام شده است توسط یک تابع زیان[۳] (مثل تابع زیان ام اس ای[۴]) میزان اشتباه بودن پیش بینی اندازه‌گیری می شود و در طول تمرین تصاویر سعی می شود که مقدار خروجی تابع زیان کمتر و کمتر شود. حال اگر حمله کننده از همین روش استفاده کند که یک تصویر را با استفاده از به وجود آوردن نویز های برنامه ریزی شده طبق تابع زیان به سمتی هدایت کند که با یک تصویر دیگر توسط مدل اشتباه گرفته شود در واقع از روش نشانه‌ی گرادیان سریع استفاده شده است.



شکل ۸ چگونگی کارکرد روش نشانه گرادیانت سریع

[۱] Fast gradient sign method

[۲] Train

[۳] Loss Function

[۴] MSE

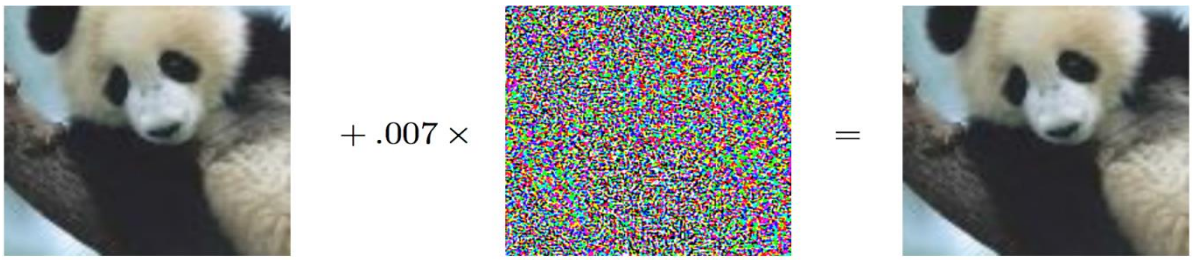
فرمول روش نشانه‌ی گرادیان سریع به صورت زیر می باشد که در آن:

- X همان تصویر اولیه می باشد که تغییر نکرده است.
- X^{adv} در واقع تصویر ما بعد از اعمال روش نشانه‌ی گرادیان سریع بر روی تصویر اولیه می باشد.
- ε یک ثابت برای تعیین شدت حمله می باشد.
- J همان تابع زیان طبقه بندی می باشد.
- Y_{true} هم برچسب صحیح تصویر X می باشد.
- ∇_x نشان دهنده‌ی شیب تابع هزینه[۱] با توجه به X می باشد.

$$x^{adv} = x + \varepsilon \cdot \mathrm{sign}(\nabla_x J(x, y_{true})),$$

فرمول ۱

در تصویر زیر مشاهده می کنید که با استفاده از روش نشانه‌ی گرادیان سریع تصویر اولیه که یک پاندا می باشد و به درستی توسط مدل تشخیص داده شده است با اضافه شدن نویزی به یک تصویر دیگر تبدیل شده است که همانطور که در شکل نشان داده است به جای پاندا مدل فریب خورده و آن را یک گیبون[۲] تشخیص داده است. همانطور که در تصویر مشخص می باشد اپسیلون مقدار بسیار کمی دارد و برابر با ۰.۰۰۷ می باشد و همانطور هم که مشخص هست وجود این نویز توسط چشم زیاد قابل تشخیص نمی باشد و فقط مدل را تحت تاثیر قرار می دهد.[۳]



شکل ۹ : اعمال نویز روی یک تصویر

[۱] Gradient Of Cost Function

[۲] Gibbon

فصل ۳:

روش حل مسئله

۳-۱- آموزش شبکه‌ی تقابلی [۱]:

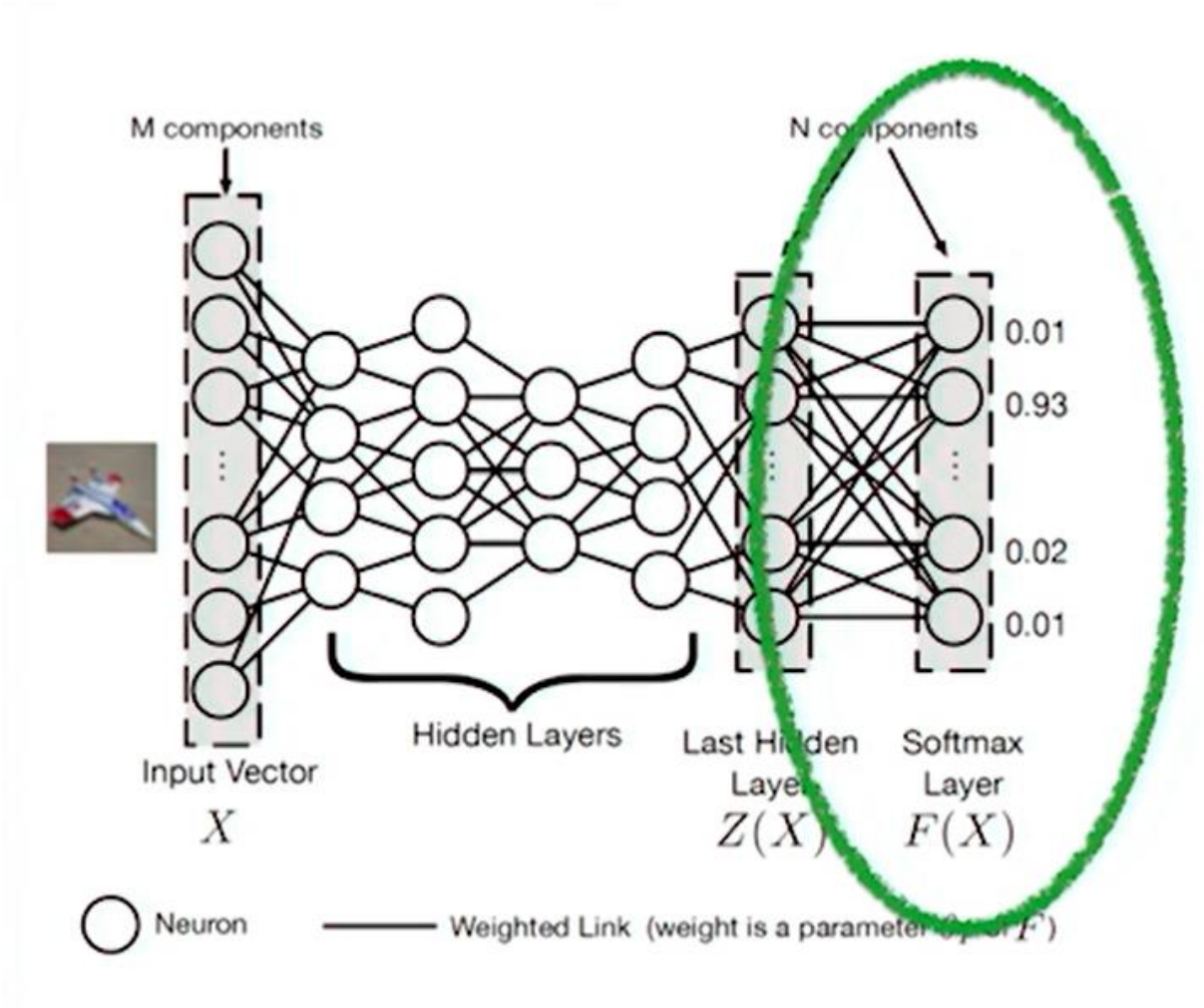
حال به سراغ مکانیزمی برای دفاع در مقابل حمله‌ی روش نشانه‌ی گرادیان سریع می رویم. برای دفاع در مقابل روش نشانه‌ی گرادیان سریع در ابتدا باید یک مدل مبتنی بر الگوریتم شبکه عصبی پیچشی برای داده های عادی مثل Mnist یا Cifire طراحی کنیم. در مرحله‌ی بعد باید این مدل را سیو کرده و به صورت یک فایل نگهداریم. حال در ادامه برنامه‌ای می نویسیم که این مدل را گرفته و با داده های دیگری آن را تمرین دهد. در واقه در این روش برای دفاع دقیقا مانند حمله کننده که از روش خود الگوریتم شبکه عصبی پیچشی برای طراحی مدل استفاده شد در اینجا نیز دفاع کننده برای دفاع در برابر حمله‌ی حمله کننده از مکانیزم خود حمله کننده برای دفاع در مقابل حمله استفاده می کند به این صورت که با استفاده از داده های اولیه‌ای که در اختیار دارد سعی می کند تصاویر اولیه را توسط روش روش نشانه‌ی گرادیان سریع نویز دار کند و سپس آن تصاویر را با برچسب های غلط که در واقع برچسب درست می باشد تمرین دهد.

۳-۲- دفاع تقطیری

دفاع تقطیری[۲] یک مکانیزم دفاعی دیگر پیاده‌سازی شده برای مقابله با حمله کننده ها می باشد. راه حل آموزش شبکه‌ی تقابلی که در بحث قبل مطرح شد در برابر حمله های روش نشانه‌ی گرادیان سریع یه خوبی دفاع را انجام می دهد و با دقت خوبی تصاویر نویز دار شده با این روش را به درستی تشخیص می دهد و برچسب درست آن تصویر را اعلام می کند ولی راه حل دیگری که پیاده سازی آن انجام شد دفاع تقطیری می باشد که در این روش ما اصلا مانند روش قبل به داده های نویز دار شده کاری نداریم و با یک روش مستقل از داده های نویز دار شده و فقط با داده های عادی از حمله‌ی مهاجم جلوگیری می کنیم. در این روش در ابتدا یک مدل برای الگوریتم شبکه عصبی پیچشی با داده های عادی (مثلا داده های Mnist) طراحی می کنیم ولی این مدل یک تفاوت با مدل های عادی الگوریتم شبکه عصبی پیچشی دارد و این تفاوت این است که دما[۳] در این مدل بجای عدد یک برروی اعداد بالاتری خواهد بود. دما در واقع یک معیار برای تعیین مقدار دقیق بودن و با اطمینان بودن پیش بینی های انجام شده توسط مدل است. در واقع مدل های ما یک تابع SoftMax دارد که نورون های لایه یکی مانده به آخر که لایه‌ی آخر لایه های پنهانی می باشد را به نورون های لایه‌ی آخر تبدیل می کند که هر نورون در لایه آخر نشان دهنده‌ی احتمال هر برچسب می باشد. مثلا در داده های Mnist لایه آخر ۱۰ تا نورون دارد که هر کدام نشان دهنده‌ی یک

[۱] Adversarial Training
[۲] Defensive Distillation
[۳] Temperature

درصد می باشند که این درصد ها درواقع پیشبینی آن مدل از یک تصویر می باشد. به عنوان مثال اگر یک تصویر عادی از عدد دو را به یک مدل عادی از داده های Mnist بدهیم احتمالا نورون مربوط به پیشبینی اینکه این تصویر دارای برچسب عدد دو می باشد عددی نزدیک به صد درصد خواهد بود و بقیه نورون های مربوط به بقیه اعداد عددی ناچیز خواهد بود. در اینگونه مدل ها دما به طور پیش فرض روی عدد یک می باشد ولی وقتی دما عددی مثل ۴۰ باشد پیشبینی ها دقت کمتری خواهند داشت و درصد های در بقیه نورون ها هم پخش خواهد شد و پیشبینی به اصطلاح نرم تر خواهد شد.

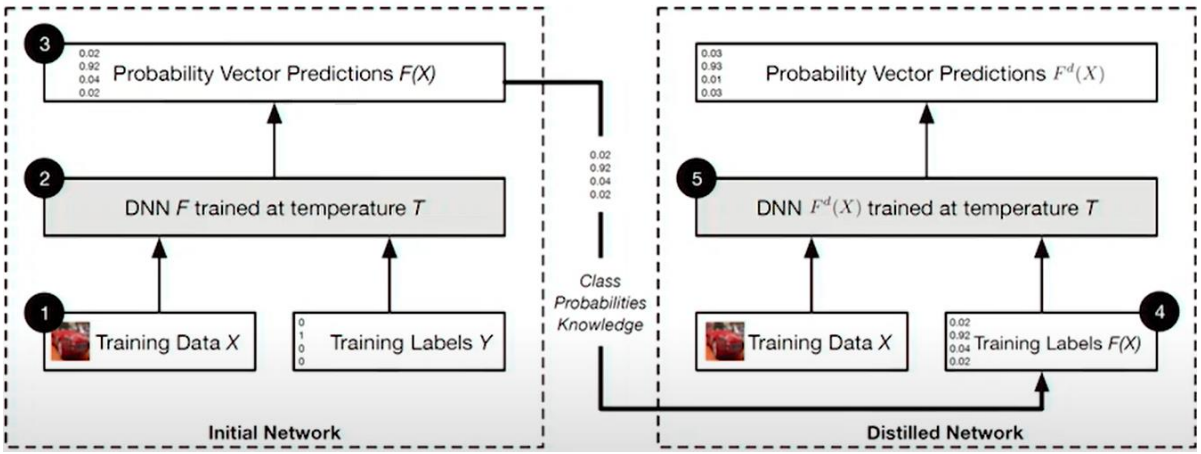


شکل ۱۰ : لایه های مختلف الگوریتم شبکه عصبی پیچشی و لایه‌ی سافت مکس

بعد از اینکه یک مدل با دمای بالا ساختیم باید یک مدل دیگر نیز بسازیم. مدل جدید یک مدل مبتنی بر الگوریتم شبکه عصبی پیچشی با داده های عادی (مثلا داده‌ی عادی Mnist) می باشد که برچسب هایش برای تمرین داده های خود را از پیش بینی های بردار احتمال[۱] مدل قبلی که خروجی مدل قبلی می باشد می گیرد و دما اش همان عدد یک می باشد.

[۱] Probability Vector Predictions

در واقع برچسب هایی که در مدل های عادی به یک مدل می دهند یک برچسب قطعی می باشد مثلا وقتی تصویر یک عدد پنج در مدل های داده‌ی Mnist به مدل داده می شود برچسب آن به طور قطعی عدد پنج می باشد ولی در این روش مدل دوم که برچسب هایش بر اساس پیش بینی های انجام شده توسط مدل قبل است، برچسب های مورد استفاده برای تمرین مدل برچسب هایی قطعی نمی باشند. به عنوان مثال وقتی تصویر عدد پنج به مدل داده می شود برچسب آن عدد پنج به طور قطعی نمی باشد بلکه برچسب آن به این صورت است که مثلا به احتمال نود درصد این تصویر پنج است و به احتمال های دیگر نه عدد دیگر (مثلا به احتمال دو درصد این تصویر هشت می باشد، به احتمال سه درصد این تصویر عدد هفت می باشد و ....) که این باعث می شود مدل ما در مقابل داده های نویز دار شده برای فریب دادن مدل مقاوم شود.[۴]



شکل ۱۱ : نمای کلی از الگوریتم دفاع تقطیری

فصل ۴:

نتیجه‌گیری و پیشنهادات

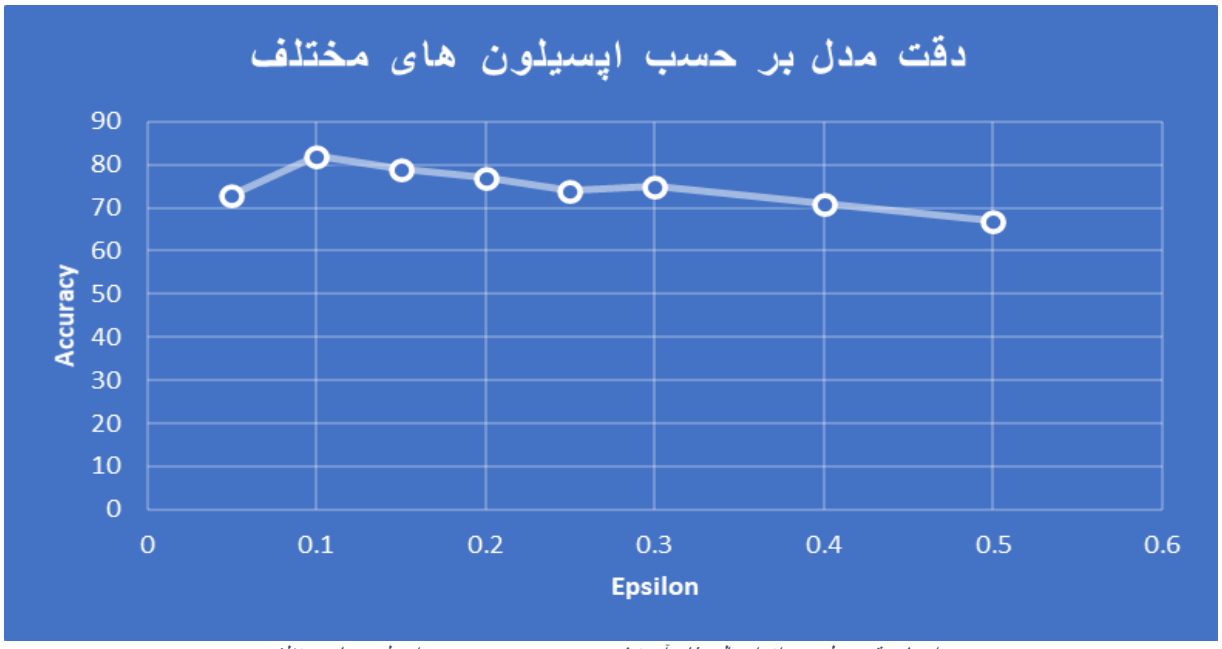
۱-۴- نتایج

نتایج بدست آمده نتیجه‌ی ساختن مدل های مختلف با اپسیلون ها و دما های مختلف با استفاده از کد های پیاده سازی شده توسط اینجانب به زبان پایتون با توجه به دفاع های مطرح شده در گزارش و آزمایش مدل ها با استفاده از حمله‌ی شبکه‌ی تقابلی می باشد.

نتیجه‌ای که حاصل شد این است که با وجود اینکه حمله های مربوط به فریب دادن الگوریتم های یادگیری عمیق می توانند بسیار خطرناک باشند. ولی با استفاده از راه های دفاعی مختلف می توان جاوی این حملات را گرفت.

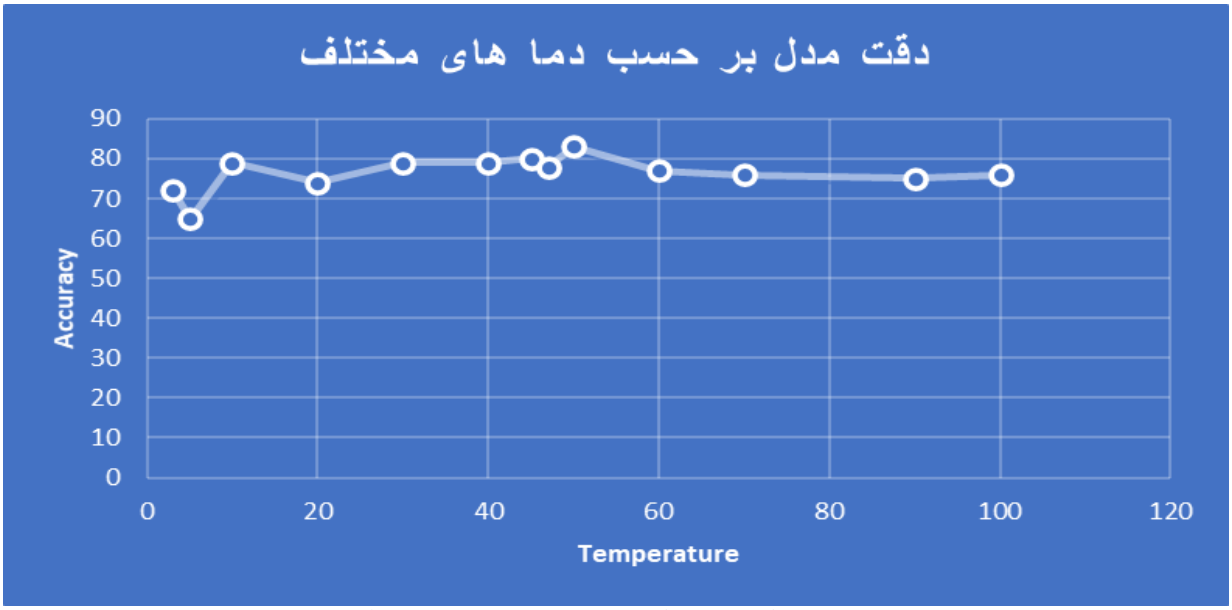
نتایج حاصل شده از پیاده سازی دو روش دفاعی آموزش شبکه‌ی تقابلی و دفاع تقطیری برای دفاع در مقابل فریب دادن الگوریتم شبکه عصبی پیچشی توسط روش نشانه‌ی گرادیان سریع نشان داد که اگر روی تصاویر مجموعه داده‌ی Mnist با استفاده از روش روش نشانه‌ی گرادیان سریع نویز به وجود بیاوریم می توان با دقت ۸۳ درصد هم چنان پیشبینی درستی از اعداد تصاویر این مجموعه داشت. و عملیات طبقه بندی با دقت ۸۳ درصدی انجام خواهد شد.

در نمودار زیر دقت مدل های مختلف که دفاع آموزش شبکه‌ی تقابلی با اپسیلون های مختلف روی آن ها اعمال شده است، به نمایش در آمده است. همانطور که مشخص است حداکثر دقت مدل در اپسیلون ۰.۱ می باشد.



نمودار ۱: دقت مدل پس از اعمال دفاع آموزش شبکه‌ی تقابلی بر حسب اپسیلون های مختلف

در نمودار زیر دقت مدل های مختلف که دفاع تقطیری با دما های مختلف روی آن ها اعمال شده است، به نمایش در آمده است. همانطور که در نمودار مشخص می باشد حداکثر دقت مدل در دمای ۵۰ می باشد.

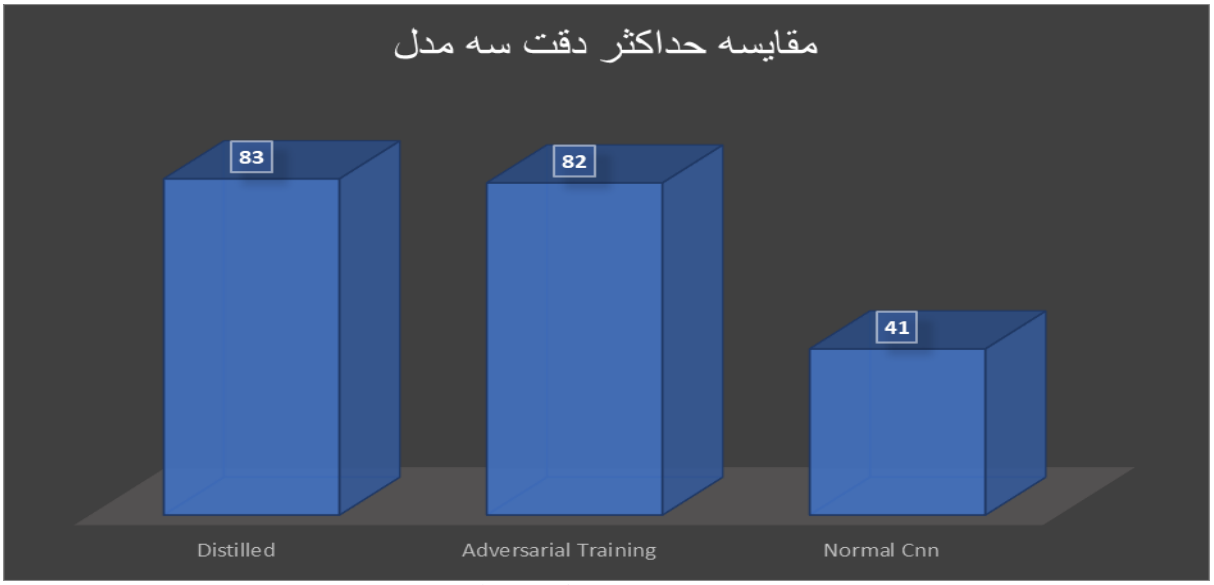


نمودار ۲: دقت مدل پس از اعمال دفاع تقطیری بر حسب دما های مختلف

در نمودار زیر یک مقایسه بین حداکثر دقت مدل های این سه نوع مقایسه شده است:

- مدل الگوریتم شبکه عصبی پیچشی تمرین داده شده با استفاده از روش دفاع تقطیری با دمای ۵۰
- مدل الگوریتم شبکه عصبی پیچشی تمرین داده شده با استقاده از روش آموزش شبکه‌ی تقابلی با اپسیلون ۰.۱
- یک مدل معمولی الگوریتم شبکه عصبی پیچشی

همانطور که در نمودار هم مشخص می باشد دقت مدل با استفاده از این روش های دفاعی تقریبا دو برابر شده است و به خوبی جلوی حمله‌ی انجام شده توسط روش نشانه‌ی گرادیان سریع برای فریب دادن مدل گرفته شده است.



نمودار ۳: مقایسه دقت مدل های مختلف

حال به مزایای این دو روش می پردازیم:

از مزایای روش دفاع آموزش شبکه‌ی تقابلی می توان به موارد ذیل اشاره کرد:

- در صورت اطلاع از اپلیسون الگوریتم اعمال شده برای حمله‌ی شبکه‌ی تقابلی می توان همان اپسیلون را برای دفاع آموزش شبکه‌ی تقابلی اعمال کرد تا بتوانیم دقت مدل را بسیار بالا ببریم و به طور میانگین دقت حاصل شده از این الگوریتم از الگوریتم های دیگر بالاتر می باشد زیرا این مدل عملیات تمرین داده را بر روی همان داده های نویز دار شده انجام می دهد.

معایب این روش نیز موارد ذیل می باشد:

- در صورت ندانستن اپلیسون الگوریتم اعمال شده برای حمله‌ی شبکه‌ی تقابلی کار سخت تر می شود و باید یک اپسیلون که با توجه به اطلاعات حملات قبلی محتمل تر می باشد انتخاب شود که باعث کمتر شدن دقت پیشبینی ها می شود ولی همچنان پیشبینی ها از دقت خوبی برخوردار هستند. از دیگر معایب این روش در صورت دانستن حدود اپسیلون الگوریتم اعمال شده برای حمله‌ی شبکه‌ی تقابلی به توان به این مورد اشاره کرد که هر بار که اپسیلون الگوریتم اعمال شده برای حمله‌ی شبکه‌ی تقابلی تغییر کند باید الگوریتم دفاعی را با یک اپسیلون متفاوت دوباره اجرا کنیم همچنین این روش کارایی پایینی نیز دارد.

از مزایای روش دفاع تقطیری می توان به موارد ذیل اشاره کرد:

- از این روش می توان بدون اینکه اطلاعات زیادی درباره‌ی حمله‌ی شبکه تقابلی داشته باشیم استفاده کنیم. زیرا این روش بدون اینکه اطلاعاتی درباره‌ی حمله‌ی اعمال شده (مثل اپسیلون در حمله‌ی شبکه تقابلی) داشته باشد بر روی الگوریتم شبکه عصبی پیچشی مورد نظر پیاده‌سازی می شود.

معایب این روش نیز موارد ذیل می باشد:

- این روش نسبت به روش دفاع آموزش شبکه‌ی تقابلی معایب خاصی ندارد و بهتر از روش دفاع آموزش شبکه‌ی تقابلی می باشد ولی همچنان معایبی دارد مثلا در این روش افزونگی وجود دارد و عملیات تمرین داده ها دو بار صورت می گیرد و همچنین بخاطر همین موضوع دارای کارایی پایینی می باشد.

۲-۴- پیشنهادات

در آینده تحقیقاتی بر روی روش دفاعی مدل های مولد[۱] انجام خواهد شد. این روش اکثرا برای کاربرد طبقه بندی در یادگیری عمیق استفاده می شود. در این روش دفاع در برابر حملاتی که برای فریب دادن یک مدل یادگیری عمیق می باشد هیچ تغییری روی مدلی که حمله قرار است روی آن صورت گیرد انجام نمی شود. و مدل تمرین داده شده با الگوریتم های یادگیری عمیق به همان صورت باقی می ماند. در واقع این روش دفاعی بجای اینکه مدل را تغییر بدهد تصویری که نویز بر روی آن اعمال شده است را بازسازی میکند. این روش به این صورت است که یک مدل جدید ساخته می شود که بتواند تصویر نویز دار شده را دریافت کرده و یک تصویر دیگر که به آن شباهت دارد و بدون نویز است را پیدا کند. سپس تصویر جدید تولید شده که دارای نویز نمی باشد به مدل اولیه‌ی ما جهت طبقه بندی داده می شود. از مزایای این روش می توان به این نکته اشاره کرد که مدل اولیه را دچار تغییر نمی کند و مدل اولیه در این روش می تواند از هر روشی در یادگیری عمیق استفاده کند.

[۱] Defense-GAN

مراجع

مراجع

[۱] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach," *Procedia Computer Science,* vol. 132, pp. 679-688, 01/01/2018/ .2018

[2] S. A. Fezza, Y. Bakhti, W. Hamidouche ,and O. Déforges, "Perceptual Evaluation of Adversarial Attacks for CNN-based Image Classification," in 2019*Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*, 2019, pp. .1-6

[3] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial Attacks and Defenses in Deep Learning," *Engineering,* vol. 6, no. 3, pp. 346-360, 01/03/2020/ .2020

[4] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, *Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks*. 2016, pp.582-597 .

واژه‌نامه فارسی به انگلیسی

برنامه های کاربردی...Applications

وسایل نقلیه خود مختار.....Autonomous vehicles

الگوریتم شبکه عصبی پیچشی..Convolutional Neural Network

الگوریتم یادگیری عمیق...Deep Learning

لایه های پنهان...Hidden layers

نگاشت ویژگی..Feature Map

برچسب.....Label

حمله‌ی شبکه‌ی تقابلی..Adversarial Attack

مرز تصمیم گیری.....Decision boundary

نمونه های مخالف.....Adversarial Examples

اپسیلون.....Epsilon

حمله با استفاده از نمونه های مخالف..Adversarial attack

روش نشانه‌ی گرادیان سریع.....Fast gradient sign method

تمرین...Train

تابع زیان..Loss Function

شیب تابع هزینه.....Gradient Of Cost Function

آموزش شبکه‌ی تقابلی...Adversarial Training

دفاع تقطیری.....Defensive Distillation

دما...Temperature

پیش بینی های بردار احتمال.....Probability Vector Predictions

روش دفاعی مدل های مولد...Defense-GAN

Abstract: In the first step of this report we are introducing Convolutional Neural Networks (CNN) and its utilities in daily life for human. Following by that there is going to be an introduction about Mnist dataset. In the next step there is an overview about fooling generative adversarial networks (GAN) on CNN s and specifically about fooling CNN s through fast gradient sign method. Finally defending methods against fooling of CNN s has been studied and some implementations has been exhibited on Python for different ways of fooling and defenses against them. Results has shown that attacks can be defended in significant amount of experiments. These results are extracted from implementing, attacking and defending algorithms on Mnist data set. These algorithms include adversarial networks and distilling defense. furthermore, results have shown that defending on fast gradient sign method can successfully neutralize their effect but there are more recent methods that can still have their effects. And there should be even more and newer ways to be contributed in defending these fooling effects.

Keywords: CNN Fooling – Adversarial Attack – Fast Gradient Sign Method – Adversarial Training - Defensive Distillation

Iran University of Science and Technology
Computer Engineering Department

Robust Convolutional Neural Network

Bachelor of Science Thesis in Computer Engineering

By:
Amirhossein Aminimehr

Supervisor:
Dr. Naser Mozayeni

August 2020