ششمین همایش بین‌المللی افق های نویـن در
مهندسی برق، کامپیوتـر و مکانیک
6th International Conference on the New Horizons in
Electrical Engineering, Computer and Mechanical
www.mhconf.ir

بهبود تشخیص حملات در شبکه با استفاده از شبکه‌های عصبی عمیق و ترکیب تکنیک‌های بیش نمونه‌گیری و کم نمونه‌گیری

مهدی اکبری کوپایی (نویسنده مسئول) ، فاطمه کریمی

موسسه آموزش عالی جهاددانشگاهی اصفهان ، اصفهان f.karimi.555@gmail.com

دانشگاه آزاد اسلامی واحد نجف آباد ، نجف آباد mehdi_akbari@hotmail.com

چکیده

امروزه بدلیل افزایش میزان حملات و نفوذ به شبکه‌ها، وجود سیستم‌های تشخیص نفوذ، بیش از پیش ضروری به نظر می‌رسد. یک سیستم تشخیص نفوذ با نظارت بر ترافیک شبکه، فعالیت‌های مشکوک را شناخته و هشدار می‌دهد. اهمیت بالای این سیستم‌ها در تشخیص نفوذ باعث گردیده که در سال‌های اخیر مطالعات مختلفی در این زمینه ارائه شود. اغلب مطالعات ارائه شده در این زمینه برای ارائه یک سیستم تشخیص نفوذ از ترکیب روش‌های خوشه‌بندی و طبقه‌بندی استفاده می‌کنند. اما غالبا این روش‌ها دارای نقاط ضعف مشترکی هستند، تعیین تعداد خوشه مناسب، تعیین نقاط بهینه اولیه مراکز از جمله محدودیت‌های است که نتایج آن‌ها را تحت تاثیر قرار می‌دهد. از طرف دیگر مشکل عدم توازن در داده‌های حملات و وجود حملاتی با حداقل داده ممکن باعث می‌شود روش‌های یادگیری ماشین در مواجه با این نوع حملات با مشکل مواجه باشند و قادر به تشخیص آن‌ها نباشند. از اینرو در این تحقیق هدف ارائه یک سیستم تشخیص نفوذ در راستای رفع دو مشکل فوق است. در این روش از یادگیری عمیق و شبکه‌های عصبی عمیق بازگشتی LSTM برای تشخیص نفوذ استفاده می‌شود. از طرف دیگر برای رفع مشکل عدم توازن داده‌ها از ترکیب تکنیک‌های بیش نمونه‌گیری و کم‌نمونه‌گیری استفاده می‌شود. در این روش برای ۳ نوع از حملات نادر که دارای تعداد نمونه‌های کمی هستند، داده‌های مصنوعی با استفاده از الگوریتم SMOTE از داده‌های حقیقی حمله، تولید می‌شود و به تعداد افزوده شده به داده‌های کلاس‌های اقلیت، از تعداد داده‌های کلاس اکثریت به صورت تصادفی حملاتی حذف می‌شود. بررسی روش پیشنهادی بر روی سه پایگاه حملات NSL-KDD، UNSW-NB15_4 و CICIDS-2017 نشان می‌دهد که راهکار مطرح شده توانسته است به صحت بالای ۹۷٪ در تشخیص حملات دست یابد که در مقایسه با روش پیشین بالغ بر ۳٪ تشخیص نفوذ را بهبود داده است.

واژه‌های کلیدی

شبکه‌های عصبی عمیق بازگشتی، تشخیص نفوذ، بیش نمونه‌گیری، کم نمونه‌گیری.

۱. متن مقاله

افزایش وابستگی فعالیت‌های تجاری، مالی، پزشکی و غیره به سیستم‌ها و سرویس‌های کامپیوتری منجر به افزایش لزوم حفظ امنیت شبکه‌های کامپیوتری در مقابل نفوذها شده است. کاربرد وسیع سیستم‌های کامپیوتری مسئولیت ناظران این امکانات را افزایش داده است؛ به این دلیل که کوچکترین خطایی در اطلاعات موجود می‌تواند خسارت جبران ناپذیری را به بار آورد. تاثیرات مخرب فرآیندهای نفوذی باعث شده، طراحی یک سیستم تشخیص نفوذ موفق و کارآمد جهت تشخیص حملات مخرب و دسترسی‌های غیر مجاز به یکی از پر اهمیت‌ترین مباحث مطرح در امنیت شبکه تبدیل شود. سیستم‌های تشخیص نفوذ یا به اختصار IDS[1] سیستم های نرم افزاری یا سخت افزاری هستند که فرآیند نظارت بر وقایع رخ داده در یک سیستم رایانه‌ای یا شبکه را به صورت خودکار انجام می دهند و آن‌ها را از نظر وجود علائم امنیتی تجزیه و تحلیل می‌کنند. با افزایش تعداد و شدت حملات شبکه طی چند سال گذشته، سیستم‌های تشخیص نفوذ به یکی از موارد ضروری در زیرساخت‌های امنیتی اکثر سازمان‌ها تبدیل شده‌اند (Aldweesh, Derhab, & Emam, 2020). یک سیستم تشخیص نفوذ با نظارت بر شبکه‌های کامپیوتری، فعالیت‌های مخرب و یا نقض سیاست‌های مدیریتی و امنیتی را بررسی کرده و به مدیر شبکه گزارش می‌دهد. این سیستم‌ها سه وظیفه اصلی نظارت و ارزیابی، تشخیص نفوذ و پاسخ را بر عهده دارند و پاسخ در آن‌ها به تولید اخطار محدود می‌شود (Al-Hadhrami & Hussain, 2020). در سال‌های اخیر به دلیل اهمیت این حوزه مطالعات مختلفی جهت ارائه سیستم‌های تشخیص نفوذ انجام شده است. کلیه روش‌های ارائه شده تا کنون مبتنی بر انواع مختلف روش‌های داده‌کاوی از جمله طبقه‌بندی و خوشه‌بندی و دیگر روش‌های یادگیری ماشین هستند. اما مطالعات نوین نشان می‌دهد که استفاده از تکنیک‌های یادگیری عمیق یا به اختصار DL[2] برتری خود را در زمینه‌های کلان داده و امنیت سایبری نشان داده است (Yin, Zhu, Fei, & He, 2017). تکنیک‌های یادگیری عمیق قابلیت استخراج و یادگیری ویژگی‌های عمیق از حملات شناخته شده و شناسایی حملات ناشناخته را بدون نیاز به استخراج ویژگی دستی دارند، که باعث محبوبیت آن‌ها شده است (Hassan, Gumaei, Alsanad, Alrubaian, & Fortino, 2020).

با توجه به مشکل داده‌های نامتوازن در داده‌های حملات شبکه، هدف در این تحقیق ارائه یک روش تشخیص حملات شبکه مبتنی بر شبکه‌های عصبی عمیق بازگشتی است که در آن برای رفع چالش داده‌های نامتوازن از ترکیب تکنیک‌های بیش نمونه‌گیری و کم نمونه‌گیری استفاده می‌شود. طبق مطالعات انجام شده تاکنون روشی از ترکیب روش‌های کم نمونه گیری و بیش نمونه گیری و شبکه های عصبی بازگشتی برای بهبود نرخ تشخیص در کلاس‌های اقلیت استفاده نکرده است.

ادامه این پایان نامه به صورت زیر سازماندهی شده است:

---

[1] Intrusion Detection Systems
[2] Deep Learning

فصل دوم؛ ادبیات تحقیق و پیشینه تحقیق: شامل دو زیر بخش اصلی است، در زیر بخش ادبیات تحقیق روش‌های تشخیص نفوذ و داده‌کاوی و روش‌های طبقه‌بندی و یادگیری عمیق معرفی می‌شوند. در زیر بخش دوم و یا پیشینه تحقیق تعدادی از الگوریتم‌های تشخیص نفوذ و مطالعاتی که تاکنون در این زمینه انجام شده است ارائه می‌شود.

فصل سوم؛ راهکار پیشنهادی: در این فصل الگوریتم پیشنهادی به طور کامل معرفی می‌گردد و فلوچارت و آن ارائه خواهد شد و بخش های مختلف آن به صورت کامل شرح داده می‌شود.

فصل چهارم؛ نتایج و ارزیابی الگوریتم: در این فصل با هدف بررسی عملکرد راهکار پیشنهاد شده و مقایسه آن با روش پایه، تعداد مختلفی از آزمون‌ها بر روی سه پایگاه داده جمع‌آوری شده انجام شده است که نتایج تغییرات در متغیرهای مستقل بر روی متغیرهای وابسته به صورت نمودار نمایش داده شده و در رابطه با علت حصول نتایج نیز بحث می‌گردد. در این فصل پس از ارائه نتایج، اهداف تحقیق نیز بررسی می‌شود.

فصل پنجم؛ بحث و نتیجه گیری: در این فصل نتیجه کلی از تحقیق و پیشنهاداتی برای مطالعات آتی ارائه می‌شود.

۲. مروری بر کارهای گذشته

در این بخش قصد بر آن است تا چند نمونه از روش‌های تشخیص نفوذ که بین سال‌های ۲۰۱۵ تا ۲۰۲۰ ارائه شده‌اند معرفی شوند. عملکرد کلی روش‌ها به همراه داده‌های مورد استفاده در هر روش و نتایج حاصل شده توسط آن‌ها در جدول (۲-۱) به صورت کلی نمایش داده شده است.

شنفیلد و همکاران[3] در سال ۲۰۱۸یک روش تشخیص نفوذ مبتنی بر شبکه‌های عصبی ارائه دادند. در این روش از یک شبکه عصبی با دو لایه پنهان استفاده شده است و شبکه فقط قادر است دو کلاس را طبقه‌بندی کند و نوع حملات در این مقاله بررسی نشده است. نتایج این روش بر روی پایگاه داده جمع‌آوری شده به صحت ۹۸٪ رسیده‌اند (Shenfield, Day, & Ayesh, 2018).

کبیر و همکاران[4] در سال ۲۰۱۸ یک روش تشخیص نفوذ جدید مبتنی بر ماشین بردار پشتیبان حداقل مربعات با نام LS-SVM[5] ارائه دادند. در این روش تشخیص نفوذ در دو مرحله انجام می‌شود. در مرحله اول، کل مجموعه داده به تعدادی زیر گروه از پیش تعیین شده تقسیم می‌شود، سپس الگوریتم، نمونه‌های نماینده را از این زیرگروه‌ها انتخاب می‌کند به گونه‌ای

[3] Shenfield et al.
[4] Kabir et al.
[5] Least Square Support Vector Machine

که نمونه‌ها کل مجموعه داده را منعکس می‌کنند. در مرحله دوم، ماشین بردار پشتیبان حداقل مربعات برای تشخیص نفوذ به نمونه‌های استخراج شده اعمال می‌شود. برای نشان دادن اثربخشی روش پیشنهادی، آزمایشات بر روی پایگاه داده KDD 99 انجام شده است که بیانگر صحت ۹۹,۷۸٪ است (Kabir, Hu, Wang, & Zhuo, 2018).

مارمول و همکاران[۶] در سال ۲۰۱۸ یک روش ترکیبی مبتنی بر الگوریتم ژنتیک و درخت تصمیم برای تشخیص نفوذ استفاده کردند. روش آن‌ها با نام Dendron با تولید قوانین جدید شناسایی قادر به طبقه بندی انواع حملات معمول و نادر است. این روش همچنین با هدف مقابله با ماهیت چالش انگیز ترافیک شبکه، روش‌های فراابتکاری را بکار می‌برد. نتایج تجربی، با استفاده از مجموعه داده‌های KDDCup99، NSL-KDD و UNSW-NB15، نشان می‌دهد که این روش قادر است با چندین معیار طبقه بندی، برتری بیشتری نسبت به سایر تکنیک های پیشرفته و قدیمی داشته باشد، این روش در بهترین حالت به صحت ۹۰,۵۴٪ رسیده است (Papamartzivanos, Mármol, & Kambourakis, 2018).

کاراتاس و همکاران[۷] در سال ۲۰۱۸ برای تشخیص نفوذ از شبکه عصبی با چندین تابع یادگیری مختلف استفاده نمودند. هدف در این تحقیق مقایسه توابع یادگیری شبکه عصبی از جمله Trainlm، Trainc، Trainbfg، Trainscg، Traincgp، Trainoss و غیره یا یکدیگر بود. شبکه عصبی استفاده در این مقاله دارای دو لایه مخفی با ۱۰ نرون است و لایه طبقه‌بندی و یا لایه خروجی دارای ۵ نرون است که داده‌ها را به ۵ حمله مختلف طبقه‌بندی می‌کند نتایج این روش بر روی پایگاه داده KDD Cup'99 که بهترین زمان اجرا متعلق به تابع trainscg است و کمترین نرخ خطا برابر با ۲,۵۸ متعلق به تابع یادگیری trainlm است (Karatas & Sahingoz, 2018).

احمد و همکاران[۸] در سال ۲۰۱۸برای ارائه یک سیستم تشخیص نفوذ از سه الگوریتم یادگیری از جمله ماشین بردار پشتیبان، جنگل تصادفی[۹] و ماشین یادگیری افراطی[۱۰] استفاده کردند و نتایج این سه الگوریتم را با یکدیگر مقایسه نمودند. نتایج این روش بر روی پایگاه داده NSL–KDD حاکی از آن است که الگوریتم ماشین یادگیری افراطی با صحت ۹۹,۵٪ در مقایسه با ماشین بردار پشتیبان و جنگل تصادفی عملکرد موفق‌تری در تشخیص نفوذ داشته است (I. Ahmad, Basheri, Iqbal, & Rahim, 2018).

---

[6] Marmol et al.
[7] Karatas et al.
[8] Ahmad et al.
[9] Random Forest
[10] Extreme Learning Machine

مارینو و همکاران[11] در سال ۲۰۱۸ از تکنیک ماشین یادگیری خصمانه[12] برای تشخیص نفوذ استفاده کرده‌اند. آن‌ها یک روش برای تشریح طبقه‌بندی‌های نادرست توسط رویکرد خصمانه ارائه دادند. اگرچه معمولاً از یادگیری ماشین خصمانه برای طبقه‌بندی استفاده می‌شود، اما در این مقاله با استفاده از یافتن حداقل تغییرات لازم برای طبقه‌بندی صحیح نمونه‌های طبقه‌بندی نشده، توضیحاتی ایجاد می‌شود. تفاوت بین نمونه های اصلی و اصلاح شده اطلاعات مربوط به ویژگی‌های مربوط به طبقه بندی غلط را فراهم می‌کند از داده‌های اصلاح شده برای تشخیص نفوذ با استفاده از الگوریتم شبکه عصبی چند لایه پرسپترون[13] استفاده می‌شود. نتایج آن‌ها بر روی دو پایگاه داده NSL-KDD و KDD99 نشان می‌دهد در بهترین حالت به صحت ۹۵,۵٪ رسیدند (Marino, Wickramasinghe, & Manic, 2018).

سحانی و همکاران[14] در سال ۲۰۱۸ از الگوریتم درخت تصمیم C4.5 برای تشخیص نفوذ استفاده کردند. این روش از دو مرحله تشکیل شده است در مرحله اول حملات به دو دسته نرمال و حمله طبقه‌بندی می‌شوند و در مرحله دوم فقط داده‌های حمله به یک ساختار لایه‌ای ارائه می‌شوند که در هر لایه یک نوع از حملات تشخیص داده می‌شود. نتایج این روش بر روی پایگاه داده KDD99 نشان می‌دهد که صحت ۹۹,۷۹٪ حاصل شده است (Sahani, Rout, Badajena, Jena, & Das, 2018).

الجوارن و همکاران[15] در سال ۲۰۱۹ از الگوریتم بهبود یافته درخت تصمیم J48 برای تشخیص نفوذ استفاده کردند. در درخت‌های تصمیم برای تفکیک گره‌ها و تشکیل شاخه‌های فرعی از معیار gain استفاده می‌شود اما در این مقاله برای بهبود درخت تصمیم J48 از تخمین IG و GR برای ساخت درخت تصمیم‌گیری استفاده شده است. در این روش ویژگی‌های دارای حداکثر IG نرمال شده مورد استفاده قرار می‌گیرد و الگوریتم با استفاده از زیر مجموعه‌های کوچک تکرار می‌شود. این روش بر روی پایگاه داده NSL KDD در مقایسه با روش‌های جنگل تصادفی و بیز ساده به صحت ۹۰٪ رسیده است (Aljawarneh, Yassein, & Aljundi, 2019).

روات و همکاران[16] در سال ۲۰۱۹ دو دسته روش‌های یادگیری ماشین سنتی و روش‌های عصبی را با یکدیگر مقایسه کردند. در این روش ابتدا داده‌ها توسط الگوریتم تحلیل مؤلفه اصلی[17] به ۱۵ بعد، کاهش داده شده و سپس از الگوریتم شبکه

---

11 Marino et al.
12 adversarial machine learning
13 Multi-Layer Perceptron (MLP)
14 Sahani et al.
15 Aljawarneh et al.
16 Rawat et al.
17 Principal Component analysis (PCA)

عصبی برای تشخیص نفوذ در داده‌های کاهش بعد داده شده استفاده شده است. نتایج این روش بر روی پایگاه داده NSL-KDD نشان می‌دهد که صحت برابر با ۷۹,۳٪ حاصل شده است (Rawat, Srinivasan, Vinayakumar, & Ghosh, 2019).

ژی و همکاران[18] در سال ۲۰۱۹ برای ارائه یک سیستم تشخیص نفوذ موفق از ترکیب شبکه عصبی پیشرو[19] و الگوریتم فراابتکاری بهینه سازی ازدحام ملخ [20] استفاده نمودند. در این روش از الگوریتم فراابتکاری ملخ برای بهینه‌سازی مرحله یادگیری شبکه عصبی پیشرو استفاده شده است. روش آن‌ها بر روی پایگاه داده UNSW-NB15 به صحت ۹۹,۳۳٪ و بر روی پایگاه داده NSL-KDD به صحت ۸۹,۸۳٪ رسیده است (Benmessahel, Xie, Chellal, & Semong, 2019).

جو و همکاران[21] در سال ۲۰۱۹ از روش‌های گروهی برای تشخیص نفوذ استفاده کرده‌اند. در این روش الگوریتم ماشین بردار پشتیبان به صورت گروهی برای تشخیص نفوذ برروی خوشه‌های تشکیل شده توسط الگوریتم خوشه‌بندی فازی Cmeans استفاده می‌شود و طبقه‌بندی فقط در دو حالت نرمال و غیر نرمال انجام می‌شود و از انواع مختلف حمله صرف نظر شده است. این روش بر روی پایگاه داده NSL-KDD و KDD'99 به صحت ۹۹,۳۶٪ رسیده است (Gu, Wang, Wang, & Wang, 2019).

احمد و همکاران در سال ۲۰۱۹ از سه الگوریتم طبقه‌بندی ماشین بردار پشتیبان، k نزدیکترین همسایه و بیز ساده[22] برای تشخیص نفوذ استفاده کرده‌اند. در این روش ابتدا زیر مجموعه‌ای بهینه از داده‌ها با استفاده از الگوریتم فراابتکاری ازدحام ذرات انتخاب می‌شوند پس از مرحله انتخاب ویژگی از سه الگوریتم مذکور برای طبقه‌بندی بهترین ویژگی‌های انتخاب شده استفاده می‌شود . نتایج این مقاله بر روی پایگاه داده KDDCup99 نشان می‌دهد که الگوریتم ماشین بردار پشتیبان با صحت ۹۹,۹۲٪ بهترین نتایج را ارائه داده است (Ahmad,T & Aziz, 2019).

چویی و همکاران[23] در سال ۲۰۱۹ در این تحقیق با استفاده از الگوریتم یادگیری بدون نظارت، یک سیستم تشخیص نفوذ شبکه ایجاد و عملکرد آن مورد بررسی قرار گرفته است. در این روش از معماری عمیق خودرمزگذارها[24] که در دسته

---

18 Xie et al.
19 Feed - Forward Neural Network (FNN)
20 locust swarm optimization
21 Gu et al.
22 Naive Bayes
23 Choi et al.
24 autoencoders

روش‌های بدون ناظر قرار می‌گیرند برای تشخیص نفوذ استفاده شده است. این روش بر روی پایگاه داده NSL-KDD به صحت ۹۱,۷۰٪ رسیده است (Choi, Kim, Lee, & Kim, 2019).

عبدالمحمد و همکاران[۲۵] در سال ۲۰۱۹ از دو روش کاهش بعد و چندین الگوریتم طبقه‌بندی برای ارائه یک سیستم تشخیص نفوذ استفاده کردند. در این روش از معماری خود رمزگذارها و تحلیل مؤلفه اصلی برای کاهش ابعاد داده استفاده کردند و برای طبقه‌بندی ویژگی‌های منتخب از الگوریتم‌های جنگل تصادفی، شبکه بیزین[۲۶]، تحلیل تفکیک خطی[۲۷] و آنالیز افتراقی درجه دو[۲۸] استفاده کردند. نتایج آن‌ها در پایگاه داده CICIDS2017 نشان می‌دهد که بهترین صحت برابر با ۹۹,۶٪ است که متعلق به ترکیب الگوریتم جنگل تصادفی و روش کاهش بعد تحلیل مؤلفه اصلی است (Abdulhammed, Musafer, Alessa, Faezipour, & Abuzneid, 2019).

لانگ و همکاران[۲۹] در سال ۲۰۱۹ از ترکیب مدل مخلوط گاوسی و الگوریتم طبقه‌بندی k نزدیکترین همسایه برای طبقه‌بندی و تشخیص حملات استفاده کرده‌اند. این روش اگر چه دارای عملکرد نسبتا مناسبی است اما بدلیل عدم رفع مشکل داده‌های نامتوازن یا نامتعادل قادر به پیشگویی موفق در حملات نادر نیست. این روش بر روی پایگاه داده NSL-KDD به صحت ۹۹,۳۱٪ رسیده است (Long et al., 2019).

کیم و همکاران[۳۰] در سال ۲۰۲۰ یک روش تشخیص نفوذ مبتنی بر الگوریتم‌های ژنتیک و خوشه‌بندی فازی Cmeans برای کاهش بعد و انتخاب ویژگی و شبکه‌های عصبی عمیق کانولوشن ارائه کرده‌اند. در این روش برای طبقه‌بندی ویژگی‌های انتخاب شده توسط الگوریتم‌های فوق از روش‌های طبقه‌بندی k نزدیکترین همسایه، جنگل تصادفی استفاده شده است. نتایج بر روی پایگاه داده NSL-KDD نشان می‌دهد که صحت برابر با ۹۸,۲۴٪ بدست آمده است (Nguyen & Kim, 2020).

حسن و همکاران[۳۱] در سال ۲۰۲۰ یک مدل یادگیری عمیق ترکیبی را برای شناسایی موثر نفوذها براساس یک شبکه عصبی کانولوشن و یک شبکه حافظه بلند مدت کوتاه[۳۲] ارائه می‌دهد. در این روش از شبکه کانولوشن برای استخراج ویژگی‌های معنی‌دار استفاده می‌شود. این روش بر روی پایگاه داده UNSW-NB15 آزمایش شده است و نتایج آن نشان

---

25 Abdulhammed et al.
26 Bayesian Network
27 Linear Discriminant Analysis (LDA)
28 Quadratic Discriminant Analysis (QDA)
29 Long et al.
30 Kim et al.
31 Hassan et al.
32 Weight - Dropped, Long Short - Term Memory

می‌دهد که در حالت دو کلاسه صحت برابر با ۹۷,۱۷٪ و در حالات چندکلاسه صحت برابر با ۹۸,۴۳٪ شده است (Hassan et al., 2020).

کلوری و همکاران[۳۳] در سال ۲۰۲۰یک روش تشخیص نفوذ ارائه نموده‌اند که در ان برای کاهش ابعاد از ترکیب الگوریتم تحلیل مؤلفه اصلی و الگوریتم فراابتکاری کرم شب تاب[۳۴] استفاده شده است. در ادامه برای طبقه‌بندی داده‌های جدید کاهش بعد یافته از الگوریتم طبقه‌بندی XGBoost استفاده شده است. این روش بر روی پایگاه داده Kaggle به صحت ۹۹,۹٪ رسیده است (Bhattacharya, Kaluri, Singh, Alazab, & Tariq, 2020).

۳. روش پیشنهادی

مبحث امنیت در زمینه‌های گوناگونی از زندگی بشر امروز در رأس توجه قرار دارد. کامپیوتر و به خصوص شبکه‌های کامپیوتری نیز از این قضیه مستثنی نیستند. با توسعه سریع شبکه‌های کامپیوتری و فراگیر شدن برنامه‌های مختلف، امنیت در شبکه‌ها بیش از پیش مورد توجه قرار گرفته است. سیستم‌های تشخیص نفوذ روش‌های بسیار حیاتی هستند که در زمینه حفظ امنیت شبکه‌ها به جز لاینفک آن‌ها تبدیل شده‌اند؛ که هدف آن‌ها محافظت از میزبان در برابر حملات مختلف است. در سال‌های اخیر مطالعات غنی برای بهبود عملکرد سیستم‌های تشخیص نفوذ انجام شده است، اما هنوز محدودیت‌های زیادی در این بین مطرح می‌باشد. اگر چه روش‌های بسیاری تاکنون در این زمینه ارائه شده است اما همچنان نیاز به ارائه سیستم‌هایی با قابلیت بالاتر و توانایی مواجه با داده‌های بزرگ به شدت مطرح می‌باشد.

۱,۳ وجه تمایزهای راهکار ارائه شده و روش پایه

روش ارائه شده در این تحقیق و روش پایه لانگ و همکاران ارائه شده در سال ۲۰۱۹ هر دو روش‌های تشخیص نفوذ با ناظر می‌باشند. به این معنی که در هر دو روش داده‌های حملات باید دارای کلاس از پیش مشخص شده باشند. وجه تمایز دو روش فوق در موارد زیر خلاصه می‌شود:

۱) روش پیشنهادی بر خلاف روش لانگ از تکنیک یادگیری عمیق و شبکه‌های عصبی عمیق بازگشتی برای تشخیص نفوذ استفاده می‌کند. در صورتی که روش لانگ از الگوریتم k نزدیکترین همسایه استفاده می‌کند که نتایج آن بسیار وابسته به مقدار k است.

---

[33] Kaluri et al.
[34] firefly

۲) روش ارائه شده در این تحقیق مشکل عدم توازن داده‌ها را تا حدودی بر طرف ساخته است. در این روش با ترکیب دو تکنیک بیش نمونه‌گیری و کم نمونه‌گیری تعداد حملات کلاس‌های اقلیت افزوده شده است و در مقابل تعداد کلاس اکثریت نیز کاهش یافته است. در صورتی که در روش لانگ مشکل عدم توازن داده‌ها برطرف نشده است.
۳) راهکار مطرح شده مرحله ارزیابی را برای هر دو حالت دو کلاسه و چند کلاسه انجام داده است در صورتی که روش لانگ و همکاران فقط بروز حمله را تشخیص داده‌اند و از تشخیص نوع حملات صرف نظر کرده‌اند.

۲,۳ *نیازمندی‌های راهکار پیشنهادی*

نیازمندی‌های راه‌کار پیشنهادی شامل ورودی‌های روش می‌باشد. که در ادامه به شرح آن پرداخته می‌شود.

ورودی‌های روش به شرح زیر است:

۱) پایگاه داده‌ها: اولین و اصلی‌ترین ورودی روش، پایگاه داده حملات و نفوذ در شبکه می‌باشد. در این روش از سه پایگاه داده مختلف برای بررسی بهتر مدل پیشنهاد شده استفاده شده است. در شکل (۳-۱) بخشی از پایگاه داده NSL-KDD نشان داده شده است. این ورودی به صورت تصادفی با توجه به نرخ مشخصی به دو بخش آموزش و آزمایش تقسیم می‌شود و از یک بخش برای ارزیابی و از بخش دیگر برای آموزش شبکه عصبی استفاده می‌شود.

۲) نرخ بخش‌بندی داده‌ها: دومین ورودی اصلی راهکار مطرح شده، نرخ تقسیم‌بندی داده‌ها به دو بخش آموزش و آزمایش است. این نرخ مقداری بین ۰,۱ تا ۰,۹ می‌گیرد.

۳) پارامترهای بیش نمونه‌گیری: تعداد همسایگان در روش بش نمونه‌گیری از دیگر پارامترهای ورودی روش است که مقدار آن تا حدودی می‌تواند در نتایج مدل موثر باشد.

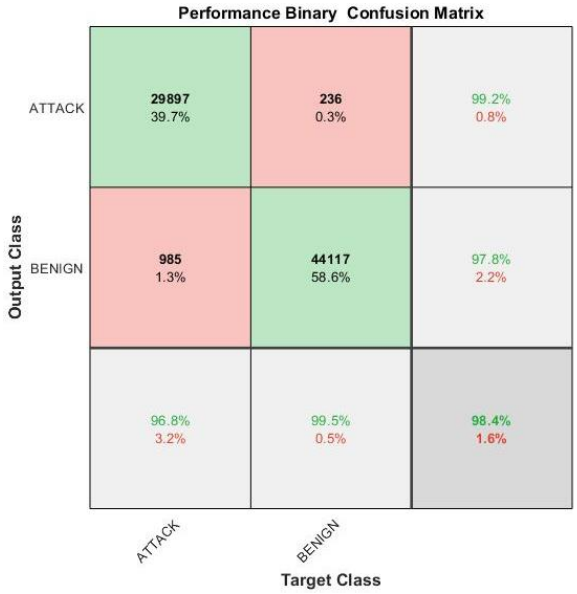
۴) پارامترهای شبکه عصبی بازگشتی: پارامترهایی مثل تعداد Epoch، حداقل سایز داده در هر گردش Mini batch size و تابع بهینه ساز سه پارامتر در شبکه‌های عصبی هستند.

```
0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,
0.03,0.17,0.00,0.00,0.00,0.00,0.05,0.00,normal,20
0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00,0.00,0.00,0.00,0.08,0.15,0.00,255,1,0.00,0.6
0,0.88,0.00,0.00,0.00,0.00,0.00,normal,15
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.00,1.00,0.00,0.00,0.05,0.07,0.00,255,26,0.10,0
.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,http,SF,232,8153,0,0,0,0,1,0,0,0,0,0,0,0,0,0,5,5,0.20,0.20,0.00,0.00,1.00,0.00,0.00,30,255,1.00,0
.00,0.03,0.04,0.03,0.01,0.00,0.01,normal,21
0,tcp,http,SF,199,420,0,0,0,0,1,0,0,0,0,0,0,0,0,0,30,32,0.00,0.00,0.00,0.00,1.00,0.00,0.09,255,255,1.00
,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0.00,0.00,1.00,1.00,0.16,0.06,0.00,255,19,0.07
,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.00,1.00,0.00,0.00,0.05,0.06,0.00,255,9,0.04,0.
05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.00,1.00,0.00,0.00,0.14,0.06,0.00,255,15,0.06,
0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23,1.00,1.00,0.00,0.00,0.09,0.05,0.00,255,23,0.
09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
```

شکل ۳-۱: بخشی از داده‌های نفوذ در پایگاه NSL-KDD

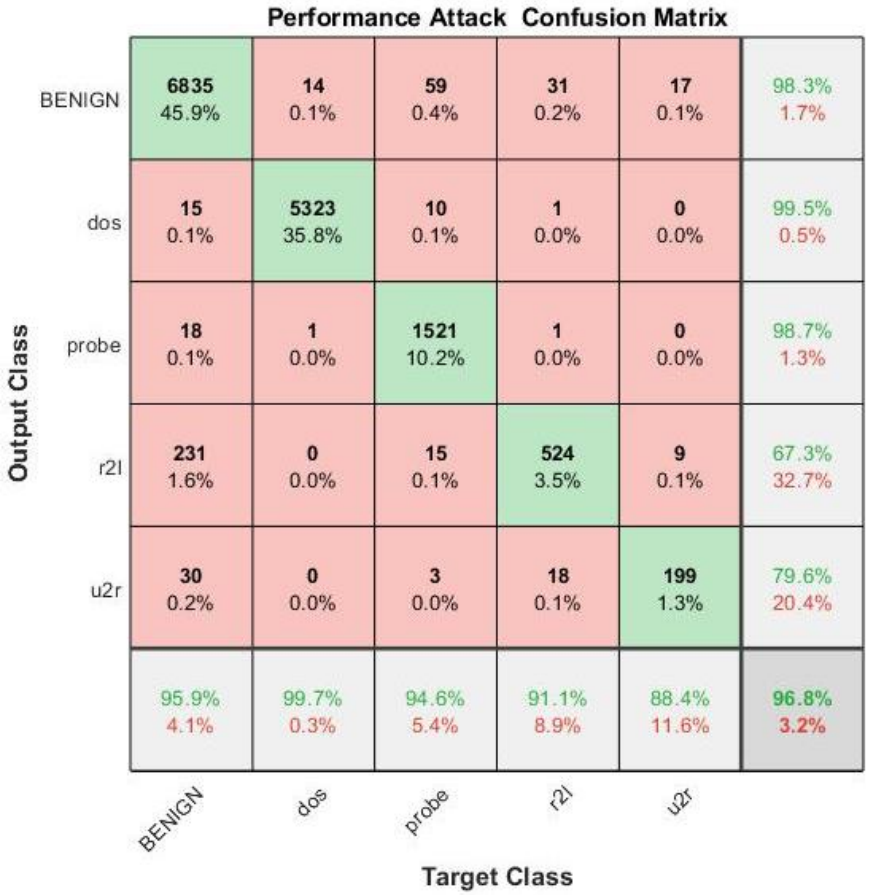
خروجی‌های راهکار پیشنهادی به شرح زیر است:

۱) ماتریس درهم ریختگی[۳۵] در دو حالت، دو کلاسه و چند کلاسه: پس از ارزیابی مدل و پیشگویی کلاس حملات توسط شبکه عصبی عمیق بازگشتی اصلی‌ترین خروجی مدل که کلیه متغیرهای وابسته مطرح در این تحقیق با استفاده از آن محاسبه می‌شوند ماتریس درهم ریختگی است که در شکل (۳-۲) و (۳-۳) حالت دو کلاسه و چند کلاسه آن برای یک پایگاه داده نشان داده شده است.



شکل ۳-۲: ماتریس درهم ریختگی روش در حالت دو کلاسه

[35] Confusion Matrix



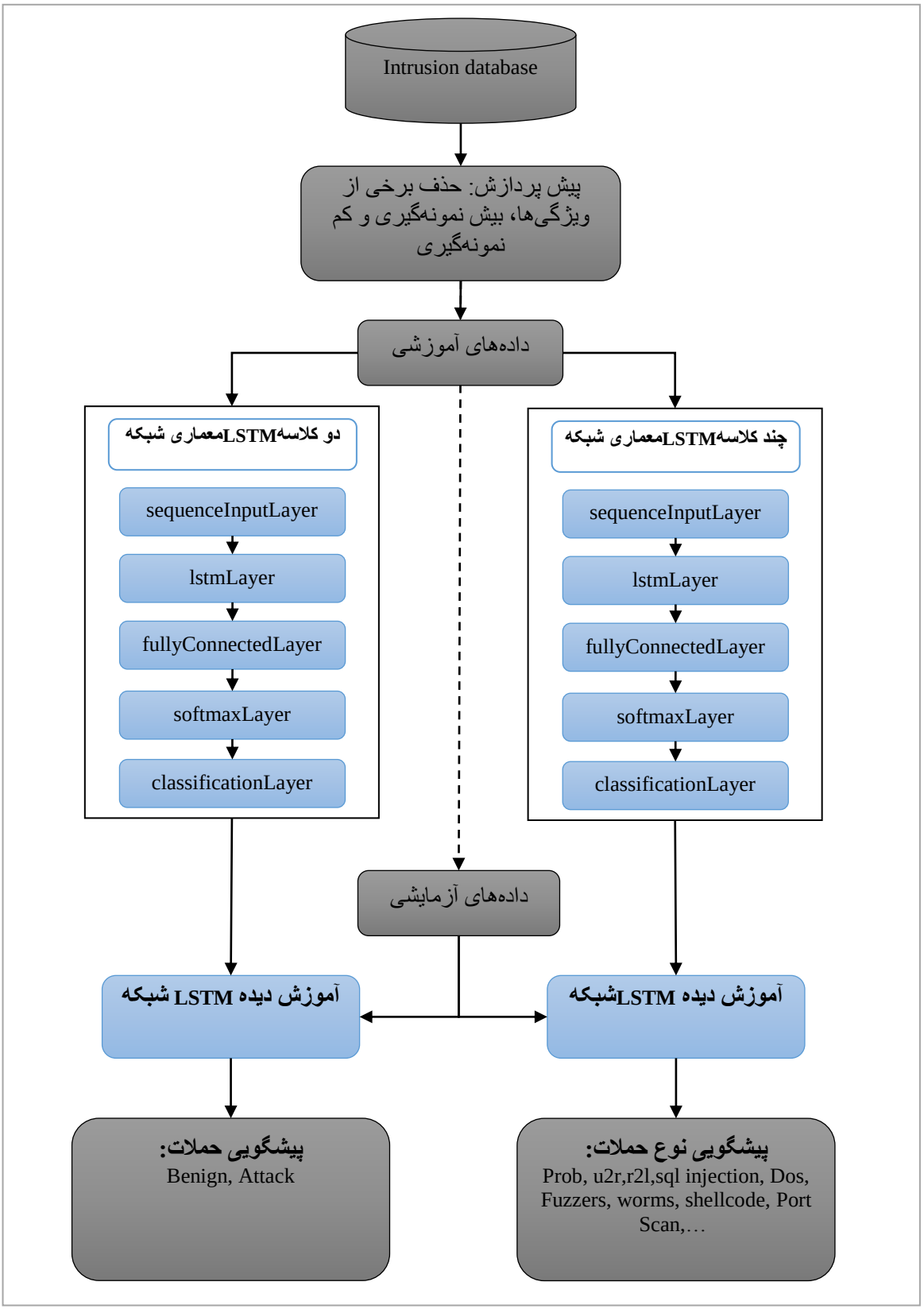
شکل ۳-۳: ماتریس درهم ریختگی روش در حالت چند کلاسه

۲) مقادیر متغیرهای وابسته تحقیق محاسبه شده در مرحله ارزیابی مدل: متغیرهای صحت، دقت، فراخوان و میانگین f از جمله خروجی‌های روش پیشنهاد شده است که برای هر دو حالت دو کلاسه و چند کلاس محاسبه شده و ارائه می‌شوند.

۳,۳ *چارچوب راهکار پیشنهادی*

راهکار مطرح در این تحقیق یک روش تشخیص نفوذ در شبکه است که از روش‌های یادگیری عمیق شامل دو شبکه عصبی عمیق بازگشتی LSTM[۳۶] متشکل از ۵ لایه و ترکیب آن با تکنیک‌های بیش نمونه‌گیری و کم نمونه‌گیری برای تشخیص نفوذ و تشخیص نوع نفوذ و حملات استفاده می‌کند. روش پیشنهاد شده شامل ۳ گام اصلی پیش پردازش و آماده‌سازی داده‌ها، آموزش شبکه و ارزیابی مدل می‌باشد. در شکل (۳-۴) دیاگرام روش ترسیم شده است.

[36] Long Short-Term Memory


Intrusion database
پیش پردازش: حذف برخی از ویژگی‌ها، بیش نمونه‌گیری و کم نمونه‌گیری
داده‌های آموزشی
دو کلاسه LSTMمعماری شبکه
sequenceInputLayer
lstmLayer
fullyConnectedLayer
softmaxLayer
classificationLayer
چند کلاسه LSTMمعماری شبکه
sequenceInputLayer
lstmLayer
fullyConnectedLayer
softmaxLayer
classificationLayer
داده‌های آزمایشی
آموزش دیده LSTM شبکه
آموزش دیده LSTMشبکه
پیشگویی حملات:
Benign, Attack
پیشگویی نوع حملات:
Prob, u2r,r2l,sql injection, Dos, Fuzzers, worms, shellcode, Port Scan,…

شکل ۳-۴: دیاگرام راهکار پیشنهادی

۴،۳ مرحله پیش پردازش داده‌ها و رفع مشکل عدم توازن

مرحله پیش پردازش در روش پیشنهادی از سه گام به شرح زیر تشکیل شده است:

در برخی از پایگاه داده‌های حملات مثل پایگاه داده UNSWNB15 استفاده شده در این تحقیق اطلاعاتی وجود دارد که به صورت ویژگی‌های اسمی[۳۷] در پایگاه ذخیره شده‌اند. از جمله این ویژگی‌های می‌توان به ویژگی سرویس[۳۸] و ویژگی حالت[۳۹] اشاره نمود که به ترتیب دارای مقادیری اسمی مثل irc, ftp-data, dns, ssh, smtp, ftp, http و ACC، CLO، CON، CEO و غیره هستند؛ که روش‌های یادگیری ماشین و یادگیری عمیق قادر به طبقه‌بندی چنین داده‌هایی نیستند. در مرحله پیش پردازش در اولین گام داده‌های اسمی با شاخص نشان دهنده آن‌ها جایگزین شده‌اند. به این معنی که مقدار یکتا برای هر یک از ویژگی‌های اسمی استخراج شده است و برای هر ویژگی مقادیر شاخص هر یک از مقادیر یکتای استخراج شده جایگزین شده است. به عنوان مثال اگر مقدار شاخص متغیر اسمی سرویس برای مقدار ftp برابر با ۳ باشد، کلیه رکوردهای پایگاه داده که در ویژگی سرویس دارای مقدار ftp می‌باشند، مقدار ۳ در آن‌ها به جای ftp درج می‌گردد.

در دومین گام از مرحله پیش پردازش مشکل داده‌های از دست رفته[۴۰] برطرف شده است. داده‌های از دست رفته از دیگر مشکلات مطرح در پایگاه داده‌ها هستند که در صورت عدم رفع آن‌ها نتایج روش‌های یادگیری ماشین کاهش می‌یابد. برای این بخش، در رکوردهای هر سه پایگاه داده که دارای مقادیر از دست رفته هستند، مقدار صفر درج شده است.

در آخرین گام از روش پیشنهادی مشکل عدم توازن داده‌ها برطرف شده است. در این مرحله برای جلوگیری از تغیرات بسیار زیاد در داده‌ها از برقراری توازن کامل در داده‌ها اجتناب شده است. غالبا در داده‌های حملات در دسترس، کلاس نرمال و یا Benign در مقایسه با کلاس حملات دیگر اختلاف بسیار زیادی دارد. به عنوان مثال در پایگاه داده UNSWNB15 بالغ بر ۳۰۰ هزار داده به کلاس نرمال تعلق دارند در صورتی که برخی از حملات در این پایگاه داده مثل Shellcode و Worms به ترتیب دارای ۳۷۱ و ۴۳ نمونه داده هستند. در نتیجه برقراری کامل توازن در این پایگاه داده‌ها باعث تغیرات بسیار زیاد در پایگاه می‌شود. از اینرو در روش پیشنهادی در راستای بهبود تشخیص نوع حملات به تعداد محدودی، به کلاس

---

[37] Nominal
[38] Service
[39] State
[40] Missing Value

برخی از حملات که دارای نمونه‌های بسیار کمی هستند افزوده شده است. تا به این صورت شبکه عصبی عمیق پیشنهادی بتواند در مرحله آموزش با داده‌های حملات نادر نیز مواجه شده و داده‌های آن‌ها را نیز بیاموزد. برای برقراری توازن نسبی در راهکار پیشنهادی به صورت زیر اقدام شده است:

ابتدا برای هر پایگاه داده تعداد داده‌های هر کلاس استخراج می‌شود. لازم به ذکر است که تعداد کلاس (نوع حملات) در سه پایگاه داده مورد بررسی در این تحقیق متفاوت می‌باشد به عنوان مثال در یک پایگاه داده ۵ نوع حمله وجود دارد و در پایگاه دیگر ۱۰ نوع حمله مطرح می‌باشد. سپس حملات براساس تعداد نمونه‌های آن‌ها به صورت نزولی مرتب می‌شوند. در بین نادرترین حملات، ۳ حمله که در مقایسه با دیگر حملات دارای نمونه داده‌های بسیار کمی می‌باشند انتخاب می‌شود. سپس با استفاده از الگوریتم SMOTE بین ۵۰۰ تا ۳۰۰۰ داده به هر ۳ کلاس حمله نادر افزوده می‌شود. در راهکار پیشنهادی تعداد همسایگان در الگوریتم SMOTE برابر با ۵ در نظر گرفته شد و برای محاسبه تشابه همسایگان از فاصله اقلیدسی ارائه شده در رابطه (۳-۱) استفاده گردید. با ذکر یک مثال نحوه عملکرد الگوریتم SMOTE شرح داده می‌شود:

$$eucledian = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (۱-۳)$$

در رابطه بالا n تعداد ابعاد داده است و x و y دو داده از داده‌های کلاس اقلیت هستند. اگر فرض شود i یک نمونه داده از کلاس اقلیت دارای مقادیر (۴، ۶) باشد و j نیز اولین نزدیکترین همسایه آن دارای مقادیر (۳، ۴) باشد، آنگاه js که یک داده مصنوعی تولید شده توسط الگوریتم SMOTE برای i می‌باشد به صورت زیر محاسبه می‌شود.

$$feature_{1i} = 6\,, feature_{2i} = 4 \;\rightarrow feature_{1j} - feature_{1i} = -2$$
$$feature_{1j} = 4\,, feature_{2j} = 3 \;\rightarrow feature_{2j} - feature_{2i} = -1$$
$$JS: synthetic\ Sample:$$
$$(feature_1', feature_2') = (6,4) + rand(0,1) \times (-2,-1)$$

پس از بیش نمونه‌گیری برای هر یک از سه کلاس دارای حملات نادر، تکنیک کم نمونه‌گیری تصادفی در روش پیشنهادی انجام می‌شود. در این بخش به تعداد داده‌های افزوده شده به سه کلاس اقلیت به صورت تصادف داده‌هایی در کلاس اکثریت نرمال انتخاب شده و حذف می‌شوند. به عنوان مثال اگر به هر کلاس اقلیت ۱۰۰۰ داده مصنوعی افزوده شده باشد در مجموع ۳۰۰۰ داده به صورت تصادفی از داده‌های کلاس نرمال انتخاب شده و حذف می‌شوند. در شکل (۳-۵) برای مثال وضعیت کلاس حملات در پایگاه داده UNSWNB15 قبل از باز نمونه‌گیری و بعد از بازنمونه‌گیری نشان داده شده است.

UNSW-NB-2015		UNSW-NB-2015	
Analysis	670	Analysis	670
Backdoor	1832	Backdoor	666
DoS	4907	DoS	4907
Exploits	11439	Exploits	11439
Fuzzers	5390	Fuzzers	5390
Generic	61878	Generic	61878
Reconnaissance	3530	Reconnaissance	3530
Shellcode	1408	Shellcode	371
Worms	1044	Worms	43
BENIGN	347956	BENIGN	351150
BENIGN	347956	BENIGN	351150
ATTACK	92098	ATTACK	88894

شکل ۳-۵: وضعیت حملات نادر قبل و بعد از بیش نمونه‌گیری

در شکل فوق تصویر سمت چپ پیش از بازنمونه‌گیری و تصویر سمت راست پس از بازنمونه‌گیری است. در این شکل سه نوع حمله نادر Worms، Shellcode و Backdoor که دارای حداقل نمونه ممکن هستند با استفاده از الگوریتم SMOTE افزایش یافته‌اند و در مقابل کلاس اکثریت نرمال کاهش یافته است. در واقع در این بخش هدف افزایش تعداد حملات نادر است بنحوی که شبکه عصبی بازگشتی بتواند داده‌های حملات نادر را نیز پیشگویی نماید.

۵,۳ مرحله یادگیری دو شبکه عصبی عمیق بازگشتی

در این بخش پس از پیش پردازش داده‌ها و آماده‌سازی آن‌ها و رفع مشکل حملات نادر و برقراری توازن نسبی در داده‌ها، با استفاده از دو شبکه عصبی عمیق بازگشتی LSTM فرآیند طبقه‌بندی حملات در دو حالت دو کلاسی و چند کلاسی انجام می‌شود. در این مرحله یکی از شبکه‌های LSTM برای پیشگویی وقوع حمله آموزش می‌بیند و شبکه دیگر برای تشخیص نوع حمله آموزش می‌بیند. معماری تعریف شده در این تحقیق برای هر دو شبکه LSTM یک معماری ۵ لایه‌ای است که در شکل (۳-۶) نشان داده شده است. تفاوت شبکه‌ها فقط در برچسب کلاس‌هایی است که به آن‌ها ارائه می‌شود. در واقع هر دو شبکه بر روی داده‌های آموزشی یکسان آموزش می‌بینند و فقط برچسب کلاس داده‌ها برای یک شبکه دو کلاسی و برای شبکه دیگر چند کلاسی می‌باشد. لازم به ذکر است که کلیه روش‌های یادگیری ماشین و یادگیری عمیق با توجه به برچسب کلاس داده‌های آموزشی قابلیت پیشگویی دارند به این معنی که نمی‌توان به یک الگوریتم طبقه‌بندی داده‌هایی با برچسب دو کلاسی ارائه نمود و از آن انتظار داشت که قابلیت پیشگویی چند کلاسی را نیز داشته باشد. در ادامه ۵ لایه تعریف شده در هر دو معماری شرح داده می‌شود. در هر دو معماری هر یک از لایه‌ها به شرح زیر می‌باشند:

۱) اولین لایه در دو معماری Sequence Input است. این لایه مسئول دریافت ورودی‌ها در قالب پایگاه داده‌ای از توالی‌ها می‌باشد. بردارهای ویژگی آماده شده در مرحله قبل، بدون برچسب کلاس آن‌ها به عنوان ورودی به این لایه داده می‌شود. لازم به ذکر است که در این بخش طول بردارهای ویژگی باید برابر باشند.

۲) دومین لایه در هر دو معماری لایه LSTM است. یک لایه LSTM وابستگی‌های طولانی مدت بین مراحل زمانی در سری زمانی و داده‌های توالی را می‌آموزد. این لایه فعل و انفعالات افزودنی را انجام می‌دهد، که می‌تواند به بهبود جریان گرادیان توالی‌های طولانی در مرحله یادگیری شبکه و همگرایی شبکه کمک کند. در این لایه تعداد واحدهای پنهان یا نرون‌های پنهان به عنوان ورودی شبکه مشخص می‌شود. در روش پیشنهادی ۲۵۶ واحد پنهان برای این لایه در هر دو شبکه در نظر گرفته شده است. همانطور که در فصل دوم نیز بیان گردید تابع فعال‌ساز واحدهای پنهان در این لایه، تابع tanh است.

۳) سومین لایه در هر دو معماری لایه تمام متصل است. لایه‌های تمام متصل ساختاری مشابه با شبکه‌های عصبی پیشخور[41] دارند. در این لایه به عنوان ورودی تعداد کلاس داده‌ها دریافت می‌شود که خروجی لایه تمام متصل برابر با تعداد کلاس ورودی است. بطور دقیق تعداد کلاس در شبکه بازگشتی دو کلاسی ۲ می‌باشد، در نتیجه خروجی لایه تمام متصل در این شبکه برابر با ۲ خواهد بود. در این لایه ورودی‌ها با بردار وزن‌های تصادفی تعریف شده توسط شبکه ضرب می‌شوند و سپس با مقدار bias جمع می‌شوند.

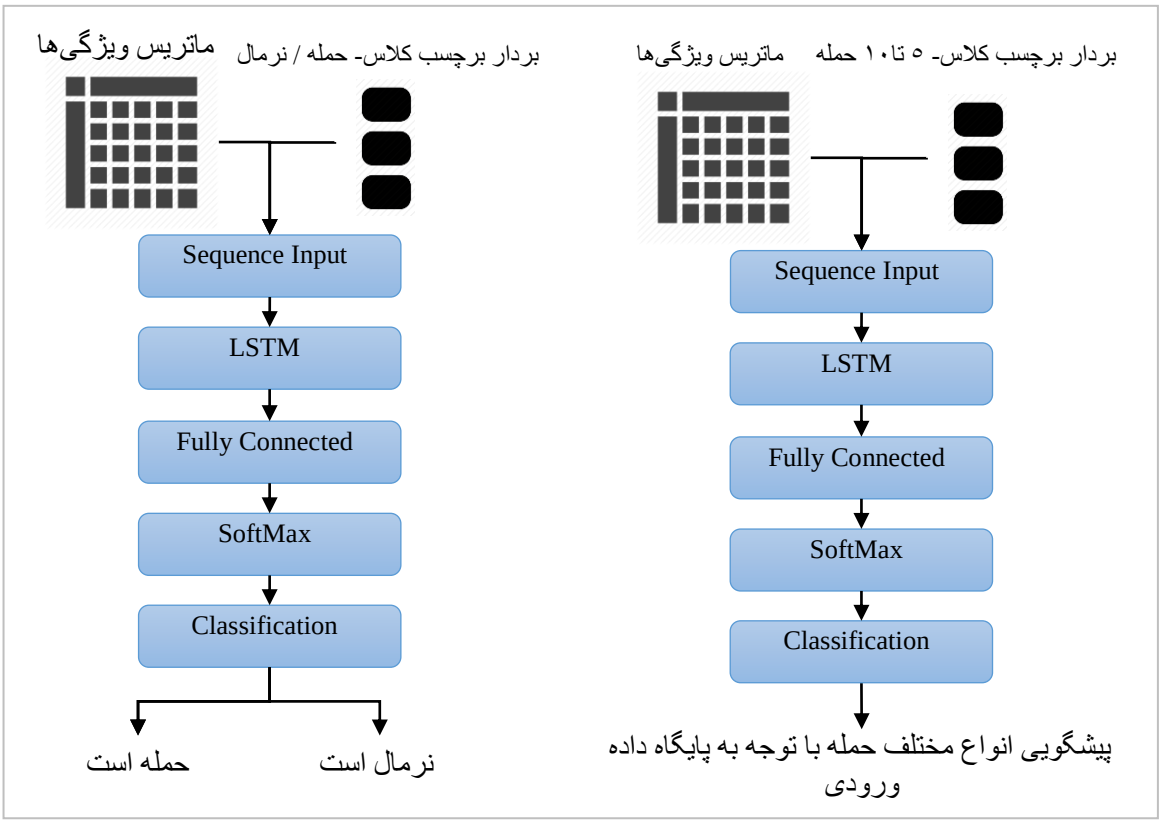
۴) چهارمین لایه در هر دو معماری لایه SoftMax است؛ این لایه در واقع یک تابع فعال‌ساز[42] است که از رابطه (۳-۲) محاسبه می‌شود. این تابع داده‌های ورودی را به مقادیری بین ۰ تا ۱ نگاشت می‌کند که مجموع احتمالات و امتیازهای محاسبه شده توسط آن برای هر نمونه داده برابر با ۱ خواهد بود (Bebis & Georgiopoulos, 1994). وظیفه این لایه پیشگویی کلاس داده‌ها می‌باشد. عملکرد این لایه به این صورت است که برای ویژگی‌های ورودی، میزان تعلق یا امتیاز آن‌ها به هر کلاس محاسبه می‌شود و سپس بردار ویژگی مورد نظر به کلاس با بیشترین امتیاز تخصیص داده می‌شود. این لایه فقط در شرایطی که شبکه برای مسائل طبقه‌بندی تعریف شده باشد قرار می‌گیرد و در شبکه‌های عمیق رگرسیون قرار نمی‌گیرد.

---

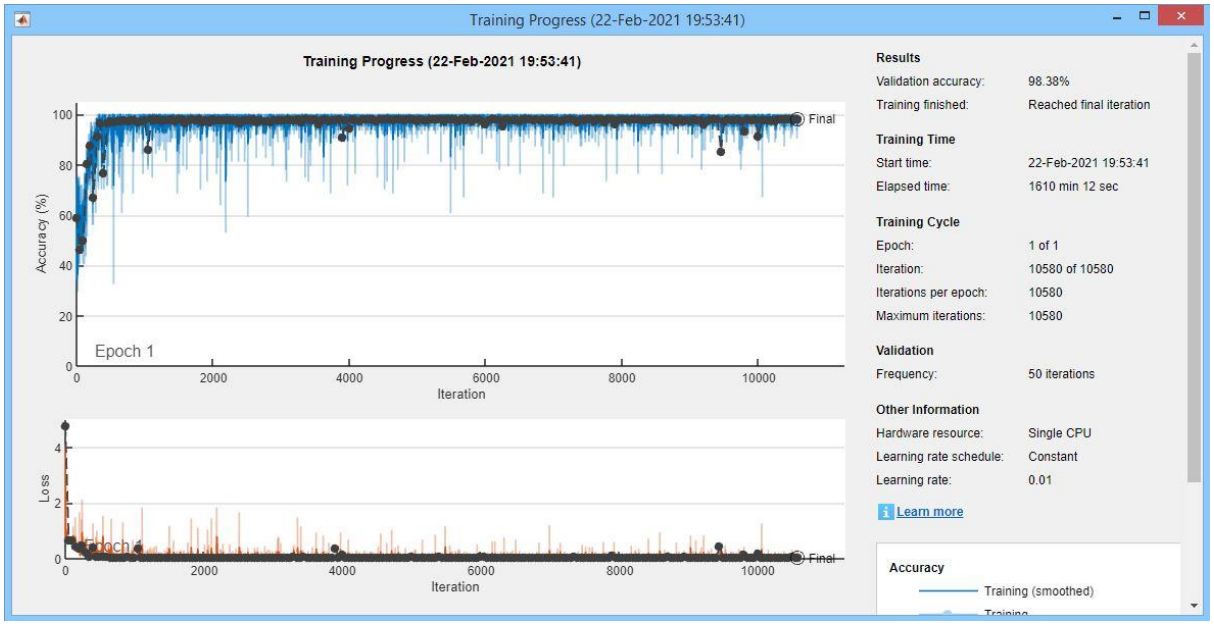
41 Feedforward
42 Activation Function

$$\sigma(Z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad ,where\ j = 1, \dots, K \qquad (۲-۳)$$

۵) آخرین لایه در هر دو معماری لایه طبقه‌بندی است. این لایه دارای یک تابع loss است که میزان خطای شبکه در مرحله یادگیری را محاسبه می‌کند. مقدار خطای شبکه در مرحله یادگیری به شبکه باز می‌گردد تا در بروزرسانی بهینه وزن‌های لایه‌های شبکه در راستای کاهش خطای شبکه استفاده شود. لازم به ذکر است که در هر دو شبکه پیشنهادی در این تحقیق مقادیر لایه‌های مختلف غالبا به صورت پیش فرض در نظر گرفته شده‌اند. در فصل چهارم از این تحقیق پارامترهای تنظیمات شبکه‌ها به صورت جدول قبل از ارائه نتایج بیان می‌شوند. در شکل (۳-۷) نمودار مرحله آموزش شبکه LSTM برای یک گردش در حالت دو کلاسه نشان داده شده است. در این شبکه نمودار بالایی بیانگر صحت عملکرد مدل در مرحله یادگیری است و نمودار پایین بیانگر میزان خطای شبکه در مرحله یادگیری است که با افزایش تعداد گردش شبکه میزان صحت افزایش یافته و مقدار خطای شبکه کاهش یافته است که این بیانگر عملکرد صحیح شبکه در مرحله یادگیری است. در این شکل نقاط مشکی رنگ نشان دهنده ارزیابی شبکه در داده‌های Validation در حین فرآیند یادگیری است. در این شکل مقدار صحت درج شده در سمت راست بالای تصویر بیانگر صحت عملکرد الگوریتم در داده‌های validation است که در این شکل برابر با ۹۸,۳۸٪ برای مقدار Epoch برابر با ۱ حاصل شده است. لازم به ذکر است که در روش پیشنهادی مقدار Epoch آموزش هر دو شبکه برابر با ۵۰ در نظر گرفته شده است.



شکل ۳-۶: معماری دو شبکه عصبی عمیق بازگشتی LSTM



شکل ۳-۷: نمودار مرحله آموزش شبکه عمیق LSTM

۳,۶ مرحله ارزیابی دو شبکه عصبی بازگشتی

آخرین از مرحله در روش پیشنهادی مرحله ارزیابی است. در این بخش پیشگویی‌های انجام شده توسط هر دو شبکه با برچسب حقیقی کلاس داده‌ها مقایسه می‌شوند و میزان موفقیت مدل در تشخیص صحیح حملات و نوع حملات محاسبه می‌شود. با مقایسه پیشگویی‌های دو شبکه مقادیر دقت و صحت و فراخوان که از جمله متغیرهای وابسته این تحقیق هستند حاصل می‌شود. نتایج این بخش از مدل در فصل آتی گزارش می‌شود و در رابطه با آن بحث خواهد شد. در جدول (۳-۱) یک مثال فرضی از این مرحله از روش ارائه شده است. در این جدول از بین ۱۰ نمونه داده فرضی در دو حالت دو کلاسی و چند کلاسی مدل توانسته است کلاس ۹ نمونه داده را به درستی پیشگویی کند و یک نمونه داده را به کلاس اشتباه تخصیص دهد در نتیجه صحت عملکرد مدل در این مثال فرضی در هر دو شبکه برابر با ۹۰٪ شده است. مرحله ارزیابی هر دو شبکه مشابه به روش زیر محاسبه می‌شود:

جدول ۳-۱: بررسی کلاس پیشگویی شده توسط دو شبکه در مرحله ارزیابی مدل

ارزیابی مدل در حالت چند کلاسی			ارزیابی مدل در حالت دو کلاسی		
میزان موفقیت	کلاس حقیقی	کلاس پیشگویی شده	میزان موفقیت	کلاس حقیقی	کلاس پیشگویی شده
×	Dos	Probe	✓	Benign	Benign
✓	Dos	Dos	✓	Benign	Benign
✓	Dos	Dos	✓	Benign	Benign
✓	Dos	Dos	✓	Benign	Benign
✓	Dos	Dos	✓	Benign	Benign
✓	Generic	Generic	✓	Benign	Benign
✓	Generic	Generic	×	Benign	Attack
✓	Generic	Generic	✓	Attack	Attack
✓	Generic	Generic	✓	Attack	Attack
✓	Generic	Generic	✓	Attack	Attack
٪ ۹۰			٪ ۹۰		

در شکل (۳-۸) شبه کد روش پیشنهادی نیز بیان شده است. هر خط از شبه کد ارائه شده براساس شماره آن شرح داده می‌شود. در خط اول هر یک از سه پایگاه داده دریافت می‌شود. در خط دوم داده‌های اسمی مشابه با آنچه در بخش پیش پردازش بیان شده به اعداد تبدیل می‌شوند و مقدار هر داده اسمی با شاخص آن در لیست مقادیر یکتا جایگزین می‌شود. در خط سوم داده‌های از دسته رفته با مقدار صفر جایگزین می‌شوند. در خط چهارم برای حل مشکل حملات نادر و رفع نسبی مشکل عدم توازن داده‌ها از الگوریتم SMOTE برای افزایش تعداد داده‌های سه کلاس اقلیت استفاده می‌گردد. در خط پنجم پس از تولید داده‌های مصنوعی جدید، با استفاده از روش کم نمونه‌گیری تصادفی به تعداد افزوده شده به سه کلاس اقلیت، به تصادف تعدادی از داده‌های کلاس اکثریت انتخاب شده و حذف می‌شوند. در خط ۶ برای آموزش و آزمایش شبکه، داده‌های نهایی شده به دو بخش داده‌های آزمایش و آموزش تقسیم‌بندی می‌شوند. در خطوط ۷ و ۸ معماری هر دو شبکه تولید می‌شود. در خطوط ۹ و ۱۰ با توجه به پارامترهای تعریف شده برای دو شبکه و معماری ساخته شده برای دو شبکه و داده‌های آموزشی، دو شبکه عمیق LSTM آموزش داده می‌شوند. در خطوط ۱۰ و ۱۱ از شبکه‌های آموزش داده شده برای پیشگویی کلاس داده‌های بخش آزمایش استفاده می‌شود. خروجی این دو خط برچسب کلاس حملات و نوع آن‌ها می‌باشد. در خطوط ۱۳ تا ۱۵ با مقایسه کلاس‌های پیشگویی شده توسط دو شبکه LSTM و کلاس حقیقی داده‌ها ماتریس درهم ریختگی برای دو کلاس و چند کلاس تشکیل می‌شود و با استفاده از این دو ماتریس خروجی‌های نهایی روش یعنی صحت، دقت، فراخوان و میانگین f محاسبه می‌شوند و به عنوان خروجی به کاربر ارائه می‌شوند.

Improving Network Intrusion Detection Using Deep Neural Networks and Combining Over-Sampling and Under-Sampling Techniques
Begin

Input: 3 Database, 2 LSTM option, Data Ratio Partitioning, K and Ratio to add in SMOTE.
Output: Confusion matrix, Accuracy, Precision, Recall, Fmeasure.

// Step1: Preprocessing, Correct Imbalance Data

1- Data=import (Database);
2- Data=Convert Nominal data to real data;
3- Data=Replace Missing Value with Zero;
4- [New Data] = ***SMOTE*** (Data, Class Attack, Ratio to addendum Neighbors);
5- [Final Data, Final Label] =***Randomly Under Sampling*** (New Data, Class Binary and Attack, Sum Oversampling Data);

// Step2: Train 2 LSTM Network

6- [Train Data, Train Label, Test Data, Test Label] = ***Partition Data*** (Final Data, Final Label);
7- Layers Binary = [Sequence Input Layer (1), Lstm Layer (Num Hidden Units=256), Fully Connected Layer (Numclasses Binary=2), SoftMax Layer, Classification Layer];
8- Layers Attack = [Sequence Input Layer (1), Lstm Layer (Num Hidden Units=256), Fully Connected Layer (Numclasses Binary=N), SoftMax Layer, Classification Layer];
9- Lstmnet Binary = ***Train Network*** (Train Data, Train Label, Layers Binary, Options);
10- Lstmnet Attack = ***Train Network*** (Train Data, Train Label Attack, Layers Attack, Options);

// Step 3: Validation LSTMs

11- Prediction Binary = ***Classify*** (Lstmnet Binary, Test Data);
12- Prediction Attack = ***Classify*** (Lstmnet Attack, Test Data);
13- Matrix Binary=***Confusion matrix*** (Prediction Binary, Test Label Binary);
14- Matrix Attack= ***Confusion matrix*** (Prediction Attack, Test Label Attack);
15- Result=***Calculate*** (Accuracy, Precision, Recall, Fmeasure);

End

شکل ۳-۸: شبه کد روش پیشنهادی

۴. ارزیابی و نتایج تجربی

۱،٤ معیارهای ارزیابی

در تکنیک‌های طبقه‌بندی از معیارهای مختلفی برای ارزیابی روش‌ها و بررسی میزان موفقیت آن‌ها استفاده می‌شود که از جمله متداولترین آن‌ها معیارهای صحت، دقت، فراخوان و میانگین f هستند (Ravipati & Abualkibash, 2019).

معیارهای فوق هر یک بخشی از مدل را بررسی می‌کنند که محاسبه همه آن‌ها در یک مدل می‌تواند باعث شود ابعاد مختلف مدل به خوبی بررسی شوند. تعاریف و روابط هر یک از ۴ متغیر فوق به شرح زیر می باشد:

۱) صحت: این معیار میزان موفقیت کلی مدل در کل پیشگویی‌های انجام شده در داده‌های آزمایشی را نشان می‌دهد. این معیار دید کاملی از نحوه عملکرد مدل ارائه می‌دهد و از رابطه (۴-۱) محاسبه می‌شود (Ravipati & Abualkibash, 2019)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (۱-٤)$$

۲) دقت: این معیار بیانگر تعداد نمونه داده‌هایی است که در هر کلاس به درستی پیشگویی شده‌اند و از رابطه (۴-۲) محاسبه می‌شود. مقدار این معیار برای هر کلاس متفاوت است (Ravipati & Abualkibash, 2019).

$$Precision = \frac{TP}{TP+FP} \quad (۲-٤)$$

۳) فراخوان: این معیار تعداد نمونه‌های درست پیشگویی شده در هر کلاس به کل داده‌های کلاس مورد نظر را بیان می‌کند و از رابطه (۴-۳) محاسبه می‌شود. این معیار نیز برای هر کلاس دارای مقدار متفاوتی است (Ravipati & Abualkibash, 2019).

$$Recall = \frac{TP}{TP+FN} \quad (۳-٤)$$

۴) میانگین f: آخرین معیار مورد بررسی در این تحقیق میانگین دو معیار دقت و فراخوان است که از رابطه (۴-۴) حاصل می‌شود (Ravipati & Abualkibash, 2019).

$$F_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (٤-٤)$$

در کلیه روابط فوق متغیرها استفاده شده با توجه به ماتریس درهم ریختگی ارائه شده در شکل (۴-۱) به شرح زیر هستند:

- True Positive یا به اختصار TP، بیانگر تعداد داده‌هایی است که کلاس حقیقی آن‌ها مثبت بوده و مدل به درستی آن‌ها را مثبت پیشگویی کرده است.
- False Positive یا به اختصار FP، بیانگر تعداد داده‌هایی است که کلاس حقیقی آن‌ها مثبت نیست و مدل به اشتباه کلاس آن‌ها را مثبت پیشگویی کرده است.

- True Negative یا به اختصار TN، بیانگر تعداد داده‌هایی است که کلاس حقیقی آن‌ها منفی بوده و مدل نیز به درستی کلاس آن‌ها را منفی پیشگویی کرده است.
- False negative با به اختصار TF، نشان دهنده داده‌هایی است که کلاس حقیقی آن‌ها منفی نبوده ولی مدل به اشتباه آن‌ها را به کلاس منفی تخصیص داده است.

		Actual Class	
		Positive	**Negative**
Predict Class	**Positive**	True Positive **TP**	False Positive **FP**
	Negative	False Negative **FN**	True Negative **TN**

شکل ۴-۱: ماتریس درهم‌ریختگی دو کلاسی

۲,٤ *مشخصات پایگاه داده‌ها*

در حوزه تشخیص نفوذ و حملات در شبکه، پایگاه داده‌های بسیاری مطرح می‌باشند که در مطالعات مختلف از آن‌ها استفاده می‌شود. در این تحقیق نیز برای بررسی جامعیت مدل پیشنهاد شده از سه پایگاه داده مختلف استفاده شده است تا به این صورت بتوان میزان موفقیت مدل را در داده‌های مختلف ارزیابی نمود. برخی از مدل‌های تشخیص نفوذ از داده‌ای به داده دیگر عملکرد بسیار متفاوتی از خود ارائه می‌دهند؛ در نتیجه بررسی یک مدل بر روی چندین پایگاه داده با خصوصیات و حملات مختلف می‌تواند نشان دهد که روش تا چه میزان در داده‌های مختلف مقاوم بوده و عملکرد موفقی از خود نشان می‌دهد (Hindy et al., 2020).

- پایگاه داده اول با عنوان UNSW-NB15 است که برای ارزیابی روش‌ها استفاده شده است. این پایگاه داده توسط یک موسسه امنیت سایبری در استرالیا تولید شده است. این پایگاه داده دارای ۴ نسخه مختلف می‌باشد که در این تحقیق از نسخه ۴ آن استفاده شده است. داده‌های این پایگاه شامل ۴۷ ویژگی و برچسب کلاس‌ها هستند. این پایگاه داده بالغ بر ۴۴۰۰۰۰ نمونه داده و ۹ کلاس حمله مختلف شامل fuzzers, analysis, backdoor, DoS, exploits, generic, reconnaissance, shellcode و worms دارد (Moustafa & Slay, 2015).
- پایگاه داده دوم در این تحقیق با نام CICIDS-2017 برای ارزیابی استفاده شده است. این مجموعه داده در موسسه کانادایی امنیت سایبری تولید شده و شامل حملات متداول و بروزی مانند Web-based , brute

bot , heart-bleed , infiltration , DDoS , DoS , force و scan attacks می‌باشد. فایل‌های این پایگاه داده حاوی ۸۰ ویژگی ترافیکی و برچسب کلاس حملات در حالت دو کلاسه و چند کلاسه هستند و دارای ۷۴۵،۴۲۳ داده می‌باشد (Sharafaldin, Lashkari, & Ghorbani, 2018).

- پایگاه داده سوم با عنوان NSL-KDD نیز آخرین پایگاه در این تحقیق برای ارزیابی روش‌ها می‌باشد این مجموعه داده در فرمت چندگانه، مانند فرمت‌های TXT و ARFF برای آموزش و آزمایش مدل‌ها در دسترس است که در این تحقیق از نسخه TXT استفاده شده است . این پایگاه داده شامل ۴۱ ویژگی و کلاس داده‌ها می‌باشد و شامل ۵ نوع کلاس مختلف از جمله DoS, U2R, R2L, probe attacks و normal می‌باشد این پایگاه بالغ بر ۱۴۸۰۰۰ داده دارد (Tavallaee, Bagheri, Lu, & Ghorbani, 2009). در جدول (۴-۱) مشخصات هر سه پایگاه داده بیان شده است.

جدول ۴-۱: مشخصات سه پایگاه داده

Name	Normal	Dos	DDos	Probe	U2R	R2L	Infiltrating/Scanning	SSH	FTP	Heartbleed	Brute Force	XSS	Sql Injection	Webshell	Port Scan	Analysis, Reconnaissance	Worms, Shellcode,Backdoor	Fuzzer,Expliots, Generic
CICIDS2017	✓	✓	✓	-	-	-	✓	✓	-	✓	✓	✓	✓	-	✓	-	-	-
NSL-KDD	✓	✓	-	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-
UNSW-NB15	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	✓

۳,٤ *طراحی آزمایش‌ها، پارامترها و محیط پیاده‌سازی*

در این تحقیق برای بررسی راهکار پیشنهاد شده ۶ گروه اصلی آزمایش انجام شده است. در دو گروه اول از آزمون‌ها تاثیر دو متغیر مستقل تحقیق بر روی اصلی‌ترین متغیر وابسته تحقیق یعنی صحت بررسی شده است. در این آزمایش‌ها به ترتیب نرخ داده‌های آموزشی بین ۰,۱ تا ۰,۹ از کل داده‌ها در نظر گرفته شده‌است و مقدار صحت برای هر سه پایگاه داده محاسبه شده است. نتایج این آزمون نشان می‌دهد که تا چه میزان مدل پیشنهاد شده در داده‌های کم نیز مقاوم بوده و عملکرد خوبی از خود نشان می‌دهد. در آزمون بعدی تعداد همسایگان بین ۱ تا ۵ همسایه در نظر گرفته شده است و میزان صحت برای مدل در هر سه پایگاه داده محاسبه شده است. این آزمون برای یافتن بهترین تعداد همسایه در روش پیشنهادی جهت افزایش داده‌های کلاس اقلیت انجام می‌شود. در ۴ گروه بعدی از آزمایش‌ها مقایسه متغیرهای وابسته تحقیق و روش پایه انجام

می‌شود. مقایسه هر یک از این متغیرها نشان می‌دهد که کدام روش در کدام پایگاه داده می‌تواند نتایج بهتری از خود ارائه دهد. برای پیاده‌سازی روش‌ها از نرم افزار متلب نسخه 2020b استفاده شده است. کلیه آزمون‌های این تحقیق در یک سیستم ۷ هسته‌ای با سرعت ۲٫۸ گیگا هرتز و حافظه اصلی ۸ گیگا بایت انجام شده است. در جدول (۴-۲) مقادیر پارامترهای پیاده‌سازی گزارش شده است.

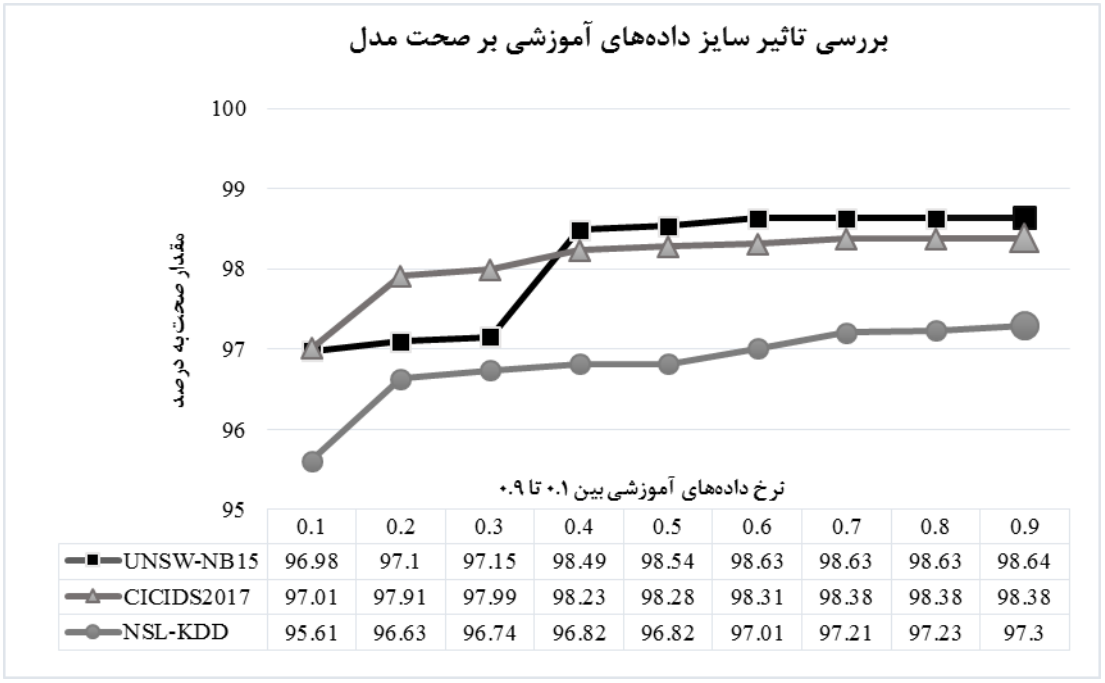
جدول ۴-۲: پارامترها و محیط پیاده‌سازی

پارامتر	مقدار
شبیه‌ساز	MATLAB 2020b
محیط ارزیابی	Ram 8 gig. CPU corei7
پارامترهای شبکه بازگشتی LSTM	
تعداد گردش(Epoch)	۵۰
Mini Batch Size	٦٤
Num Hidden Units	۲٥٦
نرخ یادگیری	۰,۰۱
تابه بهینه سازی	Root Mean Square Propagation
پارامترهای تولید داده مصنوعی	
تعداد همسایگان	٥-۱
نرخ افزایش	٥۰۰ تا ۳۰۰۰
تعداد کلاس نادر	۳
معیار فاصله	اقلیدسی
پارامترهای تقسیم‌بندی داده	
نرخ داده‌ها	۰,۹ -۰,۱

۴,۴ بررسی تاثیر سایز داده‌های آموزشی بر نتایج مدل پیشنهادی

اولین دسته از آزمون‌ها، برای بررسی تاثیر نرخ داده‌های آموزشی بر عملکرد مدل انجام شده است. با توجه به اینکه معیار صحت کلی‌ترین معیاری است که می‌تواند به خوبی عملکرد مدل را ارائه دهد در این آزمون برای هر سه پایگاه داده تاثیر سایز داده‌های آموزشی بر صحت مدل بررسی شده است. در این آزمایش نرخ داده‌های آموزشی بین ۰,۱ تا ۰,۹ در نظر گرفته شده است. تعداد همسایگان در این آزمایش برابر با ۵ تعیین شده و تعداد ۲۰۰۰ نمونه داده برای هر سه پایگاه داده تولید شده است. در روش‌های یادگیری اعم از عمیق و ماشین، نرخ داده‌های آموزشی بر نتایج مدل موثر می‌باشد. در این روش‌ها هرچه سایز داده‌های آموزشی بیشتر باشد مدل تولید شده در مرحله آموزش کامل‌تر بوده و در نتیجه این مدل در مرحله آزمایش نیز عملکرد بهتری از خود ارائه می‌دهد. در برخی از روش‌های طبقه‌بندی تاثیر سایز داده به شدت در نتایج مدل موثر است و در

مقابل در برخی دیگر تاثیر آن کمتر می‌باشد. از اینرو در اولین بخش از ارزیابی مدل پیشنهادی، هدف بررسی میزان تاثیر داده‌های آموزشی در نتایج صحت مدل است. نتایج این آزمایش در نمودار (۴-۱) گزارش شده است.



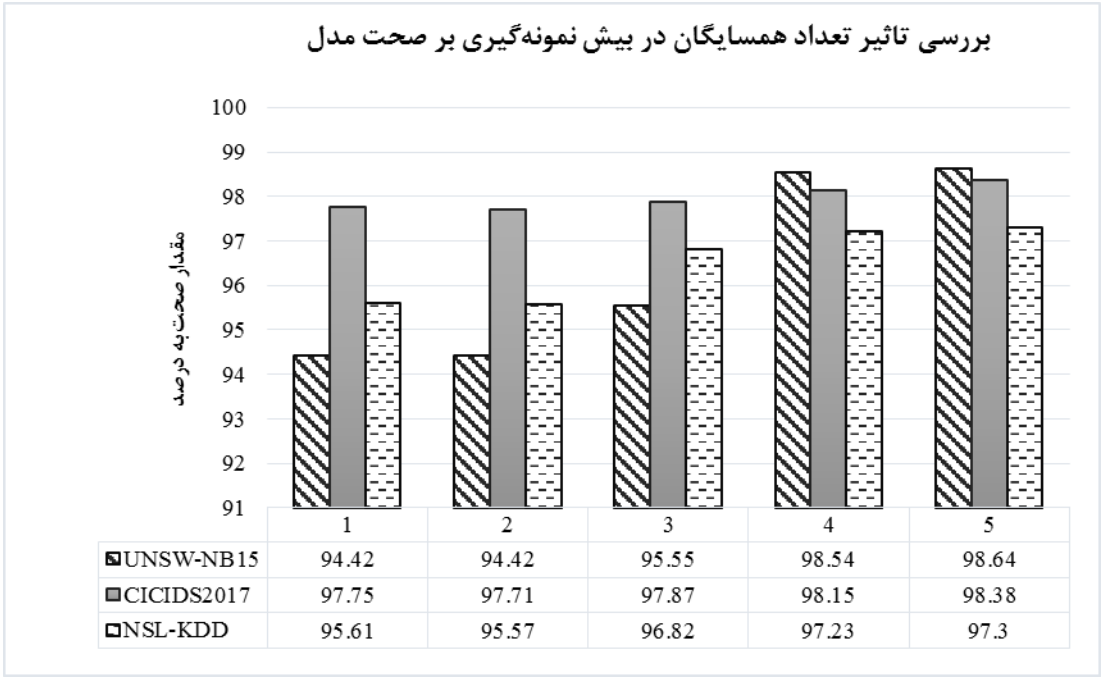
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UNSW-NB15	96.98	97.1	97.15	98.49	98.54	98.63	98.63	98.63	98.64
CICIDS2017	97.01	97.91	97.99	98.23	98.28	98.31	98.38	98.38	98.38
NSL-KDD	95.61	96.63	96.74	96.82	96.82	97.01	97.21	97.23	97.3

نمودار ۴-۱: بررسی تاثیر سایز داده‌های آزمایشی بر نتایج مدل

همانطور که در نمودار بالا گزارش شده است بهترین مقدار صحت در هر سه پایگاه داده در نرخ داده‌های آموزشی برابر با ۰,۹ حاصل شده است. به این معنی که ۹۰٪ از کل پایگاه داده به عنوان داده‌های آموزشی به شبکه عصبی بازگشتی ارائه می‌شود و ۱۰٪ باقی مانده به عنوان داده‌های آزمایشی برای ارزیابی شبکه عصبی استفاده می‌شود. در بین سه پایگاه داده فوق بهترین مقدار برابر با ۹۸,۶۴٪ است که برای پایگاه داده UNSW-NB15 حاصل شده است. بر طبق نتایج این آزمون در کلیه آزمون‌های مقایسه‌ای نرخ داده‌ها ۰,۹ به ۰,۱ در نظر گرفته می‌شود.

۵,۴ بررسی تعداد همسایگان در الگوریتم بیش نمونه‌گیری بر نتایج مدل پیشنهادی

دومین گروه از آزمون‌ها برای بررسی تاثیر تعداد همسایگان در الگوریتم بیش نمونه‌گیری SMOTE بر روی نتایج صحت مدل انجام شده است. در این آزمایش نیز فقط معیار صحت که نشان دهنده عملکرد کلی مدل می‌باشد، گزارش شده است. همانطور که در فصل دوم این تحقیق بیان شد، الگوریتم بیش نمونه‌گیری SMOTE برای تولید داده‌های مصنوعی از همسایگان هر داده واقعی استفاده می‌کند و با محاسبه اختلاف داده واقعی و همسایگان آن و افزودن مقدار اختلاف حاصل شده به همسایگان، داده‌های مصنوعی جدید را تولید می‌کند. از اینرو بررسی تعداد همسایگان می‌تواند نشان هد که تا چه میزان انتخاب درست همسایه می‌تواند در موفقیت مدل موثر واقع گردد. نتایج این آزمایش در نمودار (۴-۲) بیان شده است. در این آزمون نرخ داده‌های آموزشی برابر با ۹۰٪ کل داده‌ها تعیین شده است و ۲۰۰۰ نمونه داده مصنوعی برای هر پایگاه داده تولید شده است.



	1	2	3	4	5
UNSW-NB15	94.42	94.42	95.55	98.54	98.64
CICIDS2017	97.75	97.71	97.87	98.15	98.38
NSL-KDD	95.61	95.57	96.82	97.23	97.3

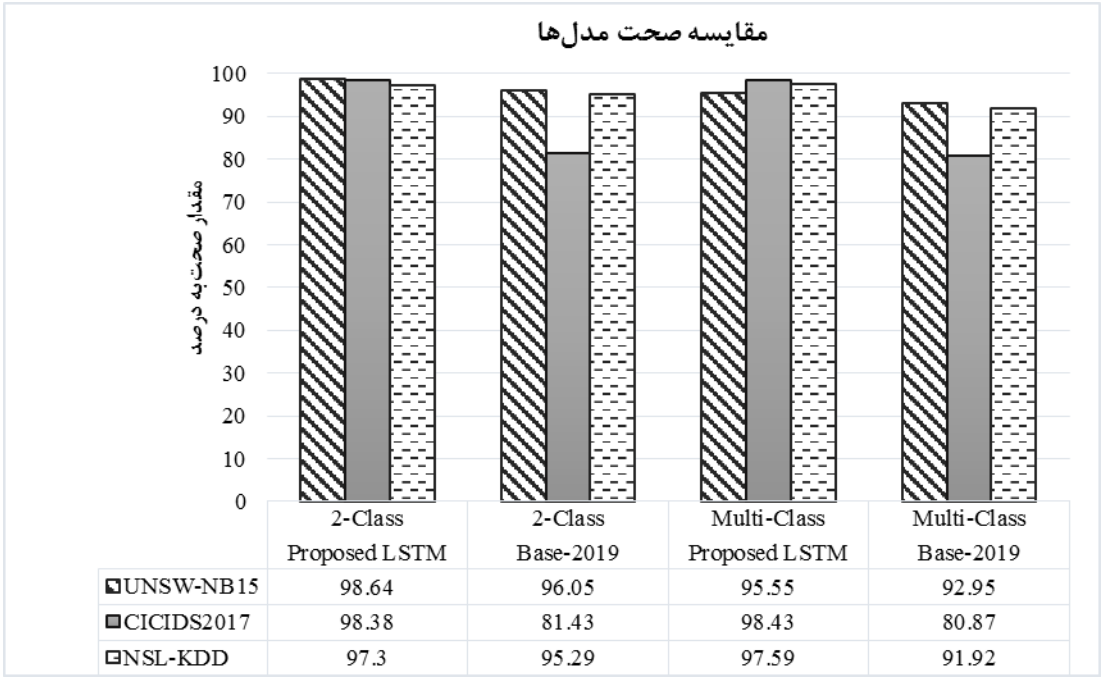
نمودار ۴-۲: بررسی تاثیر تعداد همسایگان در الگوریتم بیش نمونه‌گیری

نتایج این آزمایش نشان می‌دهد که در تعداد ۵ همسایه روش پیشنهادی توانسته است به بهترین صحت در هر سه پایگاه داده دست یابد. در پایگاه داده دوم تعداد همسایگان تاثیر چندانی بر نتایج مدل نداشته است. در واقع در برخی از پایگاه داده‌ها تراکم پایگاه داده باعث می‌شود که تعیین تعداد ۱ همسایه و یا ۵ همسایه نتایج تقریبا یکسانی داشته باشد. با توجه به این آزمایش تعداد همسایگان در کلیه آزمون‌های مقایسه‌ای برابر با ۵ در نظر گرفته می‌شود. با توجه به نتایج این دو آزمون

برای انجام آزمایش‌های مقایسه‌ای نرخ داده‌های آموزشی برابر با ۰,۹ و تعداد همسایگان برابر با ۵ تعیین می‌شود. همچنین نرخ تولید داده مصنوعی ۲۰۰۰ داده در تعیین شده است.

۶,۴ مقایسه مقدار صحت در هر سه پایگاه داده

در آزمایش‌های مقایسه‌ای، متغیرهای وابسته تحقیق با روش پایه مقایسه می‌شوند تا به این صورت بتوان دریافت در کدام معیار روش پیشنهاد شده توانسته است در مقایسه با روش پایه به موفقیت دست یابد. در این گروه از آزمایش‌ها نتایج هر یک از متغیرهای وابسته به تفکیک مقایسه شده‌اند و هر سه پایگاه داده در یک آزمون گزارش شده است تا به این صورت بتوان مقایسه راحت‌تری بین روش‌ها انجام داد. در اولین آزمایش مقایسه‌ای صحت روش پیشنهاد شده و روش پایه در هر سه پایگاه داده در دو حلت چندکلاسی و دو کلاسی با یکدیگر مقایسه شده که نتایج آن در نمودار (۴-۳) بیان شده است.



	2-Class Proposed LSTM	2-Class Base-2019	Multi-Class Proposed LSTM	Multi-Class Base-2019
UNSW-NB15	98.64	96.05	95.55	92.95
CICIDS2017	98.38	81.43	98.43	80.87
NSL-KDD	97.3	95.29	97.59	91.92

نمودار ۴-۳: مقایسه صحت مدل‌ها در سه پایگاه داده

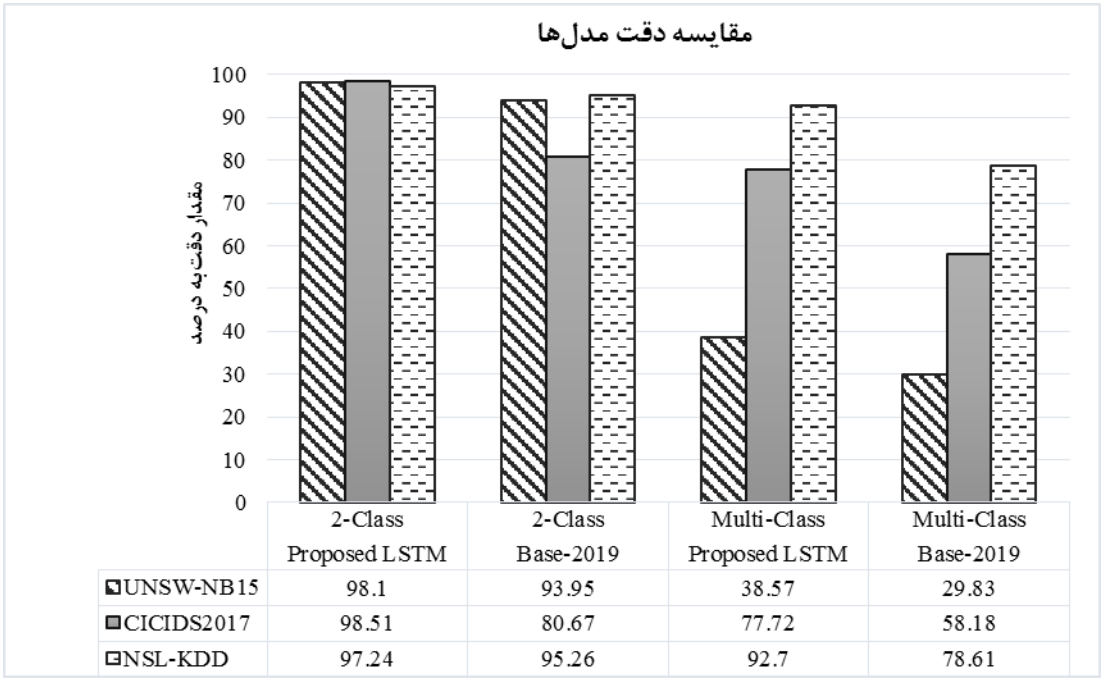
نتایج این آزمایش حاکی از آن است که در معیار صحت در هر سه پایگاه داده و در هر دو حالت چند کلاسی و دو کلاسی شبکه عصبی عمیق بازگشتی پیشنهادی توانسته است از الگوریتم k نزدیکترین همسایه استفاده شده در روش پایه بهتر عمل نماید. در واقع این نتایج نشان می‌دهد که استفاده از یادگیری عمیق در مقایسه با یادگیری ماشین می‌تواند در تشخیص حملات در شبکه موفق‌تر عمل نماید. در بین سه پایگاه داده در پایگاه CICIDS2017 موفقیت مدل در مقایسه با

روش پایه بیشتر بوده است. این مسئله بدلیل استفاده از مدل مخلوط گاوسی در روش پایه است. در روش پایه برای کاهش ابعاد به ۲ بعد از این تکنیک استفاده شده است. از آنجایی که این پایگاه داده در مقایسه با دو پایگاه دیگر بیشترین ابعاد یعنی ۸۰ ویژگی دارد، استفاده از هر تکنیک کاهش بعدی نمی‌تواند باعث موفقیت در آن شود، به این دلیل که دانش کافی باید از ۸۰ بعد دریافت شود. در نتیجه در روش پایه کاهش ابعاد از ۸۰ بعد به ۲ بعد در این پایگاه داده باعث کاهش نتایج عملکرد آن شده است. نتایج این نمودار را می‌توان به صورت زیر خلاصه نمود:

- در پایگاه داده UNSW-NB15 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۲,۵۹٪ و در حالت چند کلاسی ۲,۶٪ موفق تر از روش پایه عمل نماید.
- در پایگاه داده CICIDS2017 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱۶,۹۵٪ و در حالت چند کلاسی ۱۷,۵۶٪ موفق تر از روش پایه عمل نماید.
- در پایگاه داده NSL-KDD در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۲,۰۱٪ و در حالت چند کلاسی ۵,۶۷٪ موفق تر از روش پایه عمل نماید.

۷,۴ مقایسه مقدار دقت در هر سه پایگاه داده

دومین آزمون از این گروه به مقایسه دقت دو روش می‌پردازد. در این آزمایش نتایج دقت برای هر سه پایگاه داده برای دو کلاس و چند کلاس محاسبه شده و در نمودار (۴-۴) گزارش شده است.



	2-Class Proposed LSTM	2-Class Base-2019	Multi-Class Proposed LSTM	Multi-Class Base-2019
UNSW-NB15	98.1	93.95	38.57	29.83
CICIDS2017	98.51	80.67	77.72	58.18
NSL-KDD	97.24	95.26	92.7	78.61

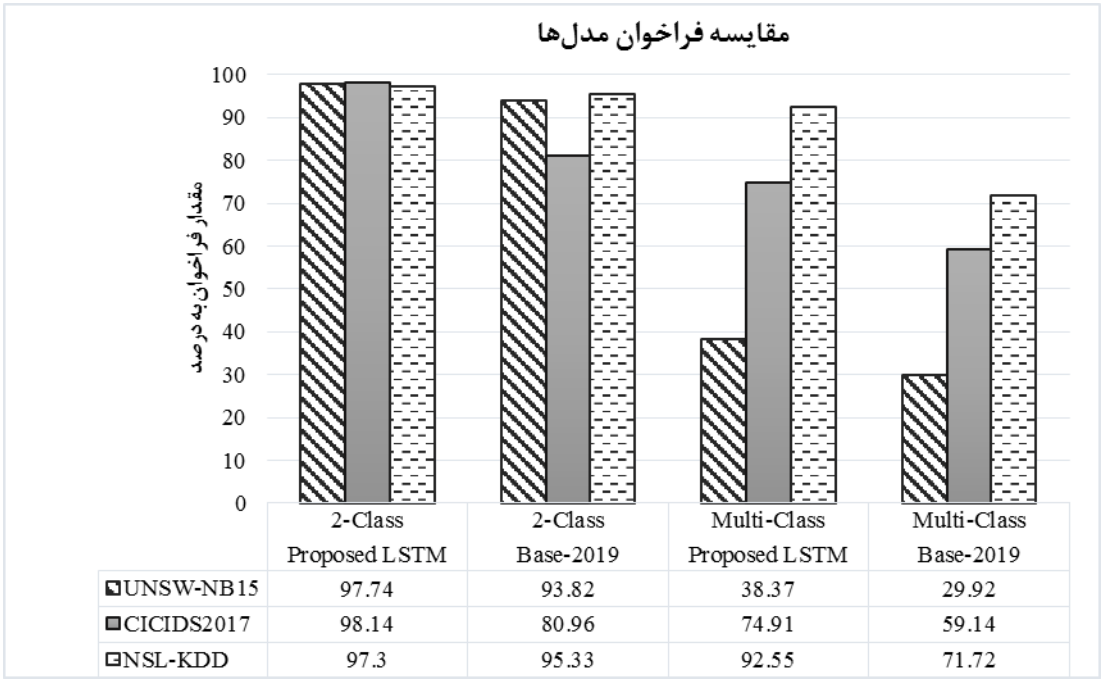
نمودار ۴-۴: مقایسه دقت مدل‌ها در سه پایگاه داده

نتایج این شکل نشان می‌دهد که استفاده از تکنیک بیش نمونه‌گیری و رفع مشکل عدم توازن داده‌ها به صورت نسبی و افزودن به کلاس حملات نادر توانسته است منجر به بهبود دقت کلی مدل شود. همانطور که در نتایج در هر سه پایگاه داده مشاهده می‌شود مقدار دقت در حال چند کلاسی به وضوح بیشتر از روش پایه است به این دلیل که روش پایه در برخی از حملات قادر به پیشگویی نبوده است و در نتیجه این مسئله باعث شده دقت کلی در کل کلاس‌ها کاهش یابد. در مقابل روش پیشنهادی بدلیل رفع مشکل حملات نادر در کلیه آن‌ها قادر به پیشگویی بوده است که این مسئله باعث افزایش دقت آن در کلیه کلاس‌ها شده است. نتایج نمودار فوق را می‌توان به صورت زیر خلاصه نمود:

- در پایگاه داده UNSW-NB15 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۴,۱۵٪ و در حالت چند کلاسی ۸,۷۴٪ در معیار دقت موفق تر از روش پایه عمل نماید.
- در پایگاه داده CICIDS2017 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱۷,۸۴٪ و در حالت چند کلاسی ۱۹,۵۴٪ در معیار دقت موفق تر از روش پایه عمل نماید.
- در پایگاه داده NSL-KDD در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱,۹۸٪ و در حالت چند کلاسی ۱۴,۰۹٪ در معیار دقت موفق تر از روش پایه عمل نماید.

۸,۴ مقایسه مقدار فراخوان در هر سه پایگاه داده

در این آزمایش معیار فراخون که در برخی منابع به آن حساسیت نیز گفته می‌شود، مقایسه شده است. در این آزمایش نتایج فراخوان برای هر سه پایگاه داده برای دو کلاس و چند کلاس محاسبه شده و در نمودار (۴-۵) گزارش شده است.



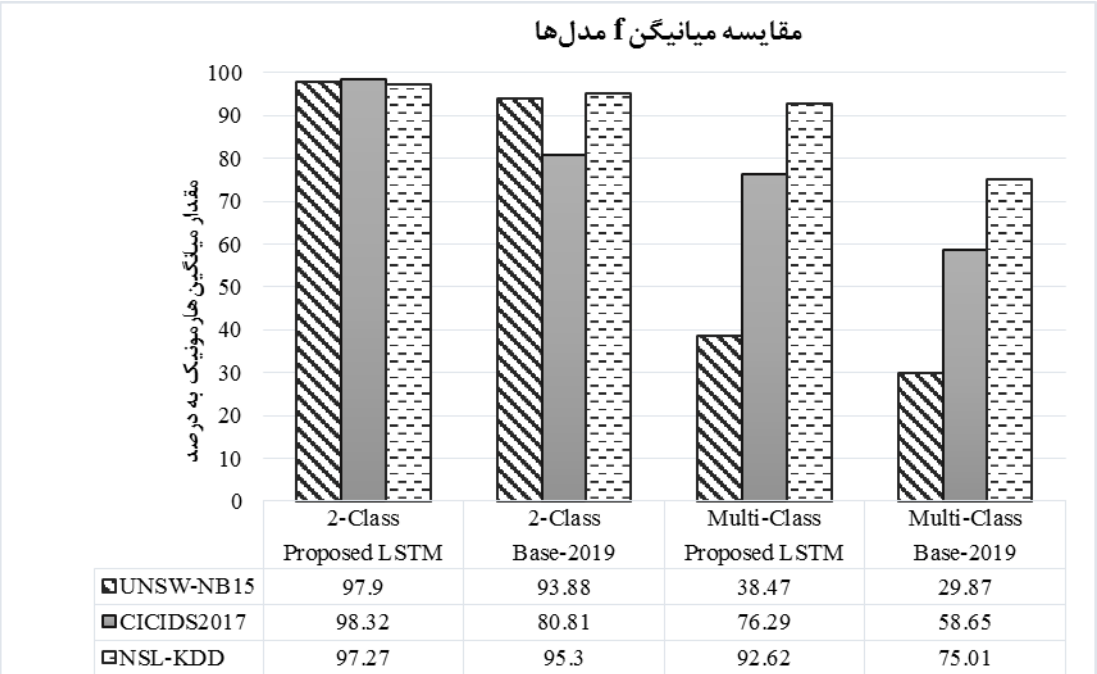
نمودار ۴-۵: مقایسه فراخوان مدل‌ها در سه پایگاه داده

کاهش ابعاد ویژگی اگرچه می‌تواند زمان مرحله یادگیری مدل را کاهش دهد اما اگر تکنیک کاهش بعد مناسبی انتخاب نشود باعث می‌شود دقت و صحت مدل به شدت کاهش یابد. این مورد در نتایج روش پایه مشاهده می‌شود. نتایج روش پایه نشان می‌دهد که ترکیب مدل مخلوط گاوسی و k نزدیکترین همسایه در همه پایگاه داده‌های حملات نمی‌تواند عملکرد موفقی از خود ارائه دهد. همانطور که در نتایج آزمایش‌ها مختلف نشان داد شد، روش پایه فقط در برخی از پایگاه داده‌ها می‌تواند عملکرد قابل قبولی از خود نشان دهد و در برخی از پایگاه‌های دیگر دقت و فراخوان چندان مناسبی ندارد؛ که این بدلیل مشکل عدم توازن داده‌ها و همچنین کاهش ابعاد ویژگی‌ها با استفاده از مدل مخلوط گاوسی است. در مقابل نتایج روش پیشنهادی که مبتنی بر تکنیک‌های یادگیری عمیق و شبکه عصبی بازگشتی LSTM می‌باشد، نشان می‌دهد که در روش ارائه شده ماهیت داده‌ها چندان تاثیری بر موفقیت مدل ندارد. در واقع یادگیری دقیق شبکه‌های بازگشتی LSTM باعث شده که این روش بتواند در هر پایگاه داده‌ای نتایج خوبی از خود ارائه دهد. استفاده از ترکیب تکنیک‌های بیش نمونه‌گیری و کم نمونه‌گیری نیز باعث شده دقت و فراخوان در هر کلاس حملات افزایش یابد و در نتیجه دقت کلی و فراخوان کلی مدل نیز افزوده شود. نتایج نمودار بالا را می‌توان به صورت زیر خلاصه نمود:

- در پایگاه داده UNSW-NB15 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۳,۹۲٪ و در حالت چند کلاسی ۸,۴۵٪ در معیار فراخوان موفق تر از روش پایه عمل نماید.
- در پایگاه داده CICIDS2017 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱۷,۱۸٪ و در حالت چند کلاسی ۱۵,۷۷٪ در معیار فراخوان موفق تر از روش پایه عمل نماید.
- در پایگاه داده NSL-KDD در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱,۹۷٪ و در حالت چند کلاسی ۲۰,۸۳٪ در معیار فراخوان موفق تر از روش پایه عمل نماید.

۹,۴ مقایسه مقدار میانگین f در هر سه پایگاه داده

آخرین گروه از آزمون‌ها نتایج میانگین هارمونیک f را برای روش پیشنهادی و پایه مقایسه می‌کند و گزارش می‌دهد. این معیار میانگین دو معیار دقت و فراخوانی است و مشابه با معیار صحت عملکرد کلی مدل را می‌توان از آن برداشت نمود. نتایج این معیار در هر سه پایگاه داده در نمودار (۴-۶) گزارش شده است.



	2-Class Proposed LSTM	2-Class Base-2019	Multi-Class Proposed LSTM	Multi-Class Base-2019
UNSW-NB15	97.9	93.88	38.47	29.87
CICIDS2017	98.32	80.81	76.29	58.65
NSL-KDD	97.27	95.3	92.62	75.01

نمودار ۴-۶: مقایسه میانگین هارمونیک مدل‌ها در سه پایگاه داده

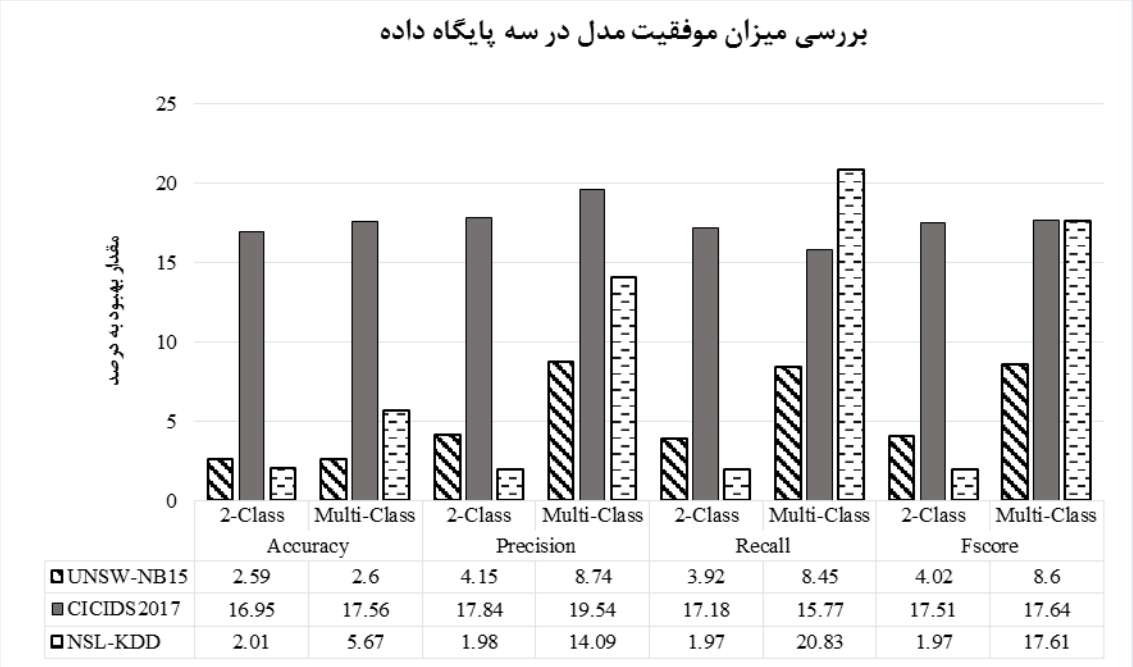
نتایج این معیار نیز مشابه با دیگر معیارها بیانگر این حقیقت است که شبکه‌های عصبی عمیق بدلیل ساختار سلسله مراتبی که دارند و همچنین معماری‌های آن‌ها در حوزه‌های مختلف همچون حوزه مورد بررسی در این تحقیق در مقایسه با روش‌های

یادگیری ماشین عملکرد بسیار موفق‌تری دارند. در روش پیشنهادی از یک شبکه عصبی LSTM با ۵ لایه مختلف برای پیشگویی انواع حملات استفاده شده است. این روش اگرچه در مرحله یادگیری زمان‌بر می‌باشد اما از آنجایی که این مرحله از روش به صورت برون خط انجام می‌شود، در نتیجه فاقد اهمیت می‌باشد و می‌توان از آن در راستای رسیدن به دقت و صحت بالاتر چشم پوشی نمود. نتایج این تحقیق نشان داد که رفع مشکل عدم توازن داده‌ها و استفاده از یادگیری عمیق در زمینه تشخیص حملات شبکه در انواع مختلف پایگاه داده‌ها می‌تواند عملکرد موفقی داشته باشد. نتایج نمودار فوق به صورت زیر خلاصه می‌شود:

- در پایگاه داده UNSW-NB15 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۴,۰۲٪ و در حالت چند کلاسی ۸,۶٪ در معیار میانگین هارمونیک موفق تر از روش پایه عمل نماید.
- در پایگاه داده CICIDS2017 در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱۷,۵۱٪ و در حالت چند کلاسی ۱۷,۶۴٪ در معیار میانگین هارمونیک موفق تر از روش پایه عمل نماید.
- در پایگاه داده NSL-KDD در حالت دو کلاسی شبکه عصبی عمیق پیشنهادی توانسته است ۱,۹۷٪ و در حالت چند کلاسی ۱۷,۶۱٪ در معیار میانگین هارمونیک موفق تر از روش پایه عمل نماید.

۱۰,۴ بررسی اهداف تحقیق

در این بخش پس از بررسی ابعاد مختلف روش پیشنهادی، می‌توان اهداف تحقیق را با بررسی وضعیت روش پیشنهادی در مقایسه با روش پایه بررسی نمود. در نمودار (۴-۷) خلاصه نتایج این فصل از تحقیق بیان شده است. در این نمودار اختلاف مقادیر شبکه عصبی بازگشتی پیشنهادی در مقایسه با روش پایه برای هر یک از معیارها در هر سه پایگاه داده محاسبه شده است. پس از بررسی نتایج کلی این فصل اهداف تحقیق مجددا بیان شده و بررسی می‌شوند تا به این صورت، بتوان دریافت کدام یک از اهداف حاصل شده است.



نمودار ۴-۷: بررسی کلی نتایج راهکار پیشنهاد شده

نتایج نمودار (۴-۷) که میزان موفقیت مدل پیشنهاد شده در مقایسه با پایه را نشان می‌دهد، حاکی از آن است که در کلیه معیارها و در هر سه پایگاه داده شبکه عصبی بازگشتی عمیق **LSTM** در مقایسه با الگوریتم یادگیری ماشین **k** نزدیکترین همسایه در تشخیص حملات شبکه بسیار موفق‌تر عمل نموده است.

۵. نتیجه گیری

۱،۵ نتیجه‌گیری به تفکیک فرضیات تحقیق

از سه پایگاه داده مختلف با نام‌های UNSW-NB15، NSL-KDD و CICIDS2017 استفاده گردید. برای ارزیابی روش‌ها ۶ گروه آزمایش انجام شد که دو گروه اول برای بررسی تاثیرات دو متغیر وابسته تحقیق بر روی متغیر مستقل تحقیق انجام شدند و چهار گروه از آزمایش‌ها در راستای مقایسه نتایج روش پیشنهادی و پایه انجام شدند. بررسی متغیرهای مستقل تحقیق نشان داد تعداد ۵ همسایه برای الگوریتم SMOTE و نرخ داده‌های ۰,۹ می‌تواند باعث شود بهترین نتایج حاصل شود. نتایج آزمایش‌های مقایسه‌ای نیز نشان داد که روش‌های یادگیری عمیق در مقایسه با روش‌های یادگیری ماشین در حوزه امنیت شبکه‌ها موفق‌تر هستند. نتایج نشان داد انتخاب تکنیک کاهش ابعاد نامناسب مثل مدل مخلوط گاوسی استفاده شده در روش پایه می‌تواند باعث شود نتایج روش‌ها در برخی از پایگاه داده‌ها به شدت کاهش یابد. در مقابل عدم کاهش ابعاد و استفاده از شبکه‌های عصبی عمیق بازگشتی می‌تواند در انواع مختلف داده‌ها موفق عمل نماید و مدلی جامع باشد. از طرف

دیگر عدم توجه به حملات نادر به شدت باعث کاهش نتایج روش‌های یادگیری ماشین سنتی مثل الگوریتم k نزدیکترین همسایه در روش پایه می‌شود و در مقابل رفع مشکلات حملات نادر در روش پیشنهادی به خوبی می‌تواند دقت و فراخوان مدل را در هر کلاس حمله و در نهایت درکل کلاس‌ها افزایش دهد.

در کنار نقاط قوت مدل پیشنهادی می‌توان محدودیت آن را به صورت زیر خلاصه نمود:

۱) زمانبر بودن مرحله یادگیری شبکه عصبی عمیق بازگشتی از جمله محدودیت‌های روش پیشنهادی می‌باشد. هر چه تعداد لایه‌های این شبکه افزایش یابد و تعداد واحدهای پنهان آن نیز افزایش یابد زمان فرآیند یادگیری آن بیشتر شده و پیچیدگی‌های محاسباتی آن افزایش می‌یابد.

۲,۵ *کارهای آینده*

با توجه به مطالعات انجام شده و تحقیق ارائه شده و محدودیت‌های مطرح در این تحقیق، کارهای آتی پیشنهادی به شرح زیر می‌باشند:

۱) استفاده از شبکه‌های عصبی بازگشتی دیگر مثل معماری بازگشتی BiLSTM[۴۳] و ترکیب معماری این تحقیق و معماری BiLSTM با استفاده از تکنیک‌هایی مثل رای‌گیری اکثریت می‌تواند باعث شود نتایج مدل افزایش داشته باشد.

۲) استفاده از یک تکنیک انتخاب ویژگی مناسب برای انتخاب زیر مجموعه بهینه از ویژگی‌ها می‌تواند پیچیدگی‌های محاسباتی و زمانی مدل پیشنهاد شده را بهبود ببخشد.

۳) ترکیب دو نوع معماری عمیق مثل شبکه‌های عمیق کانولوشن[۴۴] و شبکه‌های عمیق بازگشتی نیز از دیگر روش‌هایی است که انتظار می‌رود باعث بهبود نتایج مدل شود.

---

[43] Bidirectional Recurrent Neural Networks
[44] Convolutional Neural Networks

فهرست منابع

Aldweesh, A., Derhab, A., & Emam, A. Z. (۲۰۲۰). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues. Knowledge-based systems, ۱۸۹.

Al-Hadhrami, Y., & Hussain, F. K. (۲۰۲۰). Real time dataset generation framework for intrusion detection systems in IoT. Future Generation Computer Systems, ۱۰۸, .٤٢٣-٤١٤.

Yin, C., Zhu, Y., Fei, J., & He, X. (۲۰۱۷). A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access, ٥, ٢١٩٦١-٢١٩٥٤.

Hassan, M. M., Gumaei, A., Alsanad, A., Alrubaian, M., & Fortino, G. (2020). A hybrid deep learning model for efficient intrusion detection in big data environment. Information Sciences, 513, 386-396 .

Shenfield, A., Day, D., & Ayesh, A. (2018). Intelligent intrusion detection systems using artificial neural networks. *ICT Express*, *4*(2), 95-99.

Kabir, E., Hu, J., Wang, H., & Zhuo, G. (2018). A novel statistical technique for intrusion detection systems. Future Generation Computer Systems, 79, 303-318 .

Papamartzivanos, D., Mármol, F. G., & Kambourakis, G. (2018). Dendron: Genetic trees driven rule induction for network intrusion detection systems. *Future Generation Computer Systems*, *79*, 558-574.

Karatas, G., & Sahingoz, O. K. (2018). *Neural network based intrusion detection systems with different training functions.* Paper presented at the 2018 6th International Symposium on Digital Forensic and Security (ISDFS).

Ahmad, I., Basheri, M., Iqbal, M. J., & Rahim, A. (2018). Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. *Ieee Access*, *6*, 33789-33795.

Marino, D. L., Wickramasinghe, C. S., & Manic, M. (2018). *An adversarial approach for explainable ai in intrusion detection systems.* Paper presented at the IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society.

Sahani, R., Rout, C., Badajena, J. C., Jena, A. K & ,.Das, H. (2018). Classification of intrusion detection using data mining techniques. In Progress in computing, analytics and networking (pp. 753-764): Springer.

Aljawarneh, S., Yassein, M. B., & Aljundi, M. (2019). An enhanced J48 classification algorithm for the anomaly intrusion detection systems. Cluster Computing, 22(5), 10549-10565 .

Rawat, S., Srinivasan, A., Vinayakumar ,R., & Ghosh, U. (2019). Intrusion detection systems using classical machine learning techniques versus integrated unsupervised feature learning and deep neural network. Internet Technology Letters .

Benmessahel, I., Xie, K., Chellal, M., & Semong, T. (2019). A new evolutionary neural networks based on intrusion detection systems using locust swarm optimization. *Evolutionary Intelligence*, *12*(2), 131-146.

Gu, J., Wang, L., Wang, H., & Wang, S. (2019). A novel approach to intrusion detection using SVM ensemble with feature augmentation. *computers security*, *86*, 53-62.

Ahmad, T., & Aziz, M. N. (2019). Data preprocessing and feature selection for machine learning intrusion detection systems. ICIC Express Lett, 13(2), 93-101 .

Choi, H., Kim, M., Lee, G., & Kim, W. (2019). Unsupervised learning approach for network intrusion detection system using autoencoders. *The Journal of Supercomputing*, *75*(9), 5597-5621.

Abdulhammed, R., Musafer, H., Alessa, A., Faezipour, M., & Abuzneid, A. (2019). Features dimensionality reduction approaches for machine learning based network intrusion detection. *Electronics*, *8*(3).

Long, C., Zhang, Y., Wei, J., Wan, W., Zhao, J., & Du, G. (2019). *A Hybrid Intrusion Detection Algorithm Based on Gaussian Mixture Model and Nearest Neighbors.* Paper presented at the 2019 IEEE 44th Conference on Local Computer Networks (LCN).

Nguyen ,M. T., & Kim, K. (2020). Genetic convolutional neural network for intrusion detection systems. *Future Generation Computer Systems*, *113*, 418-427.

Hassan, M. M., Gumaei, A., Alsanad, A., Alrubaian, M., & Fortino, G. (2020). A hybrid deep learning model for efficient intrusion detection in big data environment. *Information Sciences*, *513*, 386-396.

Bhattacharya, S., Kaluri, R., Singh, S., Alazab, M., & Tariq, U. (2020). A Novel PCA-Firefly based XGBoost classification model for Intrusion Detection in Networks using GPU. *Electronics.*

Ravipati, R. D., & Abualkibash, M. (2019). Intrusion Detection System Classification Using Different Machine Learning Algorithms on KDD-99 and NSL-KDD Datasets-A Review Paper. *International Journal of Computer Science Information Technology Vol*, *11*(3).

Hindy, H., Brosset, D., Bayne, E., Seeam, A. K., Tachtatzis, C., Atkinson, R., & Bellekens, X. (2020). A taxonomy of network threats and the effect of current datasets on intrusion detection systems. *Ieee Access*, *8*, 104650-104675.

Moustafa, N., & Slay, J. (2015). *UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set).* Paper presented at the 2015 military communications and information systems conference (MilCIS).

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). *Toward generating a new intrusion detection dataset and intrusion traffic characterization.* Paper presented at the ICISSp.

detailed analysis of the KDD CUP 99 data set. Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). *A* Paper presented at the 2009 IEEE symposium on computational intelligence for security and defense applications.