سیستم تشخیص نفوذ مبتنی بر روش یادگیری عمیق

پارسا قزوینیان

زمستان ۹۷

چکیده

سیستم‌های تشخیص نفوذ (IDS) وظیفه‌ی شناسایی و تشخیص هر گونه استفاده‌ی غیرمجاز از سیستم، سوء استفاده و یا آسیب‌رسانی توسط هر دو دسته‌ی کاربران داخلی و خارجی را بر عهده دارند. تحلیل جریان‌های ترافیک شبکه جهت تشخیص نفوذ به دو روش کلی تشخیص سواستفاده و تشخیص ناهنجاری انجام می‌شود. در روش تشخیص سواستفاده با استفاده از الگوها و امضاهای از پیش تعیین شده، حملات شناخته شده فیلترشده و تشخیص داده می‌شوند. این روش وابسته به بروزرسانی دائمی پایگاه داده‌ی امضاها توسط افراد می‌باشد. این روش در یافتن حملات شناخته شده با دقت بالایی عمل می‌کند، اما در مورد حملات مشاهده نشده عملکردی کاملاً غیرموثر دارد. در مقابل روش‌های تشخیص ناهنجاری از مکانیزم‌های ابتکاری استفاده می‌کنند تا رفتار نرمال سیستم و شبکه را مدل کنند و در صورت وجود انحراف از رفتار مدل نرمال، فعالیت‌های مخرب و ناهنجار را تشخیص دهند. از این روش برای تشخیص حملات مشاهده نشده استفاده می‌شود. روش‌های یادگیری ماشین خصوصاً روش‌های یادگیری عمیق با قابلیت‌هایی که در استخراج ویژگی‌های سطح بالا و تشخیص وابستگی‌های دراز مدت در زمینه‌های دیگر مثل پردازش زبان طبیعی و ... نشان داده‌اند، می‌توانند به عنوان ابزاری مناسب برای تشخیص الگوهای حملات و نفوذها و درنتیجه در سیستم های تشخیص نفوذ به کار گرفته شوند. بنابراین در این جا با استفاده از شبکه‌های عمیق کانولوشن و بازگشتی سعی شده با شناسایی الگوها و رفتارهای ترافیک نرمال و غیرنرمال بتوانیم آن‌ها را با نرخ خطای مثبت(false positive) قابل قبول طبقه‌بندی کنیم.

کلمات کلیدی- تشخیص نفوذ، تحلیل ترافیک، شناسایی الگو، یادگیری ماشین، یادگیری عمیق، شبکه کانولوشن، شبکه بازگشتی

مقدمه

روش های سنتی تشخیص نفوذ در تشخیص حملاتی که قبلاً مشاهده نشده اند، کارایی ضعیفی از خود نشان می دهند. روش های تجاری فعلی، از اندازه گیری های آماری یا روش های محاسبه حد آستانه استفاده می کنند، که این روش ها وابسته به پارامتر های شبکه مانند طول بسته، فاصله زمانی ما بین رسیدن بسته ها اندازه جریان و ... می باشد و از این پارامتر ها برای مدل کردن ترافیک شبکه استفاده می شود، اما این روش نمی تواند راه حل موثری برای تشخیص حملات پیچیده و در حال تکامل امروزی باشد.

سیستم های خود فراگیر یکی از روش‌های موثر هستند که با استفاده از مفاهیم یادگیری ماشین مانند الگوریتم‌های نظارت‌شده و بدون نظارت قادر هستند نفوذ های شناخته شده و حتی دیده نشده را تشخیص داده و طبقه‌بندی کنند. با این حال توسعه روش‌های موثر مبتنی بر یادگیری ماشین جهت تشخیص نفوذهای شبکه و استقرار سیستم‌های تشخیص نفوذ بلادرنگ در مراحل ابتدای قرار دارد. با وجود اینکه روش‌های متعددی یافته شده است، اما جهت استفاده به صورت بلادرنگ بسیار ناکارآمد هستند. بیشتر این روش‌ها نرخ خطای مثبت (False Positive) بالایی دارند و هم‌چنین هزینه محاسباتی بالایی نیز به همراه دارند. این به این خاطر بود که در بیشتر این روش‌ها الگوهای نرمال و حمله که جهت یادگیری استفاده می‌شد محدود، محلی و در مقیاس کوچک بودند و ویژگی(Feature)های آن‌ها نیز سطح پایین بود. روش‌های جدیدتر در این حوزه یادگیری ماشین که روش‌های یادگیری عمیق نام دارند و مدل پیچیده‌تر روش‌های پیشین می‌باشد دارای دو ویژگی هستند که سبب می‌شوند قابلیت و کارایی بیشتری در این زمینه داشته باشند و رویکرد استفاده از روش های یادگیری را به سمت عملی شدن و استفاده به عنوان سیستم تشخیص نفوذ بلا درنگ با سرعت خط(line speed) ببرند که این ویژگی ها عبارت اند از: قابلیت بازنمایی سلسله مراتبی ویژگی ها و قابلیت یادگیری ارتباطات و وابستگی های طولانی مدت در الگو های زمانی در توالی داده‌های مقیاس بزرگ.

هم‌چنین پیشرفت‌های اخیر در مکانیزم روش‌های بهینه‌سازی و واحدهای پردازش گرافیکی(GPU) که از محاسبات موازی و توزیع‌شده پشتیبانی می‌کنند، قابلیت آموزش آسان الگوریتم‌های یادگیری عمیق را فراهم کرده‌اند.

بنابراین در این جا با استفاده از شبکه‌های عمیق کانولوشن و بازگشتی سعی در استخراج ویژگی‌های سطح بالای اتصالات ترافیک شبکه هستیم تا با شناسایی الگوها و رفتارهای آن‌ها بتوانیم ترافیک نرمال و غیرنرمال را طبقه‌بندی کنیم و در فاز بعدی نوع حمله را نیز بتوانیم مشخص کنیم.

تعریف مفاهیم

شبکه‌های کانولوشن یا شبکه‌های عصبی کانولوشن[1]افزونه‌ای بر شبکه سنتی (FFN)[2]هستند که با الهام گرفتن از علم عصب‌شناسی در زیست شناسی به وجود آمده‌اند. این شبکه‌ها در ابتدا برای پردازش تصاویر به کار گرفته شدند. در این جا وقایع ترافیک شبکه به صورت داده‌های سری زمانی مدل شده‌اند که به صورت میلیون‌ها رکورد اتصالات نرمال ومخرب شبکه در دسترس هستند و ترکیبی از شبکه کانولوشن وروش‌های بازگشتی بر روی آن‌ها اعمال می‌شود. CNN از یک لایه یک بعدی کانولوشن، یک لایه یک بعدی pooling، لایه کاملاً متصل[3]و یک تابع فعال‌سازی غیر خطی به نام ReLU تشکیل شده است.

داده‌های یک بعدی وقایع ترافیک شبکه که به فرم سری زمانی می‌باشد به عنوان بردار ورودی

$x = (x_1.x_2.\ldots.x_{n-1}.x_{41}.cl)$ که $x_1 \in R^d$ ویژگی‌ها و $cl \in R$ به معنای برچسب کلاس می‌باشد. لایه کانولشن یک بعدی یک نگاشت ویژگی f_m از طریق اعمال عملیات کانولوشن بر روی داده ورودی با یک فیلتر $w \in R^{fd}$ که f همان ویژگی‌ها در بسته‌های TCP/IP انجام می‌دهد که نتیجه یک مجموعه جدید از ویژگی هاست. نگاشت ویژگی جدید f_m از روی مجموعه ویژگی‌های f به صورت زیر بدست می‌آید.

$$hl_i^{fm} = tanh(w^{fm}x_{i:i+f-1} + b)$$

که $b \in R$ جمله بایاس است. فیلتر hl بر روی هر کدام از مجموعه ویژگی‌های f در یک رکورد اتصال TCP/IP $\{x_{1:f}.x_{1:f+1}.\ldots.x_{n-f-1}\}$ اعمال می‌شود تا یک نگاشت ویژگی به صورت زیر تولید شود:

$$hl = [hl_1.hl_2.\ldots.hl_{n-f+1}]$$

که $hl \in R^{n-f+1}$ و سپس عملیات max-pooling را بر روی هر نگاشت ویژه به صورت $\overrightarrow{hl} = \max\{hl\}$ می‌باشد، اعمال می‌کنیم. این کار سبب می‌شود که مهم‌ترین ویژگی‌ها یعنی ویژه با بیشترین ارزش انتخاب شود. این ویژگی‌ها وارد لایه کاملاً متصل می‌شوند. لایه کاملاً متصل شامل تابع

[1] Convolutional Neural Network
[2] Feed Forward Network
[3] Fully Connected Layer

softmax می‌باشد که توزیع احتمال هر کلاس را بدست می‌آورد. لایه کاملاً متصل به زیان ریاضیات به صورت زیر بیان می‌شود:

$$o_t = softmax(w_{h0}hl + b_0)$$

- **شبکه‌های ترکیبی**

شبکه استفاده شده ترکیبی از شبکه‌های کانولوشن و بازگشتی[4] می‌باشد. وقایع ترافیک شبکه دارای الگوی سری زمانی می‌باشند و رکورد اتصال فعلی می‌تواند بر اساس رکوردهای اتصال قبلی دسته‌بندی شود. برای یافتن الگوهای سری زمانی در طول زمان، ویژگی‌های بدست آمده از عملیات max-pooling در CNN، آن‌ها را به یک شبکه عصبی بازگشتی می‌دهیم. به صورت زیر:

$$FM = CNN(x_t)$$

در حالی که CNN از یک لایه کانولوشن یک بعدی و یک لایه Max-pooling یک بعدی تشکیل شده‌اند، x_t به معنای بردار ویژگی ورودی همراه برچسب کلاس است. بنابراین بردار ویژگی نگاشت جدید یعنی FM به RNN داده می‌شود تا روابط درازمدت زمانی را یاد بگیرد.

شبکه‌های عصبی بازگشتی شبیه به MLP[5] به اضافه یک حلقه اضافه. این حلقه اطلاعات قبلی را در طول مراحل زمان انتقال می‌دهند.

$$h_t = f(w_{FMh}FM_t + w_{hh}h_{t-1} + b_h)$$

$$o_t = w_{h0}hl + b_0$$

که f به معنای تابع فعالیت غیرخطی می‌باشد، w به معنای وزن‌ها، b به معنای اندازه بایاس و FM بخ معنای بردار ویژگی‌های هستند که توسط CNN محاسبه شده‌اند.

[4] Recurrent neural network
[5] Multi Layer Perceptron

مجموعه داده نفوذ شبکه(NIDS[6])

مجموعه داده استفاده شده در شبکه محلی پایگاه نیروی هوایی، در آزمایشگاه لینکولن MIT در سال ۱۹۹۸ توسط گروه ارزیابی تشخیص نفوذ DARPA جمع‌آوری شده است. مجموعه داده آموزشی شامل ۲۴ نوع حمله که در چهار گروه دسته‌بندی شده‌اند و مجموعه داده تست شامل ۱۴ حمله اضافه بر ۲۴ حمله می‌باشد و همگی در چهار گروه دسته‌بندی می‌شوند. ویژگی‌ها شامل اطلاعات TCP/IP که از هر دو جریان فرستنده و گیرنده با نوع پروتکل مشخص در یک بازه زمانی مشخص جمع‌آوری شده‌اند، می‌باشد. هر جریان ترافیک شامل ۱۰۰ بایت اطلاعات می‌باشد. دارای ۴۱ ویژگی که ۳۴ تای آن‌ها پیوسته و ۷ ویژگی به صورت مقادیر گسسته می باشد و به صورت زیر گروه‌بندی شده‌اند: بازه [۹-۱] ویژگی‌های اساسی، ویژگی‌های محتوایی در بازه [۲۲-۱۰]، ویژگی‌های مربوط به ترافیک در یک بازه زمانی در بازه [۳۱-۲۳] و ویژگی‌های مرتبط با هاست در بازه[۴۱-۳۲] می‌باشد.

مجموعه‌داده NSL-KDD[7] نسخه اصلاح شده KDDCup99[8] می‌باشد که جزییات KDDCup99 و NSL-KDD در جدول زیر آمده‌است:

Attack category	Full data set	10 % data set			
	KDDCup 99	KDDCup 99		NSL-KDD	
	Train	Train	Test	Train	Test
Normal	972780	97278	60593	67343	9710
DOS	3883370	391458	229853	45927	7458
Probe	41102	4107	4166	11656	2422
r2l	1126	1126	16189	995	2887
u2r	52	52	228	52	67
Total		494021	311029	125973	22544

شکل ۱: جزییات ۱۰ درصد از مجموعه داده‌های NSL-KDD و KDDCup99

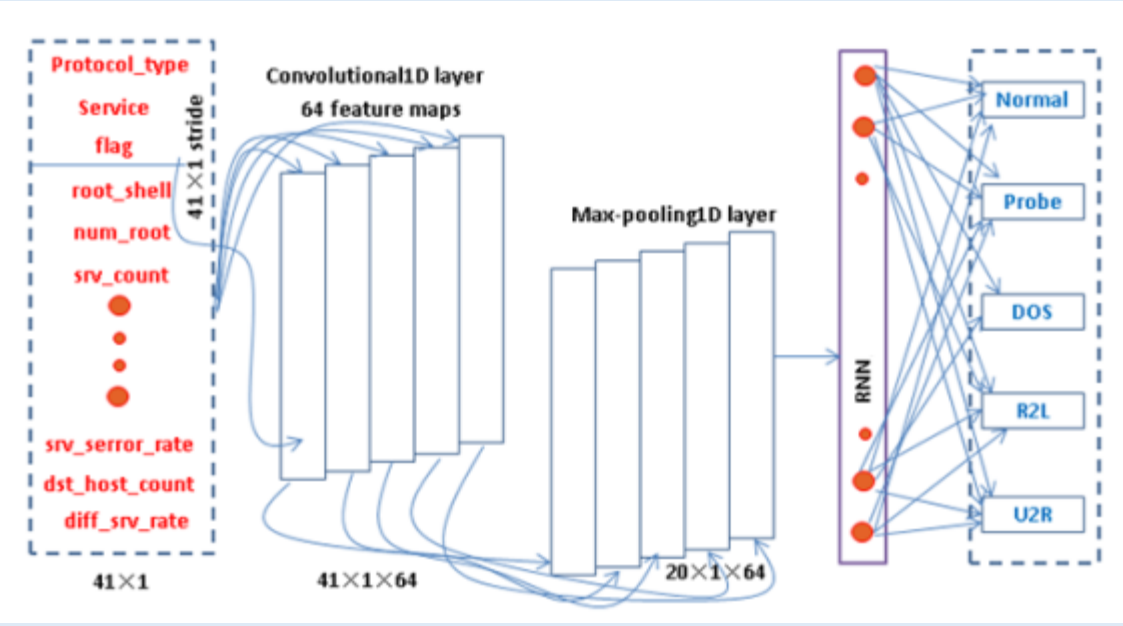
[6] Network Intrusion data set
[7] http://www.unb.ca/cic/datasets/nsl.html
[8] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

مراحل پیاده‌سازی

شبکه با با تعداد ۳۲ و ۶۴ فیلتر با اندازه‌های ۳ و ۵ تست شد که در حالت ۶۴ فیلتر با طول ۵ بیشترین دقت بدست آمد. نرخ‌های یادگیری پایین‌تر در شبکه‌های CNN کارآیی بهتری در تشخیص نوع اتصالات به همراه داشتند. البته نرخ‌های یادگیری پایین برای رسیدن به نرخ تشخیص مورد قبول در حملات با فرکانس کمتر تعداد epoch بیشتری را نیاز داشتند. با توجه به زمان فرایند آموزش، هزینه و کارآیی تشخیص، نرخ یادگیری ۰,۱ به ثابت برای یادگیری شبکه انتخاب شد.

شبکه CNN عادی کارایی خوبی در تشخیص حملات پر تکرار مانند "Normal"، "dos" و "probe" دارند. هم‌چنین شبکه هرچه ساختار پیچیده‌تری داشته باشد، تعداد epochهای بیشتری نیاز است، تا به نرخ تشخیص قابل توجهی برای حملات کم‌تکرار برسند(شاید بیشتر از ۵۰۰ epoch). تعداد epochهای لازم برای یادگیری حملات مختلف در هر نوع ساختار شبکه، متفاوت است. به علاوه، شبکه‌ها شروع به بیش‌برازش[9]حملات پرتکرار می‌کنند، وقتی الگوی آن‌ها را کامل یاد می‌گیرند. به این معناکه از یک نقطه به بعد شبکه فقط نمونه‌های آموزش را به خاطر می‌سپارد که باعث کاهش کارایی عمومیت[10]برای حملات خاص می‌شود. همه آزمایش‌های شبکه تا ۳۰۰ epoch اجرا شده‌اند. معماری کلی شبکه استفاده شده در شکل۲ دیده می‌شود که دارای یک لایه ورودی، لایه مخفی و یک لایه خروجی می‌باشد. لایه مخفی شامل یک یا چند لایه CNN می‌باشد و پس از آن نیز یک لایه FCN یا RNN می‌باشد. لایه ورودی داده را در قالب 41*1 دریافت کرده و به CNN می‌دهد. CNN یک تنسور 64*1*41 می‌سازد(۶۴ تعداد فیلترها می‌باشد) و آن را به لایه max-pooling می‌دهد که در این لایه شکل تنسور به 64*1*20 کاهش می‌یابد. حال این مرحله خروجی را می‌توان یا به یک FCN داد تا کار طبقه‌بندی انجام شود یا به یک RNN تا الگوی‌های زمانی را استخراج کند.

[9] Over Fitting
[10] Generalization Performance



شکل ۲معماری شکبه ترکیبی

نتایج ارزیابی

با توجه به جدول زیر چهار معیار زیر برای سنجش در نظر گرفته می‌شود:

جدول ۱ ماتریس اختلال (Confusion Matrix)

		Predicted class	
		Class = Yes	Class = No
Actual Class	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

- Accuracy: نرخ تعداد نمونه هایی که به درستی پیش بینی شده است به نسبت همه نمونه های مشاهده شده. Accuracy = TP+TN/TP+FP+FN+TN
- Precision: نرخ تعداد نمونه های مثبت که به درستی پیش بینی شده است به نسبت کل نمونه های مثبت. Precision = TP/TP+FP

- Recall: نرخ تعداد نمونه های مثبت که به درستی پیش بینی شده به نسبت کل نمونه های در یک کلاس.

$$Recall = TP/TP+FN$$

- F1-score: میانگین وزن دار precision و recall می باشد که هم FP و هم FN را در نظر می گیرد.

$$F1\ Score = 2*(Recall * Precision) / (Recall + Precision)$$

همان طور که قبلا اشاره شد،آزمایشات در دو حالت شبکه CNN و شبکه ترکیبی CNN+RNN انجام شد که نتایج آن بر اساس چهار معیار فوق در جدول۱ آورده شده است.

در این مرحله تنها دسته‌بندی دوکلاسه(باینری) روی مجموعه داده KDDCup99 انجام شد، در شبکه ترکیبی به جهت بیش‌برازش کارایی‌اش در حد شبکه‌های عادی پایین می‌آید.

جدول ۲ خلاصه نتایج تست بر روی مجموعه داده KDDCup99 در طبقه‌بندی رکورد های اتصال به صورت حمله یا نرمال

Algorithm	Accuracy	Precision	Recall	F-score
CNN 1 Layer	0.95	0.95	0.95	0.95
CNN 1 Layer	0.948	0.95	0.94	0.95
CNN 1 Layer-RNN	0.821	0.95	0.778	0.875
CNN 2 Layer-RNN	0.93	0.97	0.94	0.92

منابع

[1] *Roberto Jordaney, Kumar Sharad , Santanu Kumar Dash, Zhi Wang , Davide Papini ,Ilia Nouretdinov, and Lorenzo Cavallaro," Transcend: Detecting Concept Drift in Malware Classification Models", 26th USENIX Security Symposium*

[2] *N. Gao, L. Gao, Q. Gao, and H. Wang, "An intrusion detection model based on deep belief networks," in* Advanced Cloud and Big Data (CBD), 2014 Second International Conference on. *IEEE, 2014, pp. 247–252.*

[3] *Vinayakumar R, Soman KP and Prabaharan Poornachandran," Evaluating Effectiveness Of Shallow and Deep Networks to IDS-2017 Network Intrusion Detection", 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*

ضمائم

همراه این گزارش موارد زیر ضمیمه شده است:

- کدهای پیاده سازی
- بخشی از مجموعه داده *KDDCup99* که برای آموزش شبکه‌ها مورد استفاده قرار گرفته است.