Entrega 1 - Proyecto final de Deep Learning: Detección de ciberataques

Santiago Ríos Guiral
Facultad de Ingeniería - Universidad de Antioquia
Medellín, Colombia
santiago.riosg@udea.edu.co

I. Contexto de aplicación

La seguridad de las redes de computadoras es una necesidad fundamental que se debe satisfacer en el contexto actual de la evolución de las redes. En los últimos años, se ha producido un aumento en el número de ciberataques, los cuales son más frecuentes, complejos y de mayor volumen. Reportes de seguridad de Microsoft e IBM presentan un aumento de más del 40% en ciberataques entre los años 2021-2022 [1], [2]. Estas alarmantes cifras demuestran la necesidad de desarrollar nuevas estrategias de detección. En aras de mejorar la seguridad de la red se requiere la implementación de nuevas tecnologías como las redes definidas por software (SDN) y las redes autogestionables. El objetivo es desplegar redes autónomas que predicen cambios y se adapten al comportamiento de la red [3]. Esta es una característica deseable en el campo de la ciberseguridad para defenderse ante la presencia de nuevos ciberataques.

La complejidad de la red y la naturaleza dinámica de sus procesos subyacentes hace de los algoritmos de Machine Learning (ML) una herramienta natural para detectar, diagnosticar y mitigar intrusiones [4]. A través de algoritmos de ML es posible crear sistemas de detección de intrusiones (IDS). La implementación de ML en el desarrollo de aplicaciones permite incrementar la detección de ataques sofisticados, reaccionar más rápido ante una intrusión, mejorar la escalabilidad de la red, y permite automatizar las acciones de seguridad. En la revisión del estado del arte, se observa un incremento en el desarrollo de aplicaciones de seguridad que utilizan ML, incluyendo algoritmos de Deep Learning (DL). Recientemente, ML y DL son implementados para mejorar la seguridad de la red [5]. Consecuentemente, es fundamental la implementación de ML y DL en las redes de computadores con el objetivo de mejorar la eficacia en la detección y mitigación de ataques.

II. Objetivo de Machine Learning

Con el propósito de mejorar el rendimiento de los sistemas de detección de intrusiones (IDS), se ha recurrido a la implementación de algoritmos de Machine Learning (ML) y Deep Learning (DL). Estos sistemas tienen la función de clasificar el tráfico de red, estableciendo clases para el tráfico anómalo y benigno. La inclusión de algoritmos de ML o DP busca inferir sobre el tráfico que llega a una red, permitiendo así la predicción de su clase. En caso de tratarse de tráfico normal, el sistema permite su libre paso a través de la red hasta su destino, mientras que, en presencia de tráfico malicioso, se busca detectar el ataque y bloquear su acceso a la posible víctima. En este sentido, para el proyecto del curso se pretende desarrollar un clasificador haciendo uso de una base de datos con tráfico de red referente en el área de ciberseguridad. Además, se espera crear un modelo de ML que permita la clasificación del tráfico mediante diferentes características extraídas (características estadísticas, propias del tráfico, entre otras). La base de datos se utilizará para el entrenamiento y evaluación del modelo. En resumen, el modelo de ML o DP puede ser implementado como parte fundamental de un sistema de ciberseguridad robusto capaz de detectar diferentes tipos de ataques.

III. Dataset

En este proyecto se va a utilizar la base de datos *Intrusion Detection Evaluation Dataset* (CIC-IDS2017) [6]. El desarrollo de una base de datos confiable de tráfico de red es fundamental para la evaluación y mejora de los sistemas de ciberseguridad. En particular, los sistemas de detección de intrusiones (IDS) son herramientas críticas para contrarrestar los ataques cada vez más sofisticados y frecuentes. En este contexto, la base de datos CIC-IDS2017 se presenta como una solución valiosa para la comunidad de la ciberseguridad, ya que proporciona una base de datos confiable para la evaluación de IDS y otros sistemas de defensa.

CIC-IDS2017 contiene tráfico benigno y anómalo, los cuales se asemejan a paquetes reales encontrados en la redes de computadores. Esta base de datos organiza el tráfico en archivos PCAP. Asimismo, esta base de datos también se encuentra en formato CSV, donde se utilizó el extractor CICFlowMeter [7] para generar estos archivos. El extractor captura los paquetes del archivo PCAP y genera flujos de paquetes los cuales se encuentran debidamente etiquetados en un archivo CSV.

El periodo de captura de datos comenzó el lunes 3 de julio de 2017 y finalizó el viernes 7 de julio de 2017, lo que supone un total de 5 días. Incluye tráfico normal y los ataques implementados incluyen FTP de fuerza bruta, SSH de fuerza bruta, DoS, Heartbleed, ataque web, infiltración, botnet y DDoS.

En el proyecto de detección de ataques se va a utilizar la versión de CIC-IDS2017 organizada por el extractor CICFlowMeter [7]. Cada flujo de tráfico esta etiquetado, ya sea

como normal o ataque e incluye 84 características por cada flujo generado. La base de datos esta organizada en 8 archivos. A continuación se muestra su estructura.

- *Monday-WorkingHours.pcap_ISCX.csv:* 268.6MB - 529919 flujos.
- *Tuesday-WorkingHours.pcap_ISCX.csv:* 174.7MB - 445910 flujos.
- *Wednesday-workingHours.pcap_ISCX.csv:* 285.6MB - 692704 flujos.
- *Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv:* 92MB - 170367 flujos.
- *Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv:* 108.7MB - 288603 flujos.
- *Friday-WorkingHours-Morning.pcap_ISCX.csv:* 75.4MB - 191034 flujos.
- *Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv:* 101.9MB - 286468 flujos.
- *Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv:* 96.1MB - 225746 flujos.

IV. Métricas de desempeño

Para los modelos de clasificación implementados con ML y DL se puede utilizar la matriz de confusión como se presenta en la tabla I. Esta ofrece una gama de medidas de desempeño útiles en la evaluación del modelo.

TABLE I
Matriz de Confusión

Matriz de Confusión		Valores predichos	
		Benigno	Ataque
Clase	Benigno	TP	FN
	Ataque	FP	TN

TP: Positivos verdaderos.
FP: Falsos verdaderos.
FN: Falsos negativos.
TN: Positivos negativos.

Para evaluar la clasificación se utiliza la ecuación 1 que describe la precisión (*accuracy*) en la discriminación del tráfico de red. Esta ecuación indica el porcentaje de registros que fueron clasificados correctamente con respecto a todas las muestras de entrada.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{1}$$

También existen medidas que permiten evaluar la clasificación con respecto a las clases de salida. Se tiene la medida de precisión (ecuación 2) que determina el porcentaje de clasificación correcta de una determinada clase.

$$P = \frac{TP}{TPFP} \tag{2}$$

La sintonía (recall) es una medida que calcula la proporción de predicciones correctas de una clase sobre todas las posibilidades positivas o negativas. La ecuación 3 presenta el metodo para determinar la sintonía.

$$R = \frac{TP}{TP + FN} \tag{3}$$

Finalmente, se cuenta con la medida F1, la cual corresponde al promedio armónico de la medida de precisión y de sintonía para las clases de salida. En la ecuación 4 se muestra la forma como se hace su cálculo.

$$F1 = 2 \times \frac{P \times R}{P + R} \tag{4}$$

En el contexto de ciberseguridad se proyecta obtener un rendimiento alto del sistema de defensa (indicador de gestión de la defensa). Se espera una detección rápida de los ciberataques, así como disminuir el nivel de afectación de la red ante la ocurrencia de estos (actuación del proveedor de la defensa). También se espera dar un grado de confianza a los usuarios de las redes de computadores donde se está desarrollando las medidas de protección de esa información privada (satisfacción del usuario o cliente).

V. Resultados previos

Esta sección presenta un estado del arte con implementaciones de modelos de DL para detectar ciberataques. Li et al [8] comparan diferentes modelos de DL con diferentes ventanas de tiempo que agregan los paquetes de red en flujos. Utilizan CNN y RNN donde logran obtener una precisión de 88.6% y 91.5% respectivamente. En [9] los autores proponen DeepIDS. El sistema utiliza una red neuronal recurrente (GRU-RNN) donde se obtiene una precisión de 90% en la clasificación correcta de tráfico. Esta red neuronal se despliega en un sistema de detección de intrusiones para detectar tráfico anómalo. En [10], los autores utilizan redes neuronales recurrentes (RNN) para detectar ataques DDoS. El estudio presenta una red de memoria a corto plazo (LSTM) para obtener información de características temporales y espaciales del tráfico de red. Los autores obtienen una precisión de 99.9% en la clasificación de tráfico. En [11], Haider et al. proponen una red neuronal híbrida como la construcción de 2 redes neuronales, ya sean redes neuronales convolucionales (CNN) o redes neuronales recurrentes (RNN). El sistema puede utilizar 2 CNN o 2 RNN donde las salidas se combinan como entradas de una tercera red neuronal, la cual define una salida binaria. De esta forma se clasifica el tráfico como benigno o como anómalo de tipo de Denegación distribuida de servicios (DDoS). Los autores afirman lograr una precisión de 99.45% para un montaje con CNN y del 98.68% para un montaje con RNN. En [12], Wang et al. proponen un sistema de defensa el cual implementa una CNN. Para evaluar el rendimiento, los autores modifican la arquitectura cambiando el número de capas convolucionales y/o cambiando el número de capas totalmente conectadas. Se logra obtener una precisión entre el 98.7% y 99%. En [13], los autores presentan un algoritmo basado en RNN. El modelo selecciona las mejoras características de los paquetes de red de acuerdo al modelo OCSA. Posteriormente la RNN utiliza el algoritmo Propagación hacia atrás con retraso de tiempo (BPTT) para obtener una clasificación binaria. El modelo logra una precisión del 98.1% para clasificar entre paquetes normales y ataques. Finalmente, en [14] presentan una red neuronal clásica (NN)implementada en nuevas tecnologías

de red como lo son los conmutadores programables. En este sistema se logró una precisión del 96.8%.

Por lo tanto, a través de estos artículos, y otros más que se encuentra en la literatura, se materializa la importancia de aplicar modelos de ML y DL en el área de la ciberseguridad. En el proyecto final del curso de Deep Learning se espera implementar un modelo capaz de detectar la ocurrencia de ciberataques de tal forma que se pueda brindar la seguridad necesaria a una red de computadores.

REFERENCES

[1] IBM, "Cost of a data breach 2022: A million-dollar race to detect and respond," https://www.ibm.com/reports/data-breach.

[2] Microsoft, "Microsoft digital defense report 2022," https://www.microsoft.com/en-us/security/business/microsoft-digital-defense-report-2022.

[3] A. S. Jacobs, R. J. Pfitscher, R. A. Ferreira, and L. Z. Granville, "Refining network intents for self-driving networks," in *Proceedings of the Afternoon Workshop on Self-Driving Networks*, 2018, pp. 15–21.

[4] N. Feamster and J. Rexford, "Why (and how) networks should run themselves," *arXiv preprint arXiv:1710.11583*, 2017.

[5] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on sdn based network intrusion detection system using machine learning approaches," *Peer-to-Peer Networking and Applications*, vol. 12, pp. 493–501, 2019.

[6] "Canadian institute for cybersecurity. intrusion detection evaluation dataset (cic-ids2017)," https://www.unb.ca/cic/datasets/ids-2017.html.

[7] "Canadian institute for cybersecurity. cicflowmeter (formerly iscxflowmeter)," https://www.unb.ca/cic/research/applications.html#CICFlowMeter.

[8] C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang, X. Li, and L. Gong, "Detection and defense of ddos attack–based on deep learning in openflow-based sdn," *International Journal of Communication Systems*, vol. 31, no. 5, p. e3497, 2018.

[9] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, M. Ghogho, and F. El Moussa, "Deepids: Deep learning approach for intrusion detection in software defined networking," *Electronics*, vol. 9, no. 9, p. 1533, 2020.

[10] X. Liang and T. Znati, "A long short-term memory enabled framework for ddos detection," in *2019 IEEE global communications conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[11] S. Haider, A. Akhunzada, I. Mustafa, T. B. Patel, A. Fernandez, K.-K. R. Choo, and J. Iqbal, "A deep cnn ensemble framework for efficient ddos attack detection in software defined networks," *Ieee Access*, vol. 8, pp. 53 972–53 983, 2020.

[12] L. Wang and Y. Liu, "A ddos attack detection method based on information entropy and deep learning in sdn," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 1. IEEE, 2020, pp. 1084–1088.

[13] R. SaiSindhuTheja and G. K. Shyam, "An efficient metaheuristic algorithm based feature selection and recurrent neural network for dos attack detection in cloud computing environment," *Applied Soft Computing*, vol. 100, p. 106997, 2021.

[14] F. Paolucci, L. De Marinis, P. Castoldi, and F. Cugini, "Demonstration of p4 neural network switch," in *2021 Optical Fiber Communications Conference and Exhibition (OFC)*. IEEE, 2021, pp. 1–3.