UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

FACULTAD DE INGENIERÍAS Y ARQUITECTURA

CARRERA DE TELECOMUNICACIONES

Propuesta de algoritmo de control de congestión en redes de paquetes usando técnicas de machine learning

Trabajo de integración curricular previo a la obtención del título de:

INGENIERO EN TELECOMUNICACIONES

Autor: Collahuazo Balcázar, Pablo Andrés

Jiménez Soto, Francisco Xavier

Director: Ludeña González, Patricia Jeanneth

LOJA

2024

2024

Aprobación del director del Trabajo de Integración Curricular

Loja, día de mes de año

Doctor
Francisco Alberto Sandoval Noreña
Director de la carrera de Telecomunicaciones
Ciudad.-

De mi consideración:

Me permito comunicar que, en calidad de director del presente Trabajo de Integración Curricular denominado: Propuesta de algoritmo de control de congestión en redes de paquetes usando técnicas de machine learning realizado por Pablo Andrés Collahuazo Balcazar y Francisco Xavier Jiménez Soto ha sido orientado y revisado durante su ejecución, así mismo ha sido verificado a través de la herramienta de similitud académica institucional, y cuenta con un porcentaje de coincidencia aceptable. En virtud de ello, y por considerar que el mismo cumple con todos los parámetros establecidos por la Universidad, doy mi aprobación a fin de continuar con el proceso académico correspondiente.

Particular que comunico para los fines pertinentes.

Atentamente,

Director: Patricia Jeanneth Ludeña González, PhD
C.I.: 1103997530
Correo electrónico: pjludena@utpl.edu.ec

Declaración de autoría y cesión de derechos

Yo, Pablo Andrés Collahuazo Balcázar y Francisco Xavier Jiménez Soto, declaro y acepto en forma expresa lo siguiente:

Ser autor (a) del Trabajo de Integración Curricular denominado: Propuesta de algoritmo de algoritmo de control de congestión en redes de paquetes usando técnicas de machine learning, de la carrera de Telecomunicaciones, específicamente de los contenidos comprendidos en: Introducción, Capítulo 1. Antecedentes, Capítulo 2. Marco Teórico, Capítulo 3. Metodología, Capítulo 4. Análisis Comparativo de Resultados Técnicas de Machine Learning, Conclusiones y Recomendaciones, siendo Patricia Jeanneth Ludeña González, director(a) del presente trabajo; también declaro que la presente investigación no vulnera derechos de terceros ni utiliza fraudulentamente obras preexistentes. Además, ratifico que las ideas, criterios, opiniones, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad. Eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones judiciales o administrativas, en relación a la propiedad intelectual de este trabajo.

Que la presente obra, producto de mis actividades académicas y de investigación, forma parte del patrimonio de la Universidad Técnica Particular de Loja, de conformidad con el artículo 20, literal j), de la Ley Orgánica de Educación Superior; y, artículo 91 del Estatuto Orgánico de la UTPL, que establece: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la Universidad”, en tal virtud, cedo a favor de la Universidad Técnica Particular de Loja la titularidad de los derechos patrimoniales que me corresponden en calidad de autor/a, de forma incondicional, completa, exclusiva y por todo el tiempo de su vigencia.

La Universidad Técnica Particular de Loja queda facultada para ingresar el presente trabajo al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública, en cumplimiento del artículo 144 de la Ley Orgánica de Educación Superior.

.....

Autor: Pablo Andrés Collahuazo Balcázar

C.I.: 1104673338

Correo electrónico: pacollahuazo@utpl.edu.ec

.....

Autor: Francisco Xavier Jiménez Soto

C.I.: 1104292881

Correo electrónico: fxjimenez4@utpl.edu.ec

Dedicatoria

Dedico este trabajo de titulación con amor y gratitud a mi familia, mi pilar y fuente constante de inspiración. A mis padres, Ricardo y Alicia, a mi hermana Arleth y hermano Ricardo, siendo mis mayores fuentes de motivación y esfuerzo que han hecho posible este logro.

Con profunda gratitud, dedico este reconocimiento a mis padres, Francisco y Natalia, cuyo amor y apoyo incondicional han sido la piedra angular de mi formación universitaria. Su presencia constante y aliento me han moldeado tanto en mi desarrollo profesional como en mi crecimiento personal.

Agradecimiento

Quisiera expresar mi más sincero agradecimiento a mi familia, cuyo apoyo ha sido fundamental en esta etapa académica. A mis padres, Ricardo y Alicia, gracias por su amor incondicional y por ser mi constante fuente de inspiración. A mi hermana Arleth y a mi hermano Ricardo, gracias por su cariño, apoyo y compañía durante todo este tiempo. A mis queridas tías, Nilda y Olga, les agradezco sinceramente por su apoyo y afecto.

También quiero extender mi agradecimiento a nuestra directora de tesis Patricia Ludeña, por su sabiduría, paciencia y experiencia, que han sido fundamentales en la realización de este trabajo. De igual forma agradecer a Ruth Reategui, nuestra jurado, por dedicar tiempo a nuestro trabajo y por sus valiosos comentarios y sugerencias que han sido esenciales para fortalecer este proyecto.

Extiendo mi más profundo agradecimiento a todos mis familiares, cuyo apoyo incondicional me permitió estudiar en esta prestigiosa institución. Su confianza y aliento han sido fundamentales para alcanzar mi sueño de convertirme en un profesional, contribuyendo así al desarrollo de nuestro país. También quiero expresar mi sincera gratitud a todos los docentes que han sido pieza clave en mi formación académica, brindándome las herramientas y conocimientos necesarios para culminar con éxito mi trabajo de titulación.

Quiero expresar mi especial agradecimiento a nuestra tutora de tesis, la Magíster Patricia Ludeña. Su orientación experta y su dedicación semanal han sido cruciales en la evolución y éxito de nuestro proyecto. Un sincero agradecimiento a nuestra jurado, Ruth Reátegui, cuyos valiosos consejos y perspectivas enriquecieron y mejoraron significativamente nuestro trabajo de titulación. Finalmente, extiendo mi gratitud a todos los familiares que me han brindado su apoyo inquebrantable a lo largo de mi trayectoria universitaria. Cada uno de ustedes ha jugado un papel vital en este logro significativo.

Índice de contenido

Índice de tablas

Índice de figuras

Resumen

En este estudio, se aborda el desafío del control de congestión en redes de paquetes mediante la aplicación de técnicas avanzadas de machine learning. Utilizando topologías de red diseñadas en OMNeT y un conjunto de datos detallado que incluye métricas clave como capacidad, datarate, utilización, tiempo de encolamiento, y tasas de paquetes enviados, recibidos y perdidos, se desarrolla un modelo predictivo. Este modelo, evaluado mediante herramientas de machine learning como Random Forest y SVM en MATLAB, demuestra una precisión notable (94% con Random Forest, 87% con SVM y Naïve Bayes con 91%) en predecir la congestión de la red. Se establecen criterios específicos para identificar la congestión, como una utilización superior al 98%, un tiempo de encolamiento mayor a 1 ms, y una tasa de pérdida de paquetes superior al 2%. La aplicación de validación cruzada refuerza la fiabilidad de los modelos. Este enfoque no solo ofrece un método eficaz para predecir la congestión, sino que también establece un precedente en la gestión de redes, combinando análisis de datos avanzados con técnicas de aprendizaje automático.

Palabras clave: Control de Congestión, Machine Learning, Redes de Paquetes.

Abstract

In this study, the challenge of congestion control in packet networks is addressed through the application of advanced machine learning techniques. Using network topologies designed in OMNeT and a detailed dataset that includes key metrics such as capacity, datarate, utilization, queueing time, and rates of sent, received, and lost packets, a predictive model is developed. This model, evaluated using machine learning tools such as Random Forest and SVM in MATLAB, demonstrates notable accuracy (96% with Random Forest, 93% with SVM and Naïve Bayes 91%) in predicting network congestion. Specific criteria are established to identify congestion, such as utilization over 98%, queueing time greater than 1 ms, and a packet loss rate exceeding 2%. The application of cross-validation reinforces the reliability of the models. This approach not only offers an effective method for predicting congestion but also sets a precedent in network management, combining advanced data analysis with machine learning techniques.

Keywords: Congestion Control, Machine Learning, Packet Networks.

Introducción

Este estudio aborda un problema crítico en la era de la información digital: el control de congestión en redes de paquetes. Con el creciente volumen de datos transmitidos, especialmente en formatos multimedia, la congestión de red se ha convertido en un desafío significativo que afecta la calidad y eficiencia del servicio. Este trabajo propone una solución innovadora, utilizando técnicas avanzadas de machine learning para predecir y mitigar eficazmente la congestión. La aplicación de estas técnicas promete superar las limitaciones de los métodos tradicionales, ofreciendo una nueva perspectiva en la gestión de la congestión de redes.

El objetivo principal del estudio es evaluar algoritmos de control de congestión que mejoren la eficiencia y el rendimiento de las redes utilizando machine learning. La metodología empleada incluye la creación de topologías de red en OMNeT, una herramienta elegida por su flexibilidad y precisión en la simulación de redes, y el uso de MATLAB para el análisis y modelado de datos. Este enfoque combinado permite una evaluación detallada y efectiva, abordando el problema desde un punto de vista práctico y teórico.

Durante el desarrollo del estudio, se enfrentaron varios desafíos, como la complejidad de crear entornos de prueba realistas y la extracción de parámetros significativos de los escenarios de red. No obstante, el acceso a datos relevantes y a herramientas de machine learning avanzadas facilitó significativamente el progreso del proyecto.

El trabajo se estructura en cinco capítulos principales. El primer capítulo presenta los antecedentes y justifica la necesidad de este estudio. El segundo capítulo establece el marco teórico, abordando los algoritmos de control de congestión y las aplicaciones de machine learning en este ámbito. El tercer capítulo detalla la metodología adoptada, incluyendo la recopilación y análisis de datos. El cuarto capítulo describe el diseño del algoritmo de machine learning, empleando herramientas y técnicas específicas como Random Forest, SVM y Naïve-Bayes, para predecir y gestionar la congestión en las redes.

La relevancia de esta investigación es considerable tanto para la comunidad académica como para la industria. Ofrece métodos avanzados para mejorar la gestión de la congestión de la red, contribuyendo al desarrollo de infraestructuras de comunicación más robustas y eficientes, esenciales en un mundo cada vez más conectado digitalmente.

Además, la evaluación de las técnicas de machine learning se realizará utilizando métricas clave como Precisión, Precisión promedio de validación cruzada, Recall (Sensibilidad), Especificidad, F1-Score, y Área bajo la curva ROC (AUC-ROC). Estas métricas permitirán una valoración exhaustiva y objetiva de la eficacia de los modelos propuestos.

En la metodología de este estudio, se adoptó el modelo CRISP-DM para una gestión eficiente y sistemática de las diversas fases del proyecto. La primera etapa implicó una comprensión detallada del problema de congestión de la red y la definición de los objetivos del proyecto. La fase de extracción de datos fue fundamental para el entrenamiento y validación de los modelos de machine learning, seguida de una cuidadosa división de los datos en conjuntos de entrenamiento y prueba, asegurando resultados precisos y confiables.

Capítulo uno

Antecedentes

Desde la década de 1980, el campo de las redes de paquetes ha experimentado un desarrollo considerable, especialmente en el área del control de congestión. Se han propuesto numerosos algoritmos, cada uno construyendo sobre las funcionalidades de sus predecesores y evaluándose en una variedad de contextos de red, desde configuraciones simples hasta redes con topologías y necesidades más complejas. Sin embargo, la gestión del control de congestión ha enfrentado desafíos crecientes debido a varios factores, como la incorporación de enlaces satelitales en órbita LEO, que presentan dificultades significativas, especialmente en términos de retrasos en la comunicación. Además, cada segmento de la red es susceptible a fallas, lo que introduce una capa de inestabilidad. En 1982, por ejemplo, la velocidad de transferencia de datos era tan limitada que los procesadores no podían manejar adecuadamente el volumen de información recibida, lo que resultaba en cuellos de botella y pérdidas de conexión (Perrier, 2023).

El año 2020 marcó un punto de inflexión con la pandemia global y el consecuente confinamiento, lo que llevó a un aumento significativo en el uso de internet. Este aumento fue aproximadamente del 20% en comparación con 2019. Para 2021, casi el 63% de la población mundial tenía acceso a internet, lo que se traduce en una demanda extremadamente alta. En respuesta a esta situación sin precedentes, muchos países, incluyendo el nuestro, no estaban preparados. Se emprendieron grandes campañas de tendido de fibra óptica en las principales ciudades para satisfacer la creciente necesidad de acceso a internet. Un notable 75% de este incremento en el uso de internet se atribuyó a clases en línea y a la visualización de contenido multimedia (Becerra, 2021).

Estos antecedentes contextualizan la importancia y la urgencia de desarrollar algoritmos de control de congestión más eficientes y robustos. La historia de los esfuerzos en este campo refleja no solo los desafíos técnicos inherentes, sino también la evolución de las necesidades y demandas de la sociedad. Este capítulo sienta las bases para comprender

cómo los desarrollos pasados han llevado a las innovaciones actuales en el control de congestión y establece el marco para el trabajo realizado en este estudio.

1.1 Problemática

En Ecuador, con más de 18 millones de habitantes y una tasa de acceso a internet que alcanza el 53%, la demanda de servicios de comunicación, especialmente para el envío de paquetes multimedia, ha experimentado un crecimiento acelerado. Esta tendencia se intensificó a raíz de la pandemia de COVID-19, iniciada en marzo de 2020, impulsando una evolución significativa en los sistemas de comunicación, con un enfoque particular en la expansión de las redes de fibra óptica. Sin embargo, este aumento en el uso de internet no siempre se traduce en un rendimiento óptimo de la red. Diversos factores en la transmisión de datos pueden causar congestión, afectando negativamente la eficiencia y la calidad del servicio (Peña, 2021).

La congestión en las redes de paquetes representa un desafío crítico que impacta directamente en el rendimiento y la calidad del servicio. Con el incremento constante en el tráfico de datos y la demanda de ancho de banda, se hace cada vez más complicado gestionar la congestión de manera eficiente y evitar la saturación de la red. Los algoritmos tradicionales de control de congestión a menudo no logran adaptarse rápidamente a las fluctuaciones en la carga de la red y no explotan completamente las posibilidades de optimización que ofrecen las técnicas avanzadas de machine learning.

Surge la necesidad de desarrollar una solución que aprenda de forma autónoma, adaptándose a las condiciones cambiantes de la red. Esto implica el uso de conjuntos de datos históricos y características relevantes de la red para entrenar modelos predictivos capaces de tomar decisiones eficientes de enrutamiento y asignación de recursos. La ausencia de algoritmos de control de congestión que integren técnicas de machine learning restringe la capacidad de optimizar las redes de paquetes y presenta un obstáculo significativo para proporcionar un servicio de calidad en entornos con alta demanda de datos.

Por tanto, es imperativo abordar esta problemática desarrollando nuevas soluciones que permitan una gestión efectiva de la congestión en las redes de paquetes. Esto no solo

mejorará el rendimiento de la red, sino que también optimizará la experiencia del usuario, garantizando un servicio más estable y eficiente en un contexto de creciente dependencia de las comunicaciones digitales.

1.2 Objetivos

Realizar una revisiónh de literatura sobre control de congestión y técnicas de machine learning.

Implementar un algoritmo de control de congestión que use técnicas de Machine Learning.

Evaluar comparativamente el algoritmo propuesto.

1.3 Estado del Arte

El concepto de red, definido como un conjunto de dispositivos electrónicos interconectados para compartir recursos, ha evolucionado significativamente en complejidad y escala, abarcando desde pequeñas redes domésticas hasta amplios entornos empresariales. La congestión de la red, donde el tráfico excede la capacidad disponible, afecta el rendimiento y la fiabilidad de la red. Durante más de treinta años, el estudio del control de congestión de extremo a extremo ha sido esencial para garantizar un uso eficiente y equitativo de los recursos de la red entre los usuarios. Con el aumento de la complejidad de las redes modernas, los enfoques convencionales basados en reglas para controlar la congestión han empezado a mostrar limitaciones en su eficacia. En este contexto, el auge del machine learning (ML) en la resolución de problemas complejos ha inspirado a los investigadores a orientar sus esfuerzos hacia estrategias basadas en esta tecnología. Este artículo ofrece una revisión de las aplicaciones más recientes del ML en el área del control de congestión de extremo a extremo. Comenzamos evaluando cómo el control de congestión se relaciona y se beneficia del ML, seguido por un análisis de investigaciones que han incorporado estas técnicas avanzadas para mejorar las decisiones en el control de congestión y optimizar el rendimiento de la red. Finalizamos destacando desafíos actuales y explorando direcciones prometedoras para futuras investigaciones en este ámbito (Zhang & Mao, 2020)

El control del tráfico de red es un componente fundamental para los sistemas de redes, ya que facilita una entrega de información eficiente y una utilización óptima de recursos mediante la monitorización, inspección y regulación de flujos de datos. Con el desarrollo de la tecnología del Internet de las Cosas (IoT) y la llegada de la era más allá de la quinta generación (5G), dispositivos móviles inteligentes y redes de radio ultra-densas se han expandido enormemente, aumentando la escala de la red y generando cantidades explosivas de tráfico de datos. Esto ha impuesto una presión considerable sobre la gestión de Internet. Además, los avances en servicios de nube central y borde inteligente han cambiado sustancialmente los modelos de flujo de tráfico y la arquitectura de servicio de Internet. Por lo tanto, se necesitan nuevas tecnologías para manejar el control del tráfico de manera altamente escalable y adaptable. Tomando en cuenta la definición anterior, una red es propensa a tener congestión. La congestión se refiere a una situación en la que el tráfico total ofrecido en una red que posee datos sobrepasa la capacidad, lo que genera una notable disminución de su rendimiento, retrasos para la entrega de los paquetes e incluso la pérdida de los datos que se envían (Zhang & Lorenz, 2018).

El control de congestión como una estrategia esencial para prevenir la saturación en las redes. Esta técnica implica ajustar la cantidad de tráfico enviado o incrementar la capacidad de la red para procesar dicho tráfico, siendo aplicable a una variedad de redes, incluyendo aquellas de computadoras y telecomunicaciones (Tanenbaum, 2003).

La Internet Engineering Task Force describe el control de congestión como un conjunto de herramientas diseñadas para regular el flujo de datos en una red. Este proceso incluye la modificación de la velocidad de envío de paquetes para prevenir la sobrecarga de la red, asegurando así una entrega de datos confiable y puntual. Estas perspectivas subrayan la importancia de una gestión efectiva del tráfico para mantener la integridad y eficiencia de las redes modernas (Cath, 2021).

En 1986, el mundo de las redes de comunicaciones presenció un evento significativo: el primer colapso masivo de congestión. Durante este incidente, la velocidad de transferencia de datos en la red, que normalmente era de 32 Kbps, se desplomó drásticamente a solo 40

bps. Este acontecimiento marcó un punto de inflexión en el campo de las redes, despertando un interés sustancial entre los científicos y expertos en tecnologías de la información. La sorprendente disminución del ancho de banda llevó a una ola de investigaciones centradas en comprender y abordar las causas subyacentes de este tipo de colapsos en la red . Las investigaciones que siguieron se centraron principalmente en el protocolo de Control de Transmisión (TCP), que es fundamental para la gestión del tráfico en la mayoría de las redes modernas. Los estudios se orientaron hacia la identificación de vulnerabilidades y limitaciones dentro del protocolo TCP, especialmente bajo condiciones adversas que podrían desencadenar tales situaciones de congestión (Jacobson, 1998).

Los esfuerzos de estos investigadores, destacados en trabajos brindaron un conocimiento más profundo de los mecanismos de congestión de la red, sino que también sentaron las bases para el desarrollo de soluciones y mejoras en los protocolos de red existentes.

TCP Tahoe representa un hito en el desarrollo de algoritmos confiables para el control de congestión en redes. Este algoritmo introduce un sistema de reconocimiento y reenvío, donde el emisor transmite datos y el receptor acusa recibo de cada paquete. En caso de que no se reciba este reconocimiento, el paquete se reenvía. Esta metodología básica, aunque efectiva, sentó las bases para algoritmos posteriores en el campo de las redes (Floyd & Jacobson, 1993).

Posteriormente, TCP Reno emergió como el algoritmo predominante en las redes globales hasta 2019, destacándose por su fiabilidad, compatibilidad con una amplia gama de dispositivos de red y, sobre todo, su adaptabilidad. Sin embargo, este algoritmo tenía limitaciones, particularmente en redes de alta velocidad, donde su rendimiento tendía a disminuir (Kaneko et al, 2007).

Inspirado en TCP Tahoe, TCP New Reno fue desarrollado para mejorar la eficiencia, permitiendo un reenvío de datos más rápido y con menores tasas de pérdida. Su característica principal es la implementación de un mecanismo de retroceso exponencial, optimizando así el control de la congestión en la red (Hayes & Armitage, 2017).

TCP BBR (Bottleneck Bandwidth and Round-trip propagation time) representa un avance significativo en el ámbito del control de congestión, introducido por Google a finales de 2016. Este algoritmo se distingue de métodos tradicionales como CUBIC, que se centran en la detección de pérdidas para indicar congestión. En lugar de ello, BBR realiza estimaciones periódicas del ancho de banda disponible y del tiempo de ida y vuelta mínimo (RTT). El objetivo principal de BBR es funcionar en el punto óptimo de operación definido por Kleinrock, donde se busca maximizar la tasa de transferencia de datos minimizando simultáneamente la congestión. Esta estrategia permite a BBR minimizar la formación de colas y reducir significativamente los retrasos. En esencia, la finalidad de TCP BBR es mejorar sustancialmente el rendimiento de la red, asegurando una gestión eficaz de la congestión y disminuyendo los tiempos de espera (Jaeger et al., 2019).

Un elemento clave que complementa este proyecto de investigación es el uso del machine learning (ML), una disciplina fundamental dentro de la inteligencia artificial. El ML se dedica al desarrollo de algoritmos y modelos que capacitan a las computadoras para aprender de manera autónoma a partir de grandes volúmenes de datos. Esta capacidad de aprendizaje autónomo permite a las máquinas mejorar su desempeño en tareas específicas, sin necesidad de programación detallada para cada situación. El propósito principal del ML es construir sistemas capaces de autoaprendizaje y auto-mejora basados en la experiencia acumulada, en lugar de depender exclusivamente de directrices programadas. Esto se logra mediante la utilización de técnicas estadísticas y de análisis de datos para descubrir patrones y correlaciones en los datos procesados, aplicando posteriormente estos hallazgos para realizar predicciones o tomar decisiones informadas en situaciones nuevas y desconocidas (Ramos & RonaldMárquez, 2023).

En la última década, la inteligencia artificial (IA) ha experimentado un crecimiento exponencial, revolucionando las tecnologías que utilizamos cotidianamente. Este progreso se ha logrado gracias a métodos avanzados que se basan en el análisis estadístico del comportamiento, enfocándose en identificar patrones en grandes conjuntos de datos. Este

proceso suele dividirse en cuatro etapas fundamentales, cada una desempeñando un papel crucial en la transformación de datos brutos en información valiosa y aplicable.

La primera etapa es la recolección de datos, donde se acumulan grandes volúmenes de información, a menudo generados por el tráfico de la red. Esta fase es vital, ya que la calidad y la cantidad de los datos recopilados pueden influir significativamente en la eficacia de los modelos de IA desarrollados posteriormente.

La segunda etapa implica la extracción de características o métricas relevantes del conjunto de datos. Este proceso es fundamental para determinar qué aspectos de los datos son más significativos y cómo pueden ser utilizados para predecir o entender mejor ciertos fenómenos o comportamientos.

En la tercera etapa, se realiza el procesamiento de estos datos. Normalmente, se sigue una división estándar donde aproximadamente el 70% de los datos se destina al entrenamiento de modelos de IA, mientras que el 30% restante se reserva para pruebas y validación. Esta proporción ayuda a equilibrar la necesidad de entrenar modelos robustos con la necesidad de verificar su eficacia en condiciones no vistas previamente.

Finalmente, la cuarta etapa es la ejecución del algoritmo de aprendizaje. Aquí se implementan y prueban diferentes algoritmos para determinar cuál es el más adecuado para los objetivos específicos del proyecto. Esta fase permite la iteración y la optimización, asegurando que el modelo seleccionado ofrezca los mejores resultados posibles (León & Martínez, 2022).

1.4 Trabajos Relacionados

En el estudio realizado en 2012, se exploró el uso de redes vehiculares a través de software de simulación para crear escenarios que replican eventos discretos similares a los de una red de dispositivos convencionales. En este caso, se utilizaron vehículos en movimiento equipados con tecnologías inalámbricas, enfocándose en protocolos de enrutamiento, seguridad y control de congestión. Debido a la escasez de simuladores gratuitos en el mercado, se optó por emplear un software denominado VANET. Este software tenía como objetivo principal investigar el rendimiento de diferentes variantes del control de

congestión en TCP, utilizando para ello dos parámetros de evaluación: AODV (Adhoc on Demand Distance Vector) y DSR (Dynamic Source Routing), que permiten medir el retraso y el rendimiento de los parámetros de TCP. Utilizando simuladores como OMNET++ y SUMO, los resultados revelaron que el algoritmo New Reno supera claramente a Reno en rendimiento. Sin embargo, se encontró que Tahoe tiene un desempeño similar a New Reno en redes de menor tamaño, mientras que, en redes más extensas, Tahoe consigue menores retrasos y un rendimiento superior (Kaur & Josan, 2012).

En otra investigación relacionada, Abel (2014) se centró en la simulación de algoritmos de control bajo protocolos y servicios de aplicación en TCP/IP, un área que requiere una infraestructura de red cada vez más compleja y actualizada debido a la creciente demanda tecnológica. Este estudio resalta la necesidad constante de desarrollar nuevas topologías de redes y subredes repletas de computadoras, lo que implica un desafío significativo para las experiencias de laboratorio. La simulación jugó un papel crucial al facilitar la visualización de computadores y dispositivos en una red, permitiendo así la construcción de escenarios complejos sin la necesidad de contar con computadoras de recursos elevados. Este enfoque permitió experimentar con los algoritmos de control de congestión en TCP a través de UML, proporcionando insights valiosos para investigaciones como la mía, donde el enfoque se centra en mejorar el control de congestión en redes de paquetes mediante el uso de técnicas de machine learning (A. Rattalino, 2014).

Capítulo dos

Marco Teórico

2.1 Algoritmo de control de congestión

El fenómeno de la congestión en una red se produce cuando el volumen de tráfico excede la capacidad de esta, cuando la demanda de recursos supera la oferta disponible. Estos recursos incluyen aspectos cruciales como el ancho de banda, la latencia, el tamaño de los paquetes y la capacidad de procesamiento de los diferentes componentes de la red. En esencia, la congestión ocurre cuando hay más datos intentando transitar por la red de lo que esta puede manejar eficientemente, lo que conduce a una serie de problemas como la disminución del rendimiento, el aumento en los tiempos de respuesta y la pérdida potencial de paquetes.

Efectos de la congestión:

Cuando una red está congestionada, la velocidad a la que se pueden transferir los datos disminuye significativamente. Esto se debe a que más paquetes luchan por un ancho de banda limitado, lo que ralentiza el proceso de envío y recepción de datos.

La congestión puede causar un aumento en el tiempo de ida y vuelta (RTT) de los paquetes de datos, lo que significa que los paquetes tardan más tiempo en llegar a su destino y en volver. Esto se traduce en una mayor latencia, que es especialmente problemática para aplicaciones en tiempo real como juegos en línea y videollamadas.

En situaciones extremas de congestión, los paquetes pueden descartarse porque los buffers de los dispositivos de red (como los routers) se llenan completamente. La pérdida de paquetes requiere la retransmisión de esos datos, lo que agrava aún más la congestión y reduce la eficiencia de la red (Singh & Kaur, 2017).

2.2 Control de congestión en TCP

Esta técnica de control de congestión se fundamenta en el manejo de las ventanas de transmisión en TCP, que son claves para regular la tasa de envío de paquetes y ajustar progresivamente el tamaño de la ventana. Este enfoque ayuda a mantener un ancho de banda balanceado para todas las conexiones dentro de la red. Un elemento crucial en este mecanismo es el

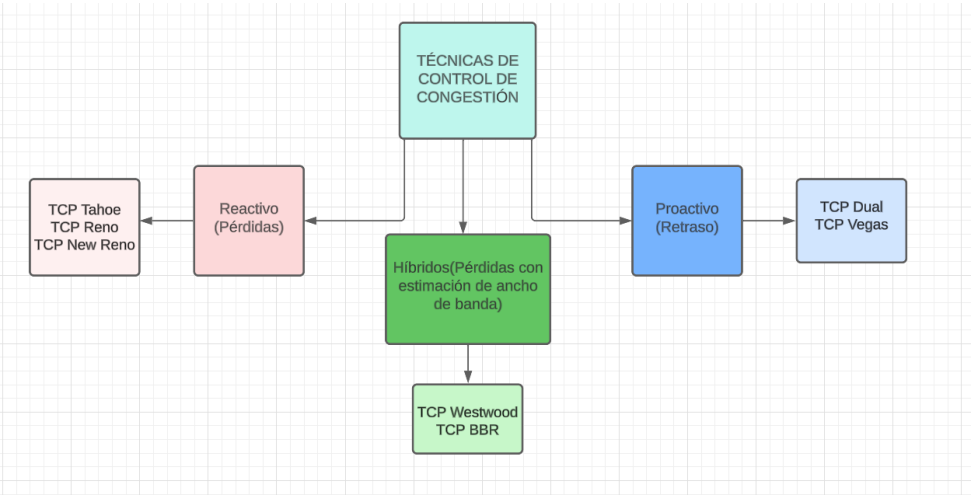
parámetro CWND (ventana de congestión) de TCP, que determina el número estimado de paquetes que pueden ser enviados y recibidos sin acusar una respuesta. TCP interpreta la pérdida de paquetes como un indicador de congestión. En TCP, hay dos mecanismos principales para el control de congestión: el incremento aditivo y el decremento multiplicativo. El incremento aditivo se emplea para aumentar la ventana de congestión cuando hay una disminución en el nivel de demanda de la red, mientras que el decremento multiplicativo reduce la ventana de congestión en respuesta a un aumento en la demanda. Este último mecanismo se activa especialmente cuando se detecta un tiempo de espera excesivo o 'timeout', indicando una posible congestión. Durante un evento de timeout, el tamaño de la ventana de congestión en el host de origen se reduce a la mitad de su valor anterior, un proceso conocido como decremento multiplicativo. Sin embargo, se mantiene la regla de que la ventana de congestión no debe disminuir por debajo del tamaño de un segmento TCP individual. Estos mecanismos son esenciales para adaptar dinámicamente el flujo de datos en la red y prevenir la congestión excesiva (Cadin & Talay, 2021).

2.3 Clasificación de técnicas de control de congestión

En el ámbito del control de congestión, se identifican tres enfoques metodológicos principales, los cuales se describen en la Figura 1.

Figura 1

Clasificación de técnicas de control de congestión



El primero es el enfoque reactivo, que se centra en la detección de la pérdida de paquetes para indicar la presencia de congestión. Este método responde a problemas ya

existentes en la red, ajustando los parámetros de control después de que la congestión ha sido detectada.

El segundo enfoque, conocido como proactivo, se esfuerza en anticipar la ocurrencia de congestión antes de que esta se manifieste. Se basa en modelos que evalúan el tiempo, particularmente prestando atención a los retrasos en la recepción de paquetes, para prever y prevenir posibles situaciones de congestión.

El tercer enfoque es el híbrido, que combina elementos tanto del reactivo como del proactivo. Este método utiliza modelos probabilísticos para estimar el ancho de banda necesario y optimizar el rendimiento de la red. Al integrar estrategias de ambos enfoques, el método híbrido busca equilibrar la eficiencia y la efectividad en la gestión de la congestión, adaptándose dinámicamente a las condiciones cambiantes de la red y anticipando problemas potenciales antes de que se conviertan en críticos (Lugones, 2006).

2.4 Machine Learning

El machine learning, también conocido como aprendizaje automático, representa un enfoque innovador en el que se capacita a las computadoras para resolver problemas de manera autónoma. Este proceso se basa en el análisis de datos obtenidos a través de la interacción humana, los cuales son fundamentales para ofrecer mejoras y soluciones efectivas, especialmente en contextos industriales. Los datos se utilizan para realizar comparaciones en tiempo real y brindar respuestas específicas adaptadas a cada situación. En el núcleo del aprendizaje automático se encuentran algoritmos diseñados para procesar y analizar los datos de entrada, generando información valiosa para abordar problemas concretos. El objetivo principal de estos algoritmos es crear un volumen ampliado de datos a partir de técnicas como la regresión lineal o los árboles de decisión. Estas herramientas permiten identificar patrones y extraer conocimientos que son cruciales para hacer predicciones precisas y tomar decisiones informadas. En resumen, el aprendizaje automático se centra en transformar los datos en inteligencia aplicable, facilitando así una toma de decisiones más eficiente y efectiva en una variedad de aplicaciones (Rojas, 2020).

2.4.1 *Aplicaciones comunes de Machine Learning*

Una gran cantidad de las plataformas y sitios web más populares del mundo implementan algoritmos de machine learning para ofrecer recomendaciones personalizadas, abarcando una amplia gama de aplicaciones cotidianas. Por ejemplo, los sistemas de filtrado de correo no deseado utilizan estos algoritmos para analizar y clasificar el contenido de los correos electrónicos. Además, en el ámbito de las redes sociales y las plataformas de contenido, el machine learning desempeña un papel crucial en la detección de patrones en imágenes y comportamientos de los usuarios, manteniendo así su atención y mejorando la experiencia en la aplicación. En el comercio electrónico, estos algoritmos se utilizan extensamente para recomendar productos, personalizando la experiencia de compra y potenciando así las ventas en línea. Estas aplicaciones del machine learning muestran su alcance e influencia, siendo herramientas esenciales en la mejora de la interacción del usuario con la tecnología en el día a día.

2.4.2 *Métodos de aprendizaje de la maquina*

En el ámbito del aprendizaje automático para computadoras, se destacan principalmente dos enfoques: el aprendizaje supervisado y el no supervisado. El aprendizaje supervisado implica entrenar algoritmos con conjuntos de datos etiquetados, donde cada entrada viene con una salida o etiqueta predefinida. Esto permite al algoritmo aprender y hacer predicciones o clasificaciones basadas en nuevos datos ingresados. Por otro lado, el aprendizaje no supervisado opera sin datos etiquetados; en su lugar, explora los datos de entrada para identificar patrones o agrupaciones intrínsecas, aprendiendo de forma autónoma sin intervención humana directa (Godoy, 2015).

2.4.3 *Árboles de decisión*

Los árboles de decisión se han establecido como un componente fundamental en el campo del aprendizaje automático, especialmente en tareas de clasificación y regresión. Estos árboles operan analizando los datos a través de una serie de evaluaciones lógicas, lo que permite tomar decisiones informadas y realizar comparaciones basadas en las características inherentes de los datos. La estructura de un árbol de decisión se compone de

nodos que dividen el conjunto de datos en subconjuntos más pequeños y manejables. Esta división facilita una evaluación más detallada y precisa de los datos. En el contexto de modelos como Random Forest, los árboles de decisión son esenciales, ya que contribuyen a la creación de múltiples árboles que trabajan en conjunto para mejorar la precisión y la robustez del modelo general. Cada árbol en un Random Forest contribuye con su perspectiva única al proceso de decisión, lo que permite un análisis más completo y preciso del conjunto de datos (Goddard et al., 1995).

2.4.4 *Clasificación y regresión*

Las funciones de clasificación y regresión son pilares fundamentales en el aprendizaje automático. La clasificación se centra en asignar categorías específicas a los datos, basándose en sus características y patrones. Esta división en categorías se realiza de acuerdo con la función específica que se está aplicando. Por otro lado, la regresión se orienta hacia la predicción de valores numéricos en lugar de categorías. Su objetivo es establecer relaciones matemáticas entre las variables de entrada y salida, permitiendo una comprensión más profunda y una predicción precisa basada en los datos analizados (Valle, 2010).

2.4.5 *Support Vector Machine*

Es un modelo de aprendizaje supervisado utilizado en el análisis de datos para tareas de clasificación y regresión. Las SVMs son particularmente conocidas por su capacidad para manejar espacios de alta dimensión y su efectividad en situaciones donde el número de dimensiones es mayor que el número de muestras. Este modelo construye un hiperplano o conjunto de hiperplanos en un espacio de alta dimensión, que se utiliza para la clasificación o regresión. En términos simples, una SVM realiza una clasificación encontrando el hiperplano que mejor separa dos clases de datos, maximizando el margen entre los puntos de las diferentes clases (Cortes & Vapnik, 1995).

Las SVMs son efectivas en situaciones donde la relación entre las clases no es lineal, ya que pueden emplear lo que se conoce como el "truco del kernel" para transformar el espacio de entrada en un espacio de mayor dimensión donde es posible una separación lineal. Son ampliamente utilizadas en aplicaciones como el reconocimiento de patrones, la

clasificación de textos y la bioinformática, debido a su robustez y precisión (Cortes & Vapnik, 1995).

2.4.6 ***Random Forest***

El Random Forest es un avanzado algoritmo de aprendizaje automático supervisado que integra tanto la clasificación como la regresión. Su principal fortaleza reside en la combinación de múltiples modelos complejos, en este caso, árboles de decisión, para crear un modelo final que es tanto más preciso como confiable. Este enfoque se basa en la creación de una "forest" o bosque, compuesto por numerosos árboles de decisión, donde cada árbol contribuye con su perspectiva individual al modelo general. Durante el proceso de entrenamiento, cada árbol de decisión en el Random Forest se construye utilizando un subconjunto aleatorio de características y datos. Esta técnica de selección aleatoria asegura que cada árbol sea único y que el modelo general sea robusto frente al sobreajuste. Además, al utilizar múltiples árboles, el algoritmo de Random Forest puede manejar eficazmente tanto datos lineales como no lineales (Breiman, 2001).

2.4.6.1 **Treebagger**

Es una función del software Matlab que facilita la implementación de random Forest utilizando múltiples árboles de decisión el cual el usuario es libre de decidir cuántos usar a mayor cantidad el resultado será más preciso, pero al mismo tiempo necesitará más capacidad de procesamiento de la máquina, ya que a cada árbol se lo entrena con un subconjunto de datos para después unir todos los árboles y combinarlos de esta manera se obtendrá una precisión alta del modelo (Yağmur et al., 2023).

2.4.7 ***Naïve-Bayes***

El Modelo Naïve Bayes, fundamentado en el teorema de Bayes, representa un enfoque probabilístico en el ámbito del aprendizaje automático, siendo particularmente eficaz en la categorización de información. Este modelo destaca por asumir que los elementos predictivos operan de manera independiente. Aunque se caracteriza por su estructura simple, ha demostrado ser altamente eficiente en varios campos, incluyendo la detección y filtrado

de correo no deseado, así como en la organización y clasificación de documentos textuales (Russell & Norvig, 2004).

2.5 Métricas de la red

Las métricas de red son esenciales para valorar la eficacia, rendimiento y salud integral de las infraestructuras de comunicación digital. Son fundamentales para monitorear el funcionamiento actual de la red y anticipar contratiempos o fallas que podrían surgir. La elección de las métricas apropiadas se alinea con los fines específicos y demandas operativas de la red. Entre estas se incluyen la tasa de pérdida de paquetes, que cuantifica los datos perdidos en transmisión; la latencia, que es el tiempo que toman los datos en llegar de un punto a otro; y el ancho de banda, que determina el volumen de información que la red puede manejar en un intervalo de tiempo. Otros indicadores relevantes son el jitter, que evalúa la variación temporal en la llegada de paquetes, y la utilización de la red, que revela qué tan intensamente se emplean los recursos de la red.

Estas métricas pueden incluir, entre otros, la tasa de pérdida de paquetes, que mide la proporción de paquetes que se pierden durante la transmisión; la latencia, que indica el tiempo que tardan los datos en viajar desde su origen hasta su destino; y el ancho de banda, que refleja la cantidad máxima de datos que puede transmitir la red en un período determinado.

2.5.1 *Paquetes Perdidos*

El porcentaje de paquetes perdidos es la comparación entre la cantidad de paquetes enviados y la cantidad de paquetes que llegan a su destino de forma correcta, en la cual a través de una simple fórmula podemos sacar el porcentaje que se pierden en el camino, como una métrica es fundamental ya que una elevada cantidad de paquetes perdidos indican que la red tiene problemas o sufre de alguna congestión además que esta métrica es usada especialmente para evaluar la calidad del servicio y la eficiencia en la transmisión de datos.

2.5.2 *Utilización*

El porcentaje de utilización permite observar la cantidad de recurso que se utiliza en tiempo real para el procesamiento de datos, comparado con todo el tiempo que tiene

disponible, esta métrica es fundamental para ver si el nivel de carga sobre una red es alto o bajo, para ser monitoreado evitando cuellos de botella y garantizar un óptimo desempeño de la red (Bonaventure, 2018).

2.5.3 *Tiempo en colas*

El tiempo de colas es utilizado para medir la cantidad de tiempo que un paquete se encuentra esperando antes de ser procesado o transmitido, cuando existe un aumento de tiempo es un gran indicador de que la red se está congestionando ya que cada paquete tiene que esperar a ser atendido.

2.6 Métricas de evaluación del algoritmo

Las métricas de rendimiento en el contexto del machine learning son herramientas esenciales utilizadas para evaluar y medir la efectividad de los modelos de aprendizaje automático. Estas métricas proporcionan una evaluación cuantitativa de cómo un modelo predice o clasifica los datos, lo que es crucial para determinar su precisión, eficiencia y utilidad en aplicaciones prácticas.

2.6.1 *Precisión*

La precisión en aprendizaje automático evalúa cuántas de las clasificaciones positivas hechas por el modelo son correctas, es decir, la proporción de verdaderos positivos entre todos los casos etiquetados como positivos (Miao & Zhu, 2022).

2.6.2 *Precisión promedio de validación cruzada*

La validación cruzada es un procedimiento para asegurar que el modelo es aplicable más allá del conjunto de datos en el que fue entrenado, dividiendo los datos en varias partes, entrenando con algunas y probando con las otras, y luego promediando los resultados para obtener una medida consistente del rendimiento.

2.6.3 *Recall (Sensibilidad)*

Mide la habilidad del modelo para identificar todos los casos positivos verdaderos dentro del conjunto de datos, reflejando la proporción de positivos que fueron correctamente identificados por el modelo.

2.6.4 ***Especificidad***

La especificidad indica la precisión del modelo al identificar los casos negativos, es decir, qué tan bien el modelo reconoce los casos que no son positivos.

2.6.5 ***F1-Score***

El F1-Score combina precisión y recall en una sola cifra que refleja la precisión y la completitud con la que el modelo realiza las clasificaciones, ideal para situaciones con clases desbalanceadas.

2.6.6 ***Área bajo la curva ROC (AUC-ROC)***

Es una métrica que muestra cuán capaz es el modelo de distinguir entre clases a través de una curva que representa la relación entre la tasa de verdaderos positivos y falsos positivos en distintos umbrales, siendo 1.0 el valor que denota la perfección y 0.5 uno que indica rendimiento aleatorio (Miao & Zhu, 2022).

Capítulo tres

Metodología

En este capítulo se detalla la metodología adoptada para abordar el reto del control de la congestión en redes de paquetes por medio de la aplicación de técnicas avanzadas de aprendizaje automático. La elección de una metodología apropiada es fundamental, pues ofrece la estructura necesaria para llevar a cabo una investigación de forma rigurosa y sistemática.

Este proceso se enmarca dentro del modelo CRISP-DM, un estándar de la industria para la minería y ciencia de datos, que, desde su creación en 1996, ha sido fundamental en la estructuración y ejecución de análisis de datos. CRISP-DM es particularmente útil en este proyecto por su capacidad de adaptarse a variados desafíos de datos, como es el caso del aprendizaje automático en redes simuladas (Schröer et al., 2021).

Este enfoque metodológico no sólo facilita la recopilación y el análisis de los datos de interés, también garantiza la exactitud y fiabilidad de los resultados obtenidos.

El estudio se ha centrado en la aplicación y evaluación de algoritmos de aprendizaje automático en un entorno de red simulado. Para ello se ha empleado OMNeT, una herramienta elegida por su eficacia y precisión en la simulación de diversas topologías de red y la creación de escenarios de congestión realistas. Por medio de estas simulaciones, se extraen métricas clave de la red como la utilización, la pérdida de paquetes y el tiempo de espera en colas. Estos parámetros son fundamentales para determinar si existe congestión en algún punto de la red y proporcionan la base para el entrenamiento y la validación de los modelos de aprendizaje automático.

Para la implementación y evaluación de estos algoritmos se utiliza MATLAB, aplicando técnicas como Random Forest, SVM y Naïve-Bayes. Este proceso comprende la recopilación y preparación de datos, un aspecto fundamental que incluye su selección, limpieza y transformación para garantizar su calidad y pertinencia.

Las métricas de evaluación empleadas para medir la eficacia de los algoritmos de aprendizaje automático incluyen la precisión, la recuperación, la especificidad, la puntuación F1 y el AUC-ROC. El uso de estas métricas constituye una base sólida para la comparación y el contraste del rendimiento de distintos modelos y garantiza una evaluación exhaustiva y objetiva.

3.1 Descripción de Herramientas y Software

En concreto, para este estudio se utiliza MATLAB para procesar y analizar los datos de rendimiento de la red obtenidos mediante OMNeT++. Esta integración facilita una evaluación completa de los modelos de aprendizaje automático en cuanto a su capacidad para predecir eficazmente la congestión en las redes de paquetes.

Se pretende que la aplicación de estas técnicas avanzadas de aprendizaje automático, contribuyan significativamente al desarrollo de métodos más eficientes y precisos para el control de la congestión en redes de paquetes.

3.1.1 ***OMNET++***

En este estudio sobre el control de la congestión en redes de paquetes mediante técnicas de aprendizaje automático, la herramienta de simulación OMNeT++ cumple una función clave, permitiendo diseñar modelos detallados de componentes de red, como nodos, enlaces, enrutadores, conmutadores y hosts.

Incluye diversas bibliotecas y módulos que facilitan la implementación y simulación de protocolos de red estándar, proporcionando un sólido punto de partida para la experimentación. En particular, para este trabajo, se destacó el uso del protocolo UDP en las simulaciones. A diferencia de TCP, que implementa mecanismos de control de congestión, el uso de UDP permite el análisis del comportamiento de la red en ausencia de estos mecanismos, favoreciendo la observación directa del impacto de la congestión y la eficacia de las técnicas de aprendizaje automático propuestas.

OMNeT++ se basa en la simulación de eventos discretos, una función esencial para modelar la transmisión de paquetes, la congestión y otros eventos críticos en el control de la congestión. Esta capacidad permite programar y observar eventos específicos, como el inicio y la finalización de transmisiones de paquetes, retransmisiones y cambios en la topología de la red, proporcionando una visión detallada y controlada del comportamiento de la red en diferentes condiciones.

Aunque el software no incluye directamente herramientas de machine learning, facilita la recolección de datos relevantes sobre el rendimiento de la red, que posteriormente pueden utilizarse para entrenar modelos de machine learning implementados en herramientas como MATLAB.

3.1.2 *MATLAB*

Para la implementación y evaluación de estos algoritmos se ha seleccionado MATLAB como herramienta principal debido a sus características y capacidades especialmente beneficiosas para este campo de investigación.

MATLAB es ampliamente reconocido por su amplio conjunto de herramientas y algoritmos integrados de aprendizaje automático. Esta característica facilita significativamente la implementación y comparación de varios modelos predictivos, como Random Forest, SVM y Naïve Bayes.

Otro aspecto fundamental son sus capacidades avanzadas para manejar, analizar y visualizar grandes conjuntos de datos; siendo estas indispensables para interpretar eficazmente los resultados obtenidos a partir de modelos de aprendizaje automático, permitiendo así una evaluación detallada y precisa del modelo de machine learning.

3.2 Adopción del Modelo CRISP-DM

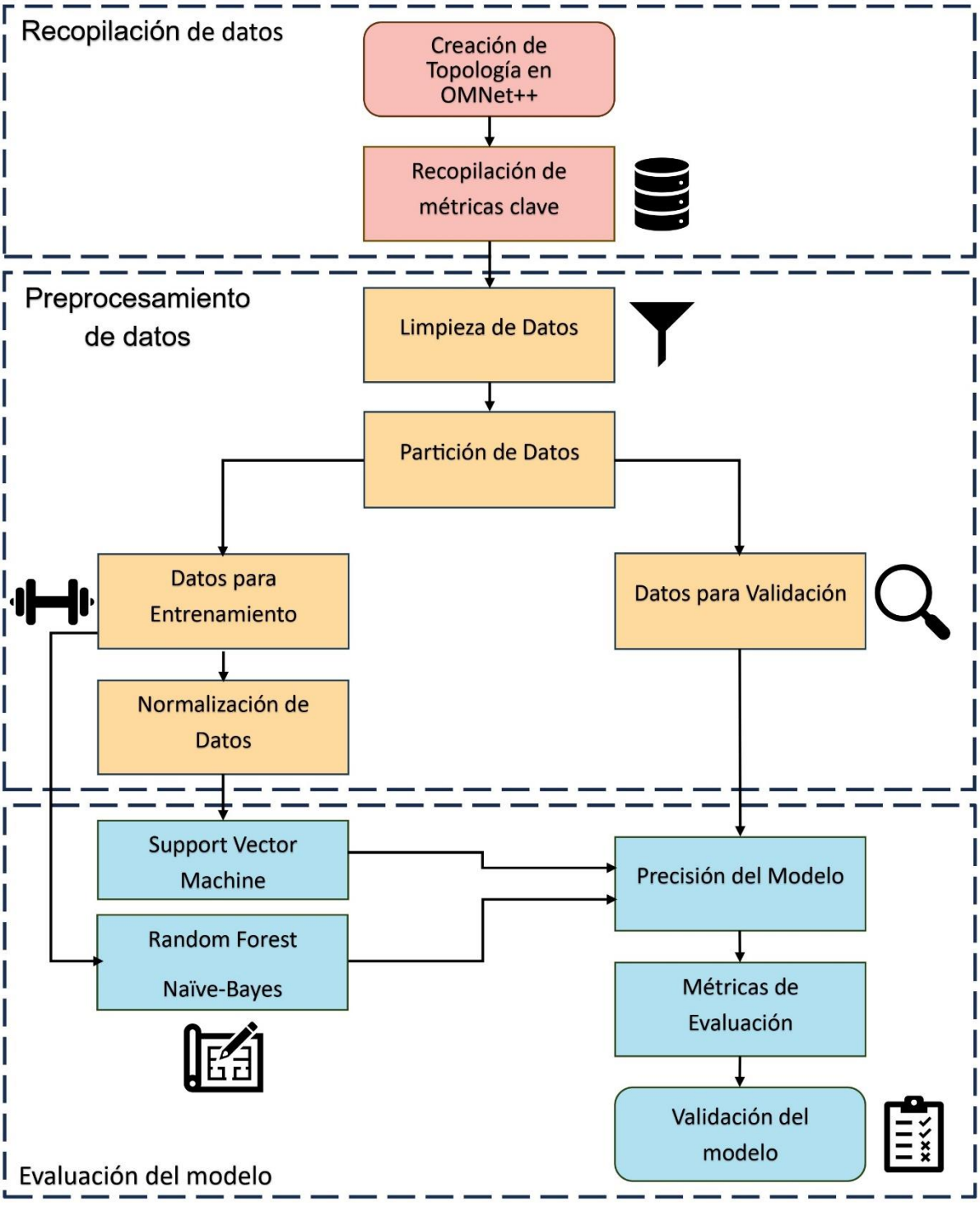
La elección del modelo CRISP-DM (Cross-Industry Standard Process for Data Mining) para este proyecto se basa en varias consideraciones importantes. En primer lugar, CRISP-DM está reconocido como una norma industrial de hecho para el desarrollo de proyectos de minería de datos independientes de la industria. Este modelo proporciona un marco metodológico estructurado que guía los esfuerzos de minería de datos a través de las fases típicas del proyecto, las tareas asociadas a cada fase y la interrelación entre estas tareas. Esta estructura probada y bien definida es esencial para el éxito de la planificación y ejecución de proyectos complejos de minería de datos como el presente proyecto.

Además, la naturaleza tecnológicamente agnóstica del CRISP-DM lo hace aplicable a todos los proyectos de minería de datos, incluyendo aquellos que incorporan técnicas avanzadas de machine learning y análisis de redes. Esto es especialmente relevante para nuestro proyecto, que busca implementar y evaluar algoritmos de machine learning en el contexto del control de congestión en redes de paquetes. La flexibilidad del CRISP-DM para adaptarse a diversas tecnologías y enfoques metodológicos es, por lo tanto, un activo significativo (Wehrstein & Bachmann, 2021).

El diagrama de flujo mostrado en la Figura 2 refleja la adaptación de la metodología CRISP-DM para el desarrollo y evaluación de algoritmos de machine learning en el estudio de la congestión en redes de paquetes.

Figura 2

Diagrama de Flujo del modelo de ML



Iniciando con la generación de topologías en OMNeT++, se recopilan meticulosamente métricas clave que, tras ser sometidas a un riguroso proceso de limpieza, se dividen en conjuntos de entrenamiento y validación. Específicamente para el algoritmo SVM, se normalizan los datos de entrenamiento para contrarrestar su sensibilidad a la escala de los atributos, asegurando así que todos los factores influyan de manera equitativa en el modelo. Luego, se procede a la fase de modelado de CRISP-DM, donde se aplican técnicas como SVM, Random Forest y Naïve-Bayes, evaluando su precisión mediante métricas establecidas. Finalmente, en la fase de evaluación, se valida la capacidad predictiva de los modelos contra el conjunto de datos de prueba, cerrando el ciclo de CRISP-DM y proporcionando información valiosa para la gestión eficiente de la congestión en redes, alineados con los objetivos de este trabajo de titulación.

3.3 Diseño y Configuración de la red de simulación

En el diseño y la configuración de la red de simulación utilizada en este estudio para investigar la extracción de métricas de rendimiento de la red, para posteriormente elaborar el conjunto de datos, se utilizó la herramienta de simulación OMNeT+.

Se desarrollo una topología de red, la cual consta de cinco computadores que transmiten datos a través de una red compuesta por 12 routers, y cinco dispositivos terminales que actúan como receptores de paquetes.

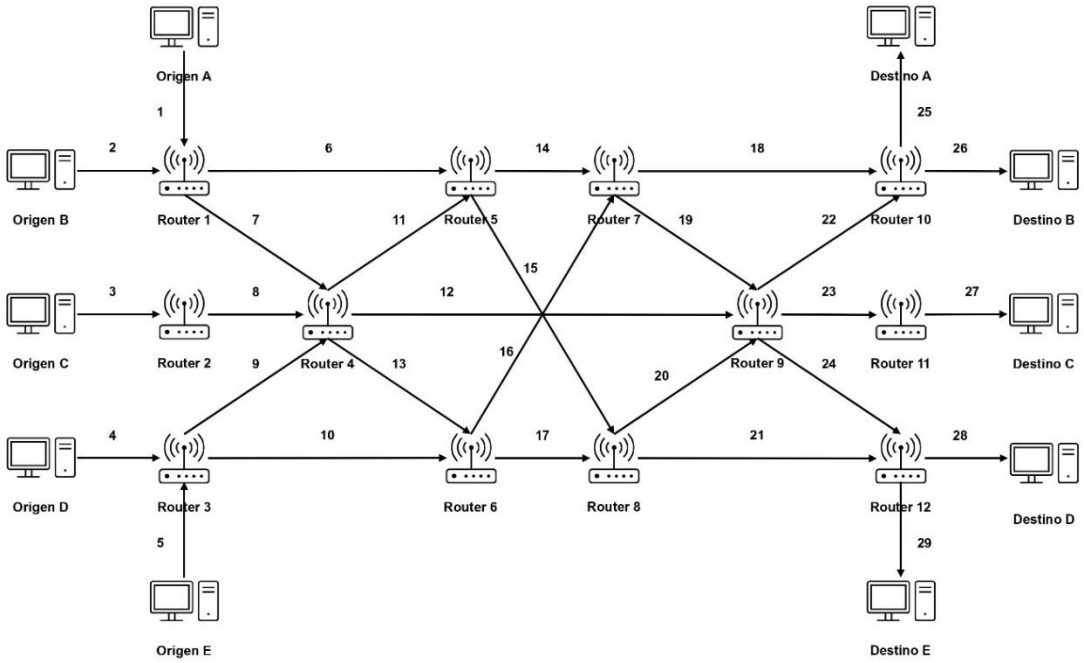
En la Figura 2 se desarrollaron tres escenarios distintos sobre una topología de red común. Cada escenario fue diseñado para probar diferentes aspectos de la congestión de la red, variando en las rutas de paquetes y las tasas de transmisión. Se optó por el uso del protocolo UDP para obtener una evaluación más precisa del rendimiento de la red sin los mecanismos de control de congestión de TCP.

Para la elaboración del conjunto de datos se diseñó una topología final la cual se muestra en la Figura 3, que consiste en usar cinco computadores que transmiten datos a través de una red compuesta por 12 routers y cinco dispositivos terminales, que actúan como receptores de paquetes. Esta configuración constituye la topología de red sobre la cual se desarrollaron 105 escenarios distintos, variando la capacidad de los enlaces y las rutas de los paquetes de envió para inducir congestión en ciertas áreas, mientras se mantenían escenarios equilibrados y óptimos en otras.

Aunque el software permitió extraer diversos parámetros, las tres métricas especificadas en el marco teórico resultaron ser las más representativas para evaluar la congestión de la red.

Figura 3

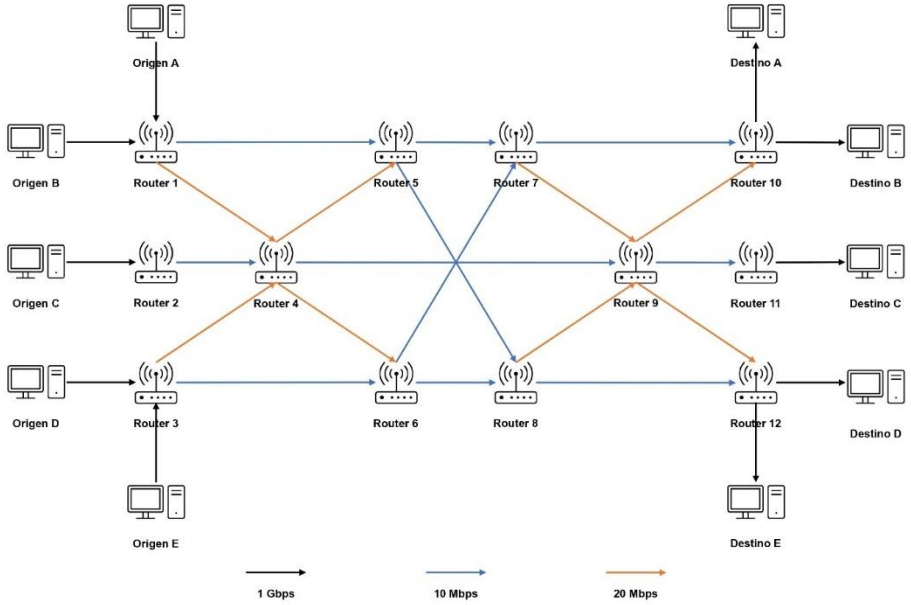
Topología final de red



La primera configuración del escenario de red se muestra en la Figura 4, en este escenario se usa enlaces con una capacidad de 10 Mbps y 20 Mbps para la transmisión de paquetes de datos.

Figura 4

Escenario de topología de red 1



En los siguientes escenarios de red que se muestran en la Figura 5 y Figura 6, se añaden en los enlaces capacidades de 5 Mbps y 15 Mbps, con la finalidad de observar el comportamiento de la red al inducir congestión en más puntos de la red a la vez.

Figura 5

Escenario de topología de red 2

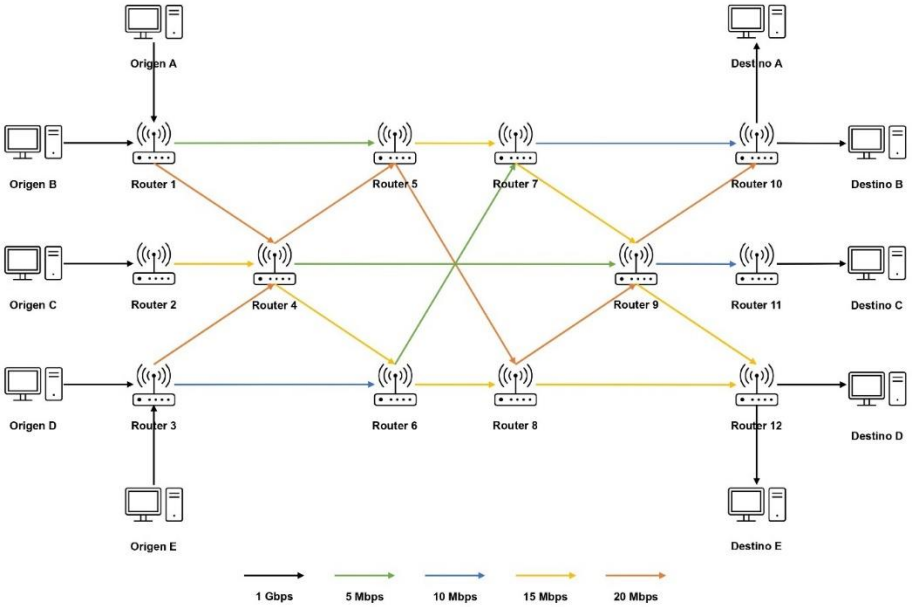
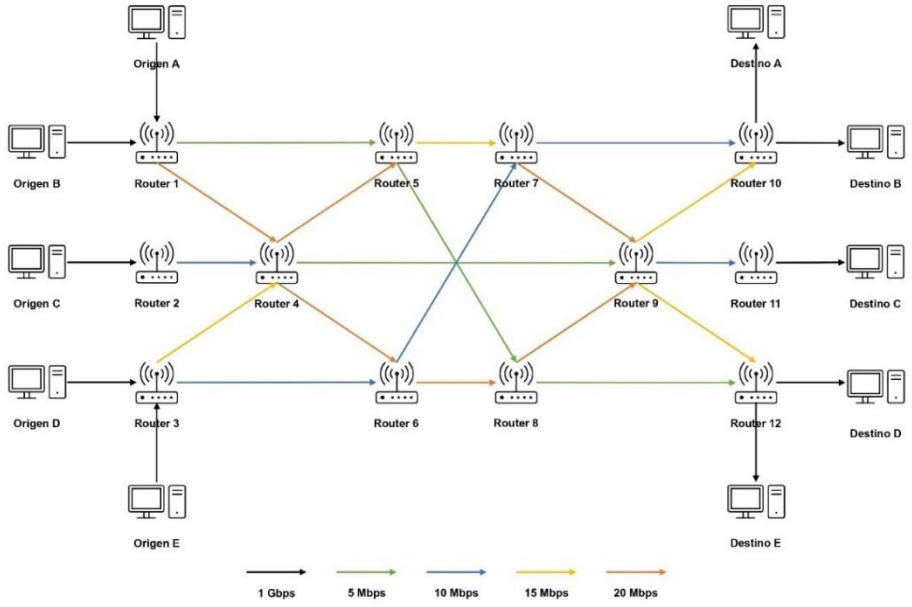


Figura 6

Escenario de topología de red 3



3.4 Recopilación y Preparación de Datos

La recopilación de datos se llevó a cabo mediante simulaciones en OMNeT++. Estas simulaciones proporcionaron un conjunto de métricas detalladas, incluyendo paquetes enviados, paquetes recibidos, tasa de transmisión de datos y tiempo de espera en colas. Estas métricas fueron exportadas a archivos XML para su posterior análisis.

A partir de estos datos, se calcularon dos métricas clave:

Utilización del Canal/Enlace: Esta métrica se obtuvo analizando la relación entre la capacidad del enlace y el volumen de tráfico transmitido, lo que proporciona una indicación directa del nivel de congestión en la red.

Tasa Porcentual de Paquetes Perdidos: Se calculó como el porcentaje de paquetes enviados que no fueron recibidos, lo cual es un indicador crucial de la congestión de la red.

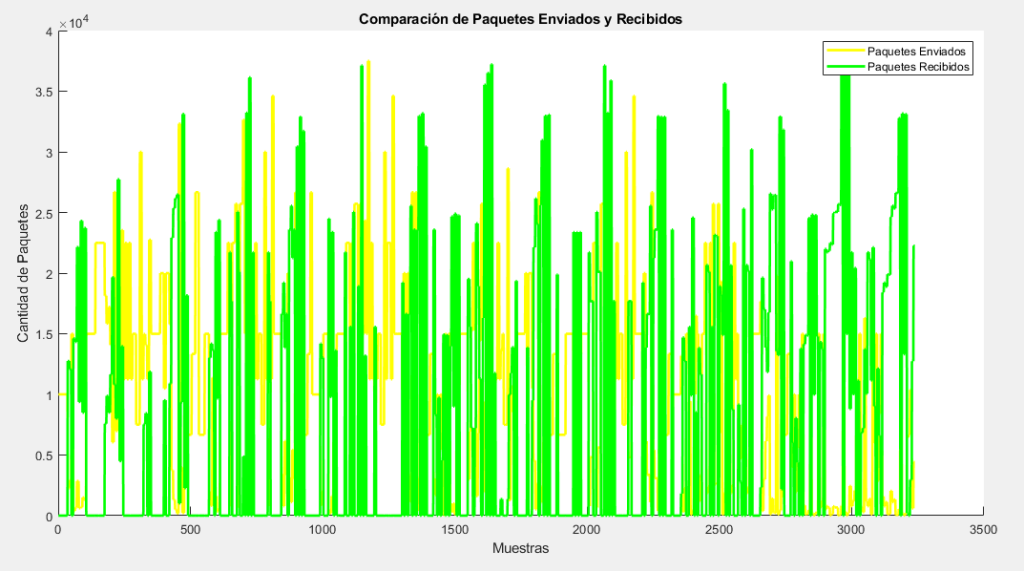
Para este estudio, de los 3 escenarios de red, se recopilaron 105 tablas o muestras de datos importantes para entrenar y validar modelos de aprendizaje automático.

El conjunto total de las muestras de datos de la red es un resultado de la variación de los escenarios, las diferentes combinaciones de rutas de envío de paquetes y tasa de transmisión de envío.

La Figura 7 ofrece una visión detallada del comportamiento de la red bajo condiciones de carga intensa, reflejadas en la creación de un conjunto de datos (data set) extenso.

Figura 7

Grafica comparativa entre paquetes enviados y recibidos



Se observa una discrepancia notable entre la cantidad de datos enviados y los efectivamente recibidos. Esta diferencia se atribuye a la pérdida de paquetes durante su tránsito a través de diversas rutas en la red. Un aspecto crucial que resalta en la gráfica es que, en ciertos intervalos, la proporción de paquetes perdidos supera el umbral del 2%. Este fenómeno es un indicador claro de congestión en la red, sugiriendo que la infraestructura de red alcanza su capacidad límite y, como consecuencia, no logra manejar eficientemente el volumen de tráfico transmitido.

3.4.1 ***Preparación y Partición de Datos***

Una vez finalizados los 100 escenarios, se obtuvo un total de 3200 registros (filas), reflejando que cada escenario se simuló durante 30 segundos, generando un dato por segundo. Para la aplicación de técnicas de aprendizaje automático, es habitual dividir los datos en un 70% para el entrenamiento del algoritmo y un 30% para la validación. Sin embargo, en nuestro enfoque, que emplea máquinas de soporte vectorial, es esencial normalizar estos datos. Además, se requiere un balance adecuado de los mismos. Observamos que, en muchos casos, hay una mayor proporción de instancias donde la red se encuentra en un estado ideal, es decir, sin congestión, en comparación con aquellos

momentos en los que sí se presenta congestión. Este desequilibrio necesita ser abordado para asegurar la eficacia y precisión del modelo de aprendizaje automático.

Una vez recopilados, los datos se prepararon para su uso en algoritmos de aprendizaje automático. Este proceso incluyó la partición de los datos, en el cual los datos se dividieron en un 70% para entrenamiento y un 30% para pruebas. Esta división garantiza que haya datos suficientes para entrenar eficazmente los modelos y que quede una cantidad significativa para probar su rendimiento.

Los datos de entrenamiento se alinearon para evitar sesgos en los modelos. Esto es especialmente importante en situaciones en las que las clases (por ejemplo, "sobreadaptadas" y "no sobreadaptadas") no están uniformemente representadas en los datos brutos.

3.4.1.1**Normalización de datos para SVM**

La normalización de los datos es un paso crucial para el algoritmo SVM (Support Vector Machine). Esta técnica es necesaria debido a la sensibilidad de las SVM a las características a escala. En los modelos SVM, las características a gran escala pueden tener una influencia desproporcionada, distorsionando el hiperplano de decisión. Esto puede dar lugar a una clasificación inexacta. La normalización de los datos garantiza que cada característica contribuya por igual al modelo, lo que permite a la SVM determinar el hiperplano que separa las clases de la forma más eficaz. La normalización equilibra las escalas de atributos y garantiza que todas las características tengan el mismo peso a la hora de determinar el hiperplano, mejorando así la precisión y la eficacia del algoritmo SVM en la clasificación de los datos.

3.5 Validación del Diseño y la Metodología

Siguiendo los principios de la metodología CRISP-DM, el diseño del estudio se somete a un proceso de validación que comprende varias etapas clave para confirmar que las técnicas de machine learning seleccionadas y la configuración de la simulación de red en OMNeT++ son adecuadas para abordar el problema de congestión en redes de paquetes.

Primero, se revisa la coherencia entre la topología de red simulada y las métricas de rendimiento recopiladas, verificando que reflejen con precisión el comportamiento de una red real bajo diversas condiciones de tráfico. Luego, se evalúa la adecuación de las divisiones de datos de entrenamiento y validación, asegurándose de que proporcionen una base sólida para entrenar y probar los modelos de manera equitativa

La normalización aplicada a los datos para el algoritmo SVM y el balanceo de clases en el conjunto de entrenamiento se justifican y validan a través de pruebas empíricas que demuestran mejoras en la precisión y la generalización de los modelos. La efectividad de estas técnicas preparatorias se confirma mediante la observación de una mejora significativa en las métricas de evaluación utilizadas, tales como la precisión, el recall y el F1-Score.

Para garantizar una evaluación integral, se implementa un sistema de validación cruzada. Este enfoque no solo mejora la robustez de los modelos al exponerlos a múltiples subconjuntos de datos, sino que también permite una evaluación más fidedigna de su rendimiento general.

Finalmente, la validación del diseño y la metodología culmina con una comparación exhaustiva de los resultados obtenidos con cada técnica de machine learning. La comparación se basa en métricas estandarizadas que proporcionan una medida objetiva de la eficacia de los algoritmos. La convergencia de los resultados obtenidos con las expectativas teóricas y las simulaciones refuerza la validez de la metodología y del diseño experimental del estudio, consolidando la base sobre la cual se construyen las conclusiones y recomendaciones finales del trabajo de titulación.

Capítulo cuatro

Análisis Comparativo de Resultados de Técnicas de Machine Learning

En el contexto de nuestro trabajo de titulación, que se centra en la implementación y evaluación de algoritmos de control de congestión en redes utilizando técnicas de machine learning. Este capítulo se enfoca en el análisis comparativo de varios algoritmos de machine learning, específicamente Máquinas de Vectores de Soporte (SVM), Random Forest y Naïve Bayes, aplicados al control de congestión en redes de paquetes. El propósito es evaluar y comparar la efectividad de estos métodos en la clasificación y predicción de la congestión de red, una tarea crucial en la gestión y optimización de redes modernas. Cada uno de estos algoritmos ofrece enfoques únicos en el tratamiento de datos y la toma de decisiones, lo que justifica su inclusión y análisis comparativo en nuestro trabajo de titulación.

En la sección de trabajos relacionados, se destacó el uso de la matriz de confusión como una herramienta valiosa para simplificar el examen de resultados. Adicionalmente, se consideró el nivel de precisión de nuestro modelo, reconociendo que una tasa superior al 70% se considera confiable.

4.1 Algoritmo Support Vector Machine

El análisis comienza con la extracción de datos del archivo que contiene todo el conjunto de datos recopilados de la simulación de red. Este archivo es una compilación de métricas clave que son críticas para entender y predecir la congestión en redes de paquetes. Las métricas incluidas son utilización, tasa porcentual de paquetes perdidos y tiempo de espera en colas, estas métricas sirven como características de entrada para los modelos de machine learning.

Los datos se clasifican en dos categorías: congestión (clase minoritaria) y no congestión (clase mayoritaria). Esta clasificación se basa en umbrales predefinidos para las métricas mencionadas, identificando situaciones de congestión de red.

Para abordar el desequilibrio en los datos, se realiza un sobremuestreo de la clase minoritaria. Esto implica generar datos sintéticos o replicar datos existentes de la clase

minoritaria para igualar el número de muestras de la clase mayoritaria, reduciendo así el sesgo en el modelo.

Los datos se mezclan aleatoriamente para evitar cualquier sesgo de ordenamiento y luego se dividen en conjuntos de entrenamiento y prueba. Esta división es crucial para evaluar la capacidad del modelo para generalizar a datos no vistos.

Los datos se normalizan para asegurar que todas las características contribuyan equitativamente al modelo. La normalización implica ajustar los valores de las características para que tengan una escala común, lo cual es esencial para modelos que son sensibles a la escala de las características, como SVM.

El modelo se entrena con los datos normalizados de entrenamiento, buscando el hiperplano que mejor separa las clases de congestión y no congestión. La evaluación se realiza en el conjunto de prueba.

Implementamos una validación cruzada para evaluar la consistencia del modelo a través de diferentes subconjuntos de datos. Esto implica repetir el proceso de entrenamiento y evaluación varias veces, cada vez con un conjunto diferente de datos de entrenamiento y prueba, lo que proporciona una estimación más confiable del rendimiento del modelo.

4.1.1 ***Resultados SVM***

En esta sección, se presentan los resultados obtenidos mediante el uso del algoritmo de máquinas de soporte vectorial (SVM); cabe destacar que este algoritmo mostró la menor precisión, equivalente a 87.5%, en comparación con los demás métodos evaluados.

La Tabla 1 refleja los resultados estadísticos clave obtenidos del modelo SVM aplicado para la predicción de congestión en redes de paquetes.

Tabla 1

Tablas de datos normalizados para SVM

Utilización	Media: 38.3316
	Desviación estándar: 34.1162
	Mínimo: 0.050605
	Máximo: 98.6936
Tiempo de espera en colas	Media: 0.013078
	Desviación estándar: 0.043494
	Mínimo: 0
	Máximo: 0.24193
Paquetes Perdidos	Media: 38.2579
	Desviación estándar: 43.3432
	Mínimo: 0
	Máximo: 99.9339

Se observa que la métrica de utilización tiene una media de 38.3316, lo que sugiere un nivel moderado de uso de recursos de la red. La desviación estándar de 34.1162, junto con el mínimo y máximo registrados, indica una amplia variabilidad, lo cual es una consideración importante para la identificación de patrones de congestión.

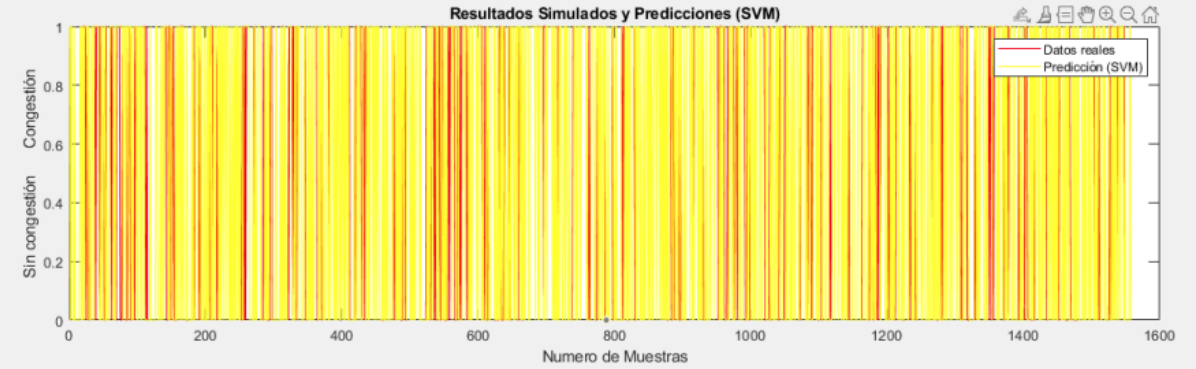
En cuanto al tiempo de encolamiento, la media reportada es de 0.013078, con una desviación estándar mínima y valores máximos que no exceden el umbral establecido para la congestión, lo que implica una gestión eficiente y una rápida respuesta del sistema ante las solicitudes de tráfico.

Por último, la métrica de paquetes perdidos muestra una media de 38.2579 con una desviación estándar significativa de 43.3432, reflejando una potencial variabilidad en la pérdida de paquetes que puede ser indicativa de eventos de congestión o problemas de estabilidad de la red. El rango de 0 a 99.9339 en esta métrica resalta la necesidad de una exploración más detallada para entender mejor las causas de estas pérdidas y su impacto en la predicción de congestión.

El clasificador SVM ha demostrado ser eficaz en la predicción de congestiones en la red, como se evidencia en la Figura 8 de resultados simulados.

Figura 8

Predicción de congestión en SVM

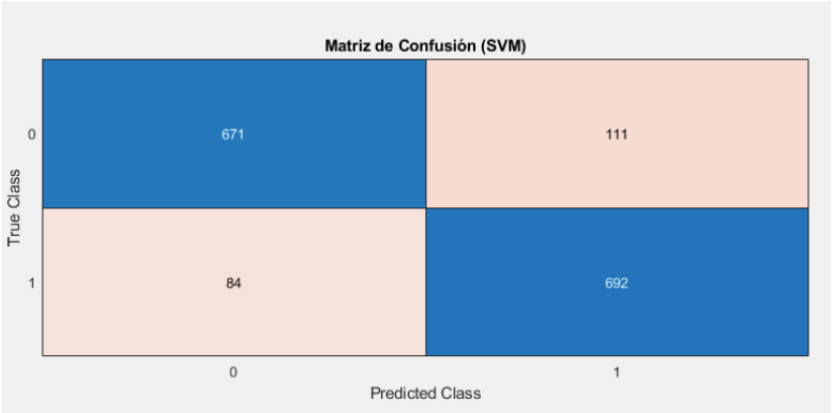


El gráfico compara las predicciones de congestión de red del modelo SVM (líneas amarillas) con los datos reales (líneas rojas) utilizando un sistema de valores discretos donde '1' indica congestión y '0' no congestión. La correspondencia entre las líneas amarillas y rojas a lo largo de las 1600 muestras indica la precisión del modelo en la detección de congestión. Las coincidencias directas reflejan predicciones correctas, mientras que las discrepancias señalan áreas para mejorar la precisión del modelo.

La matriz de confusión obtenida de la clasificación SVM (Figura 9), muestra una distinción efectiva entre los estados de congestión y no congestión de la red.

Figura 9

Matriz de confusión de SVM



Con 692 verdaderos positivos y 671 verdaderos negativos, el modelo ha identificado correctamente la mayoría de las instancias. aunque hay presencia de falsos negativos y positivos, los resultados indican un rendimiento satisfactorio, especialmente en la identificación de congestiones reales (clase 1).

4.2 Algoritmo Random Forest

La aplicación del algoritmo de Random Forest en el contexto del control de congestión en redes de paquetes comienza con el uso del conjunto de datos extraídos de la siulacion de los diferentes escenarios de red, que incluye las variables de red como utilización, tasa porcentual de paquetes perdidos y tiempo de espera en colas. Estas variables son empleadas como entradas para el modelo de Random Forest, y las etiquetas, definidas a partir de umbrales establecidos, señalan si existe o no congestión.

La metodología para la preparación de los datos comienza con la clasificación de los mismos en dos grupos: aquellos que representan congestión y aquellos que no. Se aplica un sobremuestreo a la clase minoritaria para balancear el conjunto de datos, un paso crucial para minimizar el sesgo en el modelo resultante. Una vez clasificados y balanceados, los datos se barajan y se dividen en conjuntos para entrenamiento y prueba, lo que permite evaluar la capacidad predictiva del modelo.

El modelo de Random Forest se entrena utilizando estos datos preparados. Este algoritmo se basa en la construcción de múltiples árboles de decisión, cada uno entrenado con una muestra aleatoria del conjunto de datos. La decisión final del modelo se basa en la votación mayoritaria entre todos los árboles, lo que contribuye a la robustez y precisión del modelo. Este enfoque de ensamble es particularmente efectivo para mejorar la precisión del modelo y reducir el riesgo de sobreajuste.

4.2.1 *Resultados de Random Forest*

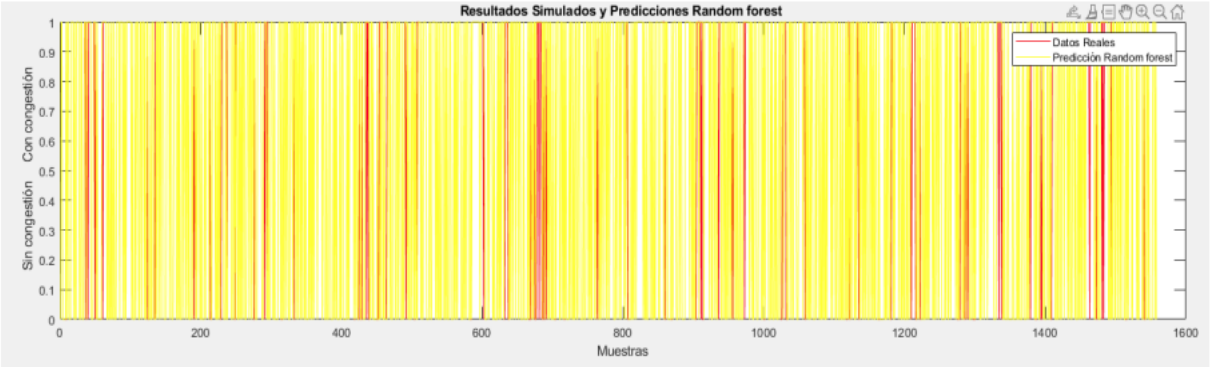
Esta sección expone los resultados obtenidos mediante el algoritmo random forest, el cual demostró ser el más preciso para este modelo con un 94.2% de precisión. Su eficacia en la clasificación es notable, y se destaca la posibilidad de aumentar el número de árboles para mejorar aún más la precisión. sin embargo, es importante mencionar el riesgo de

sobreajuste (overfitting) asociado con un incremento excesivo en el número de árboles, lo que requiere un cuidadoso equilibrio en función del tamaño del conjunto de datos.

Como se ilustra en la Figura 10, el algoritmo Random Forest sobresale por su mayor número de predicciones acertadas en comparación con otras técnicas.

Figura 10

Predicción de congestión usando el algoritmo Random Forest

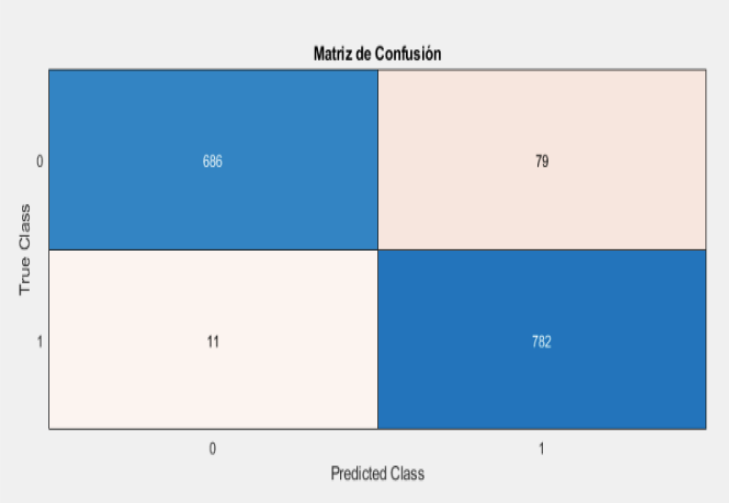


La correlación de colores entre las series indica un alto nivel de exactitud en la predicción de congestiones. Esto se debe principalmente a que el incremento en la cantidad de árboles en el modelo Random Forest contribuye significativamente a mejorar la precisión de sus predicciones

En la Figura 11 se presenta la matriz de confusión usando el algoritmo de Random Forest.

Figura 11

Matriz de confusión de Random Forest



La matriz muestra un alto número de predicciones correctas, con 686 casos correctamente identificados como clase '0' (no congestionados), y 782 casos correctamente clasificados como clase '1' (congestionados). Esto indica una alta tasa de verdaderos positivos y verdaderos negativos, reflejando la eficiencia del modelo en la clasificación de estados de la red.

4.3 Algoritmo Naïve-Bayes

La principal característica distintiva del algoritmo Naïve Bayes, en comparación con otros métodos como Random Forest o SVM, es su enfoque basado en la probabilidad y la simplicidad. Este algoritmo aplica el teorema de Bayes, asumiendo una independencia ingenua entre las características. A pesar de esta simplificación, Naïve Bayes puede ser sorprendentemente efectivo y es especialmente útil cuando se trabaja con conjuntos de datos de gran dimensión y se requiere un modelo que sea computacionalmente eficiente.

La preparación de los datos sigue un proceso similar al utilizado para otros modelos. Se clasifican los datos en categorías de congestión y no congestión, y se realiza un sobremuestreo de la clase minoritaria para equilibrar el conjunto de datos. Esta etapa es fundamental para garantizar que el modelo no esté sesgado hacia la clase mayoritaria. Posteriormente, los datos se mezclan y se dividen en conjuntos de entrenamiento y prueba.

Los resultados obtenidos con el modelo Naïve-Bayes son fundamentales para comprender cómo un enfoque basado en la probabilidad y la simplicidad puede ser efectivo en el análisis de congestión de redes. Este enfoque subraya la importancia de considerar diferentes metodologías en la investigación de este campo, especialmente cuando se requiere eficiencia computacional junto con una precisión razonable.

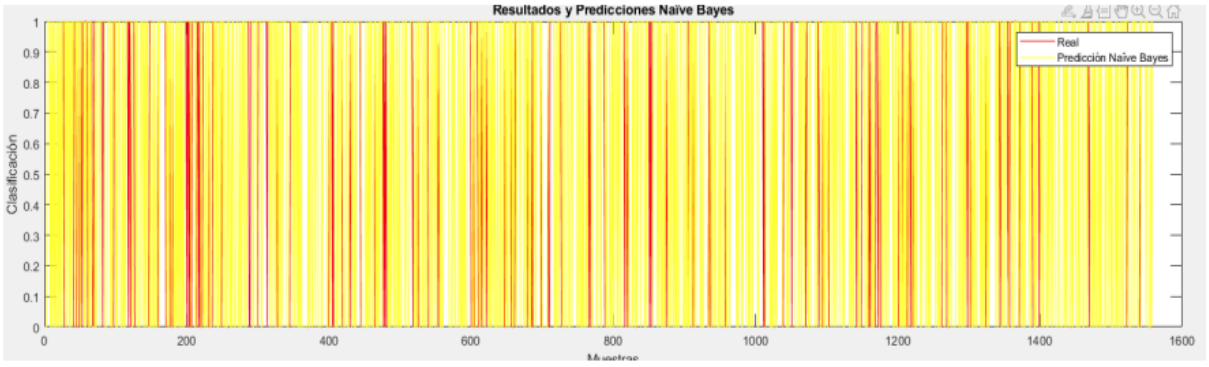
4.3.1 *Resultados de Naïve-Bayes*

El uso del clasificador Naïve-Bayes para determinar la presencia o ausencia de congestión en una red puede ser una opción eficaz dada su simplicidad y velocidad, especialmente en sistemas donde el tiempo de respuesta es crítico. Aunque este modelo se basa en la suposición de independencia entre las características, que raramente se cumple en situaciones reales, a menudo produce resultados sorprendentemente buenos en la práctica.

El clasificador Naïve-Bayes se destaca en la detección de congestiones en redes de paquetes por su rapidez y resultados fiables, a pesar de sus supuestos de independencia entre características. Demuestra un alto nivel de precisión, con un 91.6% en la predicción de congestiones.

Figura 12

Predicción de congestión usando el algoritmo Naïve-Bayes

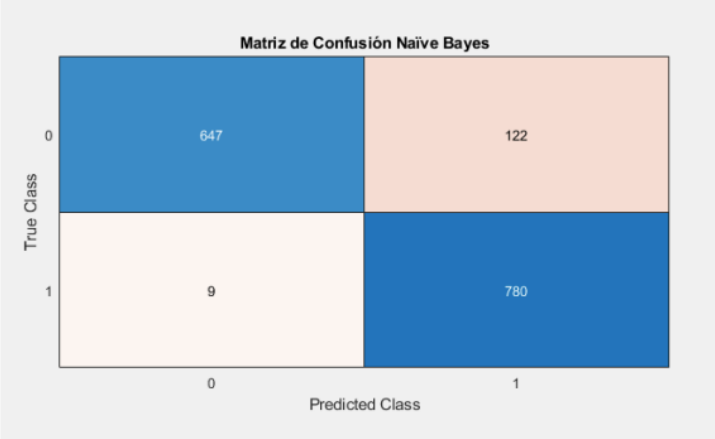


El análisis de la Figura 12 de resultados revela que el algoritmo evaluado emerge como la segunda mejor alternativa debido a su versatilidad y adaptabilidad a diversos

conjuntos de datos. Su robustez es particularmente notable ya que no requiere un equilibrio previo de los datos para funcionar eficientemente.

Figura 13

Matriz de confusión de Naïve-Bayes



La matriz de confusión (Figura 13) del clasificador Naïve-Bayes, se puede destacar que este fue el segundo mejor algoritmo que utilizamos, superando al SVM. La matriz muestra que Naïve-Bayes logró clasificar correctamente 647 casos como no congestionados (clase '0') y 780 como congestionados (clase '1'). si bien la cantidad de falsos positivos (122) es mayor en comparación con el clasificador Random Forest, el número de falsos negativos (9) es bastante bajo, lo cual es favorable ya que sugiere que el modelo es capaz de identificar de manera efectiva los casos más críticos de congestión.

4.4 Comparación de Resultados

La Tabla 2 proporciona una comparativa de tres algoritmos de clasificación SVM, Random Forest y Naïve-Bayes evaluados en base a diferentes métricas de rendimiento para la predicción de congestión en redes de paquetes.

Tabla 2

Comparativa de los algoritmos

Métricas de evaluación del algoritmo	SVM	Random Forest	Naïve-Bayes
Precisión (%)	87.5	94.2	91.6
Validación Cruzada (%)	87.2	94.4	91.1
Recall (%)	89.2	**95.7**	**92.9**
Especificidad (%)	85.8	90.1	84.1
F1-Score	0.88	0.92	0.92
AUC-ROC	**0.91**	0.94	0.92

El algoritmo Random Forest presenta el mejor rendimiento general, con una precisión del 94.2%, una validación cruzada del 94.4%, y especialmente destaca en recall con un 95.7%, indicando una alta proporción de positivos reales correctamente identificados. Su especificidad es del 90.1%, lo que demuestra su habilidad para identificar correctamente los negativos reales, y tanto el F1-Score como el AUC-ROC tienen valores altos de 0.92 y 0.94 respectivamente, lo que refleja un equilibrio entre la precisión y el recall, y una excelente capacidad de clasificación en comparación con una clasificación aleatoria.

El clasificador Naïve-Bayes, aunque con una precisión ligeramente inferior del 91.6%, mantiene resultados sólidos en todas las métricas, con un recall de 92.9% y un F1-Score de 0.92, indicando un buen balance entre precisión y recall. No obstante, su especificidad es la más baja con 84.1%, lo que podría sugerir una tasa más alta de falsos positivos comparado con Random Forest. Su AUC-ROC es de 0.92, lo que sigue siendo un resultado robusto.

Por último, el SVM muestra una precisión de 87.5%, la más baja entre los tres, con un recall y una especificidad de 89.2% y 85.8% respectivamente, lo que sugiere que mientras que es razonablemente bueno identificando los positivos reales, tiene más dificultades en la identificación correcta de los negativos reales en comparación con Random Forest. Su F1-Score y AUC-ROC son también los más bajos, con 0.88 y 0.91 respectivamente, lo que indica

que mientras que es un modelo competente, no tiene el mismo rendimiento que los otros dos algoritmos en este contexto.

Para un modelo de predicción de congestión en redes de paquetes, Random Forest es el más preciso y equilibrado en términos de rendimiento, seguido de cerca por Naïve-Bayes, con SVM como una opción más modesta. Estos resultados pueden influir en la selección del algoritmo dependiendo de la importancia relativa de las métricas para la aplicación específica y los costos asociados a falsos positivos y falsos negativos.

Conclusiones

La investigación comenzó con una exhaustiva revisión de literatura, que estableció una base sólida para comprender las dinámicas actuales del control de congestión y las aplicaciones de machine learning en este campo; lo cual confirmó la creciente importancia de los métodos de aprendizaje automático en el control de congestión y proporcionó una base sólida para el desarrollo del modelo predictivo. Se identificaron técnicas relevantes que contribuyen a una mejor comprensión y manejo de la congestión en redes de paquetes.

La implementación y configuración del modelo predictivo en MATLAB, integrando técnicas de machine learning como Random Forest, SVM y Naïve Bayes, se evaluaron con éxito mediante simulaciones en OMNeT++. Este modelo predictivo, adaptado para el control de congestión en redes de paquetes, no solo demostró su eficacia al alcanzar una alta precisión en la predicción de congestión, reflejada en tasas de 94% con Random Forest, 87% con SVM y 91% con Naïve Bayes, sino que también incorporó de manera efectiva parámetros de umbral críticos como la utilización, el tiempo de encolamiento y la tasa de pérdida de paquetes. A pesar de que el desarrollo del algoritmo tomó rumbos distintos al plan inicial, los resultados consolidaron la efectividad de las técnicas de machine learning seleccionadas, cada una aportando soluciones significativas a diferentes aspectos del problema de congestión de la red.

La evaluación comparativa de los algoritmos propuestos reveló que el modelo basado en Random Forest superó a los demás en términos de precisión general y manejo de datos no lineales. La validación cruzada reforzó la confiabilidad de los modelos, destacando la robustez de los métodos de machine learning aplicados.

Al mejorar la gestión de la congestión de la red, este trabajo tiene el potencial de impactar positivamente en la infraestructura de comunicaciones, soportando así el creciente tráfico de datos y contribuyendo al avance tecnológico y a la mejora de la calidad de vida en una sociedad cada vez más conectada digitalmente.

Recomendaciones

Se recomienda realizar revisiones de literatura periódicas para mantenerse actualizado sobre las últimas tendencias y descubrimientos en el control de congestión y machine learning, asegurando que futuros estudios estén alineados con los desarrollos más recientes en la disciplina.

Se sugiere ampliar la gama de escenarios de topologías de red en futuras investigaciones para explorar cómo los modelos de machine learning se adaptan a diversas condiciones de red y a configuraciones topológicas más complejas.

Se aconseja incrementar el número de muestras de datos para mejorar la robustez de los modelos predictivos y para validar más exhaustivamente la efectividad de las técnicas de machine learning en entornos de red variados.

Sería beneficioso adoptar un enfoque iterativo que permita ajustes y refinamientos continuos del diseño metodológico en respuesta a los resultados emergentes, mejorando así la precisión y la aplicabilidad de los modelos.

Para futuras aplicaciones del modelo SVM, se recomienda investigar y aplicar diferentes métodos de normalización y su impacto en el modelo, lo que podría optimizar aún más el rendimiento del algoritmo.

Para validar aún más la eficacia del modelo, sería ideal implementarlo en un entorno de red real y en tiempo real. Esto permitiría evaluar su capacidad predictiva y de respuesta bajo condiciones dinámicas y fluctuantes, proporcionando una prueba más rigurosa de su utilidad práctica en situaciones de congestión de red imprevistas.

Se recomienda explorar la integración de técnicas de inteligencia artificial más avanzadas, como el aprendizaje profundo, para abordar la congestión en redes de paquetes, lo que podría abrir nuevas vías de investigación y desarrollo en el campo.

Referencias

A. Rattalino, D. J. (2014). Simulación de Algoritmos de Control de Congestión enTCP bajo User Mode Linux. *Proyecto Final de la Carrera Ingeniería en Sistemas.* Universidad Nacional de La Pampa, La pampa. Obtenido de https://repo.unlpam.edu.ar/bitstream/handle/unlpam/2480/i_ratsim037_c.pdf?sequence=1

Becerra, B. (11 de diciembre de 2021). *La republica*. Obtenido de La republica: https://www.larepublica.co/consumo/consumo-de-internet-en-el-mundo-aumento-19-5-durante-la-pandemia-de-covid-19-3274945

Bonaventure, O. (2018). *Computer Networking : Principles,Protocols and Practice.* Washington: Saylor foundation.

Breiman, L. (2001). Random Forests. (R. Schapire, Ed.) *Statistics Department, University of California*, 3-28. Obtenido de https://link.springer.com/content/pdf/10.1023/A:1010933404324.pdf

Cadin, V., & Talay, C. (2021). Evolución de los algoritmos de control de congestión en las distintas variantes del protocolo TCP. *Instituto de Tecnología Aplicada*, 1-20.

Cath, C. (11 de abril de 2021). The technology we choose to create: Human rights advocacy in the Internet Engineering Task Force. *Telecommunications Policy*, 45.

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. (L. Saitta, Ed.) *AT&T Bell Labs*, 3-9. Obtenido de https://link.springer.com/content/pdf/10.1007/bf00994018.pdf

Floyd, S., & Jacobson, V. (1993). Random Early Detection Gateways. *IEEE/ACM TRANSACTIONS ON NETWORKING*, 12-17.

Goddard, J., J.M., C., F.Martínez, & Martínez, A. (1995). Redes Neuronales y Árboles de Decisión: Un Enfoque Híbrido. *emorias del Simposium Internacional de Computación organizado por el Instituto Politécnico Nacional*, 2-7.

Godoy, Á. (2015). Técnicas de aprendizaje de máquina utilizadas para la minería de texto. *Recuperación de la Información y Tecnología Avanzadas (RITA) de la Universidad Federal de Santa Catarina*, 1-24.

Hayes, D. A., & Armitage, G. (2017). Revisiting TCP Congestion Control Using Delay. *INRIA.HAL.SCIENCE*, 1-16. doi:10.1007/978-3-642-20798-3_25

Jacobson, V. (1998). Congestion avoidance and control. *ACM digital library*, 1-16.

Jaeger, B., Scholz, D., Raumer, D., Geyer, F., & Carle, G. (2019). Reproducible Measurements of TCP BBR Congestion Control. *Technical University of Munich*, 1-16.

Kaneko, K., Fujikawa, T., Su, Z., & Katto, J. (2007). TCP-Fusion: a hybrid congestion control algorithm for high-speed networks. *PFLDnet*, 31-36.

Kaur, R., & Josan, G. S. (Octubre de 2012). Performance Evaluation Of Congestion Control Tcp Variants In Vanet Using Omnet++. *International Journal of Engineering Research and Applications (IJERA)*, 1-7.

León, D. A., & Martínez, J. G. (2022). Inteligencia artificial para el control de tráfico en redes de datos una Revisión. *Entre Ciencia e Ingeniería*, 1-8.

Lugones, D. (2006). CONTROL DE CONGESTIÓN ADAPTATIVO EN REDES INFINIBAND. *Doctorado en Informática.* Universidad autonoma de Barcelona, Barcelona.

Miao, J., & Zhu, W. (2022). *Precision–recall curve (PRC) classification trees* (Vol. 3). Evolutionary Intelligence. doi:10.1007/s12065-021-00565-2

Peña, A. (2021). *Indicadores de tecnología de la información y comunicación.* Quito: INEC.

Perrier, V. (2023). LEO/GEO congestion control mechanism based on the contribution of cognitive sciences(Doctoral dissertation,ISAE). *Mécanisme de contrôle de congestion LEO/GEO basé sur l'apport des sciences.* ISAE, Toulouse.

Ramos, L., & RonaldMárquez. (2023). AI's next frontier: The rise of ChatGPT and its implications on society, industry, and scientific research. *Revista Ciencia e Ingeniería*, 1-18.

Rojas, E. M. (2020). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo. *Iberian Journal of Information Systems and Technologies*, 1-15.

Russell, S., & Norvig, P. (2004). *Inteligencia Artificial: Un Enfoque Moderno* (Segunda edición ed.). (D. F. Aragón, Ed., & J. M. Rodríguez, Trad.) Madrid: Pearson Prentice Hall. Obtenido de

https://luismejias21.files.wordpress.com/2017/09/inteligencia-artificial-un-enfoque-moderno-stuart-j-russell.pdf

Schröer, C., Kruse, F., & Gómez, J. M. (2021). *A systematic literature review on applying CRISP-DM process model.* Wolfsburg: Procedia Computer Science. doi:10.1016/j.procs.2021.01.199

Singh, Y., & Kaur, L. (2017). Obstacle Detection Techniques in Outdoor Environment: Process, Study and Analysis. *Modern Education and Computer Science Press*, 1-19.

Tanenbaum, A. (2003). *Redes de computadoras.* Amsterdam: Pearson Education India.

Valle, F. (2010). IMPLEMENTACIÓN EFICIENTE DE CLASIFICADORES PRIOR-SVM PARA MATLAB. *INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN.* UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITÉCNICA SUPERIOR, Madrid.

Wehrstein, L., & Bachmann, B. (24 de Diciembre de 2021). *CRISP-DM ready for Machine Learning Projects - Towards Data Science*. Obtenido de Medium: https://towardsdatascience.com/crisp-dm-ready-for-machine-learning-projects-2aad9172056a#:~:text=CRISP,the%20model%20were%20not%20considered

Yağmur, N., Dag, I., & Temutas, H. (2023). A New Computer-Aided Diagnostic Method for Classifying Anemia Disease: Hybrid Use of Tree Bagger and Metaheuristics. *Authorea Preprints*, 1-6. Obtenido de https://www.authorea.com/users/652444/articles/659972-a-new-computer-aided-diagnostic-method-for-classifying-anemia-disease-hybrid-use-of-tree-bagger-and-metaheuristics

Zhang, T., & Mao, S. (2020). Machine Learning for End-to-End Congestion Control. *IEEE xplore*, 52-57.

Zhang, Y., & Lorenz, P. (29 de Noviembre de 2018). AI for Network Traffic Control. *IEEE Network*, 6-7. doi:10.1109/MNET.2018.8553647