UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

MAESTRÍA EN TELECOMUNICACIONES

Tema: RUTA ÓPTIMA DE TRÁFICO DE UNA RED VIRTUAL BASADA EN ANÁLISIS DE DATOS Y ALGORITMO DE MACHINE LEARNING

Trabajo de titulación previo a la obtención del grado académico de Magíster en Telecomunicaciones
Modalidad de Titulación "Proyecto de desarrollo"

Autor: Ingeniera Carla Patricia Chávez Fuentes
Director: Ingeniero Alberto Ríos Villacorta, PhD

Ambato - Ecuador
2021

APROBACIÓN DEL TRABAJO DE TITULACIÓN

A la Unidad Académica de Titulación de la Facultad de Ingeniería en Sistemas Electrónica e Industrial.

El Tribunal receptor del Trabajo de Titulación presidido por la Ingeniera Elsa Pilar Urrutia Urrutia, Magíster, e integrado por los señores Ingeniero David Omar Guevara Aulestia Magíster e Ingeniero Víctor Santiago Manzano Villafuerte Magíster, designados por la Unidad Académica de Titulación de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, para receptar el Trabajo de Titulación con el tema: "RUTA ÓPTIMA DE TRÁFICO DE UNA RED VIRTUAL BASADA EN ANÁLISIS DE DATOS Y ALGORITMO DE MACHINE LEARNING", elaborado y presentado por la señorita Carla Patricia Chávez Fuentes, para optar por el Grado Académico de Magíster en Telecomunicaciones; una vez escuchada la defensa oral del Trabajo de Titulación el Tribunal aprueba y remite el trabajo para uso y custodia en las bibliotecas de la Universidad Técnica de Ambato.

Firmado electrónicamente por:
ELSA PILAR URRUTIA

Ing. Elsa Pilar Urrutia Urrutia, Mg.
Presidente y Miembro del Tribunal de Defensa

Firmado electrónicamente por:
DAVID OMAR GUEVARA AULESTIA

Ing. David Omar Guevara Aulestia, Mg.
Miembro del Tribunal de Defensa

Firmado electrónicamente por:
VICTOR SANTIAGO MANZANO VILLAFUERTE

Ing. Víctor Santiago Manzano Villafuerte, Mg.
Miembro del Tribunal de Defensa

AUTORÍA DEL TRABAJO DE TITULACIÓN

La responsabilidad de las opiniones, comentarios y críticas emitidas en el Trabajo de Titulación presentado con el tema: "Ruta óptima de tráfico de una red virtual basada en análisis de datos y algoritmo de machine learning", le corresponde exclusivamente a la: Ingeniera, Carla Patricia Chávez Fuentes, Autor bajo la Dirección del Ingeniero, Alberto Ríos Villacorta, PhD, director del Trabajo de Titulación; y el patrimonio intelectual a la Universidad Técnica de Ambato.

Firmado electrónicamente por:
CARLA PATRICIA CHAVEZ FUENTES

Ing. Carla Patricia Chávez Fuentes

AUTOR

Firmado electrónicamente por:
PERCY ALBERTO RIOS VILLACORTA

Ingeniero Alberto Ríos Villacorta, PhD

DIRECTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que el Trabajo de Titulación, sirva como un documento disponible para su lectura, consulta y procesos de investigación, según las normas de la Institución.

Cedo los Derechos de mi trabajo, con fines de difusión pública, además apruebo la reproducción de este, dentro de las regulaciones de la Universidad.

Firmado electrónicamente por:
CARLA PATRICIA CHAVEZ FUENTES

Ing. Carla Patricia Chávez Fuentes
c.c. 1804792602

ÍNDICE GENERAL

Contenido

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

AGRADECIMIENTO

A Dios por darme la oportunidad de vivir muchas cosas hermosas y haber conocido a mucha gente maravillosa.

A mi Familia, a mi Director de Trabajo de Titulación Dr. Alberto Ríos Villacorta por su apoyo y guía en el proceso del desarrollo del trabajo, a mis amigos, a mis compañeros, a todos quienes han hecho posible que yo siga adelante con sus consejos y cariño.

A mi novio Jesús por su cariño y apoyo, en todos los momentos difíciles de mi vida y también a su familia que siempre me ha llenado de afecto.

A toda mi familia, en especial a mi madre, por su esfuerzo soy la mujer que soy ahora. A mis hermanas y a mi sobrina por su cariño y palabras de aliento cada día.

DEDICATORIA

Dedicado a mi novio Jesús, por siempre apoyarme y ser mi compañero, en los buenos y malos momentos, este logro es de los dos.

A mi madre, que cada día se esfuerza por darnos lo mejor a mí y a mis hermanas, me faltará la vida para agradecerte.

A todos mis docentes y compañeros, de toda mi vida académica.

A mi amiga Anita, que siempre estará en mi corazón, y que con su partida se llevó un pedacito de mi vida.

UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL
MAESTRÍA EN TELECOMUNICACIONES

TEMA:

RUTA ÓPTIMA DE TRÁFICO DE UNA RED VIRTUAL BASADA EN ANÁLISIS DE DATOS Y ALGORITMO DE MACHINE LEARNING

AUTOR: Ing. Carla Patricia Chávez Fuentes
DIRECTOR: Ing. Alberto Ríos Villacorta, PhD

LINEA DE INVESTIGACION:

- TECNOLOGÍAS, SEGURIDAD Y REDES DE COMUNICACIONES

FECHA: 29 de julio de 2021

RESUMEN EJECUTIVO

El presente tema de investigación tiene como objetivo la determinación de la ruta óptima de tráfico de una red virtual basada en análisis de datos y algoritmos de machine learning mediante la predicción de tráfico. Al igual que en las redes físicas en las redes virtuales se busca el óptimo diseño de una red de comunicación. El optimizar una red de datos no es tarea fácil, debido a la complejidad y la cantidad de factores necesarios a evaluar para obtener una solución que satisfaga a los usuarios finales [1] [2]. Por lo tanto, en esta propuesta se plantea realizar una determinación de una ruta óptima de tráfico de una red virtual, con la finalidad de reducir el consumo innecesario del ancho de banda de una red y otros recursos. El análisis de tráfico de una red es un punto clave para la determinación de su arquitectura, por lo que se ha convertido en un tema de gran interés en estos años [3]. El gran avance tecnológico que se ha presentado en los últimos años, el desarrollo de la inteligencia artificial y el uso de algoritmos de aprendizaje de máquina conocidos por sus siglas en inglés como machine learning (ML), han permitido resolver varios problemas en el área de la ingeniería y de la

informática [4]. Bajo esta premisa se propone hacer uso de algoritmos de ML para determinar la ruta óptima de tráfico en una red virtual mediante su análisis de tráfico y estado de esta. Para determinar la ruta óptima del tráfico de la red se pretende realizar una función objetivo que minimice el tráfico en cada nodo aprovechando al máximo el ancho de banda determinado. El análisis de la topología y arquitectura de la red permitirá determinar la ruta óptima para el tráfico de la red con la finalidad de que el diseño satisfaga las necesidades de los usuarios finales.

Descriptores: Redes virtuales, aprendizaje de máquina, tráfico de red, flujo de tráfico, ruta óptima.

UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL
MAESTRÍA EN TELECOMUNICACIONES

THEME:

ÓPTIMAL TRAFFIC PATH OF A VIRTUAL NETWORK BASED ON DATA ANALYSIS AND MACHINE LEARNING ALGORITHM

AUTHOR: Ing. Carla Patricia Chávez Fuentes
PRINCIPAL: Ing. Alberto Ríos Villacorta, PhD
LINE OF RESEARCH:

- TECHNOLOGIES, SECURITY AND COMMUNICATIONS NETWORKS

DATE: July 29th, 2021

EXECUTIVE SUMMARY

The present research topic aims to determine the óptimal traffic route of a virtual network based on data analysis and machine learning algorithms through traffic prediction. As in physical networks, in virtual networks, the óptimal design of a communication network is sought. Optimizing a data network is not an easy task, due to the complexity and the number of factors necessary to evaluate to obtain a solution that satisfies the end users [1] [2]. Therefore, in this proposal it is proposed to determine an óptimal traffic route of a virtual network, in order to reduce the unnecessary consumption of network bandwidth and other resources. Traffic analysis of a network is a key point in determining its architecture, which is why it has become a topic of great interest in recent years [3]. The great technological advance that has occurred in recent years, the development of artificial intelligence and the use of machine learning algorithms known by its acronym in English as machine

learning (ML), have made it possible to solve several problems in the area of engineering and computer science [4]. Under this first, it is proposed to make use of ML algorithms to determine the óptimal route of traffic in a virtual network through its analysis of traffic and its status. To determine the óptimal route for network traffic, an objective function is intended to minimize the traffic in each node, making the most of the determined bandwidth. The analysis of the topology and architecture of the network will allow to determine the óptimal route for the network traffic so that the design meets the needs of the end users.

Descriptors: Virtual networks, machine learning, network traffic, traffic flow, óptimal route.

ÍNDICE DE ABREVIATURAS

WAN: Wide Area Network - Red de Área Amplia

ISDN: Red digital de servicios integrados

LAN: Local Area Network - Red de Área Local

TI: Tecnología de la información

VNT: Virtual Network Topologies – Topologías de redes virtuales

IP: Internet Protocol – Protocolo de Internet

ML: Machine Learning – Aprendizaje de Máquina

DNS: Domain Name System - Sistema de Nombres de Dominio

TCP/IP: Transmission Control Protocol/Internet Protocol - Protocolo de control de transmisión/Protocolo de Internet

TOS: Type of Service - Tipo de Servicio

MF: More Fragments – Más Fregmentos

DF: Don't Fragment – No Fragmentos

TTL: Time to live - Tiempo de vida

ICMP: Internet Control Message Protocol – Mensaje de Control del Protocolo de Internet.

UDP : User Data Protocol - Protocolo de Datos de Usuario

OSPF: Open Shortest Path First - El Camino Más Corto Primero

ARPANET: Advanced Research Projects Agency network – Red de Agencias de Proyectos de Investigación Avanzada.

S.A: Sistema Autónomo

EGP: Exterior Gateway Protocol - Protocolo de enrutamiento exterior

IGP: Internal Gateway Protocol - Protocolo de enrutamiento interior

RIP: Routing Information Protocol - Protocolo de información de enrutamiento

VLSM: Variable Length Subnet Mask - Máscara de Subred de Longitud Variable

CIDR: Classless Inter-Domain Routing - Encaminamiento inter-dominios sin clases,

EIGRP: Enhanced Interior Gateway Routing Protocol - Protocolo de enrutamiento de Gateway interior mejorado.

IS-IS: Intermediate System to intermediate System – Sistema Intermedio a Sistema Intermedio.

VMM: Virtual Machine Monitor – Monitor de Máquina Virtual

S.O: Sistema Operativo

RAM: Random Access Memory - Memoria de Acceso Aleatorio.

VPN: Virtual Private Network - Red privada virtual

VLAN: Virtual Local Area Network - Red de Área Local Virtual

VM: Virtual Machine - Máquina Virtual

VXLAN: Virtual Extensible Local Area Network - Red de Área Local Virtual Extendible

GNS3: Graphic Network Simulation - Simulación Gráfica de Redes

IOS: Internetwork Operating System - Sistema Operativo de Interconexión de Redes

DHCP: Dynamic Host Configuration Protocol - Protocolo de Configuración Dinámica de Host.

TE: Traffic Engineering – Ingeniería de Tráfico.

QoS: Quality of Service - Calidad de servicio

MTU: Maximum Transfer Unit - Unidad máxima de transmisión.

PHP: Hypertext Pre-Processor - Pre-Procesador de Hipertexto

SVM: Support Vector Machine - Máquina de soporte vectorial

GRP: Gaussian Process Regression Mode – Proceso de modo Regresivo Gausiano.

SAW: Simple Additive Weighting - Método de la Suma Ponderada

CAPÍTULO I

EL PROBLEMA DE INVESTIGACIÓN

1.1. Introducción

El mundo ha cambiado mucho en las últimas décadas y en lugar de simplemente abordar problemas locales o regionales, muchas empresas ahora deben tener en cuenta la logística y los mercados globales [1]. Muchas empresas cuentan con instalaciones distribuidas en todo el país o, incluso, en todo el mundo [5]. Pero hay algo que todas las empresas necesitan: una forma de mantener comunicaciones rápidas, seguras y fiables en donde se encuentran sus oficinas [6].

Hasta hace poco, una comunicación fiable implicaba el uso de líneas arrendadas para mantener una red de área amplia (WAN). Las líneas arrendadas, desde la Red digital de servicios integrados (ISDN, que se ejecuta a 144 KB/s) hasta la fibra óptica Carrier-3 (OC3, que se ejecuta a 155 Mbps), ofrecen a una empresa una forma de ampliar su red privada más allá de su zona geográfica inmediata [7]. Una WAN tiene ventajas claras en relación con una red pública como Internet cuando se trata de fiabilidad, rendimiento y seguridad; pero mantener una WAN, especialmente cuando se utilizan líneas arrendadas, puede volverse bastante costoso (a menudo, aumenta el costo a medida que aumenta la distancia entre las oficinas) [8].

A medida que ha ido aumentando la popularidad de Internet, las empresas han recurrido a esta como medio para ampliar sus propias redes, y se han implementado las redes virtuales [10]. En los últimos años las redes virtuales han presentado una gran acogida por empresas tanto públicas y privadas en el país. Desde la declaración de la pandemia por la covid 19, que se presentó en 2019, varias medidas sanitarias fueron implementadas por los países reducir la propagación del virus. Entre una de las medidas de emergencia se ha prohíbo a la gente asistir a lugares concurridos y cerrados, por lo que la única opción para las empresas era que sus empleados se acojan al teletrabajo. La mayoría de las empresas tanto públicas y privadas en el país sólo contemplan un trabajo de forma presencial, estas no tienen una infraestructura que permita a los empleados conectarse desde sus hogares.

Las redes virtuales se han presentado como una gran solución para el desarrollo del teletrabajo. La implementación de estas redes permite que el empleado pueda realizar las mismas actividades que las realizaba de forma presencial, permitiéndole acceder a: sistemas, programas, bases de datos, aplicaciones y recursos de la empresa. Esto exige que las redes virtuales no se ven limitadas por cableado o hardware para la gestión y uso del software de una empresa. Al igual que las redes físicas las redes virtuales se deben administrar de manera eficiente permitiendo que el usuario tenga una conexión rápida, segura y eficiente. Esto se puede lograr gracias al análisis de tráfico realizado en la red y a la implementación de nuevos e innovadores algoritmos de enrutamiento.

El desarrollo de la investigación se centra en el análisis y predicción del tráfico mediante algoritmos de machine learning para la determinación de la ruta óptima en una red virtual. En el Capítulo 1, se presenta la justificación y los objetivos a desarrollar dentro del tema de investigación. En el Capítulo 2, se presenta el estado del arte y el desarrollo del marco conceptual adecuado para conocer los conceptos más relevantes para el desarrollo del trabajo. En el Capítulo 3, se presenta la metodología usada para la investigación. El desarrollo de la tesis, determinación y aplicación de los algoritmos de machine learning para la predicción del tráfico, así como, la descripción y validación del algoritmo de la ruta óptima son validados y puestas a prueba en el Capítulo 4. Por último, en el Capítulo 5, se presentan las conclusiones y recomendaciones obtenidas a lo largo del desarrollo del trabajo.

1.2. Justificación

Las comunicaciones en estos días integran una gran cantidad de dispositivos móviles que interactúan mediante el flujo de datos. Los movimientos, adiciones y supresiones en un ambiente virtual se logran a través del software, para eliminar la tarea dispendiosa y costosa de la reconexión física [8]. La virtualización de las redes implementadas de manera adecuada permite mejorar de aproximadamente entre 5l 5 al 15% de su capacidad operativa. La virtualización de redes es mucho más que VLAN, lo que hace posible la creación de redes enteras en software, incluida la conmutación, el enrutamiento, el firewall y el equilibrio de carga, hacerlo más cerca de la aplicación y orquestar el proceso de forma predeterminada El principal hardware para brindarle

apoyo a la idea de las redes virtuales es el conmutador [11]. Los nodos y segmentos de LAN conectados a un conmutador pueden agruparse lógicamente en segmentos virtuales [12]. La necesidad de nuevos servicios y optimización de utilización de recursos de redes ha permitido que en los últimos años se dediquen varios esfuerzos al análisis de las redes virtuales abordados desde nuevos conceptos [13]. Uno de los principales problemas al hacer uso de redes virtuales es la regulación de tráfico de la red, misma que se genera la gran cantidad de máquinas y dispositivos que pueden ser virtualizadas [14] . Para afrontar este problema es que en los últimos años han surgido propuestas para la aplicación de técnicas de inteligencia artificial con la finalidad de reducir el tráfico de las redes virtuales. La sobre utilización de los enlaces y el desbalance de la carga en cada uno de ellos es una de las principales razones para el estudio de nuevos algoritmos de enrutamiento con la finalidad de determinar una ruta óptima de tráfico.

1.3. Objetivos

1.3.1. General

Determinar la ruta óptima de tráfico para una red virtual, usando análisis de datos y algoritmos de machine learning.

1.3.2. Específicos.

• Identificar las métricas-parámetros más relevantes que permitan determinar el tráfico de una red virtual.

• Analizar el tráfico generado en una red virtual en base a la arquitectura de la red.

• Evaluar un algoritmo de aprendizaje que permita determinar el tráfico de una red virtual.

• Selección de la ruta óptima de una red virtual mediante el algoritmo de aprendizaje evaluado

CAPÍTULO II

ANTECEDENTES INVESTIGATIVOS

2.1. Estado del Arte

Para mejorar la competitividad que tienen cada una de las empresas es necesario ser más ágiles en los centros de datos locales. El uso de nuevas tendencias como las redes definidas por software y la virtualización de las redes tradicionales permiten mejorar las limitaciones que presenta las redes actuales, reduciendo las amenazas de seguridad y el aumento incontrolado de los costos en el despliegue de una red física. La virtualización de las redes ha reescrito las reglas sobre los servicios de las tecnologías de la Información (TI). El nuevo y mejorado enfoque que otorgan las redes virtuales, así como los data center definidos por software (SDDC), transforman los centros de datos estáticos inflexibles e ineficientes en centros dinámicos, ágiles y optimizados [17].

En este nuevo mundo, la virtualización de datos permite que la inteligencia de la infraestructura del centro de datos pase del software al hardware, esto quiere decir que todos los elementos de la infraestructura de las TI, incluido el cómputo, las redes y el almacenamiento, se virtualiza en forma de grupo de recursos. Los recursos que se han virtualizado se pueden implementar de manera automática, con poca participación humana en comparación al despliegue de una red física. Una de las principales características de un entorno virtualizado es su flexibilidad, automatización y su control por software.

En 2014 el Grupo Taneja realizó un estudio titulado, "Transformar el centro de datos con la suite vCloud del centro de datos definido por software de VMware", en el cual se encontró que los SDDC ofrecen una reducción del 56 por ciento en los costos operativos anuales de aprovisionamiento y administración. Aún mejor, los enfoques definidos por software pueden reducir el tiempo necesario para aprovisionar una red de producción para una nueva aplicación de tres o cuatro semanas a cuestión de minutos [17].

Uno de los desafíos más importantes en el mundo actual es cómo medir el rendimiento de las infraestructuras de redes informáticas, cuando se fusionan diferentes tipos de redes. En los últimos años, las redes orientadas a datos evolucionaron hacia estructuras

convergentes, en las que el tráfico en tiempo real, como las llamadas de voz o las videoconferencias, son cada vez más importantes. La estructura está compuesta por cables tradicionales o redes virtuales que deben ser monitoreadas para analizar los parámetros específicos que permiten un servicio adecuado y estos son: el ancho de banda, el retardo, la fluctuación y la pérdida de paquetes [18].

En la actualidad, la evolución de las bases de datos sumadas a la integración de dispositivos inteligentes que tienen una conexión a internet ha provocado una explosión en la generación de datos heterogéneos incrementando el tráfico en las redes. El incremento de usuarios en la red requiere nuevas soluciones para analizar y comprender mejor el tráfico, permitiendo tener una red ágil, segura y eficiente[19]. Los estudios más relevantes realizados para determinar la ruta óptima de tráfico usando machine learning y análisis de datos son presentados en los siguientes párrafos.

En el año 2019, en [20] se presentó un estudio de Predicción del tráfico de una red inalámbrica basada en redes neuronales artificiales mediante el algoritmo de Levenberg-Marquard. En el trabajo, se afirma que las VNT topologías de red virtual permiten la introducción de nuevos servicios como televisión y video en directo, pero cada día se requiera mayor conectividad y cambios de dirección del tráfico, por lo que las VNT deben adaptarse a la variación del volumen y dirección del tráfico. En el presente trabajo se propone que la VNT se adapte mediante predicción de tráfico, en síntesis, formar un bucle para observar-analizar-actuar. Se estudian dos enfoques para reconfigurar una VNT, i) Estático, la topología se dimensiona con el máximo tráfico diario, la capacidad del VNT permanecerá mayormente infrautilizada. ii) Basada en un umbral, donde los vlinks se adaptan según la necesidad y no entregan un recurso constante, se establece un umbral de tráfico promedio (100Gb), se observa que al sobrepasar este umbral se crea un enlace dedicado para los nodos 6 y 7 donde se está generando un tráfico no habitual y este enlace se mantendrá únicamente hasta que se genere un tráfico normal. Los datos son recogidos por los enrutadores IP del borde cada cierto tiempo, dichos datos se almacenan en un depósito de datos recogidos. Un módulo de predicción basado en machine learning genera una matriz de tráfico OD, cuando el algoritmo encuentre la mejor solución, el controlador implementa los cambios en la estructura de la red

Por otro lado, en [19] se presenta una análisis centrado en los datos de la red con el objetivo de definir los segmentos con mayor tráfico en base al comportamiento del flujo de tráfico. En este estudio se analiza un conjunto de datos reales con más de 3 millones de instancias, por lo que se realiza una caracterización de las variables, de las cuales se obtienen 15 características relevantes de un total de 87. Para la caracterización de las instancias se usó el algoritmo de aprendizaje automático (ML) como es la de vecinos cercanos. La ampliación del algoritmo permite comprender y distinguir mejor los comportamientos del tráfico. Los resultados demostraron una buena correlación entre instancias en el mismo grupo generada por el aprendizaje no supervisado. Esta solución se puede integrar aún más en un entorno real mediante la virtualización de funciones de red.

Así mismo, en el año 2020 en [21], se propone la gestión de recursos de redes virtuales basada en un enfoque de aprendizaje automático. En la investigación se modela la red de sustrato como un sistema descentralizado que introduce un algoritmo de aprendizaje en cada nodo de sustrato y enlace de antenas, proporcionando capacidades de autoorganización. El algoritmo de aprendizaje multiagente que realiza la gestión de los recursos de la red de antenas de forma coordinada y descentralizada. La tarea de estos agentes es utilizar la retroalimentación evaluativa para aprender una política óptima a fin de asignar dinámicamente recursos de red a nodos y enlaces virtuales. Los agentes se aseguran de que las redes virtuales cuenten con los recursos que necesitan en un momento dado, solo se reservan para este fin los recursos requeridos. La gestión de los recursos de la red se basa en un algoritmo de aprendizaje por refuerzo que incluye políticas de inicio para mejorar la tasa de convergencia en el aprendizaje que permite la asignación de recursos de forma dinámica. Las simulaciones muestran que el enfoque dinámico mejora significativamente la tasa de aceptación de la red virtual y el número máximo de solicitudes de red virtual aceptadas en cualquier momento, al tiempo que garantiza que los requisitos de calidad del servicio de la red virtual, como la tasa de caída de paquetes y el retraso del enlace virtual, no se vean afectados.

De la misma forma en [22] , se propone una evaluación de la capacidad del aprendizaje para predecir el impacto del tráfico entre máquinas virtuales, de la misma forma se plantea una solución basada en datos que puede proporcionar parámetros óptimos de medición de redes virtuales con una mínima interferencia de tráfico. Las evaluaciones se realizan con un gran conjunto de datos de rastreo en un entorno controlado. El uso

de algoritmos de aprendizaje automático permite la predicción de tráfico y el monitoreo de la red, identificando los parámetros con mayor impacto en la red. Por otro lado, en la investigación se analiza la activación del muestreo en diferentes valores de manera regular, ya que puede ser de interés la recopilación de datos para el entrenamiento solo cuando sea relevante y estos generen una mayor cantidad de tráfico.

Por último en [23], se presenta un análisis de las características del tráfico de una red basada en funciones de redes virtuales conocidas por su término en inglés como NetworVirtualization (NFV) mientras se realiza una evaluación comparativa del comportamiento de los algoritmos ML supervisados en la clasificación del tráfico IP con respecto a su eficiencia. La eficiencia de un algoritmo depende de la compensación entre el tiempo de respuesta y la precisión. En el estudio se presenta que las redes bayesianas son uno de los mejores algoritmos de clasificación de tráfico alcanzando valores de hasta el 99,9% con una precisión de 1,1 segundos promedio. Para la evaluación de los diferentes algoritmos se analizaron ciertas características de las funciones de las redes virtuales, adaptando un conjunto de datos tradicionales con estas nuevas características con la finalidad de determinar su eficiencia. El estudio permitió evaluar las nuevas características de las funciones de las redes virtuales como el monitoreo de tráfico oculto para la mejora en la gestión del tráfico.

En la Figura 2.1, se presenta un mapa conceptual con una revisión profunda de las variables de estudio como son el tráfico y los algoritmos de machine learning. Como se puede apreciar de la investigación realizada se puede clasificar que los estudios realizados en cuanto a machinen learning para redes de datos se centran en aproximadamente en 6 temas. Por otro lado, las investigaciones realizadas en base al tráfico de redes se centran en tres puntos, la predicción, la clasificación y el enrutamiento de tráfico, Figura 2.2.

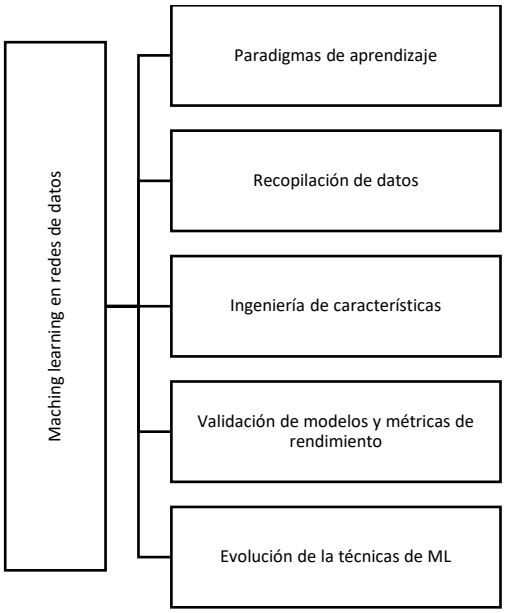


Figura 2.1 Revisión de machine learning en redes de datos.
Fuente [19], [21], [24–30]

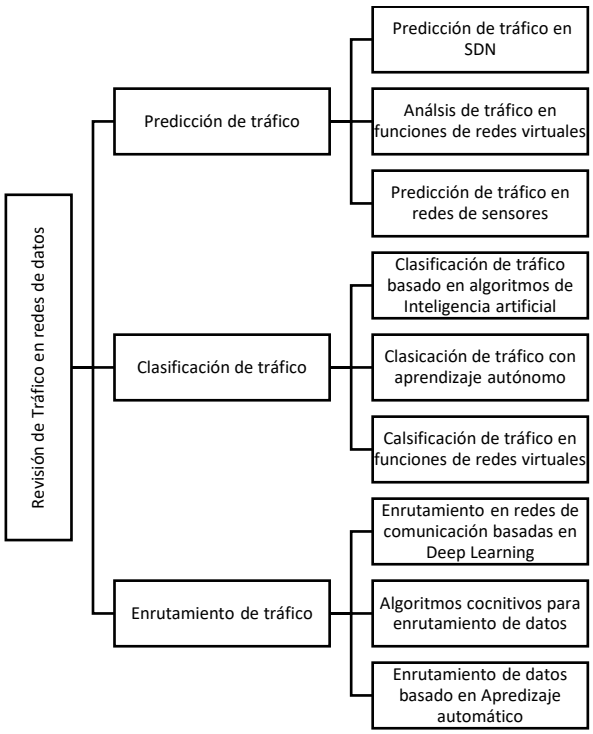


Figura 2.2 Revisión de tráfico en redes de datos
Fuente [18], [19], [21–23], [29], [31–37]

De los trabajos anteriormente mencionados se puede observar que el uso de algoritmos de aprendizaje automático, así como el uso de análisis de datos hoy en día permite la predicción del tráfico tanto en redes físicas como en redes virtuales. La predicción del

tráfico tiene como uno de sus objetivos permitir configurar la ruta óptima para reducir el tráfico en una red virtual, permitiendo tener una red más flexible y eficiente. El enrutamiento optimo del tráfico permite reducir la latencia, que se traslada como una experiencia más eficiente al usuario tanto en rendimiento como accesibilidad, permitiendo una comunicación rápida, segura confiable y de alta eficiencia. En el siguiente trabajo se analiza el uso de técnicas de machine learning así como el análisis de datos para la determinación de una ruta óptima dentro de una red virtual.

2.2. Marco Teórico

El desarrollo de la computación y del despliegue de las redes de datos representaron un valioso papel para varias empresas multinacionales que necesitaban de una red de datos para su comunicación. En sus principios la computación se basó en hardware que por lo general estaba ubicado en un único espacio físico. Las empresas se adaptaron y aprovecharon todas las capacidades de los computadores, pero estas capacidades se fueron desarrollando, entonces la centralización del hardware empezó a convertirse en un limitante. En consecuencia se reemplazó el uso de una única computadora, por el uso de múltiples computadores de menos capacidades que la central pero comunicadas entre sí, logrando incrementar las capacidades de cómputo en la red [38],[31].

Una red de comunicaciones tiene como función esencial, conectar varios usuarios con otros a través de sus dispositivos, conectar usuarios a servidores, o dispositivos entre dispositivos. La comunicación entre dos o más dispositivos se realiza utilizando las direcciones de origen y de destino. Las redes de comunicaciones pueden ser públicas (empresa pública) o privadas (empresa privada) [38]. Sin embargo, las redes empresariales de una organización son diversas y dispares que otras redes. Por lo tanto, los patrones que se ha demostrado que funcionan en una red pueden no ser factibles para otra red del mismo tipo. Esto quiere decir que la red evoluciona continuamente y la dinámica inhibe la aplicación de un conjunto fijo de patrones que ayudan en la operación y gestión de la red. Es casi imposible mantenerse al día manualmente con la administración de la red, debido al crecimiento continuo en la cantidad de aplicaciones que se ejecutan en la red y los tipos de dispositivos conectados a la red.

2.2.1. Redes de datos basadas en el protocolo de internet o IP

Las redes de datos se pueden clasificar por diferentes aspectos: según su topología física (punto a punto, punto a multipunto, bus, anillo, radial), según el medio de transmisión (alámbricas, inalámbricas, móviles, fijas), según la velocidad, protocolo, etc. Internet se clasifica según la conmutación de paquetes, que pueden ser, 3 tipos: conmutación de circuitos (redes telefónicas), conmutación de paquetes orientadas a la conexión (X.25 y X.75) y conmutación de paquetes sin conexión (redes basadas en IP) [39][40].

Una de las grandes ventajas del uso del protocolo IP es el uso del servicio de resolución del sistema de nombre de dominio, (DNS), este se utiliza principalmente en el correo electrónico y en la navegación de red. Inicialmente las redes IP eran conocidas como redes mudas, ya que cualquier modificación se realiza únicamente en los extremos y no existe centralización.[39]

2.2.2. Diseño de red

La puesta en marcha de una red de comunicaciones puede ser sumamente complicada cuando está dirigida para el uso de negocios. La instalación de equipos, conexión, programación y otras actividades simples, no son suficientes para el funcionamiento y la conectividad de una red.

La metodología para el diseño y ejecución de una red se presenta en la Figura 2.3.

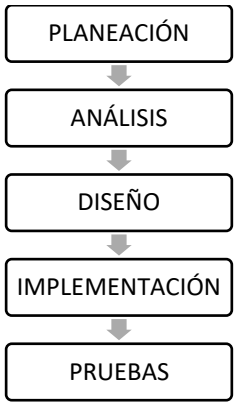


Figura 2.3 Fases para el desarrollo de una red
Fuente [31]

2.2.3. Redes Públicas y Privadas

Básicamente una red pública es la red a la cual puede acceder cualquier usuario. En contraste, las redes privadas tienen un acceso restringido de usuarios. Algunos países

tienen una reglamentación diferenciada para las redes privadas y las públicas. La red de internet se compone tanto de redes públicas como privadas, algunas redes privadas suelen permitir el acceso parcial, por ejemplo, el acceso a ciertos grupos y envió de correos [39].

2.2.4. Familia de protocolos

La denominación TCP/IP se compone de un conjunto de protocolos o normas para dar un formato a los datos que se transmiten, esto permite que las máquinas y programas puedan entenderse, todos estos protocolos permiten el soporte de un conjunto de aplicaciones.[39]

En la Figura 2.4, se detallan las capas que componen al protocolo TCP/IP.

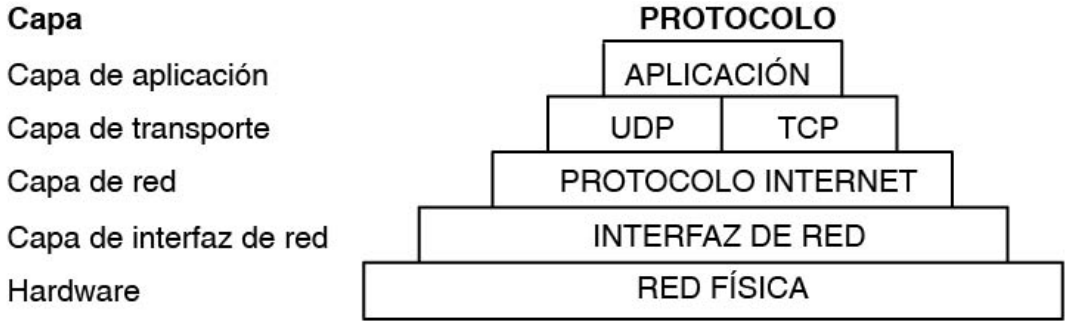


Figura 2.4 Conjunto de protocolos TCP/IP
Fuente [39]

El protocolo TCP/IP permite la comunicación entre emisor y receptor, para iniciar la transmisión la capa de aplicación envía los datos a los protocolos de la capa de transporte donde se dividen en conjuntos de datos más pequeños y se añaden identificadores. Para lograr una comunicación universal, cada computadora o dispositivo tiene su propio y único identificador, denominado dirección IP.

2.2.4.1. Protocolo IP

El internet protocol o protocolo IP es el formato de cómo se transmiten los datos de una red. El protocolo IP tiene reglas determinadas de cómo procesar los datos y manejar errores. Toda información en la red viaja a través de datagramas IP los que tiene dos partes la cabecera y el texto [41]. La cabecera cuenta con una parte fija de 20 bytes y una parte opcional entre 0 y 40 bytes que siempre deben ser múltiplos de 4. La estructura de la cabecera del datagrama IP se muestra en la Figura 2.5.

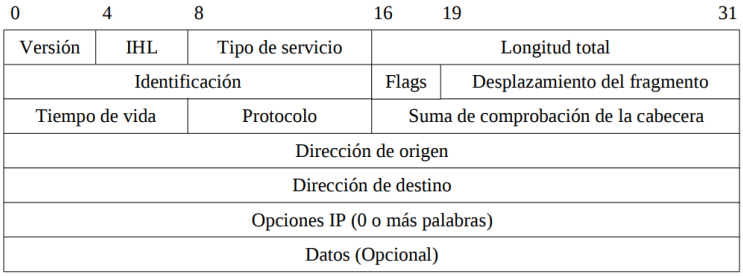


Figura 2.5 Campos del datagrama IP
Fuente [39]

La estructura del datagrama se compone de los siguientes campos [40–43]:

Versión: El encargado de permitir la coexistencia de diferentes versiones del protocolo es el campo versión. La Figura 2.5 presentada hace referencia al datagrama de estructura de la versión 4, sin embargo, la misma es base para el desarrollo del datagrama de la versión 6.

IHL: Representa la longitud de la cabecera agrupadas en palabras de 32 bits, sin embargo, cabe mencionar que este campo puede variar debido a la presencia de campos opcionales.

TOS: El campo de servicio contiene la precedencia con una longitud de 3 bits, TOS (Tipo de Servicio) con una longitud de 4 bits y reservado con una longitud de 1 bits. El subcampo de precedencia permite especificar la prioridad del dato con un valor normal de 0 y un máximo de 7. La propiedad de los paquetes se altera en base a la cola que se genera en los routers pero no modifica la ruta de estos. El subcampo TOS puede tener las combinaciones presentadas en la Tabla 2.1.

Valor TOS	Descripción
0000	Valor por defecto
0001	Mínimo costo
0010	Máxima fiabilidad
0100	Máximo rendimiento
1000	Mínimo retardo
1111	Máxima seguridad

Tabla 2.1 Combinación de subcampo TOS
Fuente [39]

Longitud total: indica la longitud (en bytes u octetos) de todo el paquete, incluidos el encabezado y los datos. Dado el tamaño de este campo, el tamaño máximo de un

paquete IP es 64 KB o 65.535 bytes. En la práctica, los tamaños de los paquetes están limitados a la unidad de transmisión máxima (MTU).

Identificación: se utiliza cuando un paquete se fragmenta en partes más pequeñas mientras atraviesa Internet, este identificador es asignado por el host transmisor para que los diferentes fragmentos que llegan al destino se puedan asociar entre sí para reensamblar.

Flags: también se utilizan para fragmentación y reensamblaje. El primer bit se denomina bit de más fragmentos (MF) y se utiliza para indicar el último fragmento de un paquete para que el receptor sepa que el paquete se puede reensamblar. El segundo bit es el bit Don't Fragment (DF), que suprime la fragmentación. El tercer bit no se utiliza (y siempre es configurado en 0)

Desplazamiento de fragmento: indica la posición de este fragmento en el paquete original. En el primer paquete de un flujo de fragmentos, el desplazamiento será 0; en los siguientes fragmentos, este campo indicará el desplazamiento en incrementos de 8 bytes.

Tiempo de vida (TTL): un valor de 0 a 255, que indica el número de saltos que este paquete puede realizar antes de descartarse dentro de la red. Cada enrutador que vea este paquete disminuirá el valor TTL en uno; si llega a 0, el paquete se descarta.

Protocolo: indica el contenido del protocolo de capa superior de los datos transportados en el paquete; las opciones incluyen ICMP (1), TCP (6), UDP (17) u OSPF (89). Puede encontrar una lista completa de números de protocolo IP en la lista de números de protocolo de la IANA. Se puede encontrar una lista específica de implementación de protocolos compatibles en el archivo de protocolo, que generalmente se encuentra en el directorio / etc (Linux / Unix / Mac OS X) oc: \ Windows \ system32 \ drivers \ etc (Windows).

Suma de comprobación del encabezado: contiene información para garantizar que el encabezado IP recibido no contenga errores. Recuerde que IP proporciona un servicio poco confiable y, por lo tanto, este campo solo verifica el encabezado de IP en lugar de todo el paquete.

Dirección de origen: dirección IP del host que envía el paquete.

Dirección de destino: dirección IP del host destinado a recibir el paquete.

2.2.4.2. Protocolo TCP/IP

El protocolo TCP/IP (Protocolo de control de transmisiones/Protocolo Internet) es uno de los protocolos más usados. Las versiones originales de TCP e IP que son de uso común en la actualidad se escribieron en septiembre de 1981, aunque a ambas se les han aplicado varias modificaciones (además, la especificación IP versión 6, o IPv6, se publicó en diciembre de 1995). En 1983, el Departamento de Defensa de los Estados Unidos ordenó que todos sus sistemas informáticos usarán el conjunto de protocolos TCP / IP para comunicaciones de larga distancia, mejorando aún más el alcance y la importancia de ARPANET. Básicamente TCP/IP es un conjunto de protocolos o normas que cumplen estándares de las comunicaciones, direccionamiento e interconexión de redes [40] .

Al utilizar TCP/IP se consigue una independencia del hardware, ya que este permite la comunicación entre sistemas conectados a una red, y esta red puede estar conectada a otras, con el uso de TCP/IP se puede:

- Realizar transferencia de archivos entre sistemas.
- Ejecutar acciones de manera remota como: iniciar sesiones, imprimir, enviar correos, tener conversaciones.
- Gestionar una red

2.2.5. Direccionamiento TCP/IP

Cada dirección IP se compone de cuatro octetos que contienen 2 identificadores: redid en la cual se identifica la red y hostid con el cual se identifica el dispositivo de la red [41], [42]. La dirección IP se clasifica en tres clases de direcciones (A,B,C), como se muestra en la Figura 2.6

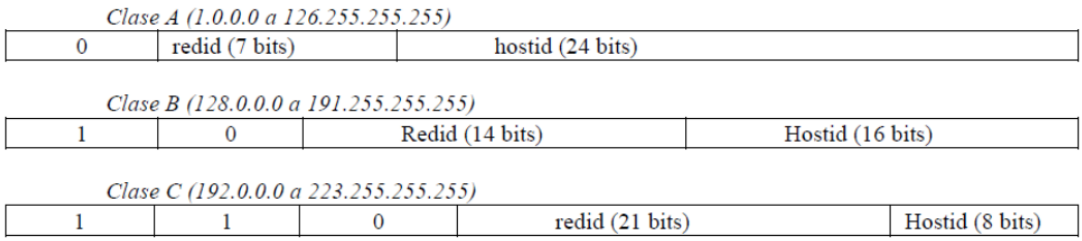


Figura 2.6 Clases de direcciones IP
Fuente [39]

La clase de una dirección IP se determina con los tres bits de orden más alto, como se observa en la figura anterior.

En la clase A, al identificador de red (redid) le corresponden 7 bits y para el identificador de host (hostid) 24 bits.

En la clase B, al identificador de red (redid) le corresponden 14 bits y para el identificador de host (hostid) 24 bits.

En la clase C, al identificador de red (redid) le corresponden 21 bits y para el identificador de host (hostid) 8 bits.

Adicionalmente hay dos clases especiales, clase D y E.

La clase D corresponde a redes multicast y la clase E es experimental, también se reservan ciertas direcciones de diferentes clases, que se muestran en la Tabla 2.2:

Address Block	Present Use	Reference
0.0.0.0/8	"This" Network	[RFC1700]
10.0.0.0/8	Private-Use Networks	[RFC1918]
14.0.0.0/8	Public-Data Networks	[RFC1700]
24.0.0.0/8	Cable Television Networks	--
39.0.0.0/8	Reserved but subject to allocation	[RFC1797]
127.0.0.0/8	Loopback	[RFC1700]
128.0.0.0/16	Reserved but subject to allocation	--
169.254.0.0/16	Link Local	--
172.16.0.0/12	Private-Use Networks	[RFC1918]
191.255.0.0/16	Reserved but subject to allocation	--
192.0.0.0/24	Reserved but subject to allocation	--
192.0.2.0/24	Test-Net	
192.88.99.0/24	6to4 Relay Anycast	[RFC3068]
192.168.0.0/16	Private-Use Networks	[RFC1918]
198.18.0.0/15	Network Interconnect Device Benchmark	[RFC2544]
223.255.255.0/24	Reserved but subject to allocation	--
224.0.0.0/4	Multicast	[RFC3171]
240.0.0.0/4	Reserved for Future Use	[RFC1700]

Tabla 2.2 Ejemplos de direcciones ip reservadas.
Fuente [39]

Los routers encaminan los paquetes en base al redid del host y no al host al que se envía los datos. El direccionamiento IP tiene la ventaja que permite el uso de broadcast, cuando todos los bits del hostid son igual a 1.

2.2.6. Enrutamiento de redes

El enrutamiento es el proceso encargado de determinar la ruta de los datos para llegar a su destino, buscando el mejor camino posible. En redes los dispositivos encargados de realizar esta tarea son los enrutadores o routers, estos utilizan tablas denominadas tablas de rutas, las mismas que se pueden llenar por dos formas: estática o dinámica [44],[40]. Este proceso es realizado por la capa de red, encargada de enrutar los paquetes desde la máquina origen a la máquina destino. Independiente del tipo de enrutamiento la premisa que se debe cumplir es la de la corrección, simplificación, robustez, estabilidad y optimización de la red en la que se transportan los paquetes [45].

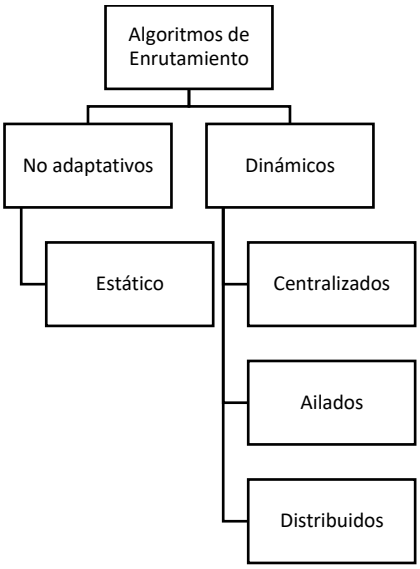


Figura 2.7 División de los algoritmos de enrutamiento
Fuente [46]

2.2.6.1. Enrutamiento Estático

En este tipo de enrutamiento la configuración la realiza el administrador de la red, el administrador configura la mejor ruta para los paquetes basándose en la topología y comportamiento de esta, es decir que el administrador conoce muy bien la red. Las rutas están definidas por adelantado y se cargan en los routers al iniciar la red [47], [48]

Las tablas de enrutamiento son llenadas de forma manual por el administrador de forma determinista, entonces si la red sufre algún cambio, estos cambios deben ser ingresados por el administrador ya que estas no se actualizan de forma automática. Para elegir la ruta más corta se puede hacer de forma manual o automática, como el algoritmo de la ruta más corta o el enrutamiento basado en flujo. Algoritmo de la ruta más corta: este se basa en una métrica establecida por el administrador, como el delay, número de saltos o el tráfico. En la Figura 2.8 se observa la ruta para llegar de A a G donde la métrica es el número de saltos.

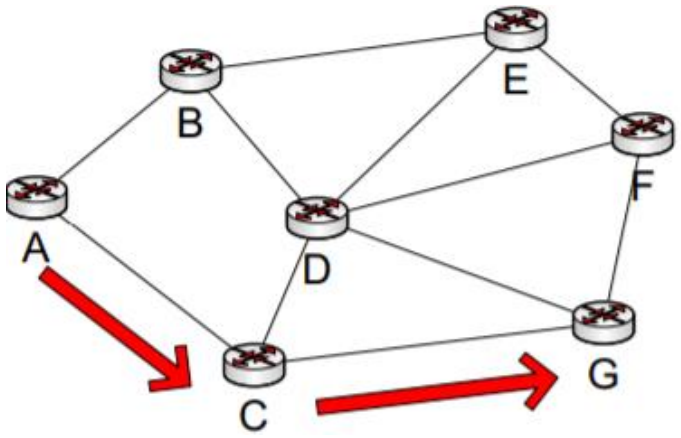


Figura 2.8 Ruta de tráfico de A a G
Fuente: [45]

Enrutamiento basado en flujo: la ruta elegida será la que tenga menor flujo de datos como se observa en la Figura 2.9

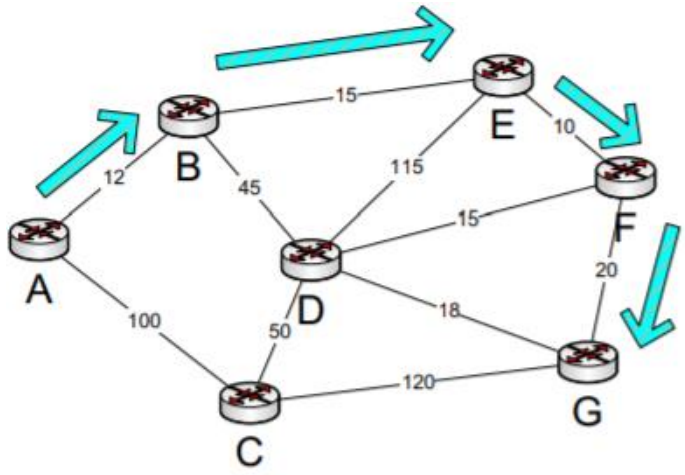


Figura 2.9 Ruta de tráfico de A a G
Fuente: [45]

El algoritmo de inundación no es muy usado, este envía los paquetes por todas las rutas de salida posible, lo que crea mucho tráfico y mensajes duplicados.

2.2.6.2. Enrutamiento Dinámico

El enrutamiento dinámico se forma por un grupo de algoritmos que buscan la mejor ruta mediante métricas específicas, aquí las tablas de enrutamiento se llenan de forma automática con los resultados de los diferentes algoritmos, estas varían según el comportamiento de la red, las decisiones se transmiten a los enrutadores mediante los protocolos de enrutamiento, la ruta de la red ya no es intervenida por el administrador

ya que el algoritmo se encarga de encontrar la ruta más corta. Entonces si la red crece o cambia el administrador de la red solo debe configurar los protocolos a utilizar.[46], [48]

Las ventajas de enrutamiento dinámico son: escalabilidad, robustez, administración más simple, mayor convergencia siempre elige la mejor ruta. Los algoritmos comúnmente usados son: algoritmo vector distancia y el algoritmo estado de enlace.

2.2.6.2.1. Protocolos de enrutamiento dinámico

Un protocolo de enrutamiento permite intercambiar la información en un sistema autónomo (S.A.), sobre el comportamiento de la red, con toda la información que se obtiene se toman decisiones para elegir la ruta de los paquetes desde su origen hasta llegar a su destino. Un S.A. o sistema autónomo, consiste en un segmento de una red que tiene un conjunto de rutas las cuales están bajo una misma autoridad administrativa. Para poder elegir el protocolo más apto para una red se debe tomar en cuenta varias características de esta:

Existen dos tipos de protocolos: los EGP o protocolos de enrutamiento exterior y los IGP o protocolos de enrutamiento interior, en la Figura 2.10 se puede observar ejemplos de EGP y de IGP [46], [48].

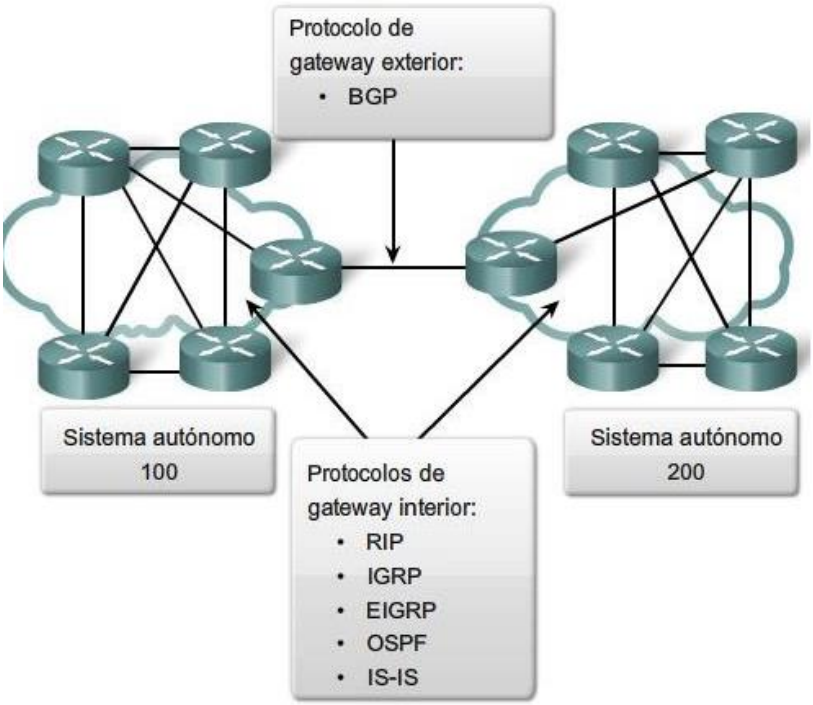


Figura 2.10 Protocolos de enrutamiento EGP e IGP
Fuente: [45]

En la Tabla 2.3 se resume la clasificación de protocolos de enrutamiento.

CLASIFICACIÓN DE LOS PROTOCOLOS DE ENRUTAMIENTO					
EGP (Protocolos de enrutamiento exterior)	**IGP (Protocolos de enrutamiento interior)**				
BGP (Protocolo de enrutamiento de borde)	Vector-distancia			Estado-enlace	

Tabla 2.3 Clasificación de los protocolos de enrutamiento
Fuente [39]

Protocolos EGP o protocolos de enrutamiento exterior: este protocolo permite establecer relaciones entre sistemas autónomos, estos pueden coexistir con varios protocolos IGP. Cada S.A. tiene una administración independiente lo que facilita el mantenimiento de la red. Para poder establecer la relación entre los S.A cada uno de estos tiene un router de borde en cada extremo como se observa en la Figura 2.11.

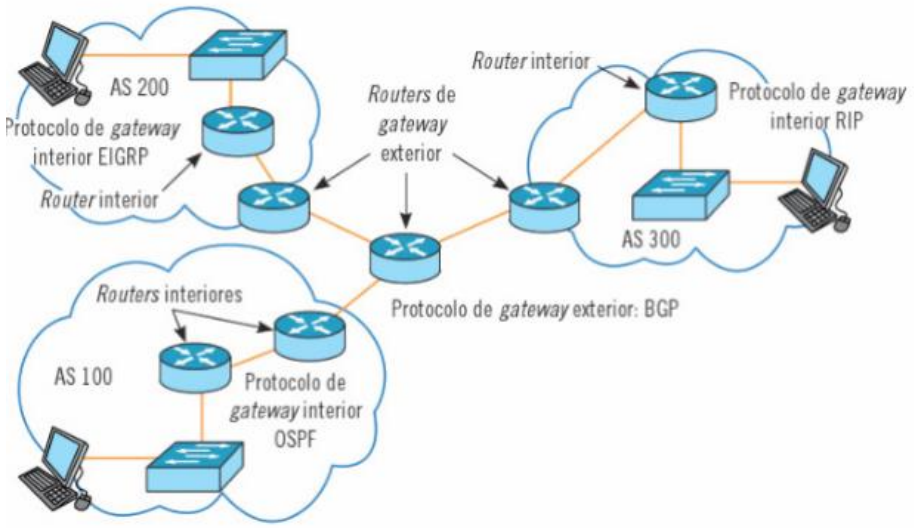


Figura 2.11 Protocolos de enrutamiento EGP
Fuente: [45]

Protocolo de enrutamiento de borde BGP o Pasarela de frontera: Este protocolo se usa para comunicar S.A, éstos definen políticas de comunicación entre las S.A como: políticas de seguridad o económicas. A este protocolo también se lo conoce como protocolo de vector de ruta, debido a que guarda las rutas más no el costo para llegar a un destino. Este protocolo hace posible la relación entre diferentes sistemas autónomos. Para su funcionamiento este protocolo utiliza 4 tipos de mensajes, sus formatos respectivos se observan en la Figura 2.12

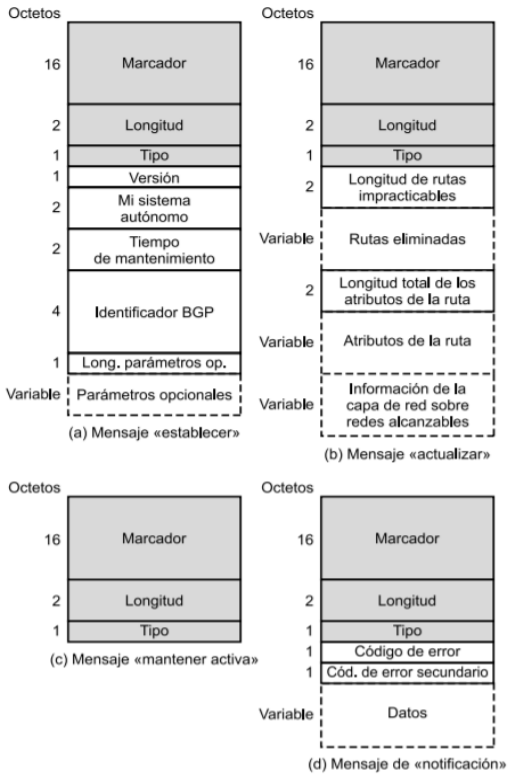


Figura 2.12 Formato de los mensajes BGP
Fuente: [48]

Protocolos IGP o protocolos de enrutamiento interior: Este tipo de protocolos trabajan para relacionar routers dentro de un mismo S.A. Estos se clasifican en dos grandes grupos: protocolos vector-distancia y estado-enlace.

Protocolos vector-distancia: Aquí los routers se comunican con los routers vecinos de forma constante y los actualizan de cambios de la topología de la red. Este está basado en dos parámetros; la distancia desde el origen hasta el destino y el vector que es la dirección del router del siguiente salto hasta alcanzar el destino. Existen varios tipos de protocolos vector-distancia, los más conocidos son: RIP, IGRP y EIGRP [45], [48]

Protocolo de información de enrutamiento o RIP: Se utiliza generalmente en redes pequeñas y homogéneas, realiza actualización o updates del estado de la red cada 30 segundos enviando datos a las tablas de enrutamiento de los router vecinos, todos los router repetirán el proceso hasta actualizar a toda la red.

La métrica utilizada en este protocolo es el conteo de saltos, sin embargo, esta se limita 15 saltos, una métrica mayor se convierte en inaccesible, por esto se considera que RIP es adecuado para redes pequeñas.

El protocolo RIP ha realizado varias mejoras y se ha convertido en RIPv2, las características más relevantes de este son:

- Este se considera un protocolo sin clase por lo que soporta VLSM y CIDR.
- La dirección de difusión es la 224.0.0.9.
- Se logra una convergencia de red en menor tiempo comparada con RIP.

Funcionamiento: Cada router envía un mensaje de solicitud de envió de su tabla a todos los routers conectados en sus interfaces y su mensaje de respuesta es la tabla de enrutamiento. La tabla de enrutamiento se actualiza cada periodo de tiempo si se encuentra una ruta al mismo destino con menor número de saltos esta se coloca en la tabla y reemplaza a la anterior [49–51]. En la Figura 2.13 se observa el formato del protocolo RIPv2.

Comando	**Versión**	**Debe ser cero (no usado)**
Identificador de familia de dirección	Etiqueta de la ruta	
Dirección IP		
Máscara de subred		
Próximo salto		
Métrica		

Figura 2.13 Formato del protocolo RIPv2
Fuente: [40]

Protocolo de enrutamiento de Gateway interior o IGRP: Este protocolo fue desarrollado por CISCO, se puede implementar en redes grandes que se encuentren dentro de un mismo S.A. En contraste con el comando RIP tiempo límite de saltos es de 255 por lo que se pueden usar en redes amplias, envía mensajes de actualización a sus vecinos cada 90 segundos. Este protocolo se basa en varias métricas: ancho de banda, retardo, confiabilidad y carga de enlace [50], [52].

Realiza una convergencia en menor tiempo que en RIP ya que aplica las actualizaciones disparo, al encontrarse con una ruta inalcanzable hace automáticamente una actualización.

Protocolo de enrutamiento de Gateway interior mejorado o EIGRP: Al igual que IGRP este fue desarrollado por CISCO pero en el 2013 se transformó en un protocolo abierto, a este protocolo se lo considera la versión mejorada del protocolo IGRP. Este protocolo reduce el tráfico redundante en una red ya que la información de las tablas de enrutamiento se actualiza únicamente con la información que es nueva para el router vecino. A estas actualizaciones se las denominan desencadenantes y hacen uso de un algoritmo de actualización por difusión o denominado DUAL, esta hace que la red tenga una convergencia en menor tiempo. A este protocolo se lo considera un protocolo híbrido ya que aplica funciones de los protocolos vector-distancia y de los protocolos estado-enlace[40], [42], [43]. El formato de un paquete EIGRP se observa en la figura

<table>
<tr><td>Versión</td><td>OPcode</td><td>Checksum</td></tr>
<tr><td colspan="3">Banderas</td></tr>
<tr><td colspan="3">Número de secuencia</td></tr>
<tr><td colspan="3">Número de reconocimiento</td></tr>
<tr><td colspan="3">Número de sistema autónomo</td></tr>
<tr><td colspan="2">Tipo</td><td>Tamaño</td></tr>
<tr><td colspan="3">Valor</td></tr>
</table>

Figura 2.14 Formato de un paquete EIGRP
Fuente: [42]

Protocolos de estado-enlace: Aquí los routers se comunican con los routers vecinos de forma constante y los actualizan de cambios de la topología de la red.

Protocolo del camino más corto o OSPF: Este protocolo es de los más robustos y es usado por grandes empresas para levantar redes muy amplias. En este protocolo se reúne toda la información de la red mediante la creación de un mapa de topología. De existir cambios en el estado de enlace la información se intercambia entre todos los routers proporcionando esta información a todos los routers de la red. La métrica en la que se fundamenta este protocolo es el costo, que se calcula a partir del valor del ancho de banda de cada enlace, como se muestra en la fórmula (1) [40].

$$Costo: 10^8/AB(de\ la\ interfaz)bps \quad (1)$$

Para calcular el costo total de una ruta desde su origen a su destino, se suma el costo de cada enlace por donde pasan los datos. El costo en cada router puede ser editado por el administrador, esto permite tener caminos con preferencia, según la necesidad [48]. OSPF envía datagramas que pueden contener 5 clases de mensajes, en la figura se resumen las características más relevantes de cada uno.

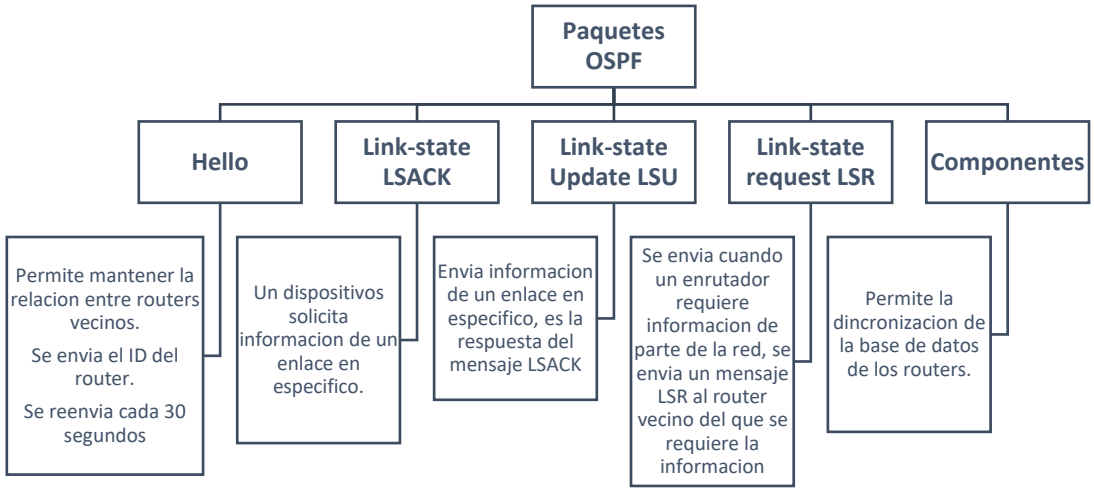


Figura 2.15 Tipos de mensaje OSPF
Fuente: [43], [48]

Protocolo de sistema intermedio a sistema intermedio o IS-IS: Este protocolo tiene características similares a las de OSPF, cada router transmite los datos del estado de su enlace y mapea el estado de la red, llenando su tabla de enrutamiento. IS-IS transporta información IP, pero no lo utiliza como protocolo de transporte, utiliza dos niveles de enrutamiento, el Nivel 1 es cuando se enruta desde la misma zona y Nivel 2 se enruta entre varias Zonas [42].

2.2.7. Virtualización

La virtualización es la división o fragmentación de los recursos que tiene una computadora, denominada Hipervisor o Virtual Machine Monitor. La capa de virtualización es una división entre el hardware de la máquina física y el sistema operativo de la máquina virtual, que permiten la creación de dispositivos y recursos en su versión virtual. Los recursos que se pueden virtualizar son: servidores, dispositivos de almacenamiento, una red o un sistema operativo (SO) [53] [54].

En la VMM se puede manejar, administrar y gestionar los recursos de una computadora, como su CPU, memoria, red y almacenamiento por lo que estos se pueden distribuir a todas las máquinas virtuales que se conectan al servidor principal.

Según VMware la "virtualización es una tecnología de software que permite ejecutar varias máquinas virtuales en una única máquina física, compartiendo los recursos de ese ordenador único entre varios entornos" [54].

2.2.7.1. Tipos o modalidades de Virtualización

La virtualización se clasifica según el recurso creado, y son:

- **Virtualización de servidores**

Este tipo de virtualización se realiza mediante el uso de un hardware con software anfitrión, se simula un entorno o máquina virtual para el software. Las aplicaciones más populares que usan este tipo de virtualización son: VMware Workstation, IBM VM, entre otros.[54], [55]

- **Paravirtualización**

La paravirtualización o virtualización asistida por sistema operativo, consiste en el trabajo compartido entre el sistema operativo huésped y el hypervisor. El kernel del sistema operativo es modificado reemplazando las intrusiones no virtuales por hypercall. El hipervisor se instala en el hardware y uno o varios SO se alojan en este.
La diferencia entre la virtualización completa y la paravirtualización radica en que el S.O huésped no conoce que está virtualizado. Un ejemplo de esta tecnología es VMware ESX [54], [55].

- **Virtualización asistida por hardware**

En esta tecnología no es necesario la modificación del sistema operativo, ya que el hipervisor tiene todos los privilegios y no tiene la necesidad de modificar al sistema operativo huésped[54].

- **Virtualización de dispositivos de entrada/salida**

Más conocida como virtualización de dispositivos I/O, comprende el enrutamiento de las solicitudes de entrada y salida de los dispositivos virtuales y el hardware compartido. El hypervisor virtualizado presenta a cada máquina virtual todos los dispositivos virtuales disponibles[54], [55].

- **Virtualización de memoria**

La virtualización de memoria supera las limitantes que tiene una memoria física. Los recursos de memoria RAM de cada máquina son agrupados en un único recurso denominado recurso pool, que está disponible para cualquier máquina que lo requiera, es decir se tiene un gran recurso de memoria, que puede ser compartida.

Esta tecnología se compone de nodos que usan el recurso y otros nodos que contribuyen a la virtualización conectándose a la red gracias a un software[53], [55].

- **Virtualización de storage o almacenamiento**

Esta tecnología permite la división del almacenamiento físico con el almacenamiento lógico. Básicamente consiste en la creación de una capa de software entre el sistema de dispositivos de storage y los S.O, permitiendo que varios dispositivos de almacenamiento se conviertan en uno incrementando su capacidad, aunque se encuentren físicamente separados [54].

- **Virtualización de aplicación**

En la virtualización de aplicación se divide la visualización de la ejecución de los programas. El sistema de software se reduce a un conjunto de paquetes pequeños, que permite garantizar, el soporte, mantenimiento y administración de la carga de la aplicación virtual. Un ejemplo de uso claro de esta virtualización es que se ejecute un editor de texto en un servidor central, pero la respuesta gráfica sea enviada a un dispositivo remoto cliente.

Para su implementación se debe considerar los siguientes pasos:

Empaquetamiento, instalación, prueba, administración y eliminación[53], [55].

Un software agente permite la creación de una capa de virtualización como se muestra en la Figura 2.16, esta ingresa a la rutina del sistema operativo y capta las operaciones sobre los ficheros realizados por la aplicación virtualizada[55].

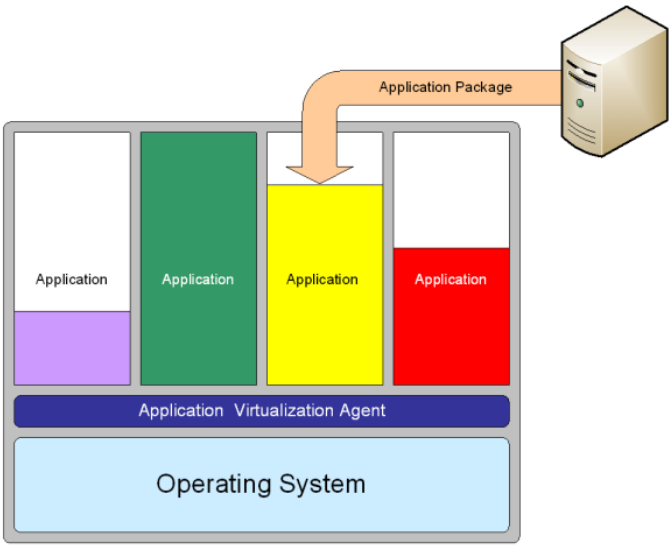


Figura 2.16 Virtualización de aplicación
Fuente: [55]

Esta virtualización se divide en 2 categorías:

1. Virtualización de aplicación hospedada, estas necesitan un software agente para ser instaladas y correr la aplicación en el dispositivo final, es decir que la aplicación se ejecuta en algún servidor.
2. Virtualización de aplicación local, las aplicaciones se ejecutan localmente, es decir en el dispositivo del usuario final[53], [55].

- **Virtualización de escritorio**

La virtualización de escritorio o desktop, es una tecnología que permite separar el ambiente del escritorio al que accede el usuario, de su máquina física y de su sistema operativo. Para esto, el escritorio virtualizado debe estar instalado en un servidor central y el usuario debe estar conectado en red con el servidor central. Todas las acciones que realice el usuario en su escritorio virtual se ejecutan en el servidor central, estas acciones son transparentes para la máquina física y sistema operativo del usuario. Esto permite al usuario acceder a máquinas virtuales en el servidor desde su propia máquina física, como si fuera su propia máquina local.

Esta virtualización permite a los usuarios acceder a aplicaciones instaladas en el servidor central sin tener que instalarlas en su ordenador[53], [55].

- **Virtualización de redes**

En la virtualización de redes se unen todos los recursos de hardware, software y funcionalidades en una red única basada en software, a esto se denomina la red virtual. Con esta tecnología se logra que diferentes cargas de red viajen por el mismo medio de forma eficiente y segura [54].

Este tipo de virtualización se clasifica en dos grupos:

1. Internas, existe un único sistema basado en software el cual simula todas las funciones de red. Este tipo de sistema se logra gracias a la virtualización de escritorios.
2. Externas, este tipo de redes consiste en la unión de varias redes o parte de ellas en una sola red virtual única o la separación de una red en varias redes virtuales. Las redes externas se componen principalmente de las redes locales virtuales y los switches de la red. En síntesis, la virtualización de redes implica unir varias redes físicas en una sola virtual, lo que reduce y mejora el proceso de gestión de red.

Las redes privadas virtuales o VPN, consiguen obtener una sola red de varias redes que se encuentran en distintos lugares geográficamente separados.

Las VLANs o redes virtuales de área local pueden configurarse según la necesidad, por lo general unen secciones específicas de red para controlar el flujo de tráfico, lo que permite tener una red robusta y segura.

2.2.7.2. Ventajas de la Virtualización

La idea de la virtualización nació en los años 50, pero en los años 60 fue aceptada la idea por IBM. VMware lanzó al mercado en 1999, la primera plataforma comercial de virtualización para la arquitectura x86. Con el pasar de los años esta plataforma tomó el nombre de VMware Workstation, desde entonces la virtualización se convirtió en una opción interesante, ya que brinda varias ventajas que se detallan a continuación:

- Incorporar nuevos recursos a los servidores virtuales, es rápido.
- Reducir costo, espacio, consumo de energía (estimación 10 a 1)
- Administrar la red es más fácil, ya que esta se encuentra centralizada.
- Todos los recursos y capacidades de la red como almacenamiento, procesamiento, memoria y red pueden ser gestionados según la necesidad.

- De ser necesario se puede realizar copias y clonaciones de manera rápida y fácil.
- Se pueden crear entornos de prueba, poniendo en marcha aplicaciones sin necesidad de afectar a la red.
- Cada máquina virtual se encuentra aislada de la otra por lo que algún fallo no afecta al resto.
- Reducción de tiempos de parada por mantenimiento ya que cualquier cambio se puede realizar en caliente, mejorando la fiabilidad de la red.
- En una red tradicional la utilización de hardware es de un 15%, con la virtualización se obtiene un 80%.

2.2.8. Redes virtuales

Las redes virtuales permiten que computadoras, servidores, entre otros dispositivos puedan comunicarse, aunque estén ubicados en diferentes sitios geográficamente. Las redes virtuales no se ven limitadas por cableado o hardware y hacen uso de gestión de software, lo que permite conectar a los dispositivos por internet.

La virtualización de redes hace posible crear, aprovisionar y administrar redes de manera programática, todo en software, utilizando la red física subyacente como una simple placa posterior de reenvío de paquetes. Los servicios de red y seguridad en software se distribuyen a hipervisores y se "adjuntan" a máquinas virtuales individuales (VM) de acuerdo con las políticas de red y seguridad definidas para cada aplicación conectada. Cuando una máquina virtual se mueve a otro host, sus servicios de red y seguridad se mueven con ella. Y cuando se crean nuevas máquinas virtuales para escalar una aplicación, las políticas necesarias también se aplican dinámicamente a esas máquinas virtuales.

De manera similar a cómo una máquina virtual es un contenedor de software que presenta servicios de cómputo lógico a una aplicación, una red virtual es un contenedor de software que presenta servicios de red lógicos: conmutación lógica, enrutamiento lógico, firewall lógico, equilibrio de carga lógico, VPN lógicas y más - a cargas de trabajo conectadas. Estos servicios de red y seguridad se entregan en software y solo requieren el reenvío de paquetes IP desde la red física subyacente. Las cargas de trabajo en sí están conectadas a través de una representación de software de un "cable" de red física. Esto permite crear toda la red en software [17], Figura 2.17.

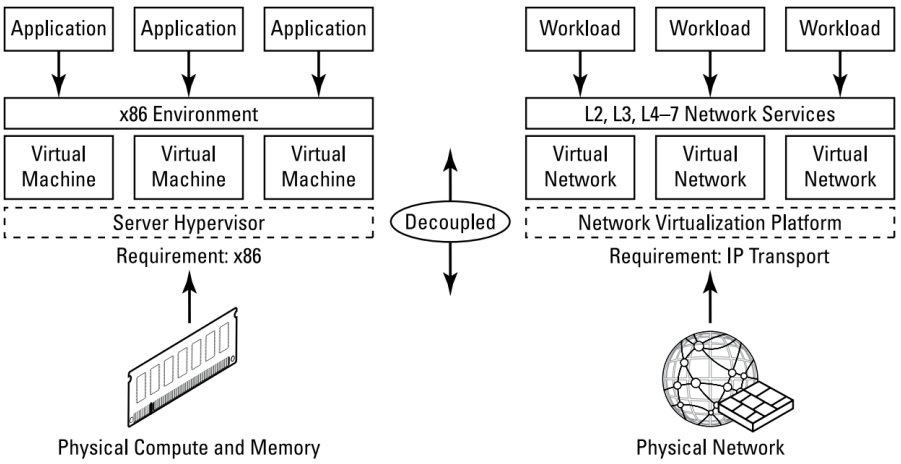


Figura 2.17 Virtualización de redes y computación
Fuente: [55]

La virtualización de la red permite coordinar los conmutadores virtuales en los hipervisores del servidor, con la finalidad de ofrecer una plataforma de hipervisor de red encargada de la creación de máquinas de forma eficiente.

Las VXLAN son otro ejemplo de redes virtuales, la Red de área local extensible, va más allá de las VLAN ya que no son divide la red en diferentes redes lógicas, sino que puede virtualizar una red entera.

Una de las formas de visualizar de forma adecuada el desempeño de una red virtual es mediante el uso de plataformas de administración en la nube. Los servicios de seguridad y la carga de trabajo se pueden configurar en este tipo de plataformas, desde la creación de redes básicas con dos nodos, hasta la validación de redes con topologías complejas que utilizan multisegmentos para la distribución de la aplicación en diferentes niveles [17].

La capa de virtualización o capa de software se compone de un supervisor más conocido como hipervisor, este asigna los recursos que contiene el hardware. El hipervisor tiene la capacidad de albergar varios sistemas operativos a los que denomina sistemas operativos huéspedes. Los SO huéspedes pueden administrar sus aplicaciones de forma tradicional es decir no virtual, con la única diferencia que se encuentran divididos del hardware por el monitor virtual. EL hipervisor determina un VMM a cada máquina virtual.[54],[55]

Con la virtualización se consigue que una máquina virtual funcione con su propio sistema operativo. La introducción de varias capas de virtualización como máquinas

virtuales (VM), redes virtuales (VN) y almacenamiento virtual proporciona dicha infraestructura bajo demanda. Esta evolución llevó a la implementación de técnicas avanzadas como entornos basados en contenedores, que crean una infraestructura dinámica como los diferentes servicios en la nube como SaS, plataforma como servicio, PaS o infraestructura como servicio IaS [56], [57].

2.2.8.1. Hipervisor

En redes al hipervisor también se lo conoce como Monitor de máquina virtual (VMM), estos se desarrollaron inicialmente en los años 70. El VMM es una capa de software que permite la separación entre el hardware y el S.O, es el elemento más básico en la virtualización. El hipervisor es una plataforma de virtualización que hace posible correr a la vez varios sistemas operativos en una misma computadora. El monitor de máquina virtual es el encargado de controlar el acceso de sistemas operativos huéspedes a los diferentes dispositivos de hardware disponibles. Para sistemas operativos de arquitectura x86, se clasifica a los hipervisores en dos tipos de arquitecturas.

Con arquitectura hospedada: el hipervisor se instala y corre como una aplicación principal como se muestra en la Figura 2.18

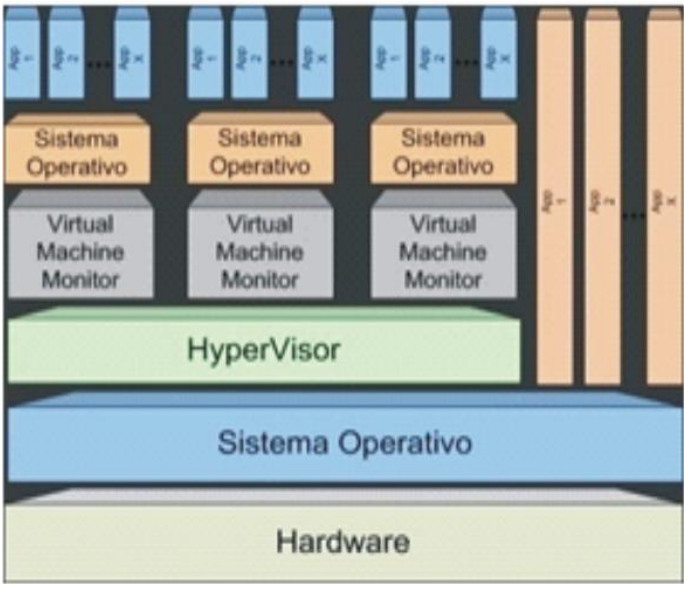


Figura 2.18 Hipervisor con arquitectura hospedada
Fuente: [55]

Con arquitectura nativa: se encuentra instalado directamente en el hardware como se muestra en la Figura 2.19 teniendo acceso directo al hardware y sus recursos, se

encuentra sobre el sistema operativo, lo que lo hace más rápido que la arquitectura anterior.

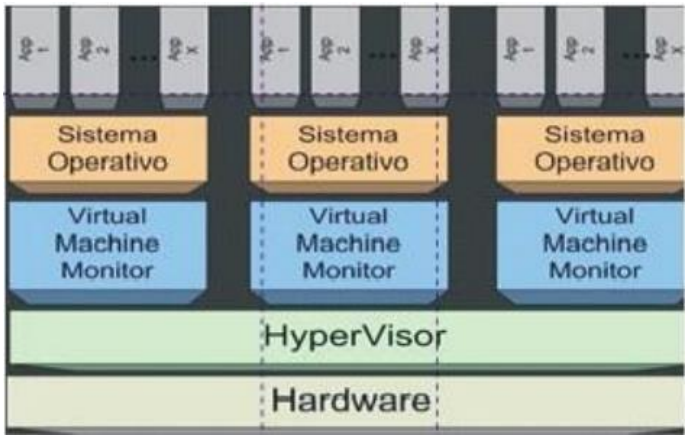


Figura 2.19 Hipervisor con arquitectura nativa
Fuente: [55]

2.2.8.2. Capas de virtualización

Emulación y Simulación

Un software de simulación es capaz de emular y simular varias tecnologías de red, como enrutamiento y conmutación, armar una topología, simular la ejecución de protocolos, funciones de servidores y soportar comunicaciones LAN y WAN.

Los softwares de simulación de redes más utilizados o tradicionales son; Cisco Packet tracer, GNS3 y NetSim, que se estudian con más detalle a continuación:

Cisco Packet Tracer: Esta herramienta fue desarrollada por CISCO, es gratuita para uso académico, pero de paga para otros fines. Esta entrega un entorno de simulación de red para configurar los dispositivos de forma correcta, eliminación de fallos de red, entre otras acciones. Cuenta con una interfaz gráfica que permite crear la topología de red deseada, y genera información detallada del comportamiento de la red en tiempo real. Packet tracer permite simular y visualizar de forma interactiva la red de estudio, en la Figura 2.21 se muestra un ejemplo de una red configurada en cisco packet tracer [53].

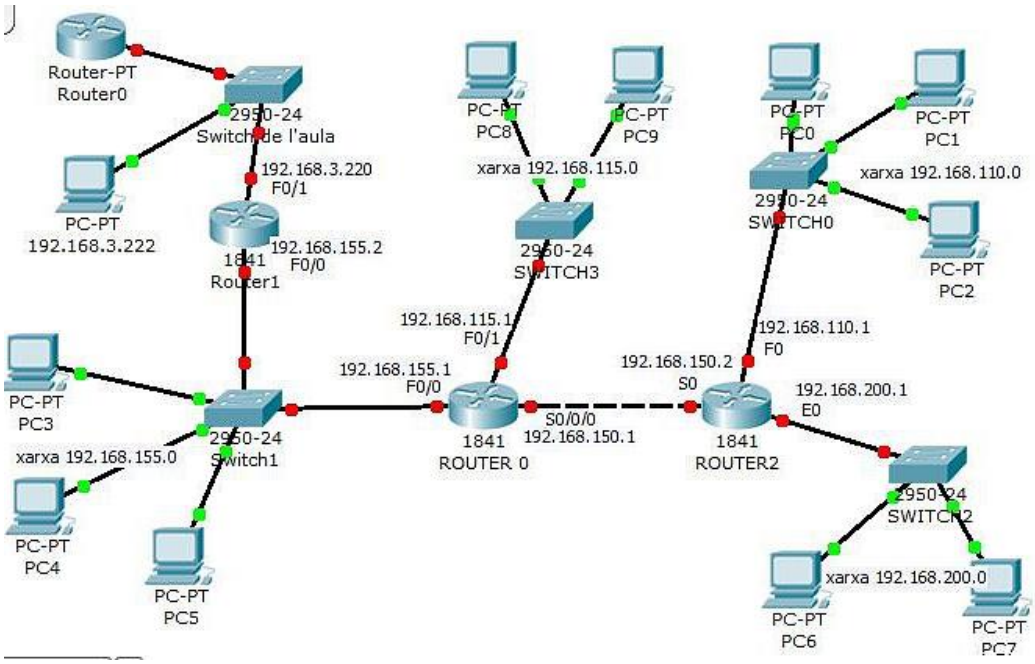


Figura 2.20 Ejemplo de red montada en Packet tracer
Fuente: [53]

Packet tracer cuenta con soporte para Windows y Linux. Tiene dos modos de trabajo: modo real y modo simulación; en el modo real se crea la red y sus configuraciones y en el modo simulación se ejecuta y pone en marcha la red simulada. En la Figura 2.21 se observa los iconos que permiten pasar de un modo a otro [54].

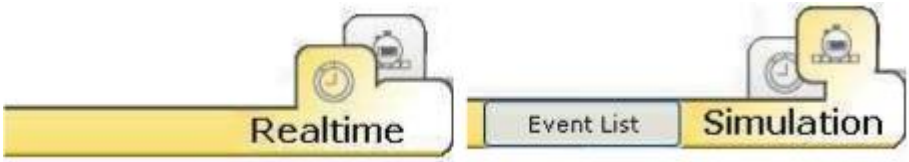


Figura 2.21 Iconos de modos de trabajo de Packet tracer
Fuente: [54]

GNS3 o Simulación gráfica de redes: GNS3 es gratuito y de código abierto, fue desarrollado hace más de 10 años, inicialmente este era capaz de emular únicamente dispositivos cisco, pero al transcurrir el tiempo ha evolucionado y hoy puede emular dispositivos de distintos proveedores como: Cisco, Brocade, Cumulus Linux, Docker, HPE entre otros. [41]

La arquitectura GNS3 se compone de 2 elementos: El software GNS3 (GUI) y la máquina virtual GN3 (MV). El cliente instala el software en su PC la cual puede contar

con un S.O. (Windows, MAC o Linux) y ya puede configurar la topología deseada, un ejemplo de una topología en GNS3 se muestra en la Figura 2.22.

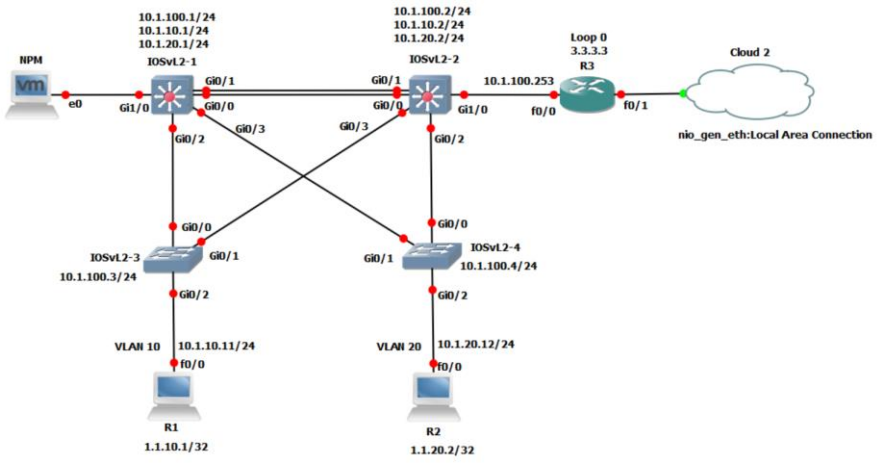


Figura 2.22 Ejemplo de topología montada en GNS3
Fuente: [58]

En GNS3 los dispositivos simulados en la topología se deben almacenar y ejecutar en servidores, gns3 cuenta con 3 opciones de servidor:

- Servidor GNS3 local, este tipo de servidor se ejecuta en la misma máquina donde fue instalado el software, es decir si se instaló en Windows todos los procesos de GNS3 se ejecutarán como procesos de Windows.
- Máquina virtual GNS3 local, este tipo de servidor es el más recomendado, aquí la máquina virtual se ejecuta en el mismo PC donde se encuentra instalado mediante un software de virtualización que puede ser VMware Workstation, VirtualBox, Citrix Xen, Proxmo, entre otros.

Con GNS3 se puede emular y simular una red, donde; la emulación se trata de imitar a un hardware y dentro copiar su IOS, en la simulación se imitan características y funciones desarrollados por GNS3 [58].

NetSim: Esta herramienta permite diseñar redes mediante una interfaz gráfica amigable pero la funcionalidad de la simulación es muy básica, pero es suficiente para el uso estudiantil.

Arquitectura básica de NetSim, su diseño se basa en capas, las capas se componen por módulos como se muestra en la Figura 2.23

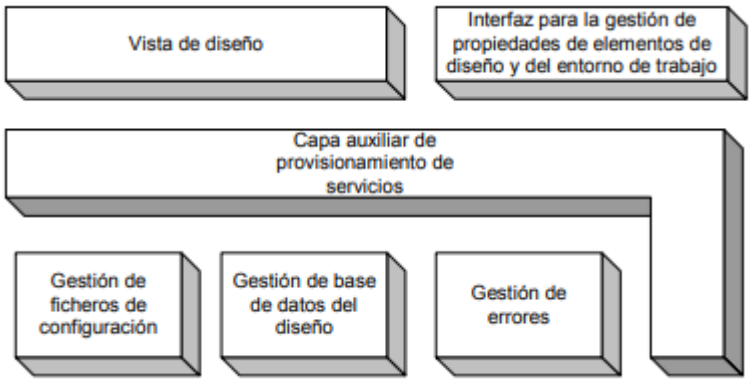


Figura 2.23 Estructura básica de NetSim
Fuente: [53]

Todos los módulos están a disposición de la capa más superior. Entre los componentes de bajo y más alto nivel esta la capa auxiliar de provisionamiento de servicios, esta debe ser considerada como un nivel lógico que permite que los elementos de las capas inferiores estén siempre disponibles para las superiores [53].

2.2.8.3. Herramientas de virtualización

Una herramienta de virtualización permite simular o emular al hardware en su totalidad y permite la ejecución de sistemas operativos. En estas se puede simular el sistema completo incluyendo instrucciones a nivel de cpu [54]

Existen distintas herramientas o plataformas que permiten la virtualización de una red, en la Tabla 2.4 se resumen dichas herramientas con sus características más relevantes.

Citrix XenServer	Tipo de virtualización: Nube, servidores y escritorios
	Gratuita y de código abierto
	Año de creación: 2003
	Ventajas: - Aumenta el porcentaje de uso de los servidores, reduciendo gastos en hardware y mano de obra - El hypervisor es gratuito y tiene un buen rendimiento. - Explota de la mejor forma las capacidades de virtualización por hardware de los procesadores de Intel y AMD - Eficiencia en el particionamiento de los recursos E/S - Migración de máquinas virtuales en caliente

	Hypervisor tipo I o Bared Metal
	Soporte para el sistema operativo invitado: Windows y linux
	Interfaz de control centralizada: Si
	Suscripción: No
	Alta disponibilidad: Si
	Características: - Cada máquina virtual se almacena como un recurso en el disco. - El particionamiento de la memoria no se hace por máquina virtual sino se comparte con las que se encuentren en ese momento en ejecución.
	RAM y CPU máxima por host: 6TB RAM y 160CPU
Proxmox VE	Tipo de virtualización: servidores
	Gratuita y de código abierto
	Año de creación: 2008
	Ventajas: - Genera respaldos de forma inmediata. - Realiza copias exactas (contenido de la RAM, configuración de los discos) - El hypervisor es gratuito y tiene un buen rendimiento. - Migración de máquinas virtuales en caliente
	Hypervisor tipo I o Bared Metal
	Soporte para el sistema operativo invitado: Windows y Linux
	Interfaz de control centralizada: Si
	Suscripción: Si, para habilitar todas las funciones
	Alta disponibilidad: Si
	Características: - Todos los nodos tienen su propio administrador - Se usa plantilla para las máquinas virtuales
	RAM y CPU máxima por host: 2TB RAM y 160CPU
VMware	Tipo de virtualización: servidores, nube, redes
	Versión completa de prueba y versión limitada gratuita
	Año de creación: 1998
	Ventajas: - Ilimitado cores por cpu - Ilimitada memoria RAM - El producto no espira

	Hypervisor tipo I o Bared Metal
	Soporte para el sistema operativo invitado: Windows, Linux, UNIX
	Interfaz de control centralizada: Si
	Suscripción: No
	Alta disponibilidad: Si
	Características: - Se basa en Linux, red hat interprise Linux - Tiene particionamiento para workstation y servidor VMware. - Proporciona una infraestructura virtual sólida, compatible y de alto rendimiento para datos empresariales del mundo real.
	RAM y CPU máxima por host: 2TB RAM y 160CPU

Tabla 2.4 Herramientas de virtualización más usadas.
Fuente: [33], [53], [54]

2.2.9. Tráfico en redes

Se define como tráfico al número de datos que atraviesan una red, en una red cada dispositivo o elemento generan datos, estos datos pueden ser:

- Alarmas: las alarmas son generadas por los dispositivos, que no son más que variables que detallan un problema como su identificador, hora del envió de la alarma. En una red moderada se generan aproximadamente de 200 a 1000 alarmas diariamente.
- Datos proporcionados por los distintos dispositivos que conforman la red como enrutadores, switches, servidores entre otros. Estos datos permiten conocer el estado de la red.

Con toda esta información se pueden realizar varias tareas de administración en la red, se deben discriminar las variables que contribuyen al tráfico de la red [23], [48], [59].

2.2.9.1. Tipos de tráfico

Generalmente el tráfico se genera en tres fuentes principales; dispositivos en la red, usuarios conectados en la red y servicios que prestan la red a los usuarios. Los dispositivos que componen una red mantienen constantemente alarmas para mantener la sincronización, configuración y relación entre otros dispositivos.

Una gran cantidad de tráfico se genera cuando los usuarios acceden a los servicios de la red, estos pueden ser:

- DHCP, este tráfico se genera cuando se adquiere, renueva y libera direcciones IP.
- Autenticación, se genera cuando se valida a un usuario en la red.
- Sesión, se genera cuando dos dispositivos inician sesión.
- Navegación en internet, se genera por ampliaciones de un navegador de internet para acceder a páginas hospedadas en el internet

Al depender el tráfico de cada dispositivo que compone la red, entre más amplia sea la red mayor será el tráfico generado, inicialmente las redes solo permitían el intercambio de información, pero paulatinamente sus servicios se fueron ampliando, estos servicios son videollamadas, voz sobre IP, aplicaciones web, entre otras [23], [48], [59]. Por lo que, según los usuarios que utilicen una red se consideran 2 tipos de tráfico: tráfico en masivo y tráfico en corporativo, como se muestran en la Figura 2.24 y Figura 2.25, respectivamente:

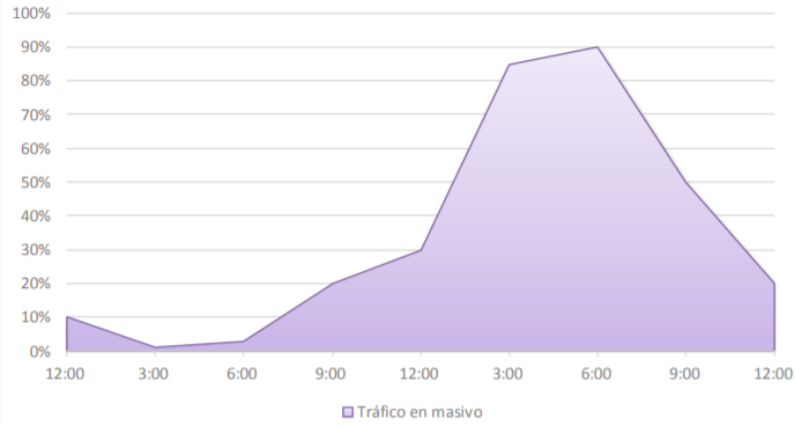


Figura 2.24 Tráfico en Masivo
Fuente: [18], [48]

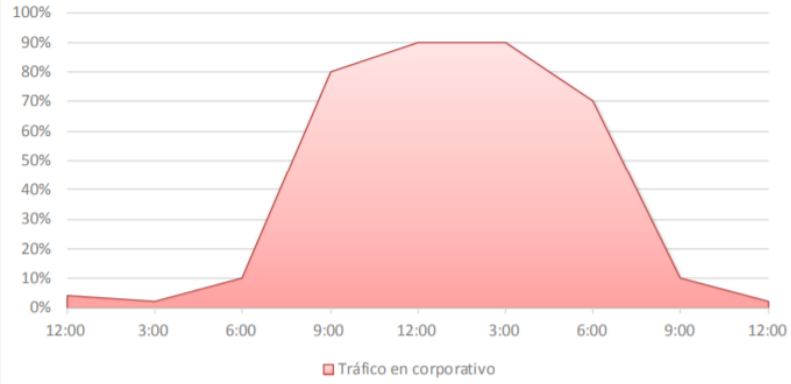


Figura 2.25 Tráfico en Corporativo
Fuente: [23], [48], [59].

2.2.9.2. Ingeniería de tráfico

Inicialmente las redes se basan en el funcionamiento del menor esfuerzo, es decir que ya es funcional si el paquete llega desde un origen a un destino. Esta comunicación únicamente cuenta con protocolos de enrutamiento, no existe prioridad en la entrega de paquetes, aquí nace la ingeniería de tráfico o más conocido por sus siglas en inglés como traffic engineering (TE).

Se genera tráfico en una red cuando fluyen datos a través de esta, todos los dispositivos en una red pueden generar tráfico, puede ser tráfico sobre su estado o el tipo de tráfico que fluye en este. Uno de los casos más comunes es que se genere congestión en un enlace determinado y otros enlaces se encuentren casi sin uso, como se muestra en la Figura 2.26.

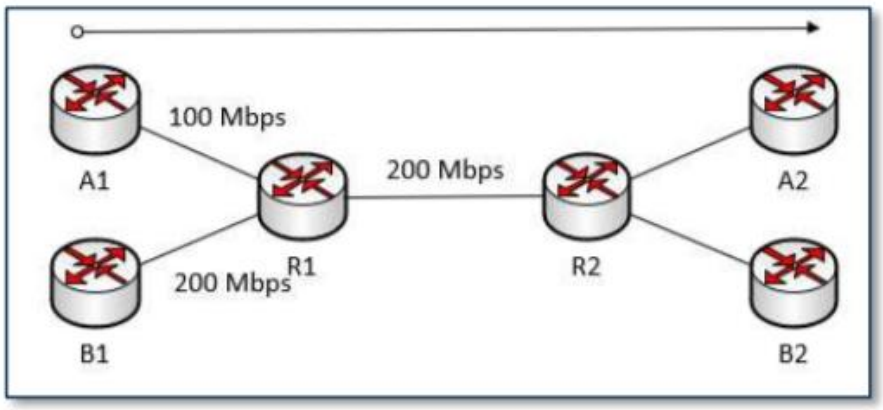


Figura 2.26 Representación de congestión
Fuente: [60]

Se puede considerar que una red se autogestiona de forma básica, ya que un dispositivo con protocolo TCP modifica su velocidad de envío de paquetes basado en el ancho de banda disponible en la ruta origen-destino. Los routers de la red son capaces de crear nuevas rutas basados en los cambios de topología de la red, con redes extensas estos mecanismos no son suficientes para un funcionamiento eficaz. La ingeniería de tráfico tiene la finalidad de adaptar el enrutamiento del tráfico según el estado de la red, cubriendo dos puntos muy importantes: un buen rendimiento del usuario y la utilización correcta de los recursos de la red [18], [60], [61]

La ingeniería de tráfico se fundamenta en la satisfacción de varios objetivos de rendimiento de la red, esto se logra con mecanismos o acciones que ayuden a los routers a escoger los caminos para cumplir con estos objetivos. La mayoría de los protocolos IGP se basan en valores de métricas (pesos), generalmente estos pesos son

establecidos por el administrador de la red, entonces los routers reúnen los pesos de cada enlace por el que deben atravesar, en una tabla se llenan la información de todas las rutas opcionales y se elige el camino más corto como se muestra en la Figura 2.31 .

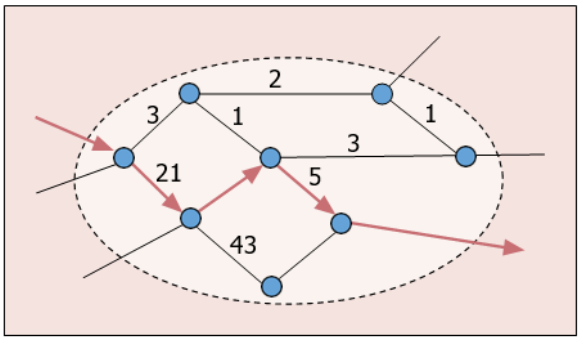


Figura 2.27 Enrutamiento del camino más corto basado en los pesos
Fuente: [48]

Estos pesos no se modifican en relación con las variaciones de tráfico o a fallas en los enlaces. La ingeniería de tráfico busca realizar las siguientes funciones:

- Gestión eficiente del ancho de banda que se tiene disponible.
- Crear más de un camino para un mismo destino cuando se presente mayor congestión
- Mejorar la eficiencia de funcionamiento de la red.
- Rendimiento del tráfico satisfactorio.
- Reducir errores por latencia, congestión y pérdida de paquetes.

Para su mejor comprensión se analizan tres escenarios: Configuración inicial con pesos de valor 1, modificación pesos en enlaces específicos y optimización global de los pesos:

- Pesos unitarios, en la Figura 2.41 se observa la ruta si todos los enlaces tuvieran el mismo peso (1), si se analiza el tráfico de q,r y s hacia t, se observa que por peso los tres se dirigirán a través de u hasta t y se generará una sobrecarga de tráfico en el enlace (u,t) [48].

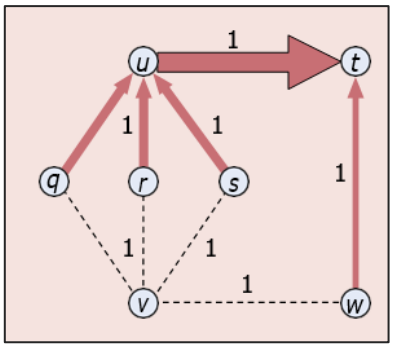


Figura 2.28 Enrutamiento con pesos unitarios en todos los enlaces
Fuente: [48]

- Modificación pesos en enlaces específicos, aquí se modifica el peso del enlace con más congestión, en este caso pasa a ser 2. Esto genera dos rutas más cortas para lograr llegar desde q,r y s hasta t esto supone 2.5 unidades de carga de enlace de (w,t) como se observa en la Figura 2.29

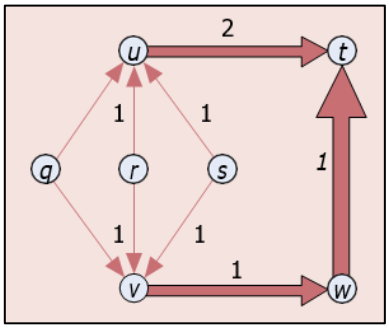


Figura 2.29 Enrutamiento con peso modificado en enlace específico
Fuente: [48]

- Optimización global de los pesos: Este ya no dependerá solo de los pesos sino también de las unidades de tráfico ya que se condicionará a que cada enlace no lleve más de 2 unidades de tráfico, este esquema es el óptimo y ya no puede ser mejorado con otro esquema. En la Figura 2.30 se muestra su funcionamiento.

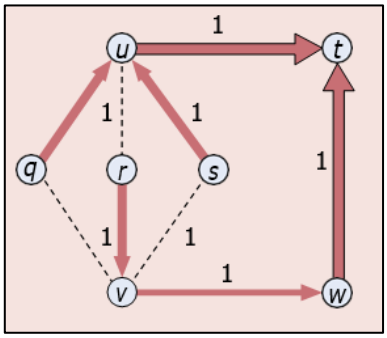


Figura 2.30 Optimización global de los pesos
Fuente: [48]

2.2.9.3. Tipos de Ingeniería de tráfico

Para poder cubrir los objetivos de la ingeniería de tráfico (TE) en una red, este tiene dos tipos de orientación:

- TE orientado al tráfico: Este se enfoca en perfeccionar las métricas relacionadas con el flujo de la información, lo que consigue reducir la pérdida de paquetes, retardo, jitter y el uso de enlace.
- TE orientado al recurso: Se basa en mejorar el uso de los recursos de la red, fundamentalmente en el ancho de banda [19], [48].

2.2.9.4. Calidad de servicio o QoS

La calidad de servicio o QoS, se orienta a proveer servicios de red sin comprometer la satisfacción del usuario, no se debe confundir con la ingeniería de tráfico. Una de las diferencias más relevantes entre la TE y QoS, es que QoS identifica la prioridad de un paquete y lo envía en el orden más conveniente para el usuario, mientras que la TE son mecanismos que aprovechan de mejor forma los recursos de la red [48].

Los parámetros de QoS son establecidos por el cliente y compartidos a su proveedor de internet o ISP, para implementar QoS en una red se deben realizar los siguientes procesos.

1. Bajo una regla se deben marcar los paquetes para poder agruparlos.
2. Se debe aplicar el tratamiento para cada grupo de paquetes y cumplir con el acuerdo del QoS.

2.2.9.5. Métricas para determinación de tráfico en una red

Las métricas son valores característicos de la red, los protocolos se basan en estas para tomar la decisión de qué ruta tomarán los datos, la métrica elegida depende de cada protocolo, las métricas comúnmente usadas son:

- **Nivel de uso:** este se mide en los enlaces, se define como el porcentaje de uso de un enlace, se calcula con la ecuación 2.2 [62].

$$\%uso = 100 * th/v \tag{2.2}$$

Donde:

$v = velocidad\ del\ enlace\ \left(\frac{bits}{seg}\right)$

$th = throughput\ o\ tasa\ de\ transferencia\ del\ enlace\ (\frac{bits}{seg})$

El throughput se usa para calcular la cantidad estimada de bits por segundo que pasan por un determinado enlace, este se calcula con la ecuación 2.3.

$$th = l/(\frac{d}{v} + \frac{l}{r}) \quad (2.3)$$

Donde:

$d = distancia\ entre\ los\ dispositivos\ que\ se\ establece\ la\ comunicación$

$th = throughput\ o\ tasa\ de\ transferencia\ del\ enlace\ (\frac{bits}{seg})$

$l = número\ de\ bits\ de\ la\ trama\ (bits)$

- **Latencia:** en está métrica se consideran dos tiempos, el tiempo que tarda un paquete en atravesar un enlace es decir desde el ingreso hasta la salida del enlace y el tiempo de procesamiento de este en cada nodo que compone la red. Aquí, es donde se identifica el paquete y los nodos toman decisiones [20], [62]. Según la aplicación los valores máximos de la latencia varían como se muestra en la Tabla 2.5.

Aplicación	**Máxima Latencia (ms)**
IpTV	100 ms
VoD	50 ms
VoIP	150 ms
Llamadas IP	150 ms
Video Streaming	300 ms
Video chat	150 ms
Video Conferencia	150 ms
Juegos	50 ms

Tabla 2.5 Máxima latencia según la aplicación.
Fuente:[33]

- **Pérdida de paquetes:** Al establecer una comunicación entre dos dispositivos se envían mensajes que se componen de varios paquetes, en ocasiones un porcentaje de estos paquetes no son capaces de llegar al destino. La pérdida de paquetes produce errores que en aplicaciones que no funcionan en tiempo real pueden ser solucionados, pero si estos ocurren en aplicaciones de tiempo real estas se afectan de forma inmediata. Esto puede afectar, por ejemplo, a una llamada de telefonía ip o a la imagen congelada en una video llamada [20], [61], [62].
- **Número de saltos:** El número de saltos es el número de routers que deberá atravesar un paquete desde su origen hasta su destino.
- **Coste:** Es un valor determinado por el administrador de la red, basado en el ancho de banda u otra característica de la red.
- **Ancho de banda:** Se define como ancho de banda a la cantidad de paquetes que atraviesan un enlace en un lapso determinado, la unidad de medida es kilobits por segundo (Kbps). El cálculo del ancho de banda se calcula:

$$B = (Cp - Ttp)8/100 \quad (2.4)$$

Donde,

$Cp\ es\ la\ capacidad\ del\ puerto$

$Ttp\ es\ la\ tasa\ de\ transferencia\ del\ puerto$

- **Confiabilidad:** Este indica las veces en que en un router a fallo se servicio este valor puede estar entre 0 o 255, entre menor sea su valor el enlace será más confiable
- **Unidad máxima de transmisión:** conocía comúnmente como MTU, es la longitud de una trama en octetos y es admitida por todos los nodos de la ruta [20], [61], [62].

2.2.9.6. Software para el análisis y monitoreo de tráfico

Las telecomunicaciones avanzan cada día y los servicios ofrecidos por la red son más complejos. Actualmente se ofrecen servicios de telefonía, de video y emisión de datos en tiempo real, esto implica que la red cada vez es más compleja de monitorear y analizar, para ello se han desarrollado herramientas que permiten efectuar funciones para hacer más fácil el monitoreo y análisis de una red, estas tareas pueden ser:

- Calcular el ancho de banda
- Determinar congestión en la red mediante patrones
- Filtrar los tipos de tráfico que fluyen en una red
- Clasificador de tipos de fallas

Entre más se obtenga del desempeño de la red y de los dispositivos, la herramienta será más efectiva, las más populares son:

- **CACTIL,** es una herramienta que permite obtener información de 4 parámetros importantes de la red que son: el tráfico de la red, uso de la PC, uso de memoria y tráfico de la red. Esta herramienta está desarrollada en PHP. Para la adquisición y presentación de los datos de la red utiliza RRDTools. Puede ser soportada por entorno Linux, solaris, BSD, Windows. Esta permite el uso de módulos externos que acceden a información de la red usando otras herramientas de gestión que permitan hacerlo.

- **SOLARWINS,** esta herramienta brinda la siguiente información de la red: tráfico de red, uso de pc, estadísticas de temperatura, eso de memoria y latencia. El uso de este software es pagado, trabajo con el protocolo SNMP.

- **NTOP,** es un sniffer, más conocido como TCPDump, que permite monitorear el tráfico y filtrarlo según el protocolo, puertos y aplicación a la que pertenecen.

- **WIRESHARK,** es uno de los sniffer más poderosos, utiliza las librerías con capturas de pccap, se puede instalar en Windows y Linux y acepta más de 300 protocolos. Las capturas del monitoreo de la red se pueden hacer en tiempo real o con datos previamente guardados. Wireshark cuenta con una interfaz al usuario intuitivo, es de código abierto [63].

2.2.10. Machine Learning

Los avances recientes en informática ofrecen las capacidades de almacenamiento y procesamiento necesarias para entrenar y probar modelos de aprendizaje automático para los voluminosos datos. Con la evolución de la computación en la nube, el uso ubicuo de computadoras, servicios y recursos digitales se vuelve cada vez más habitual en nuestra vida cotidiana. Esto ha llevado a una demanda de conexiones más rápidas y velocidades de datos más altas. Aunque existe una necesidad imperiosa de control cognitivo en la operación y gestión de la red, se plantea un conjunto único de desafíos para el aprendizaje automático [57].

El gran desarrollo de las redes virtuales en los entornos empresariales ha hecho que se genere la necesidad del uso de nuevos métodos para abordar los desafíos que se

presentan hoy en día. A diferencia de las estrategias diseñadas manualmente los algoritmos de aprendizaje automático ofrecen una enorme ventaja en los sistemas de redes.

Los algoritmos de aprendizaje automático más conocidos por su nombre en inglés como machine learning, son una rama de la computación que está diseñada para emular la inteligencia humana aprendiendo del entorno [23]. El aprendizaje automático (ML) permite que un sistema analice datos y deduzca conocimientos. Los algoritmos de aprendizaje automático utilizan métodos matemáticos computacionales para aprender sin necesidad de depender de una ecuación predeterminada. El rendimiento del algoritmo mejora de forma adaptativa con el aumento del número de muestras disponibles para el aprendizaje. Una de las ventajas de los algoritmos de aprendizaje es que las técnicas de ML pueden manejar con eficiencia problemas complejos y situaciones de gran dimensión, demostrando un éxito notable en el control de sistemas complejos, así como en los juegos de ordenador y sistemas de control en robótica. Los algoritmos de ML han sido ampliamente usados en el campo de las redes de datos (complejidad de las redes y requisitos QoS/QoE cada vez más exigentes) como para el desarrollo de tecnología como el monitoreo de redes y técnicas de análisis de big data [64],[24].

2.2.10.1. Clasificación de Algoritmos ML

Existe una amplia gama de algoritmos de aprendizaje automático, para mejorar su estudio estos se han clasificado de distintas maneras. Una de las primeras clasificaciones se presenta en [65], en donde se divide en 5 técnicas de algoritmos de aprendizaje. Sin embargo en [66], se realiza una nueva categorización y se los clasifica en 3 técnicas. En el año 2018 en [67], se presenta una revisión de las diferentes clasificaciones de algoritmos de aprendizaje, un total de 8 clases de algoritmos son presentadas en el estudio. Por otro lado, MathWorks en [68], presenta 2 clasificaciones que permiten entender el uso de los diferentes algoritmos de aprendizaje automático. Los algoritmos de aprendizaje automático se pueden clasificar en dos tipos de técnicas: aprendizaje supervisado y aprendizaje no supervisado, de las cuales se derivan varias técnicas, Figura 2.31.

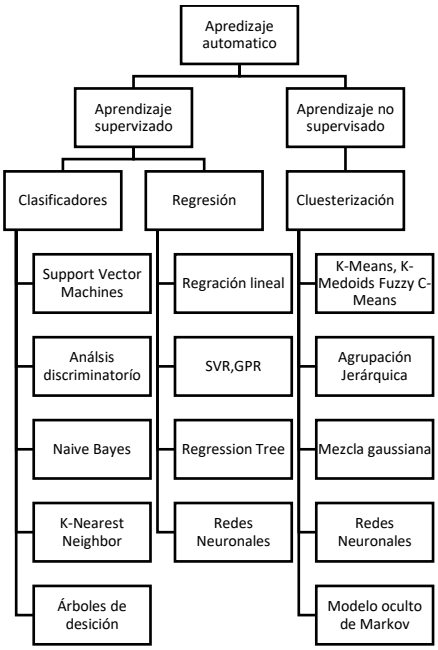


Figura 2.31 Clasificación de los algoritmos de aprendizaje automático
Fuente: [68]

2.2.10.1.1. Aprendizaje supervisado

Un modelo de aprendizaje supervisado se construye proporcionando al "sistema" un "datos de entrenamiento", es decir, entradas y sus resultados / productos conocidos para permitirle crear una relación en evidencias que presentan incertidumbre. A continuación, se proporcionan nuevas entradas para predecir en función de la relación aprendida.

Los datos de entrenamiento se describen como "etiquetados". Los problemas de aprendizaje supervisado se clasifican en problemas de "regresión" y "clasificación". En los problemas de regresión, el modelo intenta predecir resultados dentro de una salida continua, mientras que, en la clasificación, el modelo predice resultados en una salida discreta. Algunos de los algoritmos de aprendizaje supervisado comúnmente utilizados incluyen árboles de decisión, k vecinos más cercanos, bosque aleatorio, redes neuronales y máquinas de vectores de soporte [68], [70].

Algoritmos de clasificación

Son modelos que clasifican los datos de entrada en categorías en base a las características determinadas por el algoritmo. Dentro de los algoritmos de clasificación se pueden citar los más relevantes a:

SVM o Máquina de soporte vectorial

Este algoritmo de aprendizaje automático clasifica los datos una vez que encuentra el límite de decisión lineal dentro de un hiperplano separando las clases identificadas. SVM presenta un mayor margen entre 2 clases cuando los datos están linealmente separados. Cuando los datos no poseen una separación lineal se penaliza los datos con la finalidad de encontrar esa separación, Figura 2.32. Este algoritmo se puede usar en muestras que presenten una gran dimensión de datos y no sean linealmente separables, esto permite obtener un clasificador simple y fácil de interpretar [67], [68].

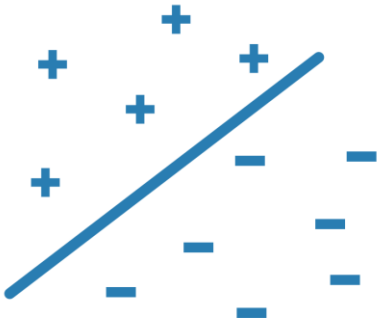


Figura 2.32 Ilustración del algoritmo de SVM
Fuente: [68]

Discriminant Analysis o Análisis discriminatorio

El algoritmo de análisis discriminatorio clasifica los datos mediante la búsqueda de combinaciones lineales de características, asumiendo una distribución gaussiana [67], [68]. Una de las desventajas de este método implica encontrar los parámetros que permiten calcular los límites de una función lineal para los datos de una distribución gaussiana para cada clase, Figura 2.33.

Figura 2.33 Ilustración del algoritmo de análisis discriminatorio.
Fuente: [68]

Naïve Bayes

Este algoritmo asume que las características de una clase no tienen relación y presencia con las de otra clase. El algoritmo clasifica los datos nuevos en función de la probabilidad más alta de que el dato pertenezca a una clase en particular, Figura 2.34. Es usualmente usada en un pequeño conjunto de datos que contienen una gran cantidad de parámetros para reducir la complejidad de clasificación. Sin embargo, el algoritmo se puede usar cuando los datos de entrenamiento no se encuentran en los escenarios deseados como son las aplicaciones médicas y financieras [67], [68].

Figura 2.34 Ilustración del algoritmo Naïve Bayes
Fuente: [68]

K Nearest Neighbor o Vecinos cercanos

Este algoritmo clasifica el conjunto de datos en base a los vecinos más cercanos, asumiendo que cada uno de los objetos son similares. Para calcular la distancia del punto centro hacia donde se encuentran los vecinos más cercanos se puede aplicar la distancia euclidiana, el coseno o la distancia de chabchev. Este algoritmo es usualmente usado cuando se necesita un algoritmo simple para establecer reglas de aprendizaje de referencia y se dispone de una gran capacidad de memoria y el tiempo no es preocupación para la resolución del problema [67], [68].

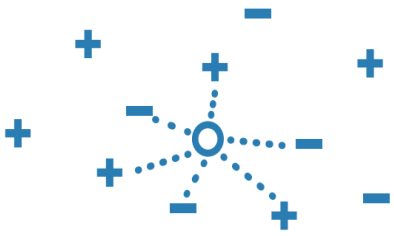


Figura 2.35 Ilustración del algoritmo K Nearest Neighbor.
Fuente: [68]

Decision Tree o Árboles de decisión

El árbol de decisiones permite decidir los datos de respuesta, siguiendo las decisiones hasta un nodo hoja. Un árbol de decisiones consiste en la condición de ramificaciones donde el valor de un predictor se compara con un peso de entrenamiento. El valor de los pesos se define en el proceso de entrenamiento, realizando modificaciones adicionales con la finalidad de simplificar el modelo, Figura 2.36. Usualmente al algoritmo de árboles de decisión se usa cuando se necesita un algoritmo que sea fácil de interpretar de rápido ajuste minimizando la capacidad de memoria y sin una alta precisión predictiva [67], [68].

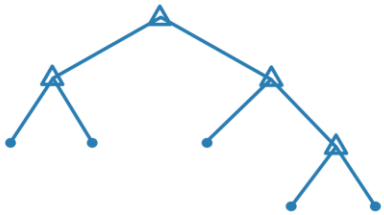


Figura 2.36 Ilustración del algoritmo árboles de decisión
Fuente: [68]

Algoritmos de regresión

Son modelos que predicen respuestas continuas las cuales se alimentan de datos históricos lo que les permite tener una mejor asertividad en el desarrollo del algoritmo.

Logistsc Regression o Regresión logística

Está técnica se basa en la predicción de la probabilidad de una respuesta binaria a una clase u otra, es usada como punto de partida para los problemas de clasificación binaria, Figura 2.37. La técnica de regresión logística se usa como un método de evaluación para el análisis de clasificaciones más complejas cuando los datos son separados claramente por un límite lineal [67], [68].

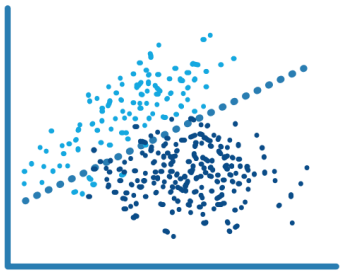


Figura 2.37 Ilustración del algoritmo de regresión logística
Fuente: [68]

SVM regression

Los algoritmos SVM de regresión funcionan como los algoritmos SVM de clasificación, pero a diferencia de los algoritmos de clasificación estos predicen una respuesta continua, Figura 2.38. El algoritmo tiene como objetivo principal minimizar la sensibilidad del error para los que el algoritmo encuentra un modelo que se desvía de los datos medidos en un valor no mayor que una pequeña cantidad, [67], [68].

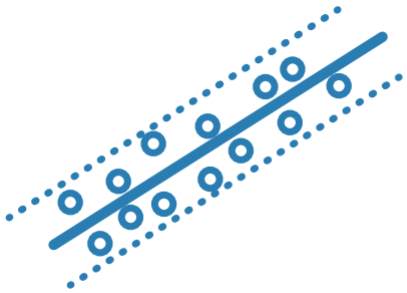


Figura 2.38 Ilustración del algoritmo de regresión SVM
Fuente: [68]

GRP, Gaussian Process Regression Mode

Este algoritmo hace usó de modelos no paramétricos que se usan para predecir el valor de una variable de respuestas continua. Son altamente usados en el análisis del campo espacial para interpolar datos en presencia de la incertidumbre. El modelo de GRP, se usa como modelo sustituto para facilitar la optimización de diseños complejos [67], [68].

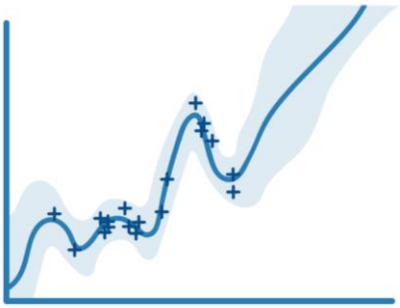


Figura 2.39 Ilustración del algoritmo de regresión logística
Fuente: [68]

Regression Tree

Este algoritmo se comporta como los árboles de decisión para clasificación a diferencia de estos los árboles de regresión se modifican para poder predecir la respuesta continua, Figura 2.40. Este algoritmo es usualmente usado cuando los predictores son categorías discretas o no se comportan de forma lineal [67], [68].

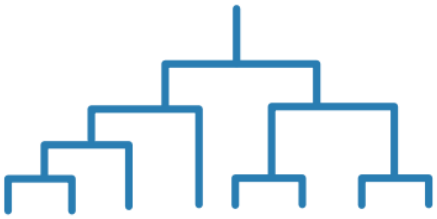


Figura 2.40 Ilustración del algoritmo de árboles de regresión
Fuente: [68]

2.2.10.1.2. Aprendizaje automático no supervisado

El aprendizaje automático no supervisado es exactamente lo contrario del enfoque supervisado. En este modelo, el sistema recibe un conjunto de datos de entrada sin sus resultados correspondientes. El modelo agrupa los datos en función de las relaciones entre las "características" "de los datos [71]. Los ejemplos populares de tales algoritmos incluyen k-medias y mapas autoorganizados.

K – means

En este algoritmo, la función de clasificación calcula similitudes entre una nueva muestra de prueba y todos los ejemplos etiquetados previamente asignados, consiste en el uso de un data set de entrenamiento, el cual define un espacio de distintas características. Luego, la muestra de prueba se asigna a la clase que tiene los K ejemplos más similares de los datos de entrenamiento [67], [68]. En caso de que no todos los K vecinos similares tengan la misma etiqueta, se aplica un voto mayoritario, Figura 2.41 .

Figura 2.41 Ilustración del algoritmo K-means
Fuente: [68]

K Medoids

De la misma forma que el algoritmo k-means, este algoritmo se centra en obtener la distancia a los demás datos de la data set con la diferencia a que el punto medio es parte de la data set, Figura 2.42 [67], [68].

Figura 2.42 Ilustración del algoritmo K-medoids
Fuente: [68]

Hierachical Clustering o Agrupación Jerárquica

La técnica de agrupación jerárquica se basa en la creación de conjuntos anidados con el análisis de las similitudes entre los pares de puntos, lo que permite agrupar los objetos en árboles binarios jerárquicos, Figura 2.43. Usualmente esta técnica es usada cuando no se tiene el conocimiento de cuantos clústeres existen en sus datos y su selección sea orientada de manera visual[67], [68].

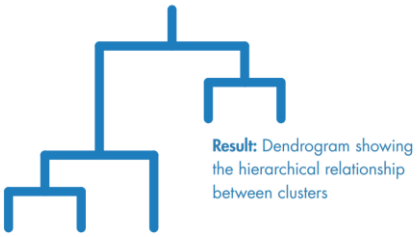


Figura 2.43 Ilustración del algoritmo agrupación jerárquica
Fuente: [68]

Gaussian Mixture o Mezcla gaussiana

La técnica de mezcla gaussiana se basa en la agrupación de datos en los cuales los puntos provienen de diferentes distribuciones normales multivariadas con un cierto grado de probabilidad, Figura 2.44. Esta técnica se usa cuando un dato puede pertenecer a más de un clúster o cuando el tamaño del clúster es diferente y presenta una estructura de correlación dentro de ella [67], [68].

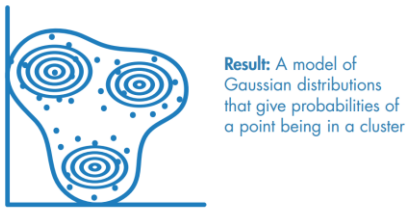


Figura 2.44 Ilustración del algoritmo mezcla gaussiana
Fuente: [68]

Neural Network o Redes neuronales

Las redes neuronales están basadas en las conexiones neuronales del cerebro humano. Las redes neuronales relacionan las entradas con las salidas deseadas. Las redes se modifican al configurar internamente los pesos de cada una de las variables, consideradas como las fortalezas de las conexiones en una red neuronal, esto permite predecir las respuestas correctas en base a las entradas dadas, Figura 2.45. Las redes neuronales son ampliamente utilizadas en sistemas incrementales que se deben actualizar constantemente [67], [68]..

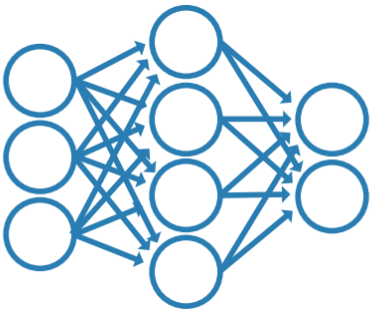


Figura 2.45 Ilustración del algoritmo de redes neuronales
Fuente: [68]

CAPÍTULO III

MARCO METODOLÓGICO

3.1. Ubicación

El trabajo de investigación se realizó en la ciudad de Ambato

3.2. Equipos y Materiales

Los equipos y materiales utilizados para el desarrollo del proyecto de investigación se describen en la Tabla 3.6.

Ítem	Descripción	Unidad	Cantidad	Precio Unitario (USD)	Precio Total (USD)
1	Computadora personal – mínimo core i7	u	1	1200	1200
2	Material bibliográfico, libros y revistas	u	1	350	350
5	Cursos virtuales para de redes virtuales y machine learning	u	1	100	100
6	Materiales: servicio de internet	u	1	100	100
				Costo Total:	1750

Tabla 3.6 Equipos y materiales utilizados

Fuente: El Autor

3.3. Tipo de investigación

3.3.1. Investigación Bibliográfica

La investigación aplicada, es una investigación bibliográfica porque la gran mayoría de datos e información utilizada para el desarrollo del trabajo, se obtuvo de libros, tesis doctorales, maestría, revistas, conferencias y artículos científicos. Esta información permite tener bases para fundamentar el presente trabajo, con la búsqueda de información se pudo conocer las herramientas más adecuadas para la simulación de la red y la aplicación del algoritmo.

3.4. Hipótesis – pregunta científica – idea a defender

¿Mediante el análisis de datos y algoritmos de machine learning se puede determinar la ruta óptima de tráfico en una red virtual?

3.5. Población o muestra:

En la presente investigación se aplica un tipo de muestreo no probabilístico de tipo discrecional, esto da la posibilidad que el autor pueda dar un juicio personal y está en la libertad de elegir el alcance del proyecto.

3.6. Recolección de la información

La recolección de información se basará en su mayoría en trabajos afines con el tema de investigación, esto permitirá encontrar datos y elementos de relevancia y de mucha utilidad para el desarrollo del trabajo de titulación. La recopilación también se hará en fuentes como libros, artículos, conferencias, entre otros.

Una fuente de investigación muy importante la completará con páginas web, específicamente páginas web de desarrolladores ya que el proyecto se basará principalmente en programación. La información recopilada permitirá hacer una red virtual, monitorearla y hacer los cambios respectivos en la ruta del tráfico y así mejorar el desempeño de la red.

3.7. Variables respuesta o resultados esperados

Los resultados esperados en el presente trabajo es que mediante análisis de datos y un algoritmo de machine learning se pueda escoger la ruta de tráfico óptima para la red virtual de estudio. Esto mediante los datos obtenidos al monitorear la red, determinar los cambios de ruta que se deban realizar en la red y comparar el comportamiento de la red antes y después de los cambios realizados y así afirmar que el comportamiento de la red mejoró al implementar la ruta óptima de tráfico.

CAPÍTULO IV

RESULTADOS E INTERPRETACIÓN DE RESULTADOS

4.1. Antecedentes

La metodología implementada para el desarrollo del proyecto de titulación se presenta en la Figura 4.46.

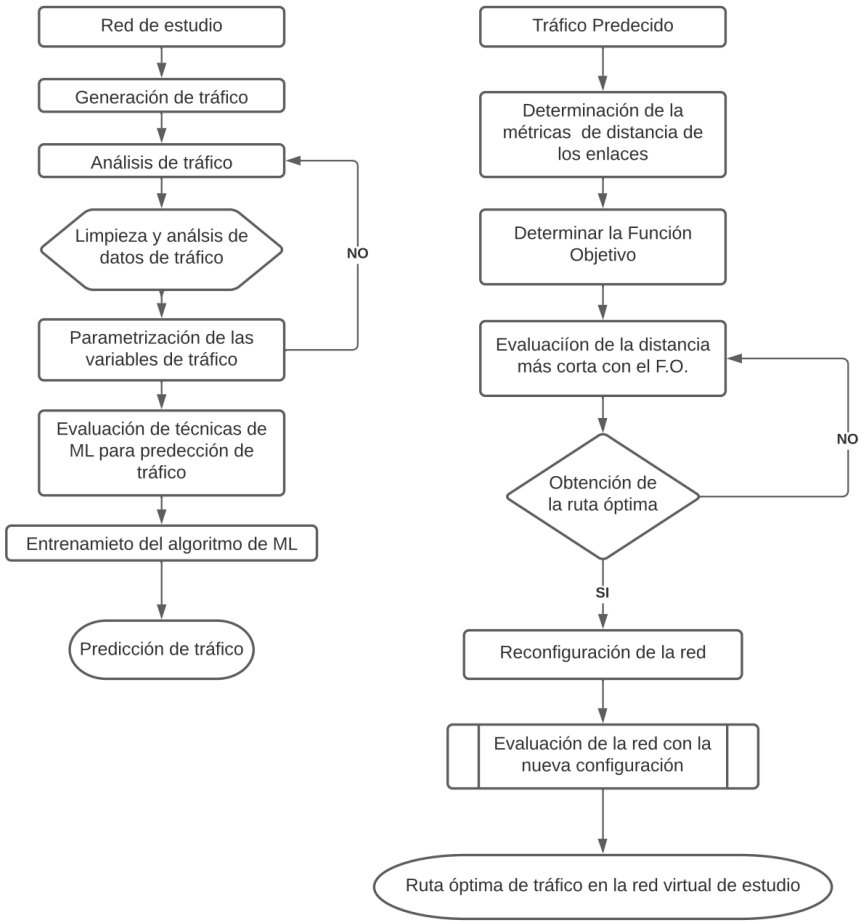


Figura 4.46 Metodología para la determinación de la ruta óptima de una red virtual
Fuente: El Autor

La metodología se divide en dos fases:

- La primera fase se centra en la predicción de tráfico de la red virtual de estudio.
- La segunda fase se enfoca en la determinación de las métricas de las distancias de cada enlace a partir de la predicción de tráfico realizada mediante el uso de una función objetivo de minimización de la métrica de la distancia para determinar la ruta óptima de la red virtual de estudio.

4.1.1. Requerimientos para el desarrollo del proyecto

Para la determinación de la ruta óptima de tráfico de una red virtual mediante el uso de análisis de datos y machine learning se ha hecho uso de una red totalmente mallada. La arquitectura de la red corporativa se basa en un ejemplo para desarrollo de habilidades de enrutamiento de la base de datos del curso de certificación en Cisco CCNA Routing and Switching, CCNA2, el ejemplo se adaptó a un entorno empresarial del Ecuador, estando la Matriz principal ubicada en Quito, y varias sucursales ubicadas en 4 ciudades (Ambato, Quito, Ibarra, Otavalo y Latacunga), cabe mencionar que la topología de la red tendrá varias oficinas en cada ciudad.

La arquitectura propuesta permite el análisis de la determinación de la ruta óptima de la red virtual. Para el desarrollo del proyecto se usó el simulador GNS3 debido a que este permite simular entornos de redes complejas y la configuración de los equipos como si fuera un entorno real. Por otro lado, GNS3 permitió simular la red propuesta sobre un entorno virtual de VMware, permitiendo validar la red virtual mediante el análisis de tráfico con la herramienta WireShark.

4.1.2. Topología de la red

La red de estudio se presenta en la Figura 4.47

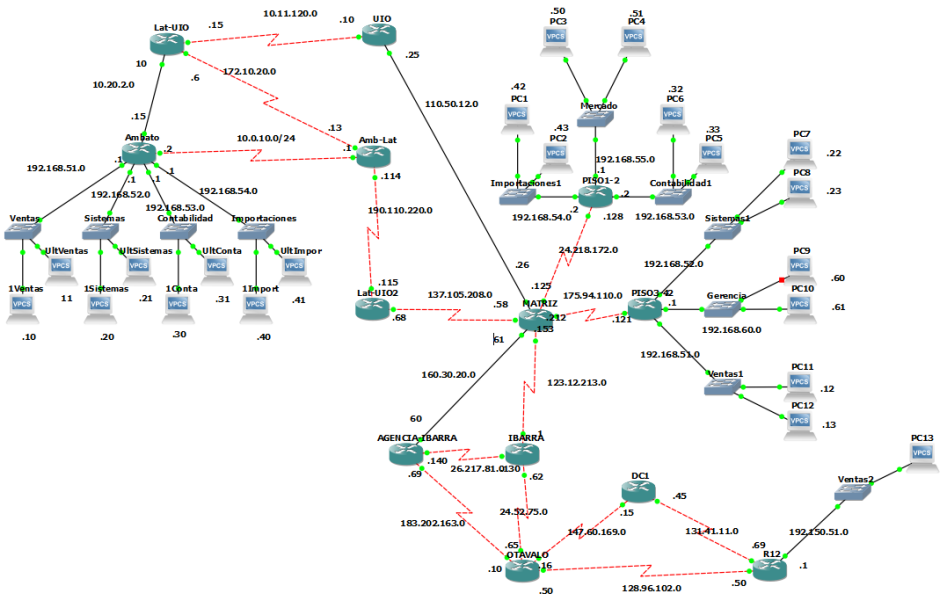


Figura 4.47 Topología de la red de estudio
Fuente: El Autor

El escenario de estudio consiste en una red de datos de una empresa que tiene la matriz en la ciudad de Quito y cuenta con 5 sucursales distribuidas en Ibarra, Ambato, Latacunga, Quito y Otavalo.

La red virtual simulada cuenta con un nodo principal denominado MATRIZ, 5 nodos que corresponden a los edificios centrales de las sucursales provinciales y 7 asignadas a edificios secundarios de las sucursales y la matriz. La red propuesta tiene como objetivo permitir el análisis de datos de tráfico para definir las métricas de cada uno de los enlaces y determinar la ruta óptima para él envió de información.

ESTRUCTURA DE LA MATRIZ:

La red de estudio tiene un total de 13 nodos: el nodo principal se denomina MATRIZ, que se compone de dos sub-redes, una designada al PISO 1-2 y otra al PISO 3-4. La red de estudio tiene un total de 3 vlans, asignadas (vlan 1 ‘‘Importaciones’’, vlan 2 ‘‘Mercado’’ y vlan 3 ‘‘Contabilidad1’’), adicional se cuenta con 1vlan para la parte de gerencia ubicada en el nodo MATRIZ.

En la Tabla 4.7 se presenta el direccionamiento de la red de estudio, interfaces y áreas asignadas.

DISPOSITIVO	INTERFAZ	DIRECCIÓN IP	MÁSCARA DE SUBRED	ÁREA ASIGNADA
ROUTER MATRIZ	Serial 1/0	123.12.213.153	255.255.255.0	IBARRA
	Serial 1/1	175.94.110.212	255.255.255.0	PISO 3-4
	Serial 1/2	24.218.172.125	255.255.255.0	PISO 1-2
	Serial 1/3	137.105.208.58	255.255.255.0	Lat-UIO2
ROUTER AMBATO	Serial 1/0	10.0.10.2	255.255.255.0	Amb-Lat
	F Ethernet 0/0	10.20.2.15	255.255.255.0	UIO
ROUTER UIO	Serial 1/0	10.11.120.10	255.255.255.0	Lat-UIO
	F Ethernet 0/0	10.20.2.25	255.255.255.0	Ambato
ROUTER Lat-UIO	Serial 1/0	10.11.120.15	255.255.255.0	Lat-UIO
	Serial 1/1	172.10.20.6	255.255.255.0	Amb-Lat
ROUTER Amb-Lat	Serial 1/0	10.0.10.1	255.255.255.0	Ambato
	Serial 1/1	172.10.20.13	255.255.255.0	Lat-UIO
	Serial 1/2	190.110.220.114	255.255.255.0	Lat-UIO2
ROUTER Lat-UIO2	Serial 1/2	190.110.220.115	255.255.255.0	Amb-Lat
	Serial 1/3	137.105.208.68	255.255.255.0	Matriz
ROUTER IBARRA	Serial 1/0	123.12.213.1	255.255.255.0	Matriz
	Serial 1/1	26.217.81.130	255.255.255.0	Agencia IBARRA
	Serial 1/2	24.52.75.62	255.255.255.0	Otavalo
ROUTER OTAVALO	Serial 1/0	24.52.75.65	255.255.255.0	Ibarra
	Serial 1/1	147.60.169.16	255.255.255.0	DC

	Serial 1/2	128.96.102.50	255.255.255.0	R12
	Serial 1/3	183.202.163.10	255.255.255.0	Agencia Ibarra
ROUTER AGENCIA IBARRA	Serial 1/0	26.217.81.140	255.255.255.0	Ibarra
	Serial 1/1	183.202.163.69	255.255.255.0	Otavalo
ROUTER DC1	Serial 1/0	147.60.169.15	255.255.255.0	Otavalo
	Serial 1/1	131.41.11.49	255.255.255.0	R12
ROUTER R12	Serial 1/0	131.41.11.50	255.255.255.0	DC
	Serial 1/1	128.96.102.69	255.255.255.0	Otavalo
ROUTER PISO 1-2	Serial 1/2	24.218.172.120	255.255.255.0	Matriz
ROUTER PISO 3-4	Serial 1/0	175.94.110.121	255.255.255.0	Matriz

Tabla 4.7 Direccionamiento de segmento 1
Fuente: El Autor

Las vlan de la red de estudio están distribuidas en los nodos principales Ambato y la Matriz (Quito). El direccionamiento de las redes virtuales asignadas para las vlans de la red se presenta en la Tabla 4.8.

Direccionamiento sucursal Ambato		
vlan	**Número de host**	**Dirección de red**
Ventas	2	192.168.51.0
Sistemas	2	192.168.52.0
Contabilidad	2	192.168.53.0
Importaciones	2	192.168.54.0
Direccionamiento sucursal Matriz		
vlan	**Número de host**	**Dirección de red**
Ventas	2	192.168.51.0
Sistemas	2	192.168.52.0
Contabilidad	2	192.168.53.0
Importaciones	2	192.168.54.0

Tabla 4.8 Direccionamiento de la VLAN de la red.
Fuente: El Autor

Como se puede observar en la Tabla 4.1 y Tabla 4.2 las máquinas fueron configuradas con una dirección de red perteneciente a cada una de las sub-redes implantadas para cada área. Por otro lado, cada uno de los router se configuraron con el protocolo de enrutamiento RIP. En la Figura 4.48, se presentan los datos más relevantes de la configuración actual del ruoter del nodo AMBATO, generada mediante en comando show running-config.

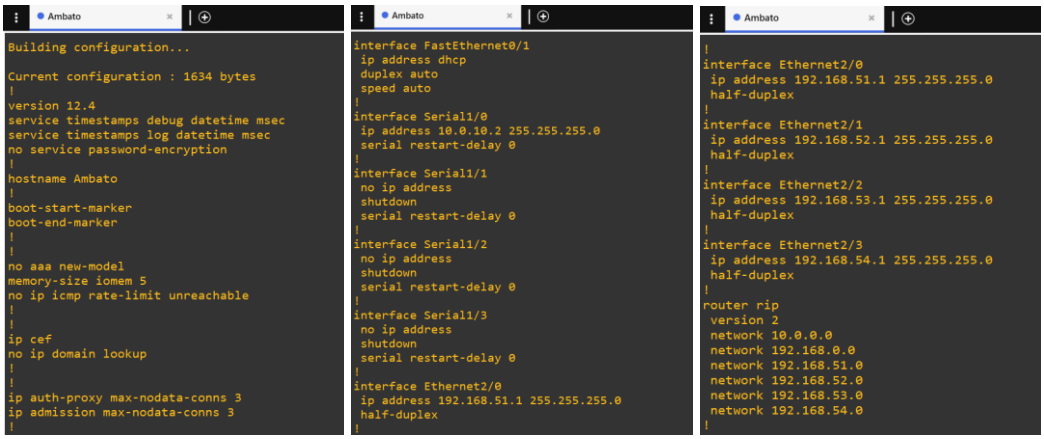


Figura 4.48 Configuraciones del router del nodo Ambato
Fuente: El autor

Posterior a la configuración de la red, se realizaron pruebas de conectividad de cada uno de los nodos. Las pruebas de conectividad se presentan en la Tabla 4.9.

Prueba de enlace	Respuesta del Ping
Ventas (AMBATO)-MATRIZ	1Ventas 1Ventas> ping ping 137.105.208.58 Cannot resolve ping 1Ventas> ping 137.105.208.58 84 bytes from 137.105.208.58 icmp_seq=1 ttl=252 time=54.728 ms 84 bytes from 137.105.208.58 icmp_seq=2 ttl=252 time=46.148 ms 84 bytes from 137.105.208.58 icmp_seq=3 ttl=252 time=64.230 ms 84 bytes from 137.105.208.58 icmp_seq=4 ttl=252 time=46.224 ms 84 bytes from 137.105.208.58 icmp_seq=5 ttl=252 time=47.170 ms
Ventas (PISO 3)-MATRIZ	1Ventas 1Ventas> ping ping 137.105.208.58 Cannot resolve ping 1Ventas> ping 137.105.208.58 84 bytes from 137.105.208.58 icmp_seq=1 ttl=252 time=54.728 ms 84 bytes from 137.105.208.58 icmp_seq=2 ttl=252 time=46.148 ms 84 bytes from 137.105.208.58 icmp_seq=3 ttl=252 time=64.230 ms 84 bytes from 137.105.208.58 icmp_seq=4 ttl=252 time=46.224 ms 84 bytes from 137.105.208.58 icmp_seq=5 ttl=252 time=47.170 ms
QUITO (SUCURSAL)-MATRIZ	1Ventas 1Ventas> ping ping 137.105.208.58 Cannot resolve ping 1Ventas> ping 137.105.208.58 84 bytes from 137.105.208.58 icmp_seq=1 ttl=252 time=54.728 ms 84 bytes from 137.105.208.58 icmp_seq=2 ttl=252 time=46.148 ms 84 bytes from 137.105.208.58 icmp_seq=3 ttl=252 time=64.230 ms 84 bytes from 137.105.208.58 icmp_seq=4 ttl=252 time=46.224 ms 84 bytes from 137.105.208.58 icmp_seq=5 ttl=252 time=47.170 ms
VENTAS (AMBATO)-VENTAS(PISO3)	1Ventas 1Ventas> ping ping 137.105.208.58 Cannot resolve ping 1Ventas> ping 137.105.208.58 84 bytes from 137.105.208.58 icmp_seq=1 ttl=252 time=54.728 ms 84 bytes from 137.105.208.58 icmp_seq=2 ttl=252 time=46.148 ms 84 bytes from 137.105.208.58 icmp_seq=3 ttl=252 time=64.230 ms 84 bytes from 137.105.208.58 icmp_seq=4 ttl=252 time=46.224 ms 84 bytes from 137.105.208.58 icmp_seq=5 ttl=252 time=47.170 ms

Tabla 4.9 Pruebas de conectividad de red virtual de estudio.
Fuente: El Autor

Una vez comprobada la conectividad se procedió a realizar varias pruebas de generación de tráfico, con la finalidad de obtener las variables que influyen en la generación de este.

Los datos utilizados para la generación de tráfico se realizaron bajo una configuración experimental controlada dentro de un escenario de simulación. Las pruebas de generación de tráfico se realizaron para un conjunto de valores de rendimiento y tamaños de paquetes predefinidos aplicando la configuración de diferentes remitentes, por lo que el total de tráfico generado en la red es proporcional al número de máquinas virtuales de la VM. Los tamaños de paquetes para la tasa de generación de tráfico son 128B, 256B, 512B, 1024B 1448B, respetando el tamaño máximo del paquete en una red ethernet de 1500 MTU en bytes, Tabla 4.10.

.

Escenario	**Tamaño de paquete**
1	128 bytes
2	256 bytes
3	512 bytes
4	10124 bytes
5	1448 bytes

Tabla 4.10 Tamaños de paquetes para los escenarios generados.
Fuente: El Autor

Las pruebas realizadas con la configuración de los distintos tamaños de paquetes plateados en los 5 escenarios se presentan en las: Figura 4.49, Figura 4.50, Figura 4.51, Figura 4.52 y Figura 4.53.

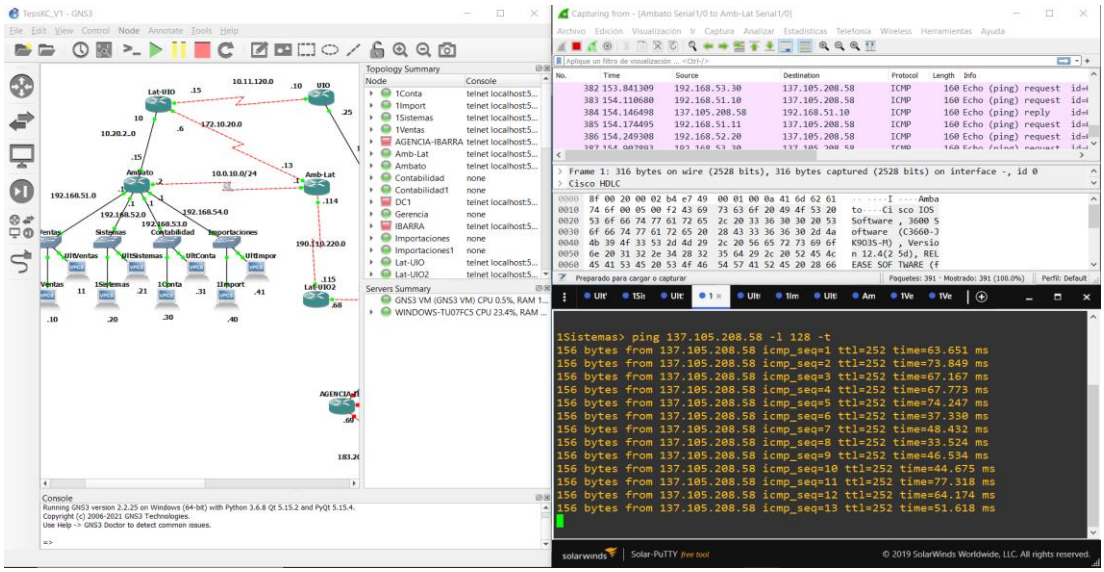


Figura 4.49 Petición múltiple con un tamaño de 128 bytes.
Fuente: El Autor

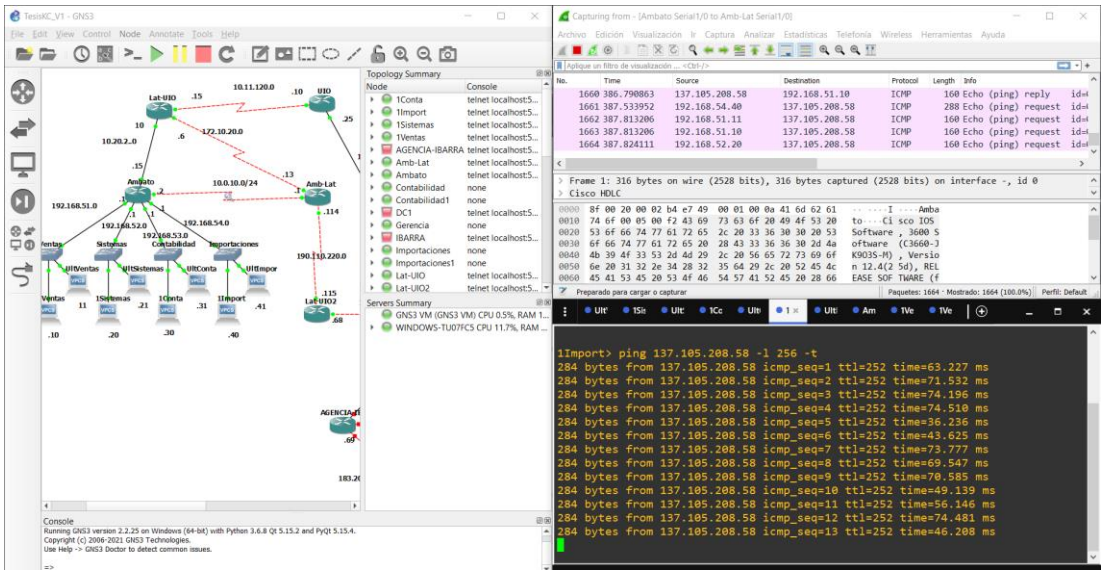


Figura 4.50 Petición múltiple con un tamaño de 256 bytes.
Fuente: El Autor

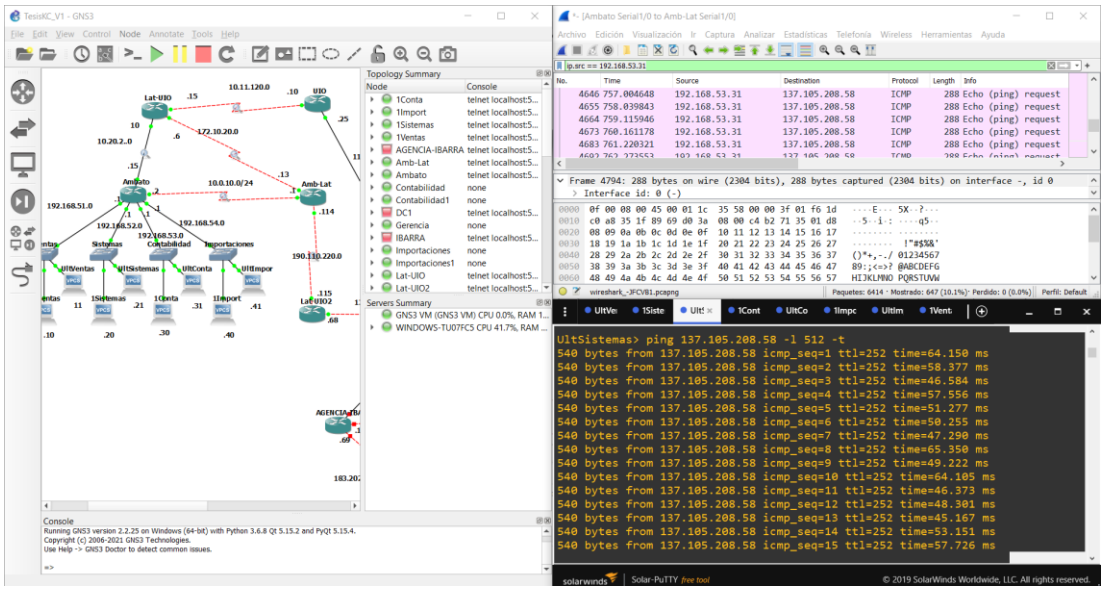


Figura 4.51 Petición múltiple con un tamaño de 512 bytes.
Fuente: El Autor

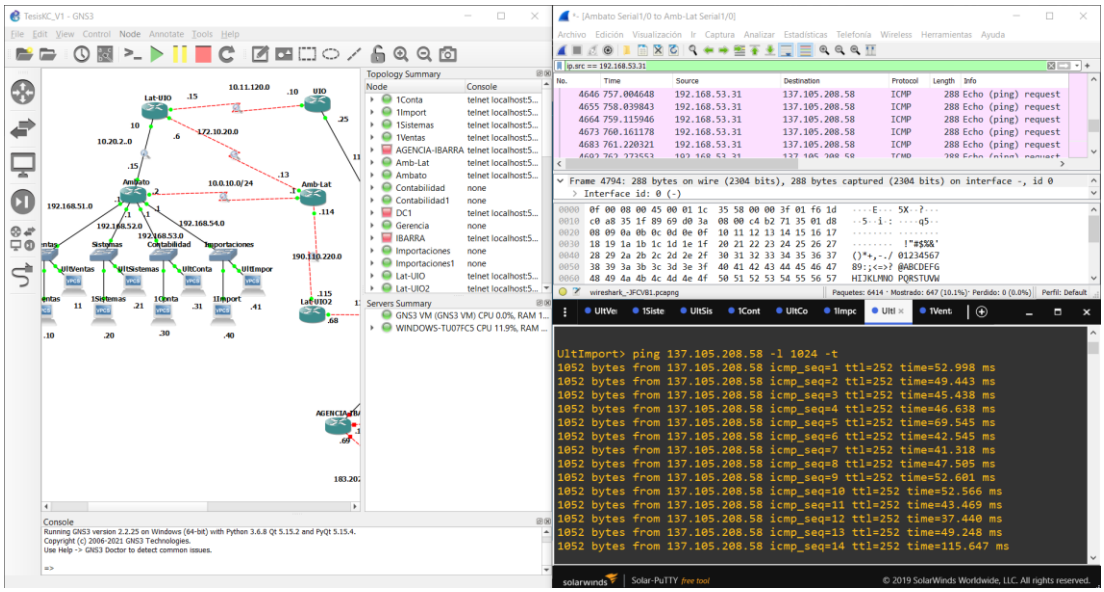


Figura 4.52 Petición múltiple con un tamaño de 1024 bytes.
Fuente: El Autor

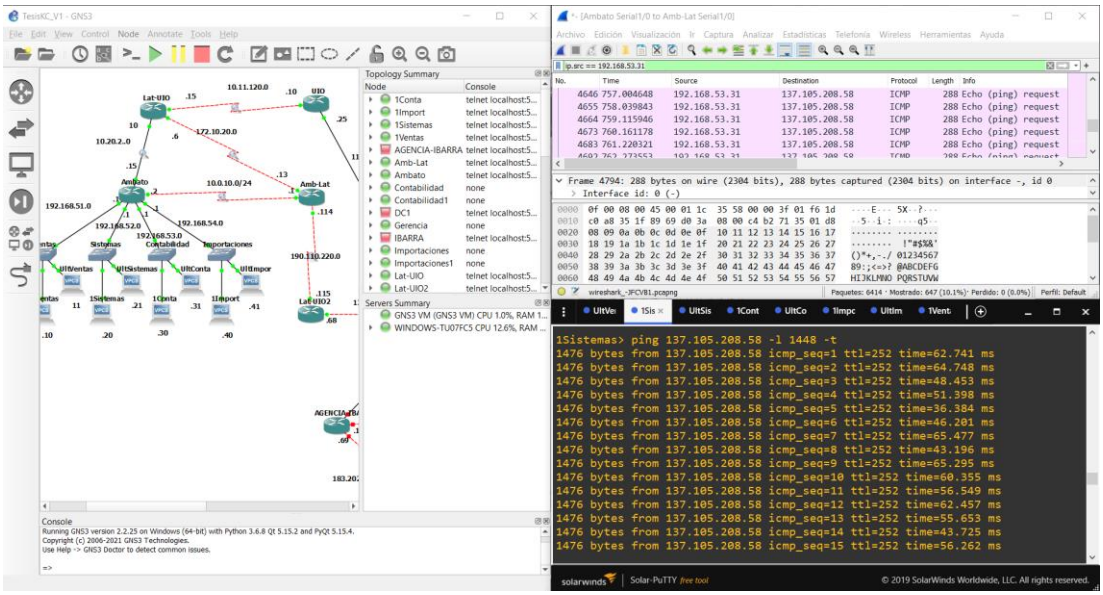


Figura 4.53 Petición múltiple con un tamaño de 1448 bytes.
Fuente: El Autor

Una vez iniciado el proceso de peticiones desde los clientes, se inicia un flujo de datos entre el nodo origen y el nodo destino. La herramienta Wireshark permite capturar el flujo de información desde que se inicia la transmisión de la información hasta el fin de la transmisión. El diagrama de flujo para el análisis de tráfico mediante la herramienta Wireshark, se presenta en la Figura 4.54.

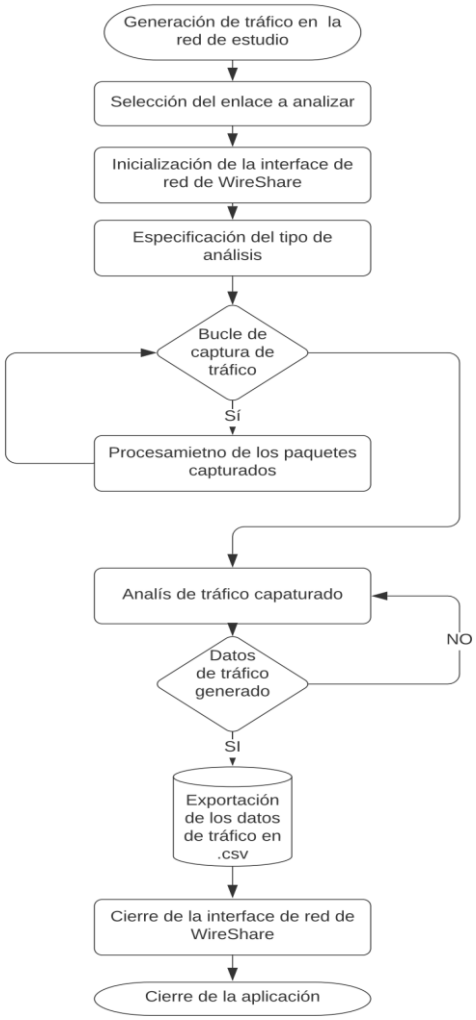


Figura 4.54 Diagrama de flujo del análisis de tráfico con la herramienta Wireshark en GNS3
Fuente: El Autor

Los datos capturados por la herramienta Wireshark se presentan en siete columnas. La primera columna contiene el número de los frames capturados en el tiempo que dura la petición desde los clientes. En la segunda columna se puede observar el tiempo de la duración de la transmisión este tiempo se contabiliza desde que la conexión ha sido aceptada. La columna tres presenta el direccionamiento IP de origen, de donde se envía el paquete. La cuarta columna muestra el direccionamiento IP destino, es decir a donde se envía la petición. En la columna cinco se presenta el tipo de protocolo de transporte que se utiliza para la generación del tráfico. La sexta columna muestra el tamaño de los bits enviados desde el origen al destino. Por último, la columna siete, entrega información adicional del paquete enviado.

En la Figura 4.55, se presenta una muestra de las capturas de tráfico realizadas en cada uno de los enlaces de la red, con la finalidad de generar una mayor cantidad de datos que permitan entrenar el modelo de la red neuronal para la predicción de este. Se capturó un total promedio de 8300 datos con un tiempo de captura de aproximadamente 20 minutos.

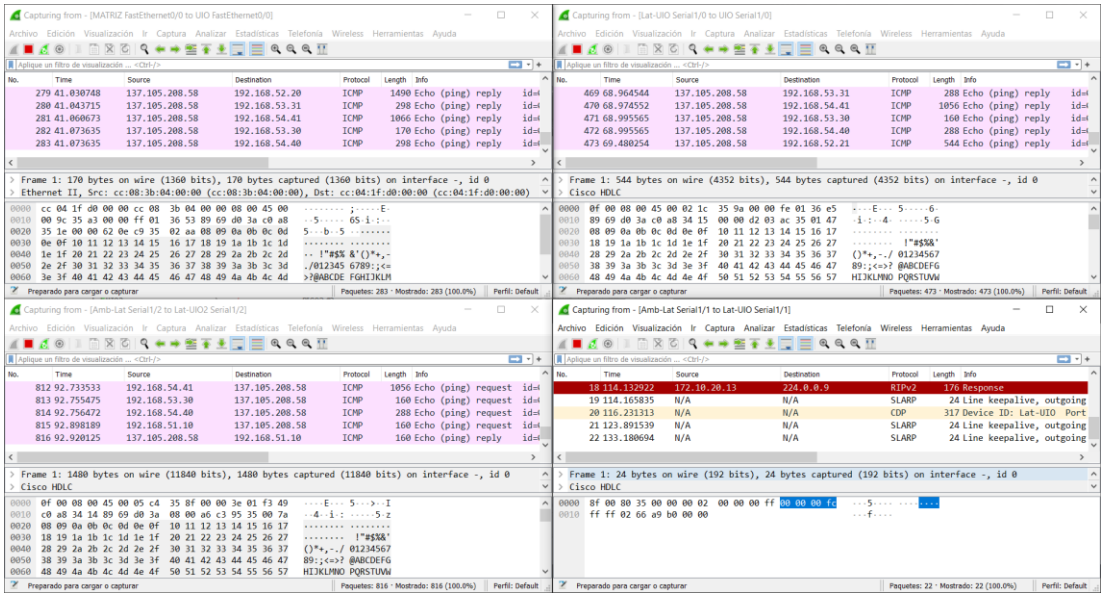


Figura 4.55 Muestra de Captura de tráfico en los enlaces de la red.
Fuente: El Autor

Desde la interfaz de la herramienta WireShark se selecciona el enlace en el cual se va a capturar el tráfico, se puede aplicar filtros de captura del tráfico con la finalidad de analizar los datos necesarios para el propósito deseado. En el caso de estudio no se utilizó ningún tipo de filtro ya que la finalidad del estudio es analizar por cuál de los enlaces fluye el tráfico a partir de la configuración de RIP. Como se puede apreciar en la Figura 4.56, el protocolo RIP, realiza su enrutamiento en base al número de saltos que se tiene desde el nodo origen al nodo destino.

El tráfico generado con la configuración del tamaño de los paquetes presentados en párrafos anteriores permitió observar el flujo total de datos entre los enlaces. Una vez realizada la captura de tráfico se procede analizar los datos capturados desde las herramientas de Wireshark. En la Figura 4.56, se presenta como se establece la conexión de nivel de red entre el cliente y el servidor. De la misma forma se puede observar cada uno de los identificadores IP que establecen la conexión.

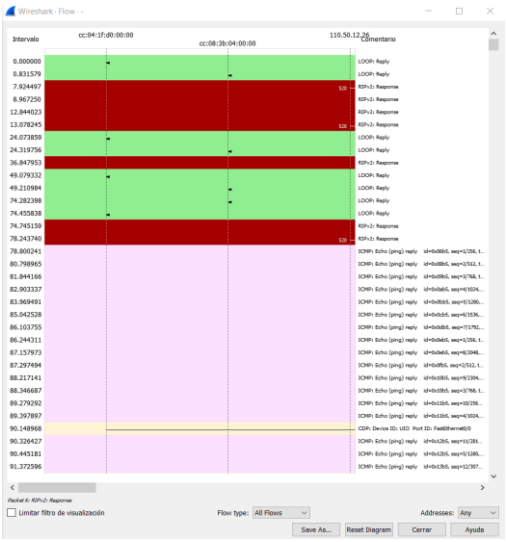


Figura 4.56 Muestra de Captura de tráfico en los enlaces de la red.
Fuente: El Autor

El tráfico generado por las peticiones desde los clientes a la MATRIZ analizados mediante la herramienta WireShark y las estadísticas de tráfico del router proporcionadas mediante el comando show ip traffic permiten identificar los diferentes protocolos y parámetros que fluyen desde la capa de red hasta la capa de enlace. Los parámetros identificados de cada uno de las capas y protocolos se presentan en la, Tabla 4.11 .

Parámetros	TCP o UDP	ICMP	IP	IPX	ETHERNET
Parámetros de capa 4	Puerto de origen	-	-	-	-
	Operador del puerto de origen	-	-	-	-
	Puerto destino	-	-	-	-
	Operador del puerto destino	Código ICMP	-	-	-
	N/A	Tipo de ICMP	N/A	-	-
Parámetros de capa 4	Bytes ToS IP	Byte ToS IP	Byte ToS IP	-	-
	Dirección IP origen	Dirección IP origen	Dirección IP origen	Red origen IPX	-
	Dirección de destino	Dirección de destino	Dirección de destino	Red de destino IP	-
	-	-	-	Nodo de destino IP	-
	TCP o UDP	ICMP	Otro protocolo	Tipo del paquete IPX	-
Parámetros de capa 2	-	-	-	-	Ethertype
	-	-	-	-	Dirección origen de las Ethernetes
	-	-	-	-	Dirección destino de las Ethernetes

Tabla 4.11 Parámetros identificados en las peticiones de los clientes.
Fuente: El Autor

En la Figura 4.57 se presenta el resumen estadístico de la longitud de paquetes para uno de los enlaces analizados. En la columna valores mínimos se puede observar los tamaños de paquetes configurados para la simulación de tráfico.

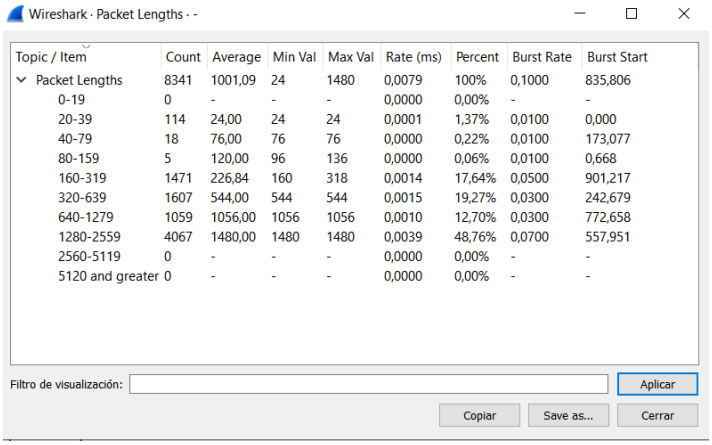
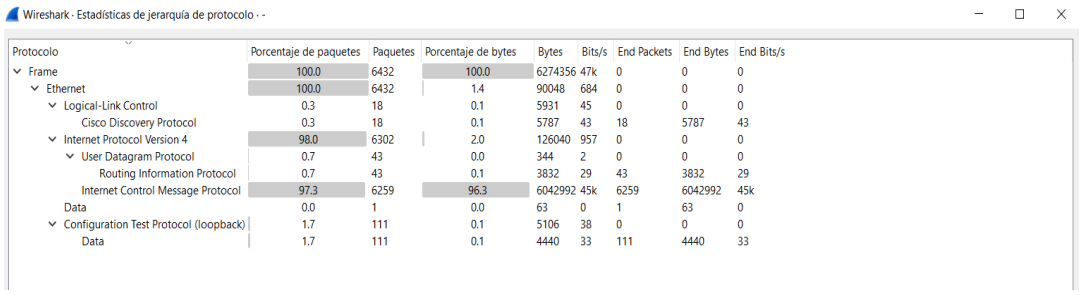
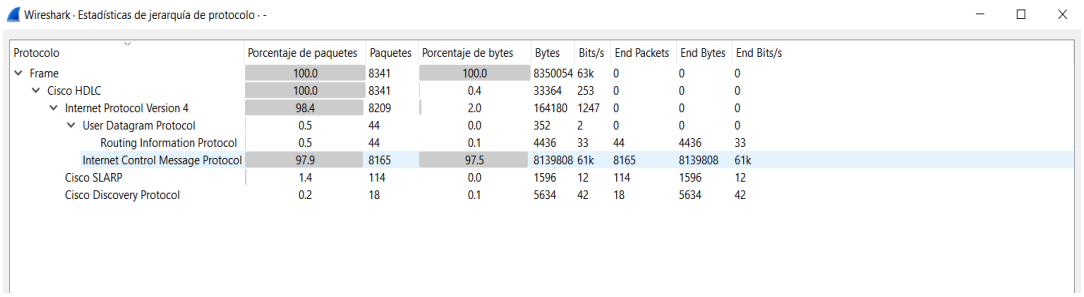


Figura 4.57 Parámetros identificados en la conexión de la comunicación.
Fuente: El Autor

Por otro lado, en las figuras: Figura 4.58 (a) y (b) se presentan las estadísticas de la jerarquía de protocolos de un enlace serial y de un enlace ethernet, respectivamente.



(a)



(b)

Figura 4.58 (a) Jerarquía de conexión de enlace ethernet (b) Jerarquía de conexión enlace serial.
Fuente: El Autor

Después de analizar cada uno de los resúmenes estadísticos generales se ha procedido a analizar el tráfico generado en cada uno de los enlaces. En la Figura 4.59 , se presenta una muestra del volumen de tráfico generado en el enlace Ambato Ambato-Latacunga.

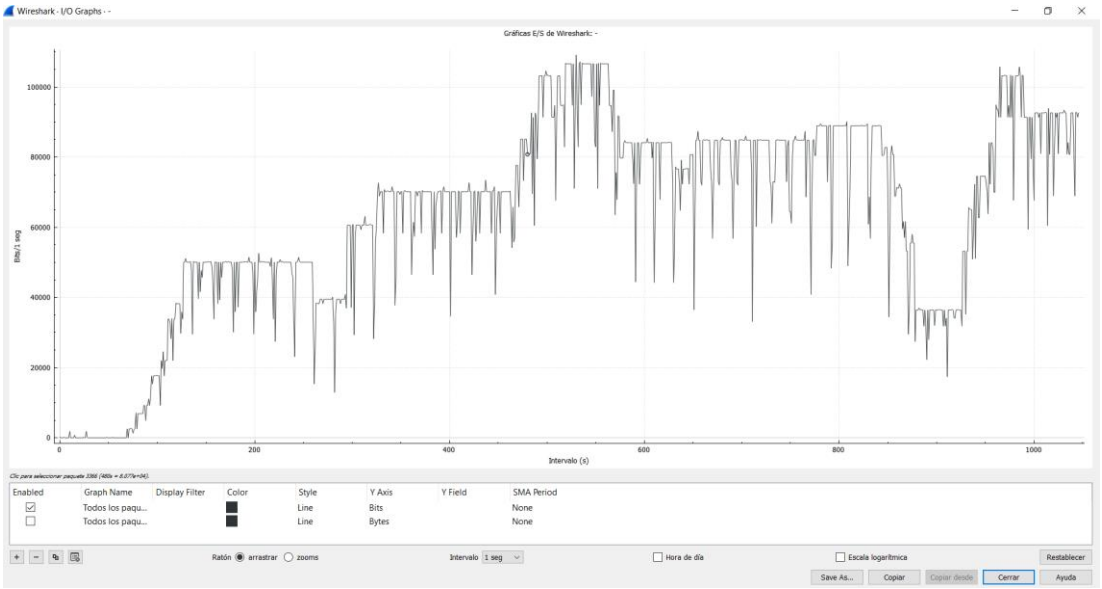


Figura 4.59 Tráfico generado en el enlace Ambato Ambato-Latacunga.
Fuente: El Autor

Obtenido el tráfico de red de cada uno de los enlaces de la red de estudio mediante la herramienta WireShark se procede a exportar los datos de tráfico a un archivo .csv para procesarlos en el programa MatLab.

El software Matriz Laboratory o más conocido por su nombre comercial como Matlab es un programa que permite el trabajo de forma matricial y sirve para resolver problemas complejos de ingeniería. En el caso de estudio, el software fue utilizado para el procesamiento de los datos de tráfico, obtenidos en la herramienta Wireshark y a su vez para el uso de las aplicaciones de machine learning que incluye dentro de sus toolbox.

Los datos se importaron a Matlab para realizar la limpieza de datos eliminando el ruido existente en los datos exportados. Una muestra del tráfico generado en la red y procesados en Matlab se presentan en la Figura 4.60.

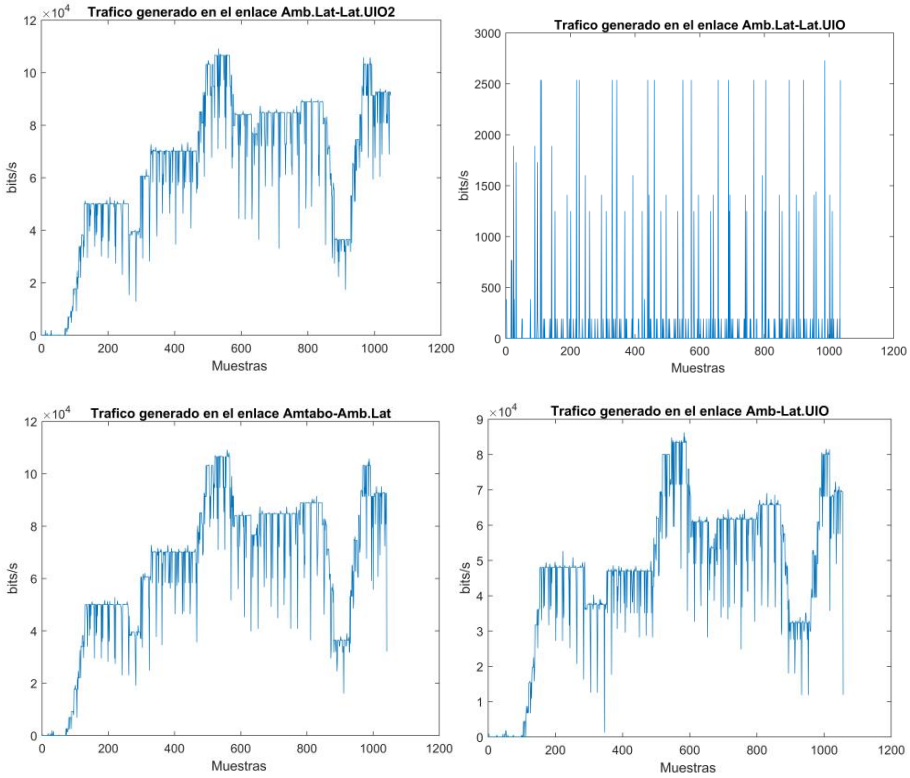
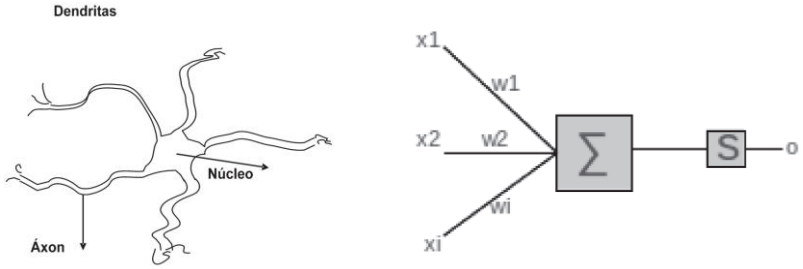


Figura 4.60 Muestra del volumen de tráfico generado en varios enlaces de la red.
Fuente: El Autor

Para la predicción de tráfico se utilizó la aplicación de redes neuronales de MatLab, una de las técnicas más usadas como mecanismo de predicción de tráfico en redes de alta velocidad para el control de congestión. Así mismo, la técnica de redes neuronales es ampliamente usada en predicción dinámica del ancho de banda, predicción de errores y clasificación de tráfico.

Como se menciona en el capítulo 2 las redes neuronales se basan en las conexiones neuronales del cerebro humano. Las redes neuronales se usan para la predicción de una salida en base a las entradas disponibles aplicándoles un peso de conexión.



(a) (b)

Figura 4.61 (a) Neurona del sistema nervioso (b) Neurona Artificial, Perceptron.
Fuente: El Autor

En la Figura 4.61. se presenta la estructura básica de una red neuronal conocida como perceptrón, donde:

xi, es el valor de entrada i-ésima

wi, es el peso de la conexión

o, es la salida de la red neuronal

s, es la función no lineal o función de activación

Usando el método de prueba y error se configuraron los parámetros de la red neuronal en Matlab hasta obtener una respuesta adecuada. Al analizar los datos obtenidos se observa que los mismos son consistentes entre la entrada y la salida por lo que se ha usado la aplicación de neuronal fitting con el algoritmo de backpropagation de Levenberg-Marquardt (trainlm). Este algoritmo normalmente requiere más memoria, pero menos tiempo. El entrenamiento se detiene automáticamente cuando la generalización deja de mejorar, como lo indica un aumento en el error cuadrático medio de las muestras de validación.

La matriz de entrada al modelo de la red neuronal entrenada está formada por el volumen de tráfico generada por cinco de los enlaces de la red y la salida de la red representa a los enlaces de borde de la red hacia la matriz. Para la configuración de los porcentajes de validación y testeo se dividió la muestra en: 65 % para el entrenamiento, 25% para la validación y 10% para el testeo de datos. Estos parámetros fueron configurados una vez detectado un fenómeno de overfitting en el entrenamiento de la red. Este fenómeno se presenta cuando la respuesta de la red neuronal se ajusta de forma exacta a las características de los datos de entrenamiento, lo que hace que la respuesta de la red no sea generalizable. Posteriormente, se realizó la configuración de las capas ocultas de la red neuronal. Las capas fueron validadas con el método de prueba y error desde 3 capas hasta 12 capas, obteniendo el mejor resultado de entrenamiento de la red con un total de 10 capas ocultas. La arquitectura de la red neuronal usada para predicción de tráfico se presenta en la Figura 4.62.

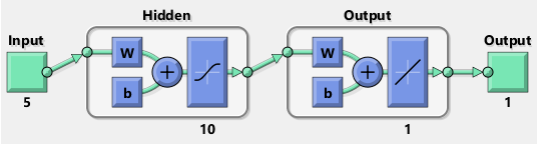


Figura 4.62 Arquitectura de la red neuronal para la predicción de tráfico.
Fuente: El Autor

La arquitectura usada en el entrenamiento de los datos presenta los resultados de regresión presentados en la Figura 4.63. Para obtener un ajuste perfecto, los datos deben caer a lo largo de una línea de 45 grados, donde las salidas de la red son iguales a los objetivos. Para este problema, el ajuste es razonablemente bueno para todos los conjuntos de datos, con valores de R en cada caso de 0,93 o más. Estos datos se han obtenido con una capacitación de la red de aproximadamente 20 veces. Cuando se vuelve a entrenar la red esta varía los pesos y sesgos iniciales de la red y puede producir una red mejorada después del reentrenamiento.

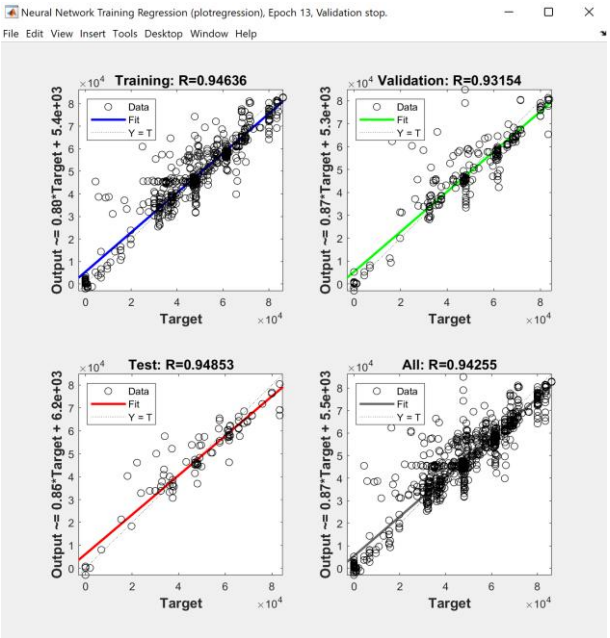


Figura 4.63 Gráfico de regresión de la red respecto a los objetivos de prueba.
Fuente: El Autor

Como se puede observar en la Figura 4.69 el error cuadrático medio va disminuyendo en cada uno de los segmentos de datos usados para el entrenamiento de la red neuronal.

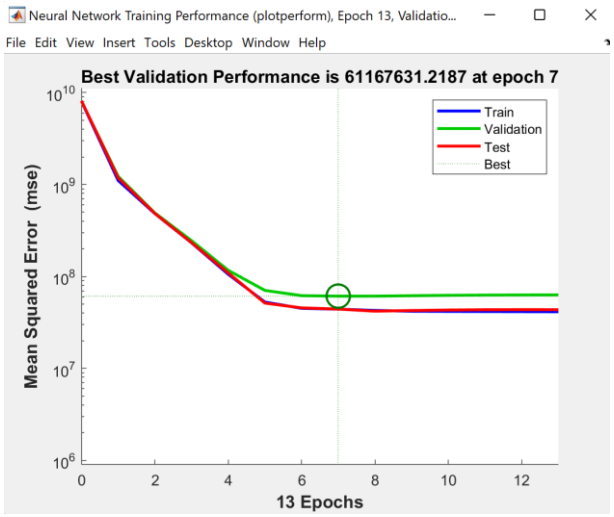


Figura 4.64 Error cuadrático medio del entrenamiento.
Fuente: El Autor

Después de realizar el entrenamiento de la red neuronal con el algoritmo de aprendizaje, se alcanza la predicción de tráfico de la red obteniendo datos de flujo adecuados y generalizables para toda la red de estudio, reduciendo la probabilidad de overfitting en el algoritmo. En la Figura 4.67Figura 4.65 Predicción del flujo de tráfico. se observa en la curva del tráfico obtenido en el software wireshark (color naranja) y la curva de predicción (color azul).

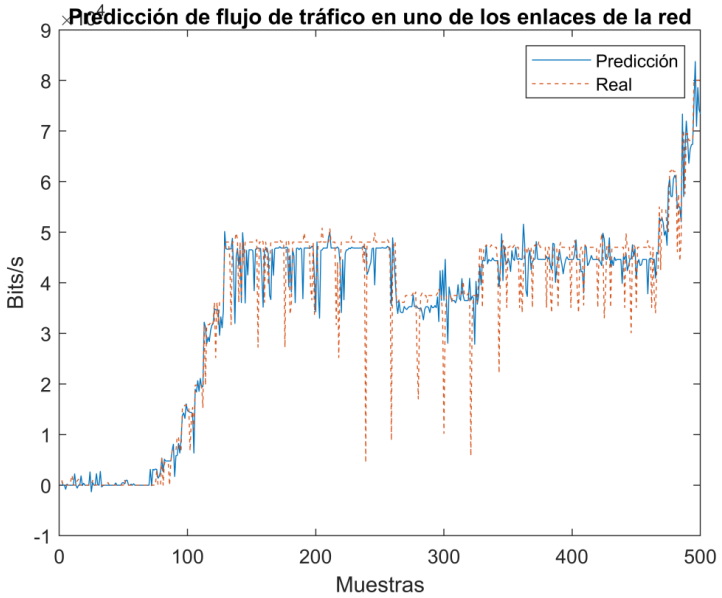


Figura 4.65 Predicción del flujo de tráfico.
Fuente: El Autor

Como se puede apreciar en la Figura 4.66, se puede ver como el error cuadrático medio se disminuye hasta alcanzar valores cercanos a cero.

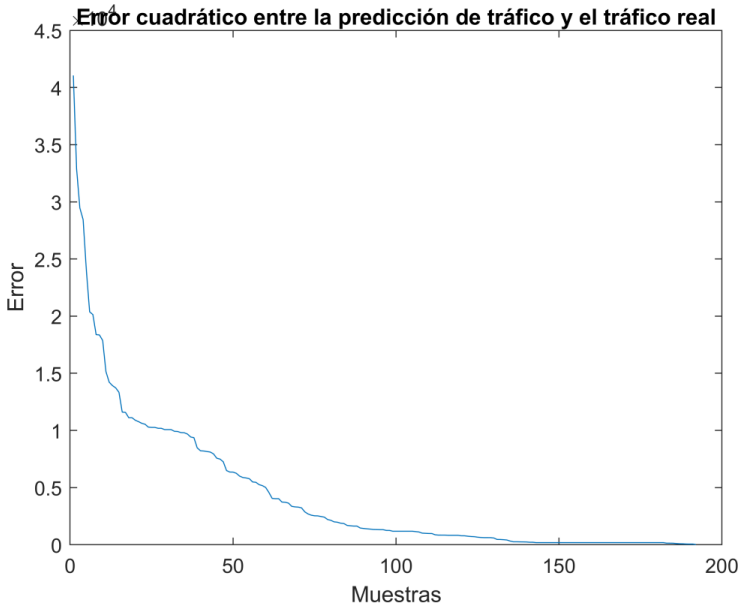


Figura 4.66 Error cuadrático entre el flujo de tráfico real y la predicción
Fuente: El Autor

Validado el algoritmo de predicción de tráfico se procedió a realizar la predicción de tráfico en los diferentes enlaces de la red con la finalidad de tener los parámetros adecuados para definir las métricas de distancia de cada uno de los enlaces.

El administrador de la red debe considerar varias variables que intervienen en la emisión y recepción correcta de un paquete en la red, los criterios generalmente considerados por un administrador de red son: número de dispositivos conectados a un nodo, latencia, ancho de banda, jitter pero hay otros valores que según el uso o servicio que preste la red deben considerar por ejemplo; el costo económico que implica la pérdida de paquetes de un área específica

Para determinar las métricas de distancia en la red de estudio se tomó en consideración:

- El tráfico promedio que tiene el enlace por lo cual se realizó la predicción de tráfico en un intervalo de tiempo de 10 minutos a carga máxima lo que permitió determinar el tráfico promedio del enlace.

- El ancho de banda disponible para cada uno de los enlaces se definió en base al tipo de comunicación.
- La disponibilidad del sistema se determinó en base al porcentaje de tráfico generado en el enlace sobre el ancho de banda disponible.
- El número de saltos es un dato proporcionado por cada uno de los routers y se lo genera utilizando el comando show ip router.
- La confiabilidad del sistema se obtiene mediante el número de paquetes perdidos, de las estadísticas generales del análisis del tráfico con la herramienta WireShark.
- El costo del enlace se asumió en base a los servicios que dispone la red y es igual a las pérdidas económicas generadas por la pérdida de comunicación del enlace.
- La prioridad del servicio es cuantificada de 1 a 4 de alta a baja prioridad en base a los tipos de servicios a los que se conecta cada uno de los routers.

En la Tabla 4.12 se presentan las variables a tomar en cuenta para determinar la métrica de distancia de cada enlace.

X	ENLACE	TRÁFICO (Kbps)	ANCHO DE BANDA DISPONIBLE (Mbps)	DISPONIBILIDAD DEL ENLACE	# DE SALTOS MAX	CONFIABILIDAD	COSTO	PRIORIDAD DE SERVICIO
x5	Amb/Lat-UIO	71,95	1,54	95,33	3	97,46	$3.000	2
x4	Amb/Amb-Lat	50,63	1,54	96,71	4	87,03	$3.000	1
x3	Amb-Lat/Lat-UIO	42,41	1,54	97,25	4	92,84	$2.000	3
x6	Lat-UIO/UIO	56,13	1,54	96,36	2	86,21	$1.500	2
x2	Amb-Lat/Lat-UIO2	41,62	1,54	97,30	2	95,14	$1.000	1
x7	UIO/MATRIZ	59,26	100	99,94	1	86,21	$5.000	2
x1	Lat-UIO2/MATRIZ	20,84	1,54	98,65	1	96,16	$5.000	1
x8	A. IBARRA/MATRIZ	50,29	100	99,95	1	86,3	$4.000	2
x10	IBARRA/MATRIZ	44,62	1,54	97,10	1	95,4	$3.000	1
x9	IBARRA/A. IBARRA	69,49	1,54	95,49	2	87	$2.000	2
x12	A. IBARRA/OTAVALO	61,09	1,54	96,03	2	87,45	$1.500	2
x11	IBARRA/OTAVALO	66,59	1,54	95,68	2	87,13	$1.500	3
x13	OTAVALO/DC	57,56	1,54	96,26	3	95,32	$1.000	4
x14	DC/R12	60,9	1,54	96,05	4	96,5	$1.000	4
x15	OTAVALO/R12	38,04	1,54	97,53	3	98,1	$1.000	4

Tabla 4.12 Tabla de parámetros para la determinación de la métrica.
Fuente: El Autor

Con los parámetros determinados en la Tabla 4.12, se procedió a calcular la métrica de cada uno de los enlaces. Al ser variables no homogéneas se aplicó el método SAW, que consiste en encontrar la suma ponderada de las calificaciones de desempeño en cada alternativa en todos los atributos. El método SAW requiere el proceso de

normalizar la matriz de decisión (X) a una escala comparable a todas las calificaciones alternativas existentes [5].

El valor de preferencia para cada alternativa (Vi) se da como se presenta en la ecuación (4.3):

$$V_i = \Sigma_j^n = 1 W_j r_{ij} \quad (4.3)$$

V_i: El valor final de la alternativa

W_j: El peso especificado

r_{ij}: Normalización de la matriz

Las ventajas del método de ponderación aditiva simple en comparación con otros modelos de toma de decisiones residen en su capacidad para realizar juicios con mayor precisión porque se basa en un valor predefinido y una ponderación de preferencia.

Una vez aplicado el método de SAW a la Tabla 4.12 se ha obtenido los valores normalizados y ponderados de cada una de las variables, lo que permite determinar la métrica de cada enlace como la suma total de todas sus variables, Tabla 4.13.

ENLACE	TRÁFICO (Kbps)	ANCHO DE BANDA (Mbps)	% DISPONIBILIDAD	# DE SALTOS MAX	% CONFIABILIDAD	COSTO	PRIORIDAD DE SERVICIO	MÉTRICA
x5	5	1	1	4	1	2	2	16
x4	3	5	3	5	4	2	1	23
x3	2	5	3	5	3	3	3	24
x6	3	5	4	3	5	4	2	26
x2	1	5	2	3	3	5	1	20
x7	2	1	1	1	5	1	2	13
x1	1	5	2	1	2	1	1	13
x8	3	1	1	1	5	1	2	14
x10	2	5	3	1	2	2	1	16
x9	5	5	5	3	4	3	2	27
x12	4	5	5	3	3	4	2	26
x11	5	5	5	3	4	4	3	29
x13	4	5	4	4	2	5	4	28
x14	4	5	4	5	1	5	4	28
x15	1	5	2	4	1	5	4	22

Tabla 4.13 Determinación de la métrica de cada uno de los enlaces.
Fuente: El Autor

En esta fase del trabajo se presenta el modelo matemático aplicado para encontrar la ruta óptima en la red de estudio, para todos los flujos que salen de un nodo x hacia el nodo Matriz. Los enlaces de la red de estudio se representan en la Figura 4.67 siendo el nodo A la oficina central denominada Matriz.

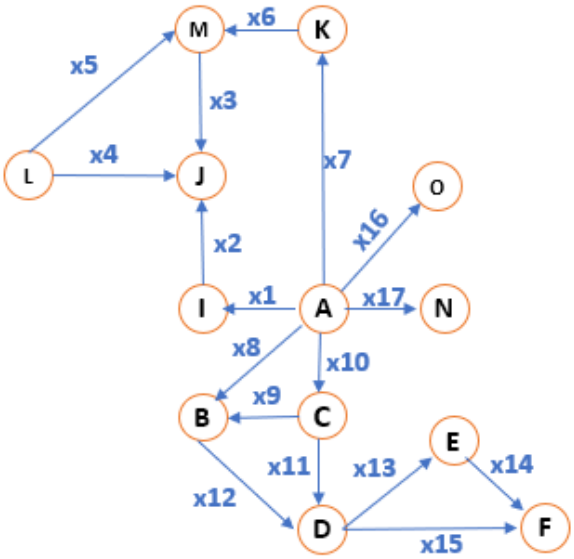


Figura 4.67 Representación gráfica de la red
Fuente El autor

El modelo tendrá como datos iniciales la matriz de flujos o rutas posibles y las métricas de distancia calculadas de cada enlace en función al tráfico de la red, obtenidas con Matlab.

La matriz de flujo o rutas posibles se obtuvo mediante el uso de un árbol de decisión, se toma como objeto de estudio las rutas más largas de ambos segmentos, en el caso del segmento 1 la ruta es desde la Matriz Nodo A hasta el Nodo G, y en el segmento 2 la ruta más crítica es desde el Nodo A hasta el Nodo F, para todos los cálculos la red se dividió en 2 segmentos, Figura 4.70

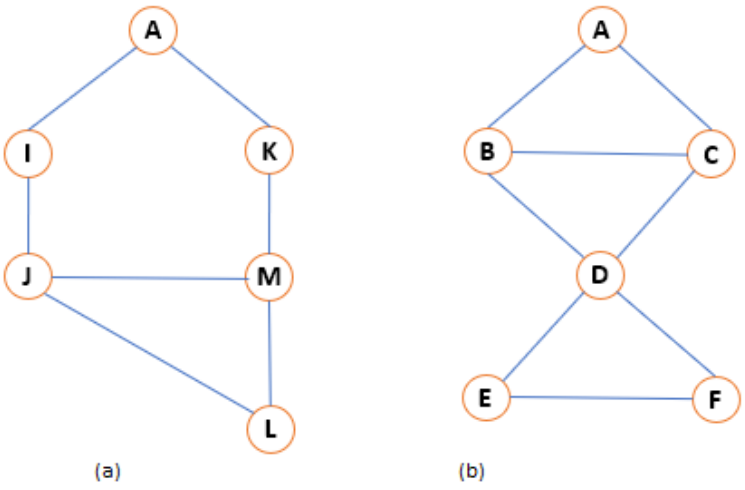


Figura 4.68 Árbol de decisión segmento 1 y segmento 2
Fuente El autor

Con el árbol de decisión se concluye que en el caso del primer segmento existen 4 rutas posibles que se ilustran a mayor detalle en la Figura 4.69 y en el segmento dos se generan 8 posibilidades de rutas como se muestra en la Figura 4.70, siendo en ambos casos el destino la MATRIZ (nodo A) y el emisor el nodo L en el segmento 1 y el nodo F en el segmento 2.

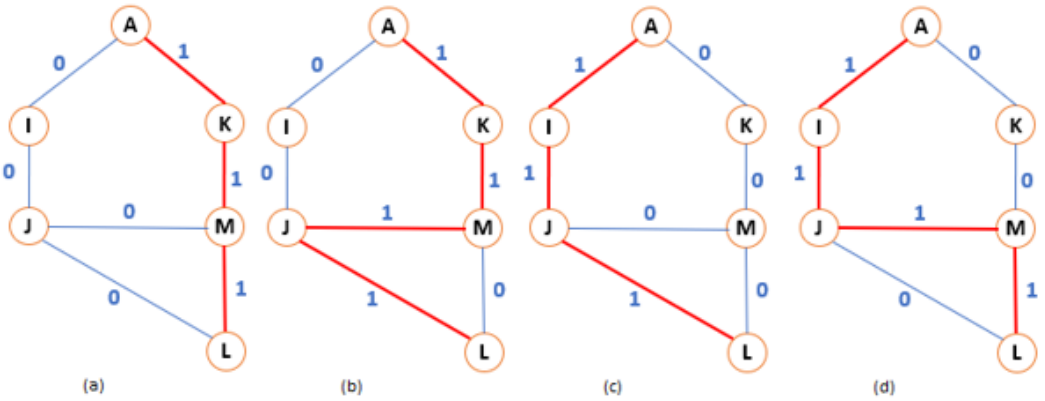


Figura 4.69 Rutas posibles para ir del nodo L hacia el A (Matriz)
Fuente El autor

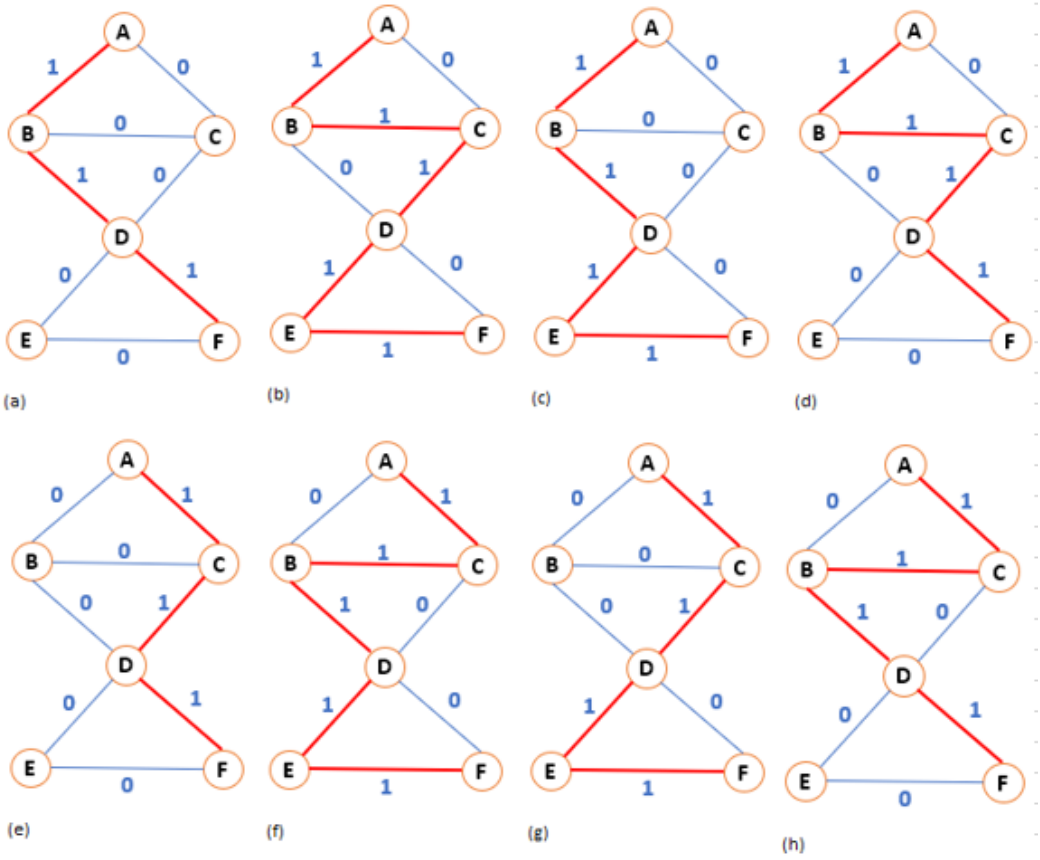


Figura 4.70 Rutas posibles para ir del nodo F hacia el A (Matriz)
Fuente El autor

Todos los enlaces pasan a tomar un valor binario donde 1 significa que el paquete pasa por ese enlace o 0 que significa que no lo hará.

En base a estos árboles de decisión se generan las matrices de todas las p-enésimas rutas posibles, tanto del segmento 1 y segmento 2, estas matrices se ilustran en la Figura 4.71

	A	B	C	D	E	F
A		P1-P4	P5-P8			
B			P2-P4,P6,P8	P1-P3,P6,P8		
C				P2,P4,P5,P7		
D					P2,P3,P5,P6,P7	P1.P4,P8
E						P2,P3,P6,P7
F						

	A	B	I	J	K	L	M
A			P3,P4		P1,P2		
B							
I				P3,P4			
J						P2,P3	P4
K					P2		P1
L							
M				P2		P4	

Figura 4.71 Matriz de posibles rutas en el segmento 1 y segmento 2, respectivamente
Fuente El autor

En el caso de los datos de las métricas de cada enlace, estos presentan en una matriz x(i,j), donde x es el valor de la métrica, i es el nodo emisor y j es el nodo receptor del enlace, en la Figura 4.72 se muestra la matriz que compone los valores de las métricas de cada enlace

	A	B	C	D	E	F
A		x8				
B			x9	x12		
C				x11		
D					x13	x15
E						x14
F						

i

j

	A	B	I	J	K	L	M
A			x1		x7		
B							
I				x2			
J						x4	x3
K							X6
L							
M				x3		x5	

i

j

Figura 4.72 Matrices de los valores de las métricas de cada enlace de la red por segmento
Fuente: El autor

Con estas dos matrices, tanto la de las rutas posibles y la de los valores de las métricas de los enlaces se procede a plantear la función objetivo, que minimiza la distancia entre el nodo origen y el nodo destino. La función objetivo que minimiza la distancia entre el nodo origen y el nodo destino para la red de estudio se definió con la fórmula (3):

$$FO = \min \sum_1^i | x_j, x_v |_1^j \tag{3}$$

$$x = (\epsilon \mathbb{R} \forall x_v) \qquad \begin{cases} x_j \notin x_v = \infty \\ x_j \in x_v = x_j \end{cases}$$

Donde;

$i = número\ de\ nodos$

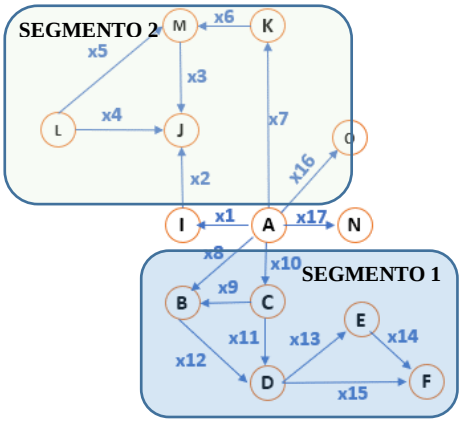
$j = número\ de\ enlace\ hasta\ el\ destino$

$v = nodo\ vecino\ de\ un\ nodo$

$x = enlace$

Para la generación de la ruta óptima de tráfico con las métricas de distancia de cada enlace determinadas en la sección anterior se procedió a validar la metodología descrita obteniendo los siguientes resultados, Figura 4.73.

VARIABLE	MÉTRICA	ENLACE
x1	13	A-I
x2	20	I-J
x3	24	J-M
x4	23	J-L
x5	16	M-L
x6	26	K-M
x7	13	A-K
x8	14	A-B
x9	27	C-B
x10	16	A-C
x11	29	C-D
x12	26	B-C
x13	28	D-E
x14	28	E-F
x15	22	D-F



SEGMENTO 1										
P1	P2	P3	P4	P5	P6	P7	P8			
62	126	96	92	67	125	101	91		FO(MIN)	62
SEGMENTO 2										
P1	P2	P3	P4							
55	86	56	73						FO(MIN)	55

Figura 4.73 Valores de la métrica de cada enlace
Fuente: El Autor

Como se observa en la Figura 4.73, la función objetivo propuesta determinó que la ruta óptima del segmento 1 y 2 es la posibilidad 1, Figura 4.74.

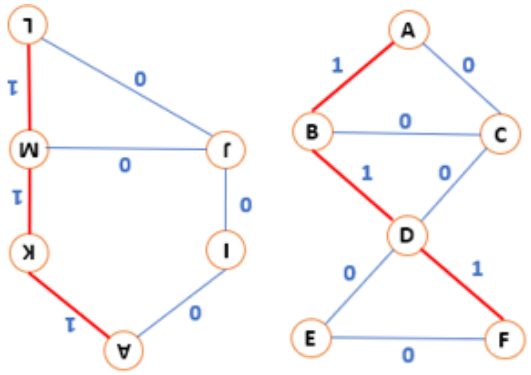


Figura 4.74 Ruta óptima determinada para la red de estudio.
Fuente: El Autor

Tomando como referencia el comportamiento de la red con el protocolo de enrutamiento RIPV2 se procedió a reconfigurar la red de forma estática en cada uno de los nodos, Figura 4.73.

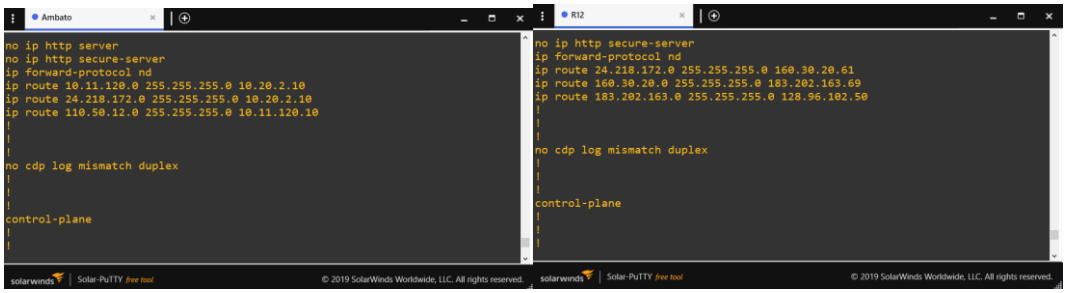


Figura 4.75 Reconfiguración de los routers con rutas estáticas.
Fuente: El Autor

Para validar el algoritmo de determinación de ruta óptima de una red virtual mediante análisis de datos y técnicas de machine learning se procedió a comparar la latencia existente en la comunicación del nodo Ambato al borde del enlace en el nodo MATRIZ y desde el nodo R12 hacia la MATRIZ.

Usando como datos de partida los tiempos de respuesta obtenidos con la configuración RIP se realizaron pruebas del algoritmo de enrutamiento óptimo de redes virtuales con análisis de datos y machine learning. Los tiempos de respuesta del protocolo RIP, y el algoritmo propuesto usando el comando traceroute se presenta en la Tabla 4.14.

.

Tipo de Enrutamiento	Origen- Destino	Tiempo de respuesta msec
RIP	Ambato - Matriz	268
	R12 -Matiz	272
Algoritmo de Enrutamiento propuesto	Ambato - Matriz	236
	R12 -Matiz	252
Porcentaje de optimización conseguida por el algoritmo [%]	Ambato - Matriz	11,94 %
	R12 -Matiz	7,23 %

Tabla 4.14 Pruebas de validación del algoritmo con el comando traceroute.
Fuente: El Autor

Para validar el algoritmo se realizó pruebas de conectividad mediante el comando ping con los tamaños de paquetes de 64, 512 y 1024 bytes obteniendo los tiempos de respuesta desde los nodos Ambato y R12 hacia la Matriz. Los resultados de los tiempos de respuesta se presentan en la Tabla 4.15.

Tipo de Enrutamiento		Tiempo de respuesta (ping)
64 bytes		
RIP	Ambato - Matriz	295,03
	R12 -Matiz	314,05
Algoritmo de Enrutamiento Planteado	Ambato - Matriz	265,45
	R12 -Matiz	298,75
Porcentaje de optimización conseguida con el algoritmo propuesto [%]	Ambato - Matriz	10%
	R12 -Matiz	4,87%
512 bytes		
RIP	Ambato - Matriz	329,98
	R12 -Matiz	334,21
Algoritmo de Enrutamiento Planteado	Ambato - Matriz	303,17
	R12 -Matiz	312,83
Porcentaje de optimización conseguida con el algoritmo propuesto [%]	Ambato - Matriz	8,12%
	R12 -Matiz	6,59%
10124 bytes		
RIP	Ambato - Matriz	378,12
	R12 -Matiz	386,98
Algoritmo de Enrutamiento Planteado	Ambato - Matriz	357,20
	R12 -Matiz	365,20
Porcentaje de optimización conseguida con el algoritmo propuesto [%]	Ambato - Matriz	5%
	R12 -Matiz	5,6%

Tabla 4.15 Pruebas de validación del algoritmo con pruebas ping.
Fuente: El Autor

Como se puede observar en la Tabla 4.15, la reducción conseguida por el uso del algoritmo se encuentra sobre un 5% de mejora en el tiempo de respuesta, sin embargo, cabe recalcar que el tiempo de respuesta de la comunicación no es motivo suficiente para validar el algoritmo. Ya que este evalúa de manera integral los diferentes parámetros de la asignación de la métrica a diferencia de los protocolos RIP y OSPF.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

La complejidad existente en la administración de las redes físicas y virtuales ha exigido el desarrollo de protocolos de enrutamiento en las redes de datos. Sin embargo, los diferentes protocolos de enrutamiento consideran ciertas métricas particulares para su funcionamiento. Esto hace que la red esté sujeta a fallas, debido a que no se incluyen variables externas como las consideradas en la ingeniería de tráfico. Dentro del estudio realizado se puede identificar varias métricas de costo del enlace las mismas que se basan en el tráfico de red, ancho de banda, costo del enlace, pérdidas de paquetes, confiabilidad del sistema, número de saltos, unidad máxima de transmisión, latencia y nivel de uso al cual se pueden sumar variables propias del tráfico de las capas 1, 2 y 3. El algoritmo propuesto integra varias de las más importantes métricas usadas por los protocolos RIP, OSPF, en base a los datos analizados con la herramienta WireShark.

El tráfico generado de una red depende de manera directa de su arquitectura, esto quiere decir que el tráfico que se genera es proporcional al número de elementos de la red (máquinas virtuales, nodos y enlaces), al tipo de comunicación (serial, ethernet, etc), al tipo de protocolo de enrutamiento (RIP, OSPF, EIGRP, etc). En el caso de estudio, la configuración del protocolo RIP, configura la ruta con el menor número de saltos desde el origen hasta el destino, sin embargo, si existen dos rutas con igual número de saltos el protocolo elige la que conoció primero, lo que puede causar la congestión en un enlace determinado de la red. El uso de herramientas especializadas para el monitoreo de tráfico como WireShark, permiten analizar el tráfico que se genera en una red, con la finalidad de obtener métricas suficientes para la configuración y administración de esta.

Los algoritmos de aprendizaje automáticos hoy en día se han vuelto una de las herramientas más útiles en cuanto a la predicción de variables estocásticas, debido a que estas usan modelos matemáticos complejos para predecir una salida en base a las variables de entrada. Las redes neuronales han sido ampliamente usadas para la

predicción de tráfico por su gran confiabilidad. Sin embargo, estudios recientes permiten analizar conceptos más específicos en el área de la inteligencia artificial como son los fenómenos de overfitting, que se presentan al entrenar una red con un porcentaje mayor al 80% de los datos de estudio. Esto usualmente genera una red neuronal artificial específica para el problema a resolver no tan adecuado en los casos que se requiera desarrollar un algoritmo que sea generalizable. Cabe mencionar que el algoritmo de redes neuronales usado se ajusta de forma adecuada para la predicción de tráfico en la red de estudio lo que permite obtener las métricas adecuadas para la determinación de la ruta óptima de tráfico.

Existen varios protocolos de enrutamiento entre los cuales destacan el de vector distancia y el de ruta de enlace, sin embargo, estos dos algoritmos presentan ciertas deficiencias ya que, estos evalúan sólo el siguiente salto para decidir la ruta más corta. En los últimos años la introducción de nuevos conceptos como los algoritmos de inteligencia artificial que se basan en procesos de optimización han sido ampliamente evaluados para determinar la ruta óptima de los enlaces. En la presente investigación se plantea una función objetivo que minimice la distancia del nodo origen al nodo destino mediante el uso de un árbol de decisiones con las p-enesimas rutas posibles de cada nodo. El algoritmo desarrollado presenta una alta tasa de asertividad en su validación por lo que se ha usado para definir las rutas óptimas de la red de estudio. Cabe mencionar que uno de los parámetros para la definición de la métrica de distancia es la predicción del tráfico realizada por la red neuronal artificial. La predicción del tráfico y el enrutamiento óptimo permitirá configurar desde el inicio una red más eficiente entregando al usuario una mejor experiencia tanto en rendimiento como accesibilidad, permitiendo una comunicación rápida, segura confiable y de alta eficiencia. El algoritmo propuesto presenta una optimización de cerca del 5% en el tiempo de respuesta lo que en comunicaciones es un gran aporte al desarrollo y validación del uso de algoritmos de inteligencia artificial en redes de datos.

5.2. Recomendaciones

La métrica o coste de un enlace debe ser determinado no solamente por el administrador de la red sino también por el gerente o personal administrativo que conozcan el enfoque de la empresa y puedan determinar la prioridad de los datos, ya sea por su contenido o por su tiempo de llegada, para alcanzar los parámetros de ingeniería de tráfico establecidos en la red.

Para implementar una red de negocios real es recomendable que la red sea en lo posible mallada, debido a que una red simple no permite tener varias rutas para llegar al mismo nodo destino y el algoritmo busca encontrar la ruta más corta para realizar la entrega de los datos a su destino, es decir debe disponer de dos o más opciones de ruta y poder compararlas y así elegir la ruta óptima.

La metodología presentada en el trabajo tiene la finalidad de ir realizando los procesos de forma sistemática los resultados de cada uno convergen y se obtiene un resultado final, esto permite ingresar los datos de tráfico y predecir su comportamiento, por lo que se podría usar y comprobar su eficiencia con la implementación de tráfico más complejo que el generado por ICMP, basta con cambiar la base de datos de entrada de tráfico y estos serán procesados, con estos datos se obtienen los valores de la métricas y la respuesta de la función objetivo debe ser implementada y validar su funcionamiento.

Como trabajo futuro se puede implementar la presente metodología en un sistema de tiempo real, es decir que la red se encuentra monitoreada todo el tiempo e ir realizando la predicción del tráfico, y poder realizar cambios oportunos como el redireccionamiento del tráfico por una ruta óptima de manera anticipada a una congestión de tráfico o fallas de la red.

REFERENCIAS

[1] M. A. A. Monroy, “Modelos de tráfico en análisis y control de redes de comunicaciones,” Ingenierıa, vol. 9, no. 1, pp. 63–87, 2004.

[2] D. Bhamare, M. Samaka, A. Erbad, R. Jain, L. Gupta, and H. A. Chan, “Óptimal virtual network function placement in multi-cloud service function chaining architecture,” Computer Communications, vol. 102, pp. 1–16, 2017.

[3] A. Gupta, M. F. Habib, U. Mandal, P. Chowdhury, M. Tornatore, and B. Mukherjee, “On service-chaining strategies using virtual network functions in operator networks,” Computer Networks, vol. 133, pp. 1–16, 2018.

[4] C. Catania and C. Garcıa Garino, “Reconocimiento de patrones en el tráfico de red basado en algoritmos genéticos,” Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 12, no. 37, 2008.

[5] J. Curcio and M. Mani, “Peer-to-peer network over a virtual private network.” Google Patents, 2015.

[6] T. Mano, T. Inoue, D. Ikarashi, K. Hamada, K. Mizutani, and O. Akashi, “Efficient virtual network optimization across multiple domains without revealing private information,” IEEE Transactions on Network and Service Management, vol. 13, no. 3, pp. 477–488, 2016.

[7] M. Sullivan and A. Heybey, “A system for managing large databases of network traffic,” in Proceedings of USENIX, 1998.

[8] T. Wood, K. Ramakrishnan, P. Shenoy, J. Van der Merwe, J. Hwang, G. Liu, and L. Chaufournier, “CloudNet: Dynamic pooling of cloud resources by live WAN migration of virtual machines,” IEEE/ACM Transactions on Networking (TON), vol. 23, no. 5, pp. 1568–1583, 2015.

[9] J. Border, D. Dillon, and P. Pardee, “Method and system for communicating over a segmented virtual private network (VPN).” Google Patents, 2015.

[10] Viavi, “The Benefits and Challenges of Virtual Networks,” Viavi Solution, 2017.

[11] X. W. Huang, G. Lochan, A. Dalal, R. Siamwalla, and J. Phillips, “Virtual network in server farm.” Google Patents, 2010.

[12] H. B. Nakil and A. Singla, “Physical path determination for virtual network packet flows.” Google Patents, 2018.

[13] C. Insfrán, D. Pinto, and B. Barán, “Diseño de topologıas virtuales en redes ópticas. un enfoque basado en colonia de hormigas,” in XXXII Latin-American Conference on Informatics, 2006, pp. 173–195.

[14] J. R. Hoy, S. R. Iyer, K. K. Kapadia, R. K. Muthukrishnan, and N. Nagaratnam, “Dynamically defined virtual private network tunnels in hybrid cloud environments.” Google Patents, 2019.

[15] R. Moreno-Vozmediano, R. S. Montero, E. Huedo, and I. M. Llorente, “Cross-site virtual network in cloud and fog computing,” IEEE Cloud Computing, vol. 4, no. 2, pp. 46–53, 2017.

[16] A. Padmanabhan, R. Zhang, P. Thakkar, T. Koponen, and M. Casado, “WAN optimizer for logical networks.” Google Patents, 2015.

[17] B. Golden, Virtualization for dummies. John Wiley & Sons, 2011.

[18] T. Bujlow, Classification and analysis of computer network traffic. 2014.

[19] O. Aouedi, K. Piamrat, S. Hamma, and J. M. Perera, “Network Traffic Analysis using Machine Learning: an unsupervised approach to understand and slice your network,” Annals of Telecommunications-annales des télécommunications, 2021.

[20] M. A. Villalba, “Predicción del tráfico de una red inalámbrica basada en redes neuronales artificiales mediante el algoritmo de Levenberg-Marquardt Ramiro Osorio D. Martha Y. Segura R. 2,” DESARROLLO E INNOVACIÓN EN INGENIERÍA, p. 27, 2019.

[21] R. Mijumbi, J.-L. Gorricho, J. Serrat, M. Claeys, F. De Turck, and S. Latré, “Design and evaluation of learning algorithms for dynamic resource management in virtual networks,” in 2014 IEEE network operations and management symposium (NOMS), 2014, pp. 1–9.

[22] K. Gogunska, C. Barakat, and G. Urvoy-Keller, “Tuning óptimal traffic measurement parameters in virtual networks with machine learning,” in 2019 IEEE 8th International Conference on Cloud Networking (CloudNet), 2019, pp. 1–3.

[23] J. Vergara-Reyes, M. C. Martinez-Ordonez, A. Ordonez, and O. M. C. Rendon, “IP traffic classification in NFV: A benchmarking of supervised Machine Learning algorithms,” in 2017 IEEE Colombian Conference on Communications and Computing (COLCOM), 2017, pp. 1–6.

[24] H. Yao, T. Mai, C. Jiang, L. Kuang, and S. Guo, “AI routers & network mind: A hybrid machine learning paradigm for packet routing,” IEEE Computational Intelligence Magazine, vol. 14, no. 4, pp. 21–30, 2019.

[25] E. Alpaydin, “Introduction to Machine Learning, 3rd Editio. ed.” The MIT Press, 2014.

[26] G. Tesauro, “Reinforcement learning in autonomic computing: A manifesto and case studies,” IEEE Internet Computing, vol. 11, no. 1, pp. 22–30, 2007.

[27] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, “Machine learning for networking: Workflow, advances and opportunities,” Ieee Network, vol. 32, no. 2, pp. 92–99, 2017.

[28] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, “State-of-the-art deep learning: Evolving machine intelligence toward tomorrow’s intelligent network traffic control systems,” IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2432–2455, 2017.

[29] J. Erman, A. Mahanti, and M. Arlitt, “Qrp05-4: Internet traffic identification using machine learning,” in IEEE Globecom 2006, 2006, pp. 1–6.

[30] P. Chemouil, P. Hui, W. Kellerer, Y. Li, R. Stadler, D. Tao, Y. Wen, and Y. Zhang, “Special issue on artificial intelligence and machine learning for networking and communications,” IEEE Journal on Selected Areas in Communications, vol. 37, no. 6, pp. 1185–1191, 2019.

[31] T. Rodrıguez and P. Jonathan, “Diseño de una red privada virtual para la optimización de las comunicaciones en la empresa comunicaciones e informática sac caso: redes de datos,” 2017.

[32] J. Kwon, D. Jung, and H. Park, “Traffic Data Classification using Machine Learning Algorithms in SDN Networks,” in 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020, pp. 1031–1033.

[33] C.-T. Yang, S.-T. Chen, J.-C. Liu, Y.-Y. Yang, K. Mitra, and R. Ranjan, “Implementation of a real-time network traffic monitoring service with network functions virtualization,” Future Generation Computer Systems, vol. 93, pp. 687–701, 2019.

[34] D. Mazón Menéndez, “Implementación de algoritmos de enrutamiento para redes inalámbricas de sensores móviles,” Universitat Politècnica de València, 2011.

[35] A. M. Ortız Torres, “Técnicas de enrutamiento inteligente para redes de sensores inalámbricas,” 2011.

[36] L. Corbalán, “Sistemas inteligentes aplicados a redes de datos,” Universidad Nacional de La Plata, 2007.

[37] E. Dıez Benito and others, “Aplicación de técnicas de Deep Learning para mejorar el enrutamiento en redes comunitarias inalámbricas basadas en OLSR,” 2020.
[38] J. A. NUÑEZ MANOHATL and others, “Medición y caracterización del jitter en tráfico de video streaming,” 2014.
[39] M. G. Bartolo, Análisis de tráfico para la red de datos de las instituciones educativas del núcleo 5 de la ciudad de Pereira. Universidad Tecnológica de Pereira. Facultad de Ingenierıas Eléctrica …, 2014.
[40] G. C. Kessler, “An overview of TCP/IP protocols and the internet,” GaryKessler website, 2007.
[41].
[42] B. A. Forouzan, TCP/IP protocol suite. McGraw-Hill Higher Education, 2002.
[43] U. Black, TCP/IP and related protocols. McGraw-Hill, Inc., 1992.
[44] S. Rendón Bernot and others, “Enrutamiento de paquetes en Redes Definidas por Software mediante Aprendizaje Automático,” Quito, 2020.
[45] J. Kurose and K. Ross, “Redes de computadores,” Madrid: Edit Pearson Addison Wesley, 2004.
[46] F. Halsall, Data communications, computer networks and open systems. Addison Wesley Longman Publishing Co., Inc., 1995.
[47] S. Rendón Bernot and others, “Enrutamiento de paquetes en Redes Definidas por Software mediante Aprendizaje Automático,” Quito, 2020.
[48] B. Fortz, J. Rexford, and M. Thorup, “Traffic engineering with traditional IP routing protocols,” IEEE communications Magazine, vol. 40, no. 10, pp. 118–124, 2002.
[49] J. A. Ulloa Márquez, “Implementación de protocolos de enrutamiento mediante un enrutador basado en software de código abierto bajo linux,” QUITO/EPN/2007, 2007.
[50] R. López Bulla, “Enrutamiento y configuración de redes,” 2018.
[51] K. EL KHADIRI, O. Labouidya, N. Elkamoun, and R. Hilal, “Comparative Study Between Dynamic IPv6 Routing Protocols of Distance Vectors and Link States,” in 2018 6th International Conference on Wireless Networks and Mobile Communications (WINCOM), 2018, pp. 1–6.
[52] W. Stallings, W. Stallings, A. Tanenbaum, K. R. Fall, and W. R. Stevens, Comunicaciones y Redes de Computadores, 6aedición. Prentice-Hall, 2000.

[53] O. Arzola Rodrıguez, “Virtualización de la red UCLV,” Universidad Central“ Marta Abreu” de Las Villas, 2011.

[54] M. D. G. Rio, Tecnologías de virtualización. 2016.

[55] K. Garcıa Pombo and Y. Fernández Romero, “Virtualización,” 2011.

[56] M. Guzek, P. Bouvry, and E.-G. Talbi, “A survey of evolutionary computation for resource management of processing in cloud computing,” IEEE Computational Intelligence Magazine, vol. 10, no. 2, pp. 53–67, 2015.

[57] D. Spiekermann and J. Keller, “Unsupervised packet-based anomaly detection in virtual networks,” Computer Networks, vol. 192, p. 108017, 2021.

[58] R. Blanco Pérez and others, “Avanzando hacia una red auto-adaptativa: simulación de redes definidas por software (SDN) mediante el simulador GNS3,” 2019.

[59] O. Aouedi, K. Piamrat, S. Hamma, and J. M. Perera, “Network Traffic Analysis using Machine Learning: an unsupervised approach to understand and slice your network,” Annals of Telecommunications-annales des télécommunications, 2021.

[60] F. Morales, M. Ruiz, L. Gifre, L. M. Contreras, V. López, and L. Velasco, “Virtual network topology adaptability based on data analytics for traffic prediction,” IEEE/OSA Journal of Optical Communications and Networking, vol. 9, no. 1, pp. A35–A45, 2017.

[61] N. Kato, Z. M. Fadlullah, B. Mao, F. Tang, O. Akashi, T. Inoue, and K. Mizutani, “The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective,” IEEE wireless communications, vol. 24, no. 3, pp. 146–153, 2016.

[62] C. E. M. Gilces and R. P. Villamar, “Aplicación de Balanceo De Carga Dinámico Para Servidores, Basada En Redes Definidas Por Software,” RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação, no. 32, pp. 67–82, 2019.

[63] E. A. Romero Rincón, E. Ramırez Garcıa, J. Pineda Rodrıguez, and D. F. Morales Yusty, “Implementación de herramientas de monitoreo de tráfico de red basado en Sofware libre en la empresa Datatech Providers,” 2016.

[64] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski, “A knowledge plane for the internet,” in Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, 2003, pp. 3–10.

[65] Y. Zhang, New advances in machine learning. BoD–Books on Demand, 2010.

[66] M. Mohammed, M. B. Khan, and E. B. M. Bashier, Machine learning: algorithms and applications. Crc Press, 2016.

[67] B. Mahesh, “Machine Learning Algorithms-A Review,” International Journal of Science and Research (IJSR).[Internet], vol. 9, pp. 381–386, 2020.

[68] S/N, Introducing Machine Learning. 2016 The MathWorks, 2016.

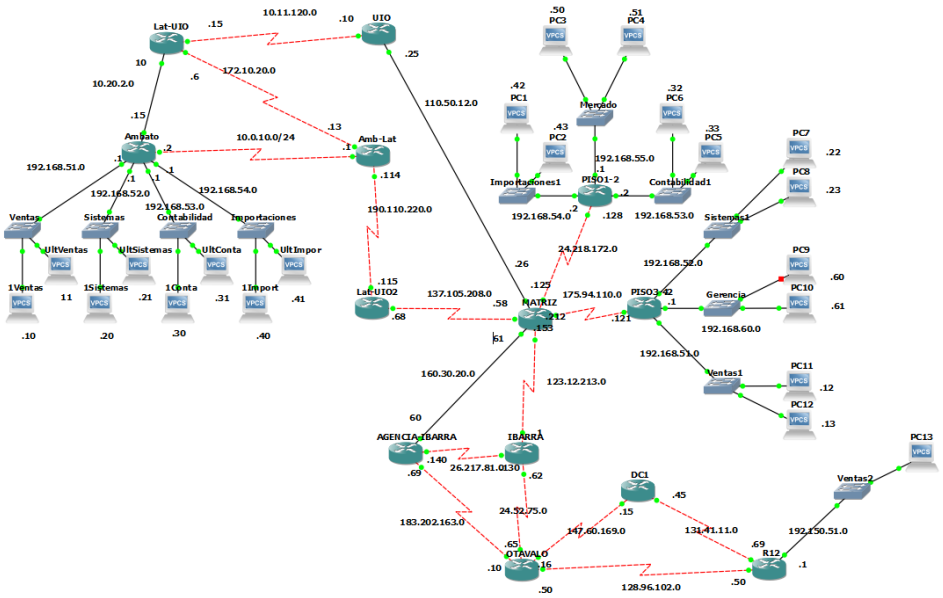
[69] J. A. G. de Mesa, J. A. R. Yanes, and J. Garvıa, “Análisis de la importancia de emplear las tecnologıas de la información y las comunicaciones en el aprendizaje: El caso de NetSim como herramienta electrónica de simulación de redes de ordenadores.”

[70] J. Han, J. Pei, and M. Kamber, Data mining: concepts and techniques. Elsevier, 2011.

[71] M. Kubat, An introduction to machine learning. Springer, 2017.

ANEXOS

Configuración de la Red de Estudio



*****///AMBATO//*****

----LEVANTAR INTERFACES-----

conf t

int s1/0

ip add 10.0.10.2 255.255.255.0

no shut

exit

int e2/0

ip add 192.168.51.1 255.255.255.0

no shut

exit

int e2/1

ip add 192.168.52.1 255.255.255.0

no shut

exit

int e2/2

ip add 192.168.53.1 255.255.255.0

no shut
exit
int e2/3
ip add 192.168.54.1 255.255.255.0
no shut
exit
int f0/0
ip add 10.20.2.15 255.255.255.0
no shut
exit
end
wr

----ENRUTAMIENTO DINAMICO-----
conf t
router rip
version 2
net 10.20.2.0
net 10.0.10.0
net 192.168.51.0
net 192.168.52.0
net 192.168.53.0
net 192.168.54.0
end
wr

----ENRUTAMIENTO ESTÁTICO-----
conf t
ip route 10.11.120.0 255.255.255.0 10.20.2.10
ip route 110.50.12.0 255.255.255.0 10.11.120.10
ip route 24.218.172.0 255.255.255.0 110.50.12.26
end
wr

*****///R12//*****

----LEVANTAR INTERFACES-----

conf t

int s1/0

ip add 131.41.11.50 255.255.255.0

no shut

exit

int s1/1

ip add 128.96.102.69 255.255.255.0

no shut

exit

end

wr

----ENRUTAMIENTO DINAMICO-----

conf t

router rip

version 2

net 128.96.102.0

net 131.41.11.0

end

wr

----ENRUTAMIENTO ESTÁTICO-----

conf t

ip route 183.202.163.0 255.255.255.0 128.96.102.50

ip route 160.30.20.0 255.255.255.0 183.202.163.69

ip route 24.218.172.0 255.255.255.0 160.30.20.61

end

wr

*****///MATRIZ//*****

----LEVANTAR INTERFACES-----

```
conf t
int f0/0
ip add 110.50.12.26 255.255.255.0
no shut
exit
int f0/1
ip add 160.30.20.61 255.255.255.0
no shut
exit
int s1/0
ip add 123.12.213.153 255.255.255.0
no shut
exit
int s1/1
ip add 175.94.110.212 255.255.255.0
no shut
exit
int s1/2
ip add 24.218.172.125 255.255.255.0
no shut
exit
int s1/3
ip add 137.105.208.58 255.255.255.0
no shut
exit
end

----ENRUTAMIENTO DINAMICO-----
conf t
router rip
version 2
network 24.0.0.0
network 110.0.0.0
network 123.0.0.0
```

network 137.105.0.0

network 160.30.0.0

network 175.94.0.0

end

wr

----ENRUTAMIENTO ESTÁTICO-----

conf t

ip route 183.202.163.0 255.255.255.0 128.96.102.50

ip route 160.30.20.0 255.255.255.0 183.202.163.69

ip route 24.218.172.0 255.255.255.0 160.30.20.61

end

wr

INFORME ANALISIS DE DATOS MATLAB

ANÁLISIS DE DATOS DE TRÁFICO

IMPORTACIÓN DE DATOS

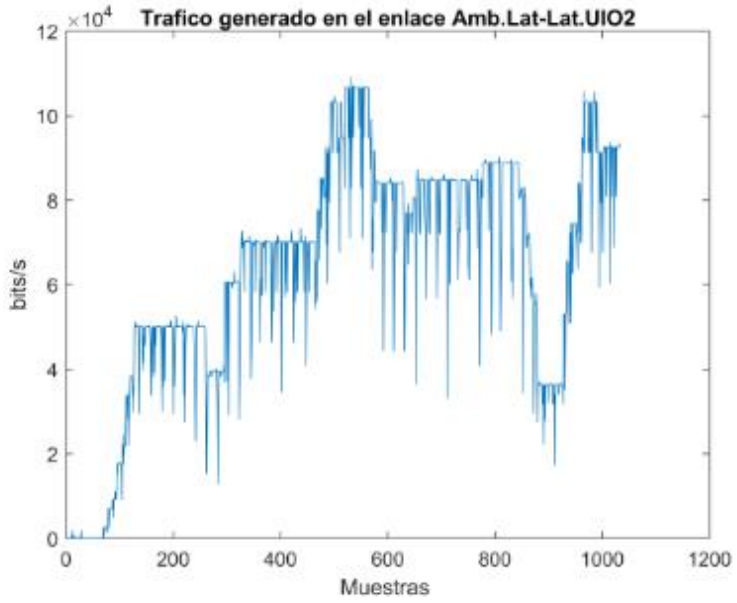
Se usa la función imporfile creada para la importación de los archivos .csv obtenidos en la herramieta WireShark al entorno de MatLab.

```
X2 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\X2.csv", [2, 1035]);
X3 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\X3.csv", [2, 1035]);
X4 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\X4.csv", [2, 1035]);
X5 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\X5.csv", [2, 1035]);
X6 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\X6.csv", [2, 1035]);
Y1 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\Y1.csv", [2, 1035]);
Y7 = importfile("C:\Users\Jeus-\OneDrive\CARLA DOCS\Datos Trafico\Y7.csv", [2, 1035]);
```

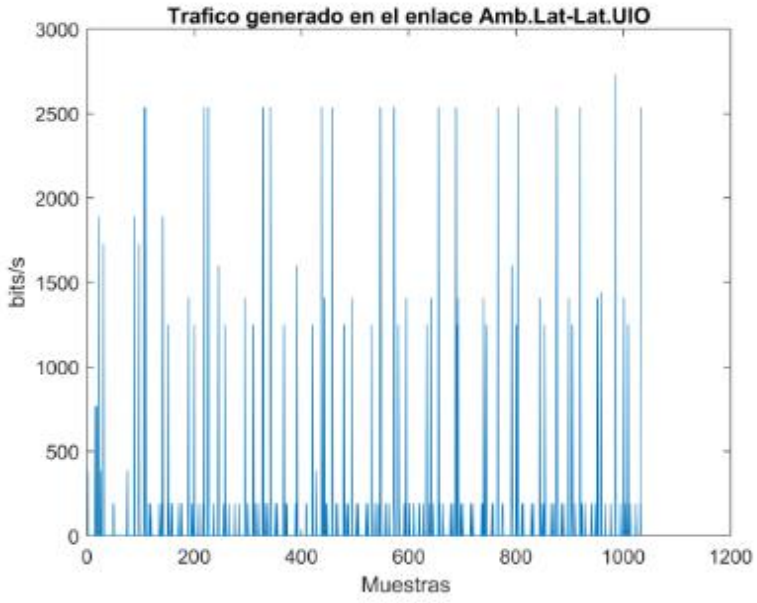
GRÁFICOS DEL TRÁFICO EN LOS ENLACES

Los datos importados se guerdan en tablas que continen vectores de las datos del tráfico obtenidos en la herramienta WireSharek. Se grafica cada uno de los datos con la finalida analizar el ruido en los datos, etapa fundamental para el análsis de datos

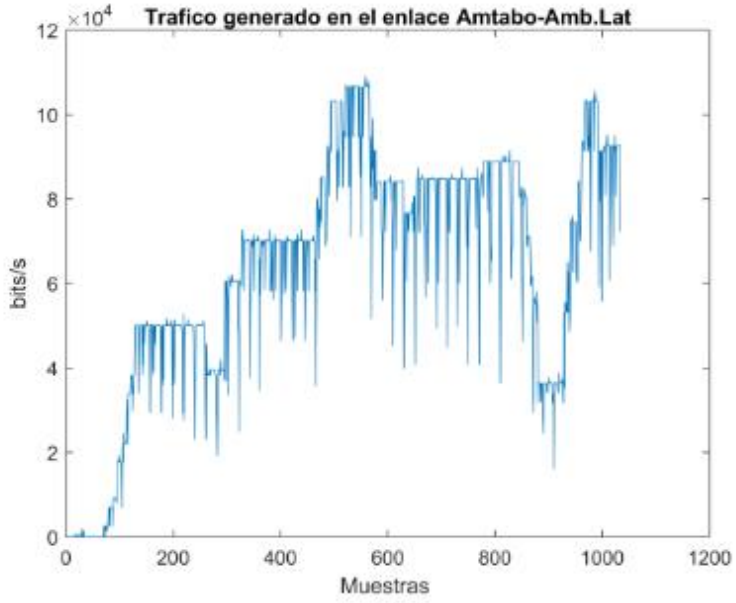
```
%% Gráfica de trafico en el enlace Amb.Lat-Lat.UIO2
plot(X2.TodosLosPaquetes)
title('Trafico generado en el enlace Amb.Lat-Lat.UIO2')
xlabel('Muestras ')
ylabel('bits/s')
```



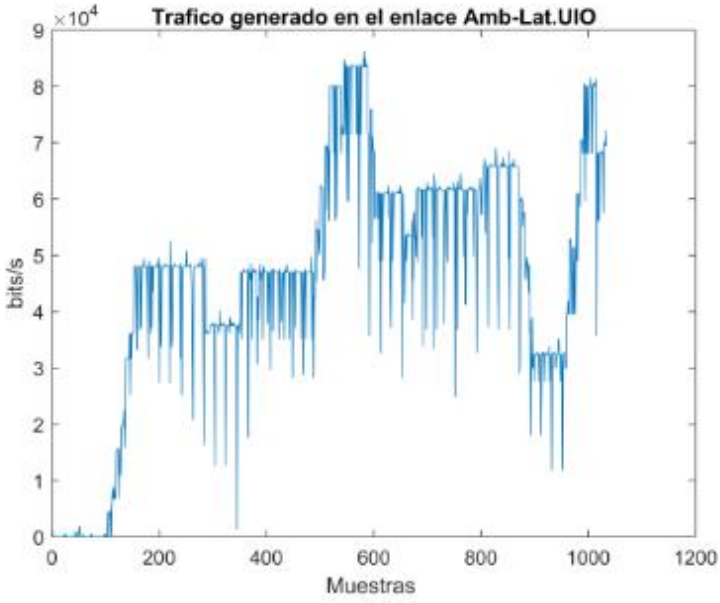
```
%% Gráfica de tráfico en el enlace Lat.UIO-Amb.Lat
plot(X3.TodosLosPaquetes)
title('Trafico generado en el enlace Amb.Lat-Lat.UIO')
xlabel('Muestras ')
ylabel('bits/s')
```



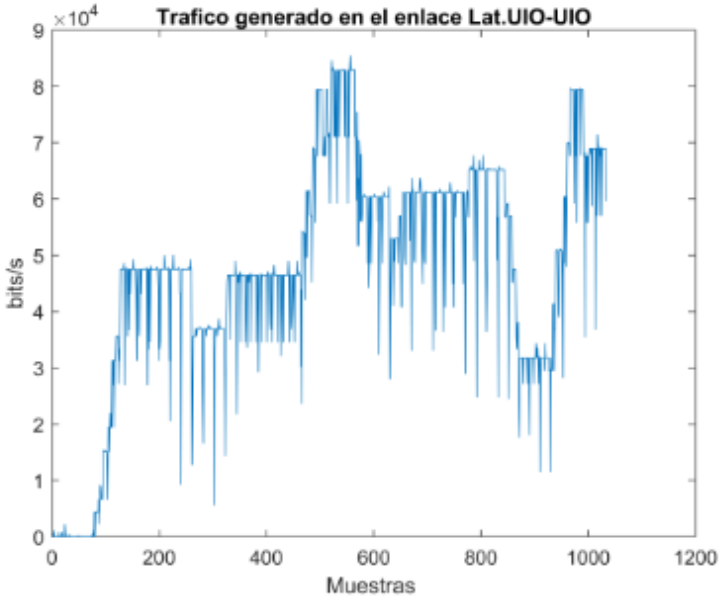

```
%% Gráfica de tráfico en el enlace Amtabo-Amb.Lat
plot(X4.TodosLosPaquetes)
title('Trafico generado en el enlace Amtabo-Amb.Lat')
xlabel('Muestras ')
ylabel('bits/s')
```



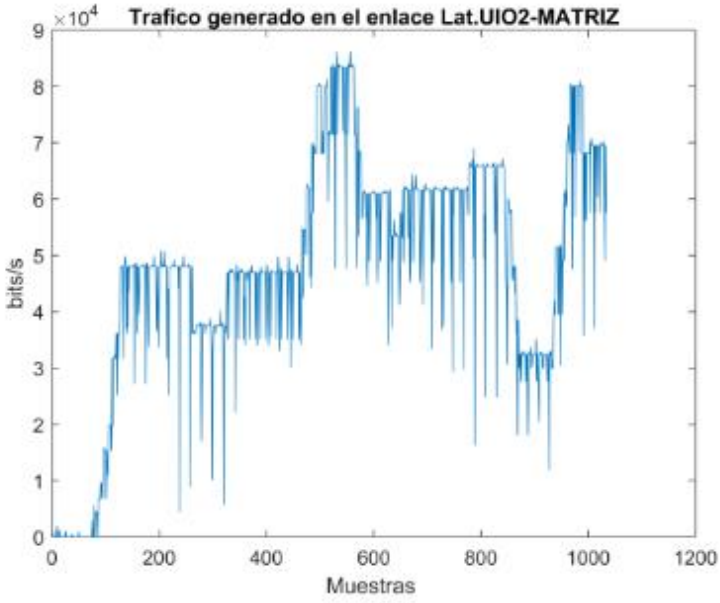

```
%% Gráfica de tráfico en el enlace Amb-Lat.UIO
plot(X5.TodosLosPaquetes)
title('Trafico generado en el enlace Amb-Lat.UIO')
xlabel('Muestras ')
ylabel('bits/s')
```



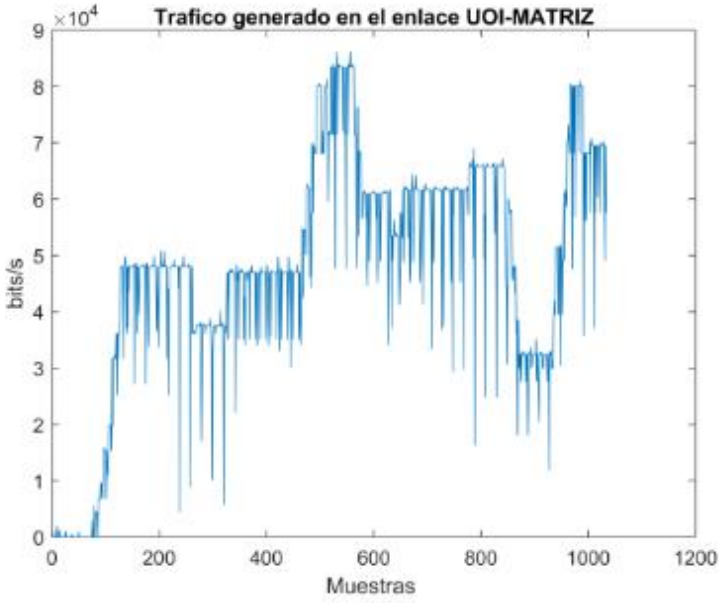

```
%% Gráfica de tráfico en el enlace Lat.UIO-UIO
plot(X6.TodosLosPaquetes)
title('Trafico generado en el enlace Lat.UIO-UIO')
xlabel('Muestras ')
ylabel('bits/s')
```




```
%% Gráfica de tráfico en el enlace Lat.UIO2-MATRIZ
plot(Y1.TodosLosPaquetes)
title('Trafico generado en el enlace Lat.UIO2-MATRIZ')
xlabel('Muestras ')
ylabel('bits/s')
```




```
%% Gráfica de tráfico en el enlace UOI-MATRIZ
plot(Y7.TodosLosPaquetes)
title('Trafico generado en el enlace UOI-MATRIZ')
xlabel('Muestras ')
ylabel('bits/s')
```



PREDICCIÓN DE TRÁFICO CON REDES NEURONALES

Para usar el toolbox de redes neuronales de matlab las variables de entrada deben ser de tipo matricial, por lo que se procede a transformar los datos importados en tablas a una matriz.

Se crea una matrix X, que contiene los datos de trafico de todos los enlaces de la red virtual de estudia generadas mediante la herramienta WireSharek. De igual forma, se crea una matriz Y con el tráfico total de los enlaces de borde es decir la Matriz.

```
X=[X2.TodosLosPaquetes,X3.TodosLosPaquetes,X4.TodosLosPaquetes,X5.TodosLosPaquetes,X6.TodosLosP

Y=[Y1.TodosLosPaquetes];
Z=[Y7.TodosLosPaquetes];
```

Generación de la funcion de red nueronal con los datos descritos en el trabajo escrito

validación de la red neuronal

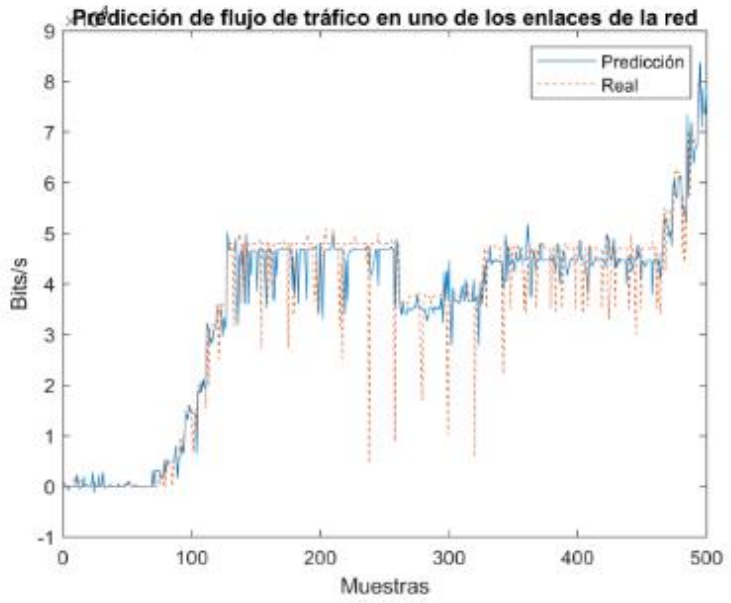
```
% Se crea un data set con los datos de flujo de trafico para verificar la
% red neuronal entranada
Xval=X(1:500,:);
Yval=Z(1:500,:);


% Funcióno de la red neuronal definida y entrenada
[Y10] = myNeuralNetworkFunction1(Xval);
```

VALIDACIÓN DEL ALGORITMO DE NNR

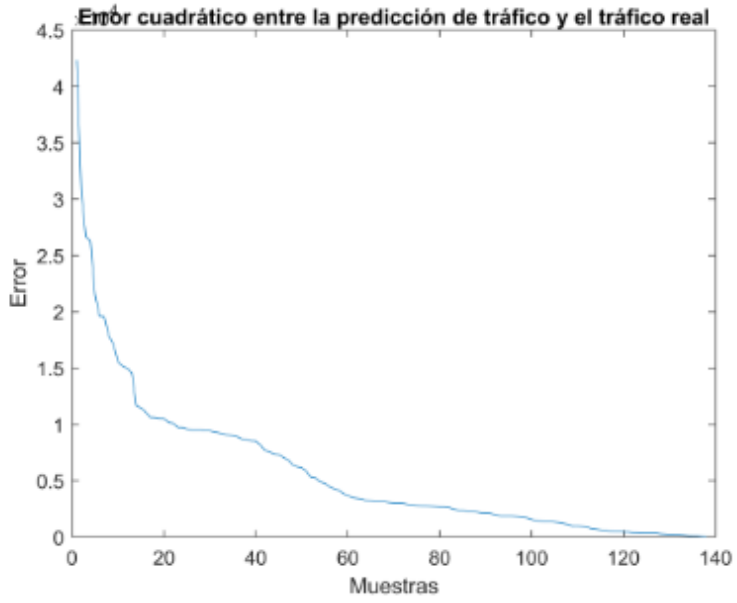
```
% Se crea un data set par la vlidacion del algoritmo
Yval=Z(1:500,:);

% Gráfico Predicción de flujo de tráfico vs trafico real
plot(Y10,'-');hold on;plot(Yval,'--');hold off;
title('Predicción de flujo de tráfico en uno de los enlaces de la red')
legend({'Predicción','Real'})
ylabel('Bits/s')
xlabel('Muestras')
```




```
% Gráfica del Error Cuadrático del volumen de la predicción de tráfico vs
% el trafico real

Error=(Y10-Yval);
Error = sort(Error,'descend');
Error = Error(Error >= 0);
plot(Error);
title('Error cuadrático entre la predicción de tráfico y el tráfico real')
xlabel('Muestras')
ylabel('Error')
```



Promedios de tráfico medio del enlace

Se obtiene el tráfico promedio generada en daca uno de los enlaces con una valores de peticiones múltiples de máxima carga con la predicción de tráfico realizado por el algoritmo de redes neuronales.

```
% Predición de tráfico de cada uno de los enlaces de la red
% Datos de entrada para la predcción
% Se otorga las variables de trafico para cada uno de los enlaces en un
% punto máximo de tráfico
X1val=X(1:200,:);
X2val=X(201:300,:);
X3val=X(301:400,:);
X4val=X(401:500,:);
X5val=X(501:600,:);
X6val=X(601:700,:);
X7val=X(701:800,:);
X8val=X(801:900,:);
X9val=X(901:1000,:);
X10val=X(350:449,:);
X11val=X(450:549,:);
X12val=X(550:649,:);
X13val=X(650:749,:);
X14val=X(750:849,:);
X15val=X(850:950,:);

% Datos de tráfico predecidos
% EL tráfico predecido en cada uno de los enlaces se nombra como Z
```

```
[Z1] = myNeuralNetworkFunction1(X1val);
[Z2] = myNeuralNetworkFunction1(X2val);
[Z3] = myNeuralNetworkFunction1(X3val);
[Z4] = myNeuralNetworkFunction1(X4val);
[Z5] = myNeuralNetworkFunction1(X5val);
[Z6] = myNeuralNetworkFunction1(X6val);
[Z7] = myNeuralNetworkFunction1(X7val);
[Z8] = myNeuralNetworkFunction1(X8val);
[Z9] = myNeuralNetworkFunction1(X9val);
[Z10] = myNeuralNetworkFunction1(X10val);
[Z11] = myNeuralNetworkFunction1(X11val);
[Z12] = myNeuralNetworkFunction1(X12val);
[Z13] = myNeuralNetworkFunction1(X13val);
[Z14] = myNeuralNetworkFunction1(X14val);
[Z15] = myNeuralNetworkFunction1(X15val);

% Tráfico promedio generado en cada uno de los enlaces con carga máxima de
% tráfico
% realizada

mean(Z1)
```

```
ans = 2.0848e+04
```

```
mean(Z2)
```

```
ans = 4.1627e+04
```

```
mean(Z3)
```

```
ans = 4.2418e+04
```

```
mean(Z4)
```

```
ans = 5.0634e+04
```

```
mean(Z5)
```

```
ans = 7.1958e+04
```

```
mean(Z6)
```

```
ans = 5.6131e+04
```

```
mean(Z7)
```

```
ans = 5.9269e+04
```

```
mean(Z8)
```

```
ans = 5.0294e+04
```

```
mean(Z9)
```

```
ans = 5.2610e+04
```

```
mean(Z10)
```

```
ans = 4.4619e+04
```

```
mean(Z11)
```

```
ans = 6.6598e+04
```

```
mean(Z12)
```

```
ans = 6.1092e+04
```

```
mean(Z13)
```

```
ans = 5.7567e+04
```

```
mean(Z14)
```

```
ans = 6.0906e+04
```

```
mean(Z15)
```

```
ans = 3.8048e+04
```

DECLARACIÓN DE FUNCIONES

Función para importar los datos de tráfico generados por la herramienta WireSharek

```
function X2 = importfile1(filename, dataLines)

%% Input handling

% If dataLines is not specified, define defaults
if nargin < 2
    dataLines = [1, Inf];
end

%% Set up the Import Options and import the data
opts = delimitedTextImportOptions("NumVariables", 2);

% Specify range and delimiter
opts.DataLines = dataLines;
opts.Delimiter = ",";

% Specify column names and types
opts.VariableNames = ["Var1", "TodosLosPaquetes"];
opts.SelectedVariableNames = "TodosLosPaquetes";
opts.VariableTypes = ["string", "double"];

% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% Specify variable properties
opts = setvaropts(opts, "Var1", "WhitespaceRule", "preserve");
```

```
opts = setvaropts(opts, "Var1", "EmptyFieldRule", "auto");

% Import the data
X2 = readtable(filename, opts);
end
```

Función del entranamiento de la red nueronal

```
function [y1] = myNeuralNetworkFunction1(x1)
%MYNEURALNETWORKFUNCTION neural network simulation function.
% [y1] = myNeuralNetworkFunction(x1) takes these arguments:
%   x = Qx5 matrix, input #1
% and returns:
%   y = Qx1 matrix, output #1
% where Q is the number of samples.

%#ok<*RPMT0>

% ===== NEURAL NETWORK CONSTANTS =====

% Input 1
x1_step1.xoffset = [0;0;0;0;0];
x1_step1.gain = [1.83311335973017e-05;0.00073313782991202 3;1.83338222352596e-05;2.320185614849
x1_step1.ymin = -1;

% Layer 1
b1 = [-0.067801183084289062819;0.41944783796894996586;-0.160538530883305 94101;0.789242501090952
IW1_1 = [0.020545365276745008315 -0.23932817399601810648 -0.4894940934762734841 0.611188240614

% Layer 2
b2 = -0.062675516817251739066;
LW2_1 = [1.332562642197080649 -0.3657297058773993359 -2.17545140609377885 1.698288498274135083

% Output 1
y1_step1.ymin = -1;
y1_step1.gain = 2.32018561484919e-05;
y1_step1.xoffset = 0;

% ===== SIMULATION ========

% Dimensions
Q = size(x1,1); % samples

% Input 1
x1 = x1';
xp1 = mapminmax_apply(x1,x1_step1);

% Layer 1
a1 = tansig_apply(repmat(b1,1,Q) + IW1_1*xp1);

% Layer 2
a2 = repmat(b2,1,Q) + LW2_1*a1;

% Output 1
```

```
y1 = mapminmax_reverse(a2,y1_step1);
y1 = y1';
end

% ===== MODULE FUNCTIONS ========

% Map Minimum and Maximum Input Processing Function
function y = mapminmax_apply(x,settings)
y = bsxfun(@minus,x,settings.xoffset);
y = bsxfun(@times,y,settings.gain);
y = bsxfun(@plus,y,settings.ymin);
end

% Sigmoid Symmetric Transfer Function
function a = tansig_apply(n,~)
a = 2 ./ (1 + exp(-2*n)) - 1;
end

% Map Minimum and Maximum Output Reverse-Processing Function
function x = mapminmax_reverse(y,settings)
x = bsxfun(@minus,y,settings.ymin);
x = bsxfun(@rdivide,x,settings.gain);
x = bsxfun(@plus,x,settings.xoffset);
end
```