Universidad de Málaga

Escuela Técnica Superior de Ingeniería de Telecomunicación
Programa de Doctorado en Ingeniería de Telecomunicación

Tesis Doctoral

Gestión avanzada de redes móviles mediante análisis de datos masivos

Autor:

Antonio Jesús García Pedrajas

Directores:

Dr. Salvador Luna Ramírez

Dr. Matías Toril Genovés

Málaga, 2024

AUTORIZACIÓN PARA LECTURA DE LA TESIS DOCTORAL

Por la presente, Dr. D. Salvador Luna Ramírez y Dr. D., Matías Toril Genovés profesores doctores del Departamento de Ingeniería de Comunicaciones de la Universidad de Málaga, certifican que el doctorando Antonio Jesús García Pedrajas ha realizado en el Departamento de Ingeniería de Comunicaciones de la Universidad de Málaga, bajo su dirección, el trabajo de investigación correspondiente a su TESIS DOCTORAL titulada

Gestión avanzada de redes móviles mediante análisis de datos masivos

Dicho trabajo ha dado lugar a las siguientes publicaciones en revistas y aportaciones a congresos que no han sido utilizadas en tesis anteriores:

1. A. García, V. Buenestado, S. Luna-Ramírez, M. Toril, J. M. Ruiz C. Gijón, M. Toril, S. Luna, M.L. Marí, "A geometric method for estimating the nominal cell range in cellular networks", Mobile Information Systems (Hindawi), vol. 2018, may. 2018.
2. A. García, M. Toril, P. Oliver, S. Luna-Ramírez, R. García, "Big Data Analytics for Automated QoE Management in Mobile Networks", IEEE Communications Magazine, vol. 57, no. 8, pp. 91-96, ago. 2019.
3. A. García, M. Toril, P. Oliver, S. Luna-Ramírez, M. Ortiz, "Automatic alarm prioritization by data mining for fault management in cellular networks", Expert Systems with Applications (Elsevier), vol. 158, 113526, nov. 2020.
4. A. García, C. Gijón, M. Toril, S. Luna-Ramírez, "Data-driven construction of user utility functions from radio connection traces in LTE", Electronics (MDPI), Special Issue on Radio Access Network Planning and Management, vol. 10, no. 7, pp. 829, mar. 2021.
5. A. García, V. Buenestado, S. Luna-Ramírez, M. Toril, A. Mendo, "Cálculo del radio de celda basado en el diagrama de Voronoi para redes móviles LTE", XXX Simposium de la Unión Científica Internacional de Radio (URSI), Pamplona (España), sep. 2015.
6. A. García, M. Toril, P. Oliver, V. Buenestado, S. Luna-Ramírez, "Estimación de umbrales de calidad de servicio para servicios móviles mediante trazas de conexión en LTE", XXXI Simposium de la Unión Científica Internacional de Radio (URSI), Madrid (España), sep. 2016.
7. A. García, C. Gijón, M. Toril, S. Luna-Ramírez, "Data-driven construction of user utility functions from connection traces in LTE", TD(20)12040 CA15104 IRACON, Louvain-la-Neuve (Bélgica), ene. 2020.
8. A. García, M. Toril, S. Luna-Ramírez, V. Buenestado, "Diagnóstico de conexiones problemáticas en redes celulares mediante herramientas de monitorización de tráfico", XXXV Simposium de la Unión Científica Internacional de Radio (URSI), Málaga (España), sep. 2020.

Por todo ello, consideran que esta Tesis es apta para su presentación al Tribunal que ha de juzgarla. Y para que conste a efectos de lo establecido, AUTORIZAN la presentación de esta Tesis en la Universidad de Málaga.

En Málaga, a 2 de Mayo de 2024.

LUNA RAMIREZ SALVADOR - 53685227F

Firmado digitalmente por LUNA RAMIREZ SALVADOR - 53685227F
Fecha: 2024.05.02 14:45:45 +02'00'

Fdo.: Dr. D. Salvador Luna Ramírez

TORIL GENOVES MATIAS - 33393202P

Firmado digitalmente por TORIL GENOVES MATIAS - 33393202P
Fecha: 2024.05.07 09:28:41 +02'00'

Fdo.: Dr. D. Matías Toril Genovés

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

D./Dña ANTONIO JESÚS GARCÍA PEDRAJAS
Estudiante del programa de doctorado INGENIERÍA DE TELECOMUNICACIÓN de la Universidad de Málaga, autor/a de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada: GESTIÓN AVANZADA DE REDES MÓVILES MEDIANTE ANÁLISIS DE DATOS MASIVOS

Realizada bajo la tutorización de SALVADOR LUNA RAMÍREZ y dirección de SALVADOR LUNA RAMÍREZ Y MATÍAS TORIL GENOVÉS (si tuviera varios directores deberá hacer constar el nombre de todos)

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 2 de MAYO de 2024

GARCIA PEDRAJAS ANTONIO JESUS - 76428178Y Firmado digitalmente por GARCIA PEDRAJAS ANTONIO JESUS - 76428178Y Fecha: 2024.05.11 11:12:30 +02'00'	LUNA RAMIREZ SALVADOR - 53685227F Firmado digitalmente por LUNA RAMIREZ SALVADOR - 53685227F Fecha: 2024.05.02 14:46:01 +02'00'
Fdo.: ANTONIO JESÚS GARCÍA PEDRAJAS Doctorando/a	Fdo.: SALVADOR LUNA RAMÍREZ Tutor/a
LUNA RAMIREZ SALVADOR - 53685227F Firmado digitalmente por LUNA RAMIREZ SALVADOR - 53685227F Fecha: 2024.05.02 14:46:10 +02'00'	TORIL GENOVES MATIAS - 33393202P Firmado digitalmente por TORIL GENOVES MATIAS - 33393202P Fecha: 2024.05.07 09:28:56 +02'00'
Fdo.: SALVADOR LUNA RAMÍREZ Y MATÍAS TORIL GENOVÉS	

Edificio Pabellón de Gobierno. Campus El Ejido. 29071
Tel.: 952 13 10 28 / 952 13 14 61 / 952 13 71 10
E-mail: doctorado@uma.es

Director/es de tesis

Edificio Pabellón de Gobierno. Campus El Ejido. 29071
Tel.: 952 13 10 28 / 952 13 14 61 / 952 13 71 10
E-mail: doctorado@uma.es

Agradecimientos

En primer lugar, quiero agradecer a mi familia por su apoyo incondicional durante todos estos años. A mis padres, no solo por haber sido fuente de inspiración, sino por todo lo que me habéis enseñado. Hoy soy lo que soy, en gran parte, por vosotros. A mi hermana, mi cuñado y mis sobrinos, por poder contar siempre con vosotros cuando lo he necesitado. Mención especial, a Macarena, mi fiel compañera. Gracias por tu comprensión, tu ayuda y tu confianza. Has sabido darme energía cuando más falta me hacía y, sin duda, una de las razones de que este trabajo se haya realizado eres tú.

También quiero agradecer a todos aquellas personas que, de un modo u otro, han estado a mi lado en este camino. A mis compañeros en la universidad, mis compañeros de trabajo y mis amigos. En especial, a Pablo, Víctor, José María, José Ángel, Alejandro, Juan y José Badía. Gracias por todo por vuestra ayuda y compañía. Me enorgullece poder consideraos, mis amigos.

Por último, agradecer especialmente a Salvador y a Matías, mis directores de tesis. Gracias por darme esta oportunidad, por todo lo que me habéis enseñado, por vuestra dedicación y esfuerzo para que hoy pueda escribir estas líneas, y por haber sido una motivación para mí. Gracias de todo corazón.

Índice general

Resumen

El crecimiento, en tamaño y complejidad, experimentado en las redes de comunicaciones móviles durante los últimos años ha evidenciado la incapacidad de los procesos actuales para una apropiada gestión de dichas redes. Además, el aumento de las expectativas de los usuarios en los actuales servicios de movilidad, ha forzado a los operadores a modernizar sus procesos de gestión, actualmente enfocados en el rendimiento de la red, para considerar la calidad de experiencia (*Quality of Experience, QoE*) y la satisfacción del usuario. Este nuevo paradigma ha popularizado el uso de técnicas de automatización para la gestión de redes, resultando en las conocidas como redes autoorganizadas (*Self Organizing Networks*, SON). Por otro lado, los últimos avances en la tecnología de la información han propiciado el desarrollo de nuevas técnicas de análisis de datos masivos (*Big Data Analytics*, BDA) permitiendo mejorar los procesos de gestión mencionados anteriormente.

Esta tesis aborda el uso de técnicas de BDA para desarrollar nuevos procesos para la gestión de la red, centrados en la satisfacción del usuario, haciendo uso de la información generada por los diferentes elementos de la red.

En primer lugar, se propone un método para ajustar los umbrales de calidad de servicio definidos en las funciones de utilidad, usadas para caracterizar la QoE de algunos de los servicios más demandados actualmente en las redes móviles. El objetivo es permitir la actualización automática de los principales modelos de QoE con funciones de utilidad. Como novedad, este modelo hace uso de técnicas de BDA para el cálculo de los nuevos umbrales mediante el uso de trazas de conexión generadas en la red, eliminando la necesidad de procesos costosos (p. ej., encuestas) o el uso de dispositivos externos a la red (p. ej., aplicaciones instaladas en terminales móviles) para evaluar la QoE.

En segundo lugar, se presenta el potencial que las aplicaciones de monitorización y análisis del tráfico de la red troncal ofrecen para la gestión de la QoE en

redes celulares comerciales. El objetivo es describir el esquema básico de este tipo de aplicaciones para la monitorización de la QoE, proponiendo una metodología genérica para validar su configuración con el fin de asegurar un correcto funcionamiento. Además, se exponen diferentes casos de uso en los que se refleja la utilidad que las técnicas de BDA pueden ofrecer en este tipo de aplicaciones en el contexto de la QoE tras su despliegue en redes móviles reales.

Por último, se propone un método automático para la identificación y priorización de alarmas, utilizadas en la gestión de fallos, en base a la necesidad de acciones adicionales para la resolución del problema subyacente en la red. El objetivo del método es simplificar el proceso de gestión de fallos reduciendo el número de alarmas que deben ser gestionadas por el personal especializado que resuelve la incidencia, y restaurar el servicio de la red tan pronto como sea posible. Para ello, el método hace uso de técnicas de BDA para predecir, de forma automática, aquellas alarmas que requieren de la generación de un ticket de incidencia para su posterior revisión.

Los métodos propuestos en esta tesis se conciben como parte de los procesos automáticos de gestión de red. Tanto para su desarrollo como para la evaluación de su desempeño, se han empleado datos de redes comerciales en las que coexisten diversos servicios y tecnologías de acceso radio.

Índice de figuras

Índice de tablas

Lista de acrónimos

AM	**A**larm **M**anagement
API	**A**pplication **P**rogramming **I**nterface
BDA	**B**ig **D**ata **A**nalytics
BSS	**B**usiness **S**upport **S**ystem
CCO	**C**overage and **C**apacitry **O**ptimization
CMIP	**C**ommon **M**anagement **I**nterface **P**rotocol
CRISP-DM	**CR**oss-**I**ndustry **S**tandard **P**rocess for **D**ata **M**ining
DASH	**D**ynamic **A**daptive **S**treaming over **H**TTP
DL	**D**own**L**ink
eNB	**e**volved **N**ode **B**
ESR	**E**xtended **S**ession **R**ecord
FM	**F**ault **M**anagement
FTP	**F**ile **T**ransfer **P**rotocol
GGSN	**G**ateway **G**PRS **S**upport **N**ode
GTP	**G**PRS **T**unneling **P**rotocol
GUI	**G**raphical **U**ser **I**nterface
HTTP	**H**yper**T**ext **T**ransfer **P**rotocol
IBM	**I**nternational **B**usiness **M**achines
IMS	**I**P **M**ultimedia **S**ubsystem
IMSI	**I**nternational **M**obile **S**ubscriber **I**dentity
IP	**I**nternet **P**rotocol
JSON	**J**ava**S**cript **O**bject **N**otation
KDD	**K**nowledge **D**iscovery in **D**atabases
KPI	**K**ey **P**erformance **I**ndicator

LSTM	**L**ong-**S**hort **T**erm **M**emory
LTE	**L**ong **T**erm **E**volution
MAC	**M**edium **A**ccess **C**ontrol
MIB	**M**anager **I**nformation **B**ase
ML	**M**achine **L**earning
MME	**M**obility **M**anagement **E**ntity
MOS	**M**ean **O**pinion **S**core
MRO	**M**obility **R**obustness **O**ptimization
NMS	**N**etwork **M**anagement **S**ystem
NOC	**N**etwork **O**peration **C**enter
OSS	**O**perations **S**upport **S**ystem
PDCP	**P**acket **D**ata **C**ontrol **P**rotocol
PEP	**P**erformance **E**nhancing **P**roxy
PM	**P**erformance **M**anagement
QCI	**Q**oS **C**lass **I**dentifier
QoE	**Q**uality **o**f **E**xperience
QoS	**Q**uality **o**f **S**ervice
QUIC	**Q**uick **U**DP **I**nternet **C**onnection
R-KPI	**R**esource-**K**ey **P**erformance **I**ndicator
RB	**R**ed **B**ayesiana
RF	**R**andom **F**orest
RLC	**R**adio **L**ink **C**ontrol
RMON	**R**emote **N**etwork **MON**itoring
RNA	**R**ed **N**euronal **A**rtificial
RNN	**R**ecurrent **N**eural **N**etwork
RRC	**R**adio **R**esource **C**ontrol
RTP	**R**eal **T**ime **P**rotocol
RTT	**R**ound-**T**rip **T**ime
S-KPI	**S**ervice-**K**ey **P**erformance **I**ndicator
SGSN	**S**erving **G**PRS **S**upport **N**ode
SGW	**S**erving **G**ate**W**ay

SNMP	**S**imple **N**etwork **M**anagement **P**rotocol
SON	**S**elf-**O**rganizing **N**etwork
SVM	**S**upport **V**ector **M**achine
TCP	**T**ransmission **C**ontrol **P**rotocol
TM	**T**icket **M**anagement
TMA	**T**raffic **M**onitoring and **A**nalysis
UDP	**U**ser **D**atagram **P**rotocol
UL	**U**p**L**ink
VoIP	**V**oice **o**ver **I**nternet **P**rotocol
VoLTE	**V**oice **o**ver **L**TE
WOM	**W**ork **O**rder **M**anagement

Capítulo 1

Introducción

En este capítulo inicial se introduce el trabajo realizado en esta tesis. En primer lugar, se explica la motivación que da lugar a la tesis. A continuación, se describen los objetivos de esta y, posteriormente, la metodología de trabajo empleada. Finalmente, se presenta la estructura de este documento.

1.1. Motivación

En los últimos años, el número de usuarios y servicios en las redes celulares se ha incrementado drásticamente. Se espera que, para el inicio de 2024, el 70 % de la población tenga algún tipo de suscripción a un servicio de movilidad y el número de dispositivos móviles alcance los 13,1 billones, de los cuales 1,4 billones tendrán soporte para la tecnología de 5G [1]. Por otro lado, el éxito de los nuevos teléfonos inteligentes y *tablets* ha hecho que las expectativas de los usuarios sean ahora un factor muy importante que considerar en los procesos de gestión de red [2]. Para satisfacer las expectativas de los usuarios, los operadores se han visto forzados a actualizar el modo en que gestionan sus redes. Dicha gestión ha pasado de estar enfocada en el rendimiento de la red y la calidad de servicio (*Quality of Service*, QoS), a una aproximación más moderna centrada en la satisfacción del usuario y la calidad de experiencia (*Quality of Experience*, QoE) [3].

Al mismo tiempo, el despliegue de las nuevas redes 5G ha añadido importantes cambios en la red con la aparición de la virtualización de la red y la comunicación máquina-a-máquina, haciendo que el proceso de gestión del tráfico sea todo un reto

para los operadores [4]. El precio a pagar es un incremento de la heterogeneidad de la red debido a la coexistencia de múltiples tecnologías de acceso radio, estaciones base de muy diferentes rangos y disparidad de dispositivos [5]. Tal diversidad ha generado un aumento considerable en la complejidad de las redes celulares, creando nuevos problemas en los procesos de gestión.

Todos estos cambios han provocado que los operadores demanden nuevas herramientas que faciliten la gestión de sus redes. Esto ha hecho que se popularice el uso de técnicas de automatización para gestionar las redes de comunicaciones móviles, dando lugar a las conocidas como redes autoorganizadas (*Self-Organizing Networks*, SON). Dichas redes proporcionan inteligencia y adaptabilidad en la red simplificando los procesos de configuración y optimización [6].

En paralelo, gracias a los continuos avances en las tecnologías de la información, ahora es posible analizar grandes volúmenes de información usando técnicas de análisis de datos masivos (*Big Data Analytics*, BDA). Esto ha permitido a los operadores hacer uso de toda la información generada en su red (p. ej., trazas de conexión o tráfico a nivel de paquete), mejorando el tiempo de reacción de los sistemas de gestión, permitiendo acciones en tiempo real y mejorando la monitorización, el control y la optimización de la QoE [7].

Por eficiencia, las herramientas SON suelen enfocarse en aquellas áreas de la gestión que tienen una mayor repercusión sobre el rendimiento de la red. Entre ellas destacan, la gestión del rendimiento (*Performance Management*, PM) y la gestión de fallos (*Fault Management*, FM) [8]. Ambas áreas tienen un impacto directo en la satisfacción final del usuario. Una gestión efectiva de la QoE es un factor diferencial en un mercado donde tanto las redes como los servicios son tan similares entre operadores. Del mismo modo, una resolución de problemas en la red rápida y efectiva evita quejas e insatisfacción del usuario final. Por tanto, ambos puntos son claves en los futuros desarrollos de herramientas de monitorización y optimización de red.

Una de las principales tareas de la gestión de la QoE es la definición de indicadores que aseguren una apropiada caracterización del rendimiento del servicio. En dicha caracterización se intentan reflejar los diferentes factores que influyen en la calidad percibida por el usuario con el objetivo de describir la relación existente entre variables medibles y calidad de experiencia, dando como resultado diferentes modelos de QoE [9]. Estos modelos a menudo consisten en funciones de utilidad

analíticas que relacionan de forma directa los valores de los indicadores de la QoS ofrecida por la red y la opinión del usuario cuantificada con algún tipo de métrica (p. ej., nota media de opinión) [10]. En su forma más simple, esta relación entre la QoS y la QoE se define mediante una función logarítmica [11] o exponencial [12]. En la literatura, se han definido modelos de QoE para algunos de los principales servicios en movilidad, tales como voz sobre el protocolo de Internet (*Voice over Internet Protocol*, VoIP) [13], descarga progresiva de vídeo (streaming) [14], protocolo de transferencia de ficheros (*File Transfer Protocol*, FTP) o navegación web [15]. Sin embargo, estos modelos pueden quedar obsoletos con la evolución continua de las redes móviles, los terminales y las expectativas de los usuarios. Por tanto, se requiere una actualización de las funciones de utilidad que definen los distintos modelos. Tradicionalmente, este ajuste se ha realizado mediante costosos ensayos en entornos de laboratorio, lo que lo descarta para el desarrollo de procesos automáticos de optimización [16]. Por tanto, es necesario encontrar métodos automáticos en la red real que permitan actualizar los modelos para una gestión apropiada de la QoE.

En los últimos años, el uso de herramientas de monitorización y análisis del tráfico (*Traffic Monitoring and Analysis*, TMA) ha llegado a ser clave en la gestión del rendimiento de la red [17]. Estas soluciones capturan y analizan el tráfico generado en la red permitiendo monitorizar, de forma pasiva, todo el tráfico con gran detalle. Esto se consigue mediante sondas que capturan el tráfico de diferentes capas de protocolo. Posteriormente, puede realizarse un análisis detallado del tráfico capturado, a nivel de paquete, con el fin de obtener indicadores específicos para cada servicio [18]. Estos indicadores pueden utilizarse para desarrollar modelos de QoE más realistas permitiendo una gestión de la QoE más precisa. Sin embargo, el despliegue y la configuración de este tipo de aplicaciones no es trivial, y no existe una metodología genérica de validación que permita asegurar su correcto funcionamiento tras despliegue. Además, el uso de este tipo de aplicaciones no está muy extendido debido a su alto coste y complejidad, por lo que aún no se han desarrollado muchos modelos de optimización que permitan corroborar sus beneficios en el proceso de gestión de la QoE [18].

Al igual que la gestión del rendimiento, la gestión de fallos es otra de las áreas de mayor impacto y con mayor posibilidad de optimización de la gestión de red con el fin de mejorar la calidad experimentada por el usuario. Los sistemas utilizados para los procesos de FM son cruciales para proporcionar información de

valor con el fin de minimizar las pérdidas físicas y económicas ante situaciones anormales en el funcionamiento de la red [19]. Sin embargo, se ha evidenciado que los sistemas tradicionales están lejos del rendimiento esperado en las actuales redes móviles [20]. La principal razón es la gran cantidad de alarmas que generan los diferentes elementos de la red, y que deben analizarse por el equipo humano involucrado en los procesos de FM para resolver el fallo [21]. A lo largo de los años, se han desarrollado diferentes métodos para reducir el número de alarmas consideradas [22]. Sin embargo, aún no se ha desarrollado un método que permita identificar aquellas alarmas más prioritarias que terminan requiriendo una acción de corrección llevada a cabo por personal especializado para resolver el fallo en la red. Por otro lado, métodos tradicionales para reducir el número de alarmas utilizan modelos de clasificación basados en simples reglas heurísticas [23] o algoritmos de aprendizaje automáticos más complejos [24]. Sin embargo, elegir el mejor algoritmo es aún un reto ya que no existe un clasificador que pueda resolver cualquier problema [25].

1.2. Objetivos de la tesis

El objetivo general de esta tesis es demostrar cómo el uso de las técnicas de BDA favorece el desarrollo de métodos y modelos automáticos que puedan ser usados para optimizar los procesos de gestión de redes móviles, enfocados en mejorar la calidad de experiencia y la satisfacción final del usuario.

Para alcanzar este fin, se plantean los siguientes objetivos:

O1. Identificar los indicadores de red que mayor impacto tienen en la QoE de los principales servicios en movilidad.

O2. Desarrollar un método heurístico para el ajuste automático de los parámetros de los modelos de QoE basados en funciones de utilidad, mediante la modificación de sus umbrales de calidad de servicio.

O3. Desarrollar una metodología de validación de herramientas de TMA para la gestión avanzada de la QoE.

O4. Identificar diferentes casos de uso para mostrar el potencial de las aplicaciones TMA en los procesos de optimización de la QoE.

O5. Desarrollar un modelo automático para la priorización de alarmas generadas en la red reduciendo los tiempos de respuesta para aumentar la satisfacción de usuario.

Las principales contribuciones de esta tesis son:

1. Un método automático para el ajuste de los umbrales de calidad de servicio en los actuales modelos analíticos de QoE analizando las trazas de conexión generadas en una red LTE (*Long Term Evolution*). Al ejecutarse de forma automática, pueden detectarse cambios en las expectativas de los usuarios o tras el despliegue de nuevos servicios en la red para actualizar los valores de dichos umbrales, eliminando la necesidad de realizar costosas encuestas a los usuarios. Gracias al uso de las trazas de conexión, el modelo considera una gran variedad de sistemas y factores externos que no pueden ser simulados en entornos de laboratorio.

2. Una metodología para validar una aplicación TMA enfocada a la gestión de la QoE mediante el uso de análisis de datos masivos. Esta validación asegura un funcionamiento adecuado de la aplicación tras su despliegue en redes celulares comerciales. Tras su validación, la aplicación se utiliza para acometer diferentes casos de uso aplicando distintas técnicas de BDA para mostrar su potencial en procesos de gestión y optimización de la QoE.

3. Un modelo para identificar y priorizar alarmas generadas por los diferentes elementos de una red móvil en base a la necesidad de acciones por parte del personal especializado. Durante el desarrollo del modelo se realiza una comparación de rendimiento de diferentes algoritmos aprendizaje automático haciendo uso de un conjunto de datos real, algo poco común por motivos de privacidad de los datos. Esta comparación puede ayudar a la construcción de nuevos métodos usando el clasificador o combinación de clasificadores que mejor se ajuste al problema de reducción de alarmas.

Los modelos propuestos hacen uso de diversas técnicas de BDA. Por un lado, el modelo de ajuste de los umbrales de calidad de servicio utiliza un algoritmo de aprendizaje no supervisado para clasificar los principales servicios en la red. Por otro lado, el conjunto de datos obtenidos por una aplicación de TMA se emplea para generar información de valor para los operadores mediante técnicas avanzadas

de análisis y predicción. En este caso, se propone un modelo para la caracterización del tiempo de inicial de buffer del servicio de vídeo, haciendo uso de un algoritmo de regresión polinómico multivariable. También, se propone un modelo de predicción basado en series temporales, haciendo uso de redes neuronales. Del mismo modo, se utilizan técnicas de análisis y exploración de datos para la monitorización y visualización de la calidad de experiencia tanto a nivel de celda como de conexión. Por último, el modelo de priorización de alarmas se basa en diferentes algoritmos de aprendizaje supervisado y métodos de ensamble para la obtención del modelo final.

Los beneficios previstos de los métodos desarrollados en esta tesis son:

- Desde el punto de vista del abonado, una mayor satisfacción al percibir una mejor calidad de experiencia en los servicios consumidos. Además, los problemas generados en la red tendrán un menor impacto en el cliente final al reducirse los tiempos de resolución.
- Desde el punto de vista de los operadores, mejora en las prestaciones de la red, con procesos automáticos que permitan ofrecer un servicio más enfocado en el usuario final. Al mismo tiempo, una reducción de los costes de inversión al disminuir el uso de recursos tanto humanos como de equipos para cursar el tráfico necesario en los procesos de gestión de sus redes al optimizar el uso de los recursos disponibles.

1.3. Metodología de trabajo

En base a los objetivos propuestos, se establece el siguiente plan de trabajo:

1. *Selección del problema y revisión de la literatura.* Primero, se identifican las áreas funcionales de la gestión de red que tienen una importancia específica en la QoE. Posteriormente, se realiza una revisión exhaustiva del estado de la técnica de los principales temas dentro del alcance de esta tesis.
2. *Formulación del problema y propuestas.* Una vez se identifican las necesidades, se formulan los principales problemas a resolver y se proponen nuevas soluciones. La atención se centra en los procesos de monitorización y optimización de red con criterios de QoE, y el proceso de gestión de fallos.

3. *Obtención de datos, preprocesado y análisis.* Los diferentes conjuntos de datos utilizados para O1, O2, O3, O4 y O5 se obtienen de redes celulares comerciales. El operador de red es responsable de recopilar los datos desde el sistema de soporte y el centro de operaciones, y proporcionarlos. Una vez los datos están disponibles, se exportan a un formato apropiado para su consumo usando herramientas proporcionadas por el operador. Posteriormente, los datos se preprocesan (p. ej., las trazas de conexión usadas para O1 y O2 se decodifican y sincronizan, los reportes generados por la aplicación de monitorización del tráfico se procesan para utilizarse en O2 y O3, etc.).

4. *Revisión de la funcionalidad ofrecida por aplicaciones de monitorización y análisis del tráfico.* Estudio del funcionamiento básico de una aplicación TMA desplegada en una red móvil real.

5. *Desarrollo y validación del modelo de ajuste de las funciones de utilidad usadas para el modelado de la QoE a nivel de servicio.* Para la evaluación del modelo, se utilizan datos obtenidos de una red LTE real.

6. *Desarrollo y validación de diferentes metodologías que evalúan los beneficios del uso de aplicaciones TMA en los procesos de gestión de la QoE.* Para el desarrollo y la validación de estos modelos, se utilizan datos obtenidos por una aplicación TMA desplegada en la red troncal de una red LTE real.

7. *Desarrollo y validación del modelo de priorización de alarmas en base a la necesidad de creación de tickets de incidencia.* Durante las pruebas, se utilizan datos de alarmas y tickets recopilados por un centro de operaciones de red real.

8. *Análisis del rendimiento.* Todos los modelos y métodos desarrollados se validan con datos obtenidos de redes comerciales, asegurando una evaluación realista de los resultados. Se utilizan diferentes herramientas para cada objetivo, concretamente Matlab (O1, O2, O4), SPSS Modeler (O5) y Python (O3 y O4). El uso de diferentes herramientas y plataformas permite identificar los principales beneficios e inconvenientes para proporcionar recomendaciones a los operadores de red.

1.4. Estructura de la memoria

Tras este primer capítulo, en esta memoria se distinguen dos partes claramente diferenciadas, dedicadas al uso de técnicas de BDA para resolver dos problemas principales en los procesos de gestión de la red. Los Capítulos 3 y 4 se centran en el proceso de caracterización, monitorización y optimización de la QoE. Por otro lado, el Capítulo 5 trata el proceso de gestión de fallos en la red.

En el Capítulo 2 se presentan los conceptos básicos necesarios para contextualizar este trabajo. En una primera sección, se introducen las técnicas de BDA. A continuación, se define el concepto de red autoorganizada y sus principales categorías. Al mismo tiempo, se repasan brevemente los procesos de gestión de red, listando sus principales áreas funcionales, y como la gestión de la QoE ha adquirido una gran importancia tanto en la gestión del rendimiento como en la gestión de fallos. Por último, se profundiza en la gestión de fallos describiendo los principales componentes y roles en esta área.

En el Capítulo 3 se presenta un método de ajuste automático de funciones de utilidad en modelos de QoE. El capítulo comienza con una revisión del estado de la técnica sobre los nuevos procesos de gestión de red enfocados en la QoE y la satisfacción de usuario, mostrando la necesidad de métodos automáticos para actualizar dichos modelos. Seguidamente, se formula el problema de la caracterización de la QoE mediante el uso de funciones de utilidad. A continuación, se presenta el método automático para la actualización de los umbrales de calidad de servicio propuesto en esta tesis, seguido de las pruebas realizadas para su validación. Por último, se presentan las conclusiones obtenidas de los resultados.

En el Capítulo 4 se revisa el uso de aplicaciones de monitorización y análisis del tráfico para la gestión de la QoE. El capítulo comienza describiendo el esquema básico de una aplicación TMA para la monitorización de la QoE en una red móvil. Seguidamente, se detalla una metodología genérica para la validación de dicha aplicación tras su despliegue. A continuación, se presentan diferentes casos de uso que permiten reflejar el potencial de estas aplicaciones en la gestión de la QoE. Cada caso de uso se describe brevemente, indicando su alcance y mostrando los resultados obtenidos tras las pruebas. Para validar los diferentes casos de uso, se emplean datos generados por una aplicación TMA en una red celular real. Por último, se presentan las conclusiones obtenidas del proceso de revisión de la aplicación y de los casos de uso analizados.

En el Capítulo 5 se presenta un modelo automático para identificar y priorizar las alarmas generadas por los elementos de la red. El capítulo comienza con una revisión del estado de la técnica sobre el proceso de gestión de fallos en redes celulares, resaltando el problema generado por la gran cantidad de alarmas a ser analizadas por el equipo especializado en el proceso. Seguidamente, se formula el problema, definiendo las diferentes etapas del proceso de gestión de fallos e identificando la posible optimización que permita reducir los tiempos de resolución. A continuación, se presenta el modelo de priorización de alarmas propuesto en esta tesis, y las pruebas realizadas para su validación. Por último, se presentan las conclusiones obtenidas de los resultados.

En el Capítulo 6 se presentan las conclusiones generales y líneas futuras de continuación del trabajo realizado en esta tesis. Además, se presentan las publicaciones que avalan este trabajo y se describen las principales aportaciones de la tesis.

Capítulo 2

Conceptos básicos

En este capítulo se describen algunos de los conceptos básicos necesarios para contextualizar esta tesis. Para ello, la Sección 2.1 introduce el uso de técnicas de análisis de datos masivos en el contexto de las telecomunicaciones, y se explican algunas de las metodologías más comunes para el desarrollo de proyectos de este tipo. A continuación, en la Sección 2.2, se presentan las redes autoorganizadas, definiendo sus categorías principales. Finalmente, la Sección 2.3 repasa el proceso de gestión de redes celulares, definiendo primero las diferentes áreas funcionales que engloban el proceso de gestión, posteriormente, explica el proceso de gestión enfocado en la calidad de experiencia en redes celulares, detallando los principales pasos de dicho proceso para, finalmente, detallar los principales componentes y roles que componen la gestión de fallos en estas redes.

2.1. Técnicas de análisis de datos masivos

En la actual era de la digitalización, en la que se ha producido un desarrollo exponencial en la tecnología de la información y las comunicaciones, se ha observado un gran crecimiento en la cantidad de datos generados, transmitidos y almacenados. Este cambio no solo se ha visto reflejado en el volumen de datos, sino en su complejidad y frecuencia de actualización, poniendo de manifiesto la incapacidad de las herramientas tradicionales para gestionarlos en este nuevo escenario. Por tanto, en los últimos años, los esfuerzos se han centrado en desarrollar nuevas herramientas y técnicas que puedan hacer frente a este reto [26].

El termino dato masivo o, más comúnmente conocido como *big data*, se usa en una gran variedad de campos. *Big data* hace referencia a datos que muestran las siguientes características [27]:

- *Volumen.* Hace referencia a la cantidad de datos recopilados, analizados y visualizados. En algunos casos, este volumen puede ser de decenas de terabytes de datos. En otros casos, incluso cientos de petabytes.
- *Velocidad.* Indica la velocidad con la que los datos son consumidos y procesados. Esta velocidad dependerá del contexto y las necesidades, siendo el caso extremo la transmisión de datos y procesamiento en tiempo real.
- *Variedad.* Hace referencia, precisamente, a la variedad en el tipo de datos disponibles. Tradicionalmente, las fuentes de datos solían ser únicas y estructuradas. Cada vez con más frecuencia, se reciben nuevos tipos de datos no estructurados. La naturaleza de estos datos requiere de acciones adicionales para procesarlos y obtener la información esperada.

Con el crecimiento reciente del uso de *big data*, no solo las tecnologías y herramientas se han visto obligadas a evolucionar, sino los procesos y técnicas para analizar y obtener valor de la gran cantidad de datos disponibles. En este contexto, surgen las técnicas de *Big Data Analytics* (BDA) [28]. BDA describe los diferentes procesos que permiten extraer información de valor de grandes volúmenes de datos (p. ej., mediante la identificación de patrones, correlaciones, predicciones ...). BDA hace referencia a todo el proceso de recolección, procesamiento, limpieza y análisis de grandes volúmenes de datos. Cómo se ha mencionado anteriormente, la capacidad de analizar grandes volúmenes de datos, en múltiples formatos y de diferentes fuentes en un tiempo reducido, aporta grandes beneficios, tales como, reducción de costes generando negocio de forma más eficiente, evolución del producto con un enfoque más centrado en las necesidades del cliente, o menores tiempos de reacción frente a nuevas oportunidades de mercado [29].

El uso de técnicas de BDA ha adquirido una importancia capital en la industria de las telecomunicaciones. Como ya se ha mencionado anteriormente, en los últimos años las redes de telecomunicación han experimentado un crecimiento exponencial tanto en el número de abonados como de servicios disponibles. Esto, sumado a la cada vez mayor complejidad de las redes por la coexistencia de múltiples tecnologías, ha hecho que los operadores centren sus esfuerzos en desarrollar

nuevas técnicas para, no solo facilitar la gestión, sino obtener valor de la cantidad de datos generados en sus redes. En este contexto, las técnicas de BDA encajan a la perfección en las demandas de los operadores, pues permiten gestionar el gran volumen de datos disponible en las redes (es decir, tanto información específica de la red como información relacionada con el abonado), y desarrollar modelos analíticos que aporten grandes beneficios (tanto económicos como operacionales).

Algunos ejemplos comunes de proyectos de BDA aplicables a la industria de las telecomunicaciones son [30]:

1. *Predicción de pérdida de clientes*: BDA puede utilizarse para identificar posibles patrones en la baja de suscripciones y, por tanto, predecir la pérdida de clientes. La retención de abonados es un factor clave para los operadores debido al alto coste que tiene la adquisición de nuevos clientes.

2. *Ofertas personalizadas*: Gracias a la información del uso del tráfico, el tipo de servicios, la zona demográfica e información individual del abonado, combinado con técnicas de análisis complejas, los operadores tienen la posibilidad de elaborar ofertas más personalizadas para aumentar la satisfacción de usuario. Este tipo de acciones tienen un impacto directo en la renovación de suscripciones, mejorando tanto la productividad como la popularidad frente a competidores.

3. *Mejora de la experiencia de usuario*: El uso de *big data* ha permitido a los operadores crear una relación más estrecha y personalizada con sus clientes, reduciendo así el número de llamadas recibidas por el centro de atención al cliente. Poder elaborar un perfil individual de experiencia del usuario permite anticipar posibles insatisfacciones y, por consiguiente, quejas.

4. *Detección temprana de fallos*: La posibilidad de manejar gran cantidad de datos de diferentes fuentes y estructuras permite al operador monitorizar todos los elementos que componen su red. Haciendo uso de técnicas sofísticas de BDA se pueden desarrollar modelos que detecten, en un tiempo reducido, fallos en la red (e incluso que puedan predecirlos) para poder realizar las correspondientes acciones correctivas sin que el impacto en el usuario sea crítico.

5. *Ahorro de energía*: El gasto energético que supone la gestión de una red de telecomunicaciones, no solo de los propios elementos que componen la

red, sino de los propios dispositivos de almacenamiento de datos, supone un gasto considerable para los operadores. Las técnicas de BDA proporcionan mecanismos para poder optimizar, de forma automática, estos elementos de la red en función de su uso y disponibilidad.

El trabajo desarrollado en esta tesis se centra, principalmente, en la mejora de la experiencia de usuario y la detección temprana de fallos.

2.1.1. Metodologías para un proyecto de BDA

La implementación de un proyecto de BDA requiere especial atención para asegurar su éxito. Este tipo de proyectos tiene un alcance mucho mayor que los proyectos de desarrollo software clásicos. La variedad de aspectos a considerar en este tipo de proyectos va desde decisiones de carácter técnico (p. ej., fuentes de datos a usar, procesamiento de datos a realizar o técnicas de modelado) hasta decisiones puramente de negocio.

Para afrontar estos retos, a lo largo de los años se han definido una variedad de metodologías que permiten desarrollar un proyecto de BDA con mayor eficiencia y seguridad. A continuación, se hace una breve explicación de las tres metodologías más adoptadas [30]:

Knowledge Discovery in Databases (KDD)

Se define como un proceso no-trivial e iterativo para identificar patrones desconocidos en los datos, proporcionando información de utilidad y novedosa con el fin de tomar decisiones en base a ello. La Figura 2.1 muestra las principales etapas de la metodología KDD. Este proceso consiste en 9 pasos:

1. *Entendimiento del dominio*, para adquirir un conocimiento del dominio de la aplicación, los objetivos y el entorno donde se realizará.
2. *Selección de datos*, para identificar los datos que se usarán para obtener el entendimiento (es decir, los resultados del proyecto).
3. *Preprocesado*, para enriquecer los datos y mejorar su fiabilidad mediante procesos de limpieza.

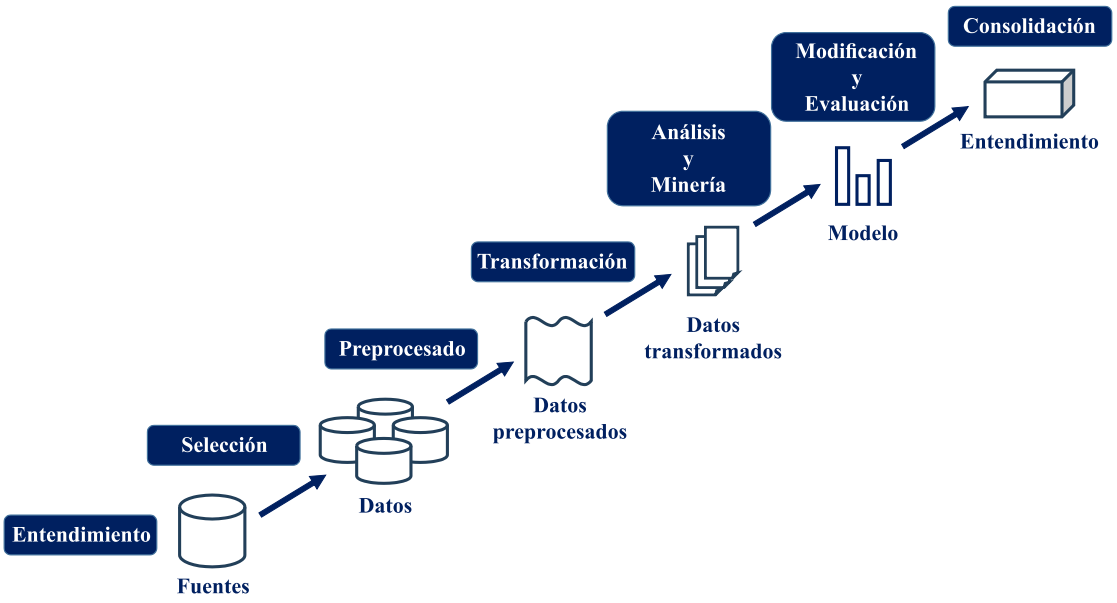


FIGURA 2.1: Visión general de la metodología KDD.

4. *Transformación*, para obtener datos más apropiados para su minería.

5. *Análisis y selección de la tarea de minería*, para determinar el tipo de modelo más apropiado.

6. *Minería de datos*, donde se aplican los diferentes algoritmos para la obtención de modelos.

7. *Modificación del modelo*, para configurar sus parámetros de configuración hasta obtener los mejores resultados.

8. *Evaluación*, para interpretar y validar los resultados obtenidos con respecto a los objetivos definidos en el primer paso.

9. *Consolidación del entendimiento*, con el fin de incorporar los resultados obtenidos en otros sistemas para acciones posteriores.

Esta metodología se considera, a su vez, la base de las otras 2 metodologías descritas a continuación [31].

Metodología SEMMA

Desarrollada por *SAS institute*, es una metodología para la selección, exploración y modelado para el descubrimiento de nuevos patrones haciendo uso de

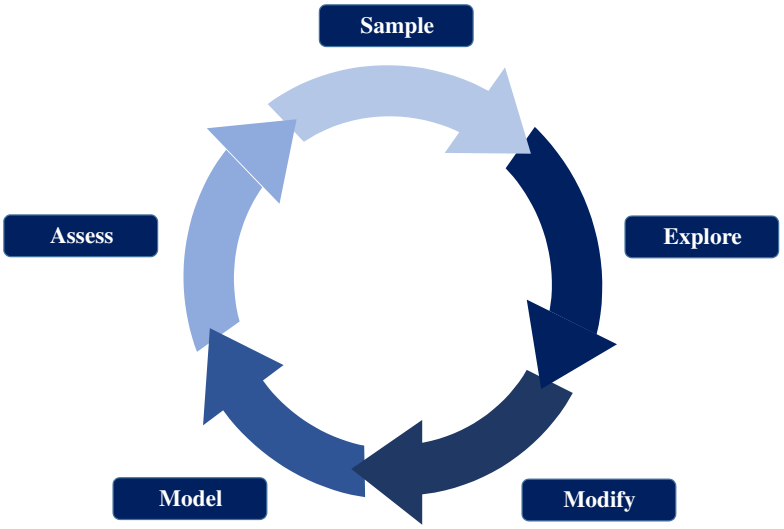


FIGURA 2.2: Visión general de la metodología SEMMA.

grandes cantidades de datos. Como definen sus siglas, se compone de 5 pasos como se muestra en la Figura 2.2 [32]:

1. *Sample*, analizar un conjunto reducido de datos de un grupo mucho mayor.
2. *Explore*, buscar patrones en los datos con el fin de obtener nueva información.
3. *Modify*, transformar los datos con el fin de obtener, modificar o eliminar variables.
4. *Model*, crear el modelo que mejor se ajusta a los objetivos del proyecto.
5. *Assess*, evaluar la utilidad y fiabilidad de los resultados.

Metodología *Cross Industry Standard Process for Data Mining* (CRISP-DM)

La Figura 2.3 detalla las 6 fases en las que se divide esta metodología [33]:

1. *Entendimiento del negocio*, para definir los objetivos y convertir este conocimiento en una definición de proyecto de minería de datos.
2. *Entendimiento de los datos*, para recopilar los datos y familiarizarse con ellos con el fin de identificar problemas o información desconocida. Esta etapa aún permite una revisión del entendimiento del negocio si la información obtenida pudiera afectar a los objetivos.

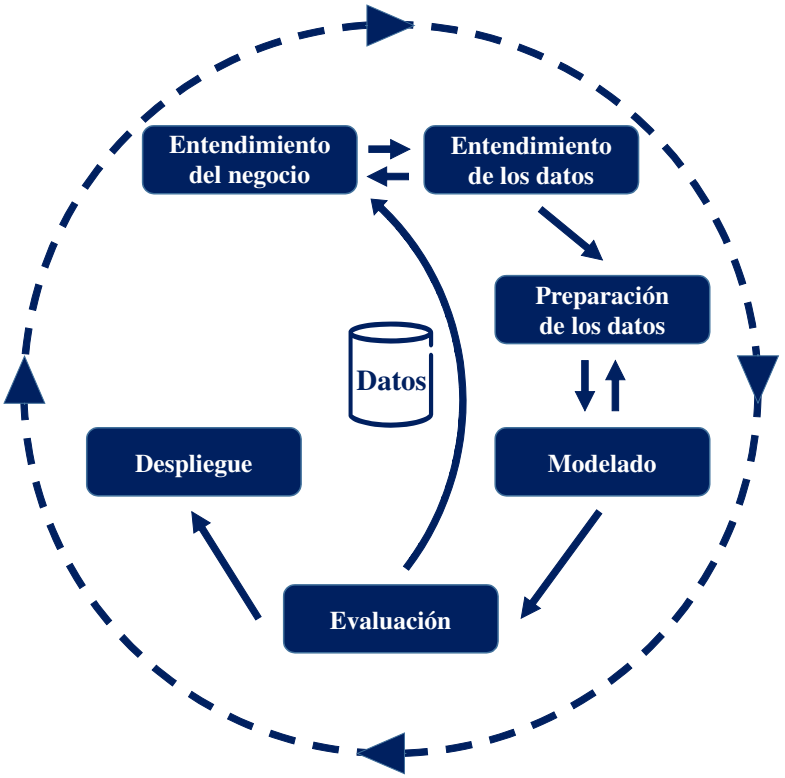


FIGURA 2.3: Visión general de la metodología CRISP-DM.

3. *Preparación de los datos*, para construir un conjunto de datos que se use como entrada del modelo posteriormente. Este proceso puede realizarse tantas veces como sea necesario para generar el conjunto más útil.

4. *Modelado*, donde se prueban y seleccionan varias técnicas de modelado, configurando sus parámetros para un funcionamiento óptimo. Este proceso puede llevar a la necesidad de nuevas transformaciones por lo que esta etapa podría generar la necesidad de repetir la fase de preparación.

5. *Evaluación*, para evaluar los resultados obtenidos tras la creación del modelo generado anteriormente, revisando los pasos seguidos para obtener el modelo y comprobando que se ajustan a los objetivos fijados. En caso contrario, podría ser necesario empezar de nuevo el proceso desde el primer paso.

6. *Despliegue*, para proporcionar los resultados del proyecto a las partes interesadas. Puede ser tan simple como un reporte para el cliente o la implementación de un proceso de minería de datos.

Como se observa en la definición de cada metodología, existen muchas similitudes entre las tres, por lo que no existe una decisión clara de cuál es la que mejor

resultados ofrece. Este tipo de decisiones deben hacerse al inicio del proyecto de BDA en base a diferentes criterios (p. ej., naturaleza del proyecto, herramientas disponibles, tiempos de desarrollo ...).

2.2. Redes autoorganizadas

Con el aumento en la complejidad de las redes celulares, las tareas de planificación, operación y mantenimiento se han complicado enormemente. Para resolver ese problema, se ha popularizado el uso de herramientas para la automatización de dichas tareas, dando lugar a las redes autoorganizadas (*Self-Organizing Networks*, SON) [34]. Del mismo modo, el uso de técnicas de BDA ha reforzado su popularidad añadiendo nuevas herramientas y mecanismos para solventar los retos planteados por el uso de nuevas tecnologías (p. ej., 5G o 6G) y nuevos servicios [35]. Los principales beneficios son [36]:

1. Reducción de costes operacionales y de inversión, disminuyendo la intervención humana en tareas manuales que han de repetirse en el tiempo y/o el espacio.
2. Reducción de los tiempos de nuevas instalaciones y despliegues.
3. Gestión de la red mucho más fiable y rápida con la incorporación de nuevos procesos automatizados.
4. Aumento de la experiencia de usuario al aumentar el rendimiento general de la red y disminuir los tiempos de resolución de fallos.

En redes móviles, las técnicas SON suelen dividirse en tres categorías principales: la planificación automática (o autoconfiguración), la optimización automática (o autoajuste) y la resolución automática de fallos (o autocuración). A continuación, se definen los principales objetivos de cada categoría.

Autoconfiguración

La autoconfiguración tiene como objetivo el despliegue y la puesta en marcha de nuevos elementos en la red de forma automática. Este proceso incluye las tareas de

despliegue, el establecimiento de la conexión con otros elementos y la configuración inicial de parámetros sin la intervención del operador. Alguno de los casos de uso más comunes son la selección de nuevos emplazamientos [37], la configuración del identificador de celda [38], la planificación de frecuencias [39], o la estructuración jerárquica de la red [40].

Autooptimización

El objetivo de la autooptimización es adaptar la configuración de los elementos de la red a los cambios que esta sufre a lo largo del tiempo. Este proceso requiere la evaluación periódica del funcionamiento de la red, la obtención de nuevos parámetros de configuración, y la posterior aplicación de dichos cambios [41]. En este proceso, los modelos de optimización implementados hacen uso de las diferentes fuentes de datos disponibles en la red como entrada de sus algoritmos. Por tanto, el uso de técnicas para recopilar y procesar grandes volúmenes de datos, de diferente naturaleza, permitirá desarrollar modelos más complejos y fiables aumentando sus beneficios [42].

Existen numerosos casos de uso de autooptimización. Algunos de los más destacados son la optimización del balance entre cobertura y capacidad (Coverage and Capaticty Optimization, CCO) [43], la optimización de la movilidad de usuarios (Mobility Robustness Optimization, MRO) [44], la optimización de la congestión [45], la optimización de asignación de recursos [46], y la optimización del reparto del tráfico [47].

Autocuración

La autocuración debe detectar fallos en la red y minimizar el impacto sobre los usuarios. Este proceso incluye la detección, notificación, compensación, diagnóstico y resolución de los problemas en el funcionamiento. Para ello, todos los elementos de la red deben monitorizarse con el fin de recopilar los datos necesarios para evaluar su estado [48]. Algunas de las aplicaciones de la autocuración son la correlación de alarmas [49], la compensación de fallos [50] o el diagnóstico automático de fallos [51].

2.3. Gestión de redes celulares

El proceso de gestión de red celular se refiere a todas las actividades, métodos, procedimientos, y herramientas que conciernen a la actividad, administración, mantenimiento y aprovisionamiento de la red. La actividad hace referencia al correcto funcionamiento de la red y sus servicios. Incluye la monitorización de la red para detectar problemas, idealmente, antes de que el usuario se vea afectado. La administración se encarga de controlar los recursos de la red y como estos son asignados. Incluye todas las tareas para mantener la red bajo control. El mantenimiento concierne a todas las acciones para reparar y mantener actualizada la red. Estas acciones también son, generalmente, medidas correctivas y preventivas para mejorar el funcionamiento y la estabilidad de la red. Aprovisionamiento se refiere a la disponibilidad y configuración de los recursos para ofrecer los distintos servicios [52].

Una correcta gestión de la red puede aportar grandes beneficios a los operadores en términos de reducción de costes, ahorro de tiempo, incremento de la productividad o retención de clientes y nuevas contrataciones. Una gestión efectiva de la red tiene un impacto directo en la calidad de experiencia del usuario, pues los problemas se identifican con mayor celeridad evitando dañar la reputación del operador. Por ello, los sistemas de gestión de redes (*Network Management Systems*, NMSs) han adquirido una gran importancia para una gestión exitosa de la red [53]. Estos sistemas se encargan del mantenimiento y la configuración de los elementos de la red, servidores, y servicios, así como de la comprobación del correcto funcionamiento de todos los dispositivos que componen la red.

A continuación, se definen algunos conceptos básicos de la gestión de redes celulares.

2.3.1. Áreas funcionales

La gestión de la red engloba diferentes áreas funcionales, siendo algunas de los más importantes la gestión de la configuración, la gestión de la contabilidad, la gestión de fallos, la gestión del rendimiento o la gestión de la seguridad [54]. A continuación, se detallan cada una de estas áreas:

- *Gestión de la configuración*: Es el proceso para organizar y mantener, de forma consistente y manejable, toda la información sobre los elementos físicos y las aplicaciones que conforman la red. Incluye todos los aspectos de configuración de los elementos de red, tales como el aprovisionamiento de la red, inventario, gestión remota, copias de seguridad, etc.

- *Gestión de la contabilidad*: Proporciona todos los mecanismos necesarios para medir el uso de los diferentes servicios de la red y determinar todo lo asociado a costes relativos al operador, y los cargos al cliente final por el uso del servicio. El objetivo es hacer un uso más efectivo de todos los sistemas disponibles, y minimizar el coste operacional. Además, es el área responsable de la facturación.

- *Gestión de fallos*: Es el proceso encargado de detectar y corregir averías en la red (tanto de elementos físicos como de las herramientas usadas). La detección de fallos se realiza mediante la gestión de alarmas o ficheros de errores en tiempo real. Cuando un fallo ocurre, se genera una notificación o alarma al administrador de red para su posterior revisión y análisis. Los objetivos de la gestión de fallos incluyen la detección temprana del fallo, la posterior generación de notificaciones o la resolución del problema, entre otros.

- *Gestión del rendimiento*: Esta área cubre todo lo relacionado con el rendimiento general de la red (es decir, detección de cuellos de botella, identificación de problemas potenciales, monitorización de indicadores ...). Uno de los principales objetivos de esta área es identificar qué optimizaciones se pueden realizar para mejorar el rendimiento general de la red. Para ello, se recopilan diferentes estadísticas y se analizan para la posterior toma de decisiones.

- *Gestión de la seguridad*: Cubre la configuración y la monitorización para minimizar y/o evitar posibles ataques de seguridad. Entre los objetivos de esta gestión se encuentran la seguridad de la red, la inspección del tráfico para protegerse frente a virus o código malicioso o la creación de políticas para limitar el incremento de tráfico a destinos particulares o de orígenes específicos.

Para una efectiva gestión de la red, las diferentes áreas deben estar integradas entre sí. Es decir, la información generada y gestionada por un área puede ser de utilidad para otra, por lo que debe estar siempre disponible.

2.3.2. Gestión de la calidad de experiencia

En los últimos años, la forma en la que se gestionan las redes celulares ha pasado de un modelo más centrado en la calidad de servicio (*Quality of Service*, QoS) y el rendimiento de la red, a uno cuyo objetivo es la satisfacción final del usuario y la calidad de experiencia (*Quality of Experience*, QoE) [55]. La gestión de la QoE ha adquirido especial importancia en dos áreas específicas de la gestión de redes celulares: la gestión del rendimiento y la gestión de fallos. Esta importancia está motivada por el hecho de que la satisfacción de usuario se ve mucho más afectada por degradaciones en la calidad no controladas (p. ej., perdida de paquetes por problemas de congestión) que por degradaciones controladas (p. ej., reducción del ancho de banda a causa del plan de tarificación). Por tanto, la gestión de la QoE es, hoy en día, una prioridad para los operadores.

La QoE se define como la aceptabilidad general de una aplicación o servicio, tal y como la percibe subjetivamente el usuario final [56]. De esta forma, la QoE describe el grado de satisfacción o descontento de un usuario final cuando usa un producto o servicio. Por tanto, se trata de un concepto amplio dependiente de factores de diferentes dominios y disciplinas. Estos factores pueden agruparse en: a) factores humanos relacionas con la propia característica del usuario (p. ej., edad, educación, situación socioeconómica ...), b) factores de sistema propios de la aplicación o servicio que se está consumiendo (p. ej., resolución, tasa de refresco, ancho de banda ...), y c) factores de contexto del entorno que rodea al usuario (p. ej., ubicación, hora del día, marca del proveedor ...) [57].

El objetivo final de la gestión de la QoE es optimizar la experiencia del usuario haciendo un uso efectivo de los recursos de la red para aumentar la satisfacción del cliente [58]. Para gestionar de forma efectiva la QoE, es necesario entender e identificar los múltiples factores (tanto objetivos como subjetivos) que afectan a cada servicio de forma específica. Este proceso puede describirse en tres pasos principales: a) la caracterización y modelado de la QoE, b) la medida y monitorización de la QoE, y c) la optimización y control de la QoE [59].

Caracterización y modelado de la QoE

Por la subjetividad de la QoE, ésta se ha evaluado, tradicionalmente, mediante encuestas realizadas directamente sobre los usuarios para conocer su grado de

satisfacción. Sin embargo, este proceso es costoso en tiempo y recursos, por lo que los operadores no pueden utilizarlo como mecanismo para tomar decisiones con el fin de mejorar la QoE una vez la red ya está en funcionamiento.

En los últimos años, se han propuesto nuevos métodos para estimar la QoE en función de determinados indicadores de rendimiento asociados a los servicios. Una solución para obtener este tipo de indicadores es mediante la instalación de agentes directamente en los terminales móviles con el fin de realizar medidas subjetivas de QoE y enviar la información a un servidor central [60]. Otras alternativas incluyen el despliegue de nuevos elementos de red que capturen el tráfico y permitan generar métricas específicas de servicio [61].

En cualquier caso, los indicadores de rendimiento de la red (p. ej., caudal de usuario) no reflejan directamente el valor de QoE percibido por el usuario. Para ello, suelen utilizarse funciones de utilidad asociadas a cada servicio para obtener un valor subjetivo de QoE a partir de un valor objetivo de QoS. En la literatura existen infinidad de trabajos que definen modelos para relacionar el estado de la red en términos de QoS con un valor de puntuación de opinión media (*Mean Opinion Score*, MOS) para un servicio específico. Por ejemplo, en [15] los autores presentan un modelo de QoE para el servicio de navegación web definido por una función de utilidad que relaciona el caudal (*throughput*) de usuario con un valor de la escala MOS. Para ello, los autores realizan una encuesta a 52 voluntarios en la que recogen la experiencia percibida para 14 páginas web diferentes. La Figura 2.4 muestra gráficamente el comportamiento de dicho modelo.

Medida y monitorización de la QoE

Para proporcionar una medida precisa de la QoE, es necesario considerar tantos factores por los que puede verse afectada como sea posible. El proceso de medida y monitorización de la QoE engloba la recopilación de datos relacionados con el estado y el rendimiento de la red, las características del terminal móvil, el usuario, el contexto, la información específica del servicio, entre otros [62]. Estos datos pueden obtenerse de diferentes puntos del sistema de comunicación (es decir, red y/o terminal móvil), en diferentes momentos y empleando diferentes métodos.

Según el origen de la información, el proceso de medida y monitorización de la QoE se clasifica en medidas en la red y medidas en el cliente. Monitorizar en el

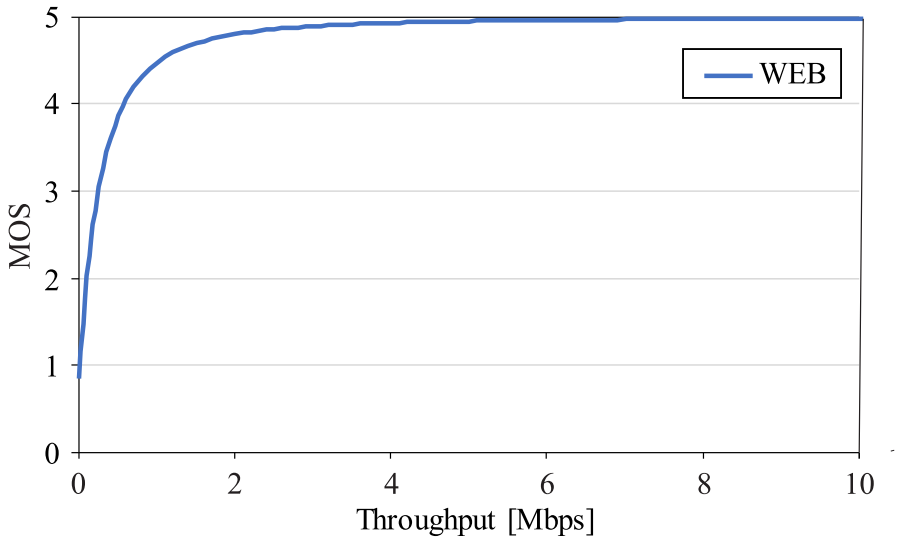


FIGURA 2.4: Función de utilidad que relaciona el *throughput* y la QoE para el servicio web.

lado del cliente proporciona los datos más realistas sobre la calidad del servicio y lo que el usuario percibe. Por ello, el 3GGP ha estandarizado diferentes métodos para medir la QoE de servicios de streaming RTP (*Real Time Protocol*) o HTTP (*HyperText Transfer Protocol*), descarga progresiva y telefonía multimedia [63][64]. El principal reto sigue siendo compartir este tipo medidas con la red y/o el proveedor del servicio, debido a problemas de privacidad, confianza e integridad, ya que el usuario podría mentir con el fin de obtener mejores prestaciones.

Según el método de obtención de la información, el proceso de medida y monitorización de la QoE puede clasificarse en activo y pasivo [65]. Los métodos activos generan un tráfico artificial en la red con aplicaciones específicas, para simular diferentes patrones de consumo de un servicio determinado, permitiendo así medir la QoE en diferentes localizaciones de la red. Sin embargo, estos métodos no pueden utilizarse en momentos de máximo uso de la red, pues podría afectar a la experiencia percibida por usuarios reales, por lo que las medidas no son tan precisas. Por otro lado, los métodos pasivos extraen información real del consumo de un servicio mediante indicadores de rendimiento del sistema o sondas en diferentes interfaces de la red. En el caso de los indicadores de rendimiento, esta información no suele estar segregada a nivel de servicio, por lo que la monitorización a este nivel no es sencilla. Por el contrario, las sondas sí permiten diferenciar entre conexiones de diferentes servicios, mediante el análisis del tráfico capturado, pero el coste es mucho mayor ya que se necesitan elementos adicionales para capturar y almacenar este tráfico.

Optimización y control

El objetivo principal de la optimización y el control de la QoE es maximizar la satisfacción del usuario y optimizar el uso limitado de los recursos de la red. Este proceso presenta grandes retos por la heterogeneidad de dispositivos móviles y servicios, el crecimiento exponencial de los datos móviles, la diversidad de contextos en los que se consumen los servicios y las diferentes expectativas de los usuarios. Por tanto, la mejor estrategia de optimización dependerá de diferentes factores como el foco de optimización (es decir, si está orientado a la red o al usuario), los parámetros elegidos para el ajuste o el momento elegido para realizar dicho proceso.

Algunos de los trabajos en la literatura sobre optimización de la QoE se centran en algoritmos de planificación dinámica de recursos para priorizar ciertas conexiones para ajustar algún indicador agregado de QoE [14][46]. Recientemente, se han realizado diferentes trabajos con el objetivo de aliviar los problemas de degradación y optimizar la QoE general de la red realizando un reparto del tráfico entre celdas [66][67].

2.3.3. Componentes y roles en la gestión de fallos

En un mercado donde las redes y los servicios son muy similares entre operadores, la forma en la que se gestiona la red para incrementar la satisfacción del usuario se ha convertido en un factor diferencial. Uno de los objetivos clave en este proceso es reducir el impacto de los fallos en la red, el cual, como se ha explicado anteriormente, es responsabilidad del área de gestión de fallos [68]. Sin embargo, el proceso de gestión de fallos es uno de los más complejos, y en el que intervienen más elementos (tanto físicos como humanos) de todos los procesos en la gestión de la red. Tradicionalmente, este proceso se ha realizado por un grupo de expertos desde el centro de operaciones de la red (*Network Operation Center*, NOC). La función de un NOC es la de monitorizar la red y gestionar los fallos que se detecten en ella. Para ello, se recopila la información de los diferentes elementos de red, mediante los NMSs, y se notifica de cualquier fallo o incidencia al personal especializado para su posterior evaluación. Esta clase de centros cuenta con la presencia de diversos tipos de ingenieros y técnicos, catalogados generalmente por niveles (de menor a mayor valor en función de su experiencia). Ellos son los

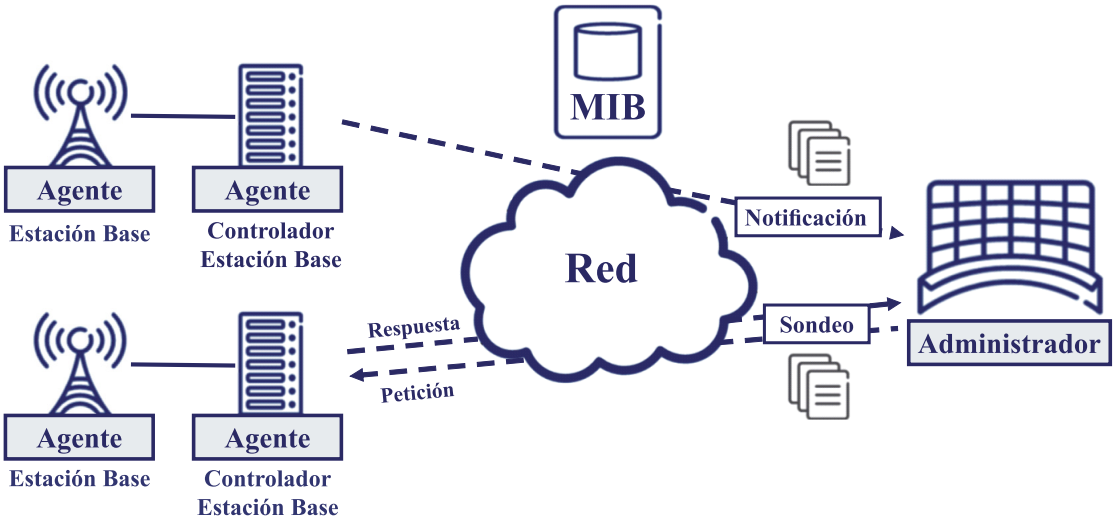


FIGURA 2.5: Arquitectura centralizada de gestión de red.

encargados de recibir las notificaciones de fallo, analizar el problema y actuar de la forma requerida para la resolución del problema.

Aunque existen diferentes arquitecturas en las redes para su gestión (p. ej., centralizada, distribuida y jerárquica), todas comparten los mismos componentes [69]. La Figura 2.5 muestra un tipo específico de arquitectura compuesta de los siguientes componentes:

1. *Administrador*: Un administrador es una herramienta con una interfaz de usuario, generalmente ubicado en el NOC, que permite la gestión de diferentes elementos en una red móvil. Sus principales funciones son: a) monitorizar los diferentes dispositivos gestionados en la red y recopilar la información generada por ellos. Dicha información se utilizará posteriormente por el equipo de gestión; b) solicitar información a los dispositivos y recibir sus respuestas; y c) configurar los dispositivos mediante variables de configuración y umbrales.

2. *Agente*: Un agente es un programa integrado en un elemento de red (p. ej., servidor, enrutador, conmutador ...) responsable de monitorizar y comunicarse con el administrador. El agente proporciona información sobre el estado de un elemento de red al administrador, ya sea de forma asíncrona o a partir de la recepción de una petición.

3. *Manager Information Base (MIB)*: Un MIB es una base de datos que recopila información de parámetros de configuración de los elementos de la red. Esta

información se recopila del agente y se comparte con el administrador. En un proceso de gestión de fallos, esta información se utiliza, generalmente, para identificar la causa del problema y para análisis posteriores.

4. Protocolo de gestión de red: El principal rol de este protocolo es el de definir un lenguaje común empleado por el agente y el administrador para intercambiar información de los elementos de la red. Los protocolos más utilizados son [70]:

   - Protocolo SNMP (*Simple Network Management Protocol*), usado para la supervisión de fallos y rendimiento. Gracias a su naturaleza, SNMP también funciona muy bien en estrategias de sondeo (solicitando información) y determinando el estado operativo de una funcionalidad específica [71].
   - Protocolo RMON (*Remote Network MONitoring*), que surge como extensión del protocolo SNMP, define una serie estadísticas y funciones que se pueden intercambiar entre controladores [72].
   - Protocolo CMIP (*Common Management Interface Protocol*), más complejo que los anteriores y usado, principalmente, para el intercambio de alguna información de proveedores de servicio [73].

5. *Modelo de comunicación*: Las principales estrategias para el intercambio de información entre el administrador y el agente en los NMSs son por sondeo o notificación. El modo por sondeo (*poll*, en inglés) se basa en una petición por parte del administrador y la posterior respuesta por parte del agente. En este modo, la comunicación es siempre iniciada por el administrador. Dicha comunicación puede configurarse de forma automática o iniciada por el usuario. Por otro lado, en el modo por notificación (*push*, en inglés) es el agente el que inicia la comunicación en base a unos parámetros previamente configurados por el usuario. Esta notificación puede ser periódica y planificada, o asíncrona en base a una serie de reglas definidas.

La principal diferencia entre los distintos tipos de arquitectura citados anteriormente es la combinación y distribución de estos elementos en la red. Por ejemplo, la arquitectura centralizada, descrita en la Figura 2.5, se compone de un único administrador controlando la red y un agente por cada elemento de la red que está siendo gestionado. Por otro lado, la arquitectura distribuida divide la red en

segmentos y cada segmento se compone de un administrador sin que haya interacción entre ellos. En el caso de la arquitectura jerárquica, se combinan ambas estructuras, donde cada administrador gestiona, de forma local, un conjunto de elementos de la red, y existe la figura de un administrador de nivel superior que actúa como controlador central.

Independientemente de la arquitectura utilizada, toda la información generada por los elementos de la red y recopilada por el administrador, es centralizada en el NOC donde se procesa. Esta acción es llevada a cabo por el equipo del NOC, el cual es responsable de monitorizar cada elemento de la red controlado por el NMS, así como de realizar acciones correctivas para resolver incidencias y fallos. Este equipo está compuesto por diferentes actores y roles que trabajan juntos para asegurar un funcionamiento óptimo de la red. Los principales roles en el proceso de gestión de fallos son:

- *Equipo de gestión*: Es el rol principal del proceso de gestión de fallos. Responsable de resolver los fallos de la red y restaurar el servicio tan pronto como sea posible. Sus principales tareas son las de analizar las alertas para detectar fallos, identificar y priorizar los fallos, y ejecutar acciones correctivas para restaurar el servicio. Cómo se ha mencionado anteriormente, este equipo está formado por ingenieros y técnicos especializados, los cuales se agrupan en diferentes niveles en base a su experiencia y habilidades para resolver problemas. Un técnico de primer nivel (L1) (también conocido como informador de fallos) es responsable de supervisar las alarmas generadas por la red, detectar y notificar los fallos. En caso de que un fallo requiera de una inspección posterior, el grupo L1 genera un ticket de incidencia, el cual es escalado a un técnico de nivel superior (es decir, L2 y/o L3). Los técnicos de niveles superiores son responsables de detectar la causa del fallo y efectuar las tareas de resolución necesarias. En la mayoría de los casos, estas acciones son remotas y permiten restaurar el servicio sin una acción directa sobre el elemento de red. En otros casos, se requiere la creación de una orden de trabajo para que un ingeniero haga las tareas de reparación requeridas en el elemento de red (p. ej., reemplazo, configuración, reparación ...). Al mismo tiempo, estos grupos superiores son también responsables de validar el resultado de la solución realizada.

- *Equipo de atención al cliente*: Este es un rol secundario que da servicio al cliente haciendo de punto de contacto para peticiones y quejas. Este equipo se

centra en recibir y registrar peticiones por parte de los clientes o empleados (por medio de llamadas telefónicas o correos) para informar de fallos que no han sido notificados por una alarma pero que han sido detectados por el usuario final. Entre sus funciones, el equipo es responsable de recibir y gestionar las llamadas del cliente, registrar el incidente mediante la creación de un ticket de incidencia que no estará asociado a ninguna alarma, solicitar soporte de un técnico especializado por medio de una orden de trabajo (en el caso de que así se requiera), y actualizar la información del ticket con el estado de la resolución.

- *Otros roles*: En ciertos procesos de gestión de fallos, es posible la presencia de otros grupos. Por ejemplo, un distribuidor es el encargo de gestionar las ordenes de trabajo generadas. Será el responsable de notificar al ingeniero de campo y de cerrar la orden de trabajo cuando se haya finalizado la acción correctiva. Del mismo modo, el ingeniero de campo es el encargado de hacer acciones directamente sobre el elemento de red defectuoso e implementar las acciones correctivas requeridas (p. ej., reemplazar una pieza del elemento, actualizar una aplicación ...) para resolver el fallo cuando este no puede hacerse de forma remota.

Capítulo 3

Construcción de funciones de utilidad para el modelado de QoE mediante trazas de conexión

La caracterización y modelado es una de las tareas principales en la gestión de la calidad de experiencia (*Quality of Experience*, QoE). En este capítulo, se evalúa el uso de la información generada en una red celular en forma de trazas de conexión para la adaptación de las funciones de utilidad empleadas para el modelado de la QoE. El ajuste de parámetros en las funciones de utilidad se hará para los servicios más populares actualmente en una red móvil.

El capítulo se divide en cinco secciones. La Sección 3.1 presenta una revisión del estado de la técnica para la caracterización y modelado de la QoE en redes móviles. La Sección 3.2 formula el problema de la caracterización de la QoE a partir de umbrales de calidad de servicio (*Quality of Service*, QoS). Dicha caracterización puede realizarse a nivel de usuario gracias a la obtención y procesado de forma masiva de trazas de conexión. Tras la formulación del problema, la Sección 3.3 describe el método automático implementado para la estimación del valor de los umbrales de QoS para cada servicio. Posteriormente, la Sección 3.4 presenta la validación del método de estimación de umbrales mediante un conjunto de trazas de conexión radio obtenidas de una red LTE real. Finalmente, la Sección 3.5 expone las principales conclusiones de este capítulo.

3.1. Revisión del estado de la técnica

Los operadores han cambiado la forma que tenían de gestionar sus redes a un enfoque más moderno y realista enfocado en la QoE y la satisfacción del usuario. Por tanto, entender y gestionar la QoE percibida por un usuario se ha convertido en un factor diferencial en un mercado tan competitivo como es el de las comunicaciones móviles. Como parte de este entendimiento, una de las principales tareas en la gestión de la QoE en redes móviles es su caracterización y modelado.

Una de las claves en la caracterización de la QoE es entender los factores que influyen en la percepción que el usuario tiene de la calidad del servicio, estableciendo la relación entre variables medibles y la experiencia percibida por el usuario final [58]. Generalmente, dicha caracterización consiste en funciones de utilidad analíticas que relacionan indicadores de QoS de la red con la opinión del usuario final, dando como resultado modelos de QoE para diferentes servicios.

La Figura 3.1 muestra la representación gráfica de una función de utilidad genérica que relaciona los diferentes valores de QoS con el valor final de QoE. Se aprecia cómo las funciones de utilidad suelen definirse a tramos con una fase final de saturación (valor constante), y, por tanto, los parámetros que componen estas funciones de utilidad consisten en umbrales de QoS que definen los límites (superior y/o inferior) a partir de los cuales la QoE se mantiene constante una vez se superan [14]. En la figura se distinguen claramente los diferentes tramos que definen los umbrales mencionados. Tradicionalmente, los valores de estos umbrales se obtienen mediante ensayos subjetivos en entornos de laboratorio con usuarios reales. Este método suele presentar muchos inconvenientes para los operadores, ya que es costoso en tiempo y precio y, además, puede no reflejar de forma precisa las condiciones de un entorno real. Como consecuencia, dichos umbrales raramente se actualizan. Sin embargo, las expectativas de QoE del usuario se siguen incrementando con la aparición de nuevos servicios, terminales más modernos, etc. Esto provoca que dichos umbrales queden obsoletos, y no acaben reflejando el grado en el que la satisfacción del usuario disminuye. Por tanto, es importante que los modelos de QoE se mantengan actualizados a través de la adaptación continua de sus parámetros. En la mayoría de los casos, modificar los parámetros del modelo es suficiente para actualizar de forma precisa la experiencia de usuario que evoluciona con el tiempo, evitando acciones más complejas y costosas. Incluso así, esto

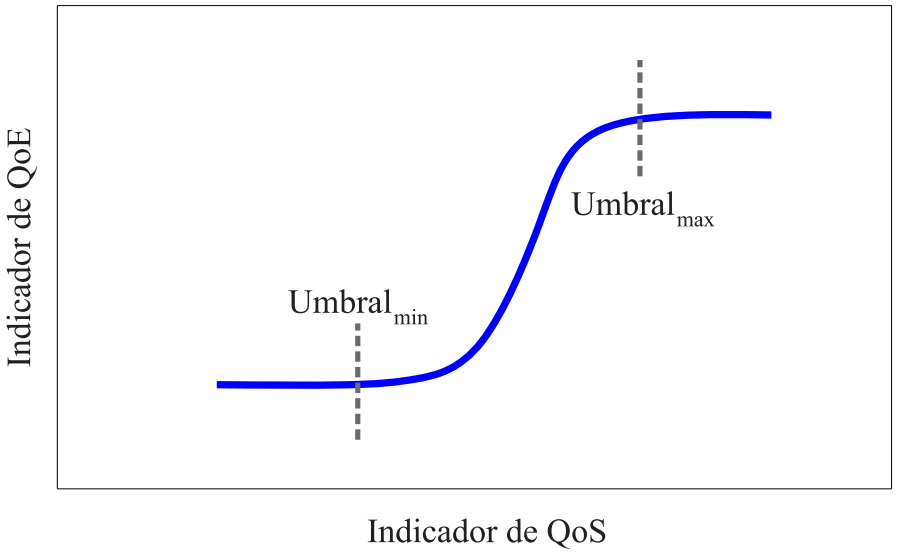


FIGURA 3.1: Función de utilidad genérica para la relación entre QoS y QoE.

requiere un proceso automático para el ajuste de dichos parámetros para evitar los ya mencionados ensayos.

Con este fin, los operadores buscan los medios necesarios que permitan realizar estas actualizaciones con el menor impacto en sus redes (es decir, sin la necesidad de cambios costosos y complejos). El uso de datos o indicadores generados en una red real es una solución efectiva para, en general, la gestión y optimización de las redes celulares, y, en particular, para la actualización de los umbrales en las funciones de utilidad. Las redes móviles actuales generan una gran cantidad de información recopilada en forma de medidas y registros. Sin embargo, la mayoría de esta información es descartada por los operadores, sustituyéndola por información agregada por estación base y periodo, que hace imposible una gestión de la QoE a nivel de usuario. Gracias a los últimos avances en las técnicas de análisis de datos masivos (*Big Data Analytics*, BDA), ahora los operadores tienen la posibilidad de analizar grandes volúmenes de datos, permitiendo el uso de información más específica a nivel de usuario, como es el caso de las trazas de conexión [74]. Dicha información puede ser de gran utilidad para el ajuste de modelos de QoE, pues registran sistemáticamente todos los eventos asociados con un usuario específico en un periodo de tiempo determinado, llegando a ser una herramienta muy poderosa para el análisis, monitorización y control de la red [35]. Sin embargo, hasta donde se conoce, en la literatura no se ha propuesto un proceso para la actualización automática de los umbrales de QoS en las funciones de utilidad de modelos de QoE.

En este capítulo, se presenta un nuevo método para el ajuste automático de los umbrales de QoS en modelos analíticos de QoE clásicos, analizando las trazas de conexión radio. Dicho ajuste se realiza para algunos de los servicios más demandados actualmente en las redes móviles: servicio de datos de búfer completo, vídeo streaming, voz sobre LTE y navegación web. Por simplicidad, el método propuesto se basa en la hipótesis de que los usuarios que experimentan una QoE no satisfactoria tienden a interrumpir de forma precoz sus conexiones con respecto a la media del resto de usuarios. Por tanto, los valores de los umbrales de QoS pueden inferirse detectando dicho comportamiento a partir de diferentes indicadores de red medidos a nivel de conexión. Las principales contribuciones de este capítulo son: a) validar la hipótesis de que la satisfacción de usuario se puede inferir a partir de indicadores de QoS modelando el comportamiento habitual según el servicio consumido, b) identificar el indicador de QoS que mayor impacto tiene en la QoE para cada uno de los principales servicios de una red móvil, c) proponer un método automático para ajustar los umbrales de QoS en funciones de utilidad para la caracterización de la QoE, y d) validar el modelo usando un conjunto de trazas de conexión obtenidas de dos redes LTE comerciales.

3.2. Formulación del problema

Durante los últimos años, los operadores han evolucionado sus redes móviles incluyendo nuevos indicadores y contadores para reflejar de forma más precisa el funcionamiento de los distintos servicios (p. ej., tiempo de descarga o tiempo de carga para los servicios web y vídeo, respectivamente). Pese a ello, la satisfacción de usuario, al tratarse de un aspecto subjetivo, no puede medirse directamente, sino estimarse a partir de otros indicadores que sí puedan medirse. Para ello, los modelos de QoE hacen uso de funciones de utilidad que trasladan el valor de los indicadores de rendimiento (*Key Performance Indicators*, KPIs) que representan la QoS, a la experiencia de usuario, o QoE [75].

Cómo se ha indicado anteriormente, es plausible asumir que la experiencia de usuario se mantiene constante a niveles máximos cuando un umbral superior de QoS se supera. Esto es, existe algún valor de QoS por encima del cual no se percibe una mejor experiencia del servicio, habiendo llegado al máximo que el usuario percibe. De forma similar, se puede definir un umbral de QoS mínimo a partir del cual el usuario interrumpe su conexión debido a un mal funcionamiento del

servicio, muy por debajo de sus expectativas. Estas asunciones se pueden reflejar con una expresión genérica que relaciona QoE e indicadores de QoS como sigue:

$$QoE = \max\{\min\{f(QoS_1, QoS_2, \ldots, QoS_N), QoE_{max}\}, QoE_{min}\}\ , \tag{3.1}$$

donde QoS_i, $\forall i \in \{1, 2...N\}$, son los N indicadores de rendimiento de la red que reflejan de manera más directa el rendimiento del servicio, QoE es el indicador que cuantifica la experiencia de usuario, f es la función de utilidad de usuario, y QoE_{max} y QoE_{min} definen los límites de QoE máximo y mínimo percibidos por el usuario.

La existencia de límites de QoE implica, de forma directa, la existencia de umbrales de QoS ($QoS_{i,umb_{max}}$ y $QoS_{i,umb_{min}}$), que generan un valor de QoE constante cuando son superados. Por tanto, la definición de QoE expresada en (3.1), puede reformularse como

$$\begin{aligned} QoE = f(&\max\{\min\{QoS_1, QoS_{1,umb_{max}}\}, QoS_{1,umb_{min}}\}, \\ &\ldots, \max\{\min\{QoS_N, QoS_{N,umb_{max}}\}, QoS_{N,umb_{min}}\})\ , \end{aligned} \tag{3.2}$$

Pese a que se disponga de una función que relaciona la experiencia de usuario con umbrales de QoS, dicha relación no puede generalizarse para todos los servicios de una red, ya que la experiencia del usuario está estrechamente relacionada con factores que dependen del tipo de servicio, y, por tanto, no son los mismos para todos los servicios. Por ejemplo, mientras que las llamadas de voz se ven fuertemente afectadas por el retardo de paquete, un usuario subiendo una foto a una red social percibirá una peor experiencia en función del *throughput* experimentado durante dicha subida. Esto demuestra la necesidad de definir diferentes funciones de utilidad para cada servicio.

Para facilitar comparativas, la QoE se suele medir de la misma manera para todos los servicios, mediante la puntuación de opinión media (*Mean Opinion Score*, MOS). La escala MOS se distribuye en valores desde el 1 (la peor experiencia) hasta 5 (la mejor experiencia), de forma que $MOS^{(s)}_{min} \geq 1$ y $MOS^{(s)}_{max} \leq 5$. De acuerdo a estas nuevas consideraciones, las definiciones (3.1) y (3.2) pueden reformularse como

$$
\begin{aligned}
MOS^{(s)} &= \max\{\min\{f^{(s)}(QoS_1, QoS_2, \ldots, QoS_{N_{QoS}}), MOS^{(s)}_{max}\}, MOS^{(s)}_{umb_{min}}\} \\
&= f^{(s)}(\max\{\min\{QoS_1, QoS^{(s)}_{1,max}\}, QoS^{(s)}_{1,min}\}), \\
&\ldots, \max\{\min\{QoS_N, QoS^{(s)}_{N,max}\}, QoS^{(s)}_{N,min}\}) \,,
\end{aligned}
\tag{3.3}
$$

donde el superíndice s se refiere al servicio considerado (es decir, $s \in \{web,\ v\acute{i}deo, \ldots\}$). De (3.3), se deduce que usuarios de diferentes servicios podrían percibir una QoE diferente para los mismos valores de QoS (p. ej., *throughput*). Esto refleja un hecho constatable: que para un mismo estado de red no todos los servicios se perciben con la misma calidad. En otras palabras, con el fin de garantizar la misma MOS para todos los servicios, es necesario ofrecer diferentes valores de QoS para cada servicio.

A pesar de la gran utilidad de los umbrales de QoS, la obtención de dichos valores supone un gran reto para los operadores de red. Dichos umbrales de QoS por servicio, $QoS^{(s)}_{i,min/max}$, dependen de muchos factores, tales como la expectativa del usuario (una propiedad subjetiva), las características de los terminales (se espera una mejor experiencia cuanto más caro es el dispositivo) o la propia evolución de la red (niveles de experiencia de usuario percibidos en redes 3G puede que ya no sean suficientes con la aparición de las nuevas redes 5G). Por ello, la capacidad de actualizar estos umbrales de QoS de forma efectiva y automatizada se ha convertido en un factor clave para una correcta gestión de la QoE en las redes móviles.

3.3. Estimación de umbrales de QoS a nivel de servicio

En esta sección, se presenta un método automático para estimar los umbrales de QoS para diferentes servicios a partir de datos de una red real. Este proceso se lleva a cabo mediante un modelo heurístico basado en el comportamiento del usuario observado en trazas de conexión. El método descrito se enfoca en la estima del umbral de QoS que produce el peor funcionamiento del servicio tolerado por los usuarios antes de decidir interrumpir la conexión. Por tanto, dependiendo del

servicio consumido, este valor puede corresponderse con $QoS^{(s)}_{i,umb_{min}}$ (p. ej., caudal o *throughput* mínimo) o $QoS^{(s)}_{i,umb_{max}}$ (p. ej., retardo o *delay* máximo), en función de la naturaleza del servicio.

El método se divide en dos fases principales para la estimación de los umbrales de QoS para cada servicio: 1) la clasificación de las trazas de conexión a nivel de servicio, y 2) la estimación de los umbrales de QoS por servicio basado en el comportamiento del usuario.

A la entrada, el método utiliza los siguientes descriptores extraídos de las trazas de conexión:

1. El valor de QCI (*QoS Class Identifier*).
2. El tiempo de conexión RRC (*Radio Resource Control*).
3. El volumen total del tráfico a nivel del protocolo de convergencia de datos en paquetes en el enlace descendente (*DownLink*, DL) y ascendente (*UpLink*, UL).
4. La proporción del volumen de tráfico DL transmitido en los últimos intervalos de tiempo de transmisión (*Transmission Time Intervals*, TTIs) con respecto al total [76].
5. El ratio de actividad DL, calculado como la relación entre los TTIs activos (es decir, aquellos con datos para transmitir) y la duración efectiva de la conexión.
6. El *throughput* de sesión DL, calculado como el volumen DL transmitido entre la duración efectiva de la conexión.
7. El retardo medio del enlace descendente, τ, definido como la suma del retardo de conexión medio DL en las capas de RLC (*Radio Link Control*) y MAC (*Medium Access Control*).
8. La media del *throughput* de conexión PDCP *(Packet Data Control Protocol)* DL, $TH_{PDCP,DL}$, excluyendo los últimos TTIs.

A la salida, el método proporciona una estimación del umbral de QoS para cada indicador i y el servicio s, $QoS^{(s)}_{i,umb_{min/max}}$.

3.3.1. Paso 1: Clasificación de las trazas de conexión

Las trazas de conexión son una fuente de datos que permite analizar el rendimiento de la conexión percibido por el usuario de forma individual. Sin embargo, en el conjunto de trazas, coexisten conexiones de múltiples servicios con características muy diferentes, y que no suelen estar identificadas inequívocamente. Por ello, los operadores de red se ven obligados a identificar cada conexión mediante algún método, para ofrecer una gestión de recursos y acceso específico a cada servicio [77].

En las redes LTE, una de las formas principales para diferenciar entre servicios es mediante el valor de QCI de la conexión [78]. Dependiendo de dicho valor de QCI, se aplican diferentes prioridades y políticas de gestión de tráfico (p. ej., umbrales de cola, configuración del protocolo de la capa de enlace ...)

Tradicionalmente, los servicios se clasifican generalmente como QCI 1 (*Voice over Internet Protocol*, VoIP), QCI 2 (videoconferencia), QCI 3 (juegos en tiempo real), QCI 4 (vídeo difusión), QCI 5 (señalización IMS), y QCIs del 6 al 9 (servicios basados en el protocolo TCP sin una tasa de bit garantizada) [78]. Es en este rango de QCI del 6 al 9 donde coexisten gran variedad de servicios, desde redes sociales a transferencia de archivos, haciendo necesario un desglose adicional al del QCI, puesto que cada uno de los servicios agrupados en la misma categoría requieren indicadores y niveles de QoS muy diferentes desde la perspectiva de la QoE. Además, algunos operadores reservan estos últimos valores de QCI para priorización de usuario (p. ej., planes específicos de pago). Por tanto, la monitorización de la experiencia de usuario para servicios clasificados en estos rangos de QCI (del 6 al 9) se vuelve imprecisa y compleja.

En los últimos años, se han empleado diferentes métodos para la clasificación del tráfico de datos. Utilizar el puerto de la conexión es el más sencillo [79]. Sin embargo, la mayoría de los servicios no se rigen por una asignación de puertos estándar y, en otros casos, estos pueden ser dinámicos. Otros métodos más sofisticados se basan en el análisis de la información intercambiada durante la sesión [80]. Sin embargo, esté método no es posible cuando el tráfico está cifrado, como es habitual. Incluso aunque no lo estuviera, estos métodos requieren de sondas de tráfico, con un alto coste, conectadas en diferentes puntos de la red para acceder a dicha información [81]. Por tanto, los operadores se ven forzados a buscar otras opciones.

Una alternativa para solventar estas limitaciones es analizar las características del tráfico generado durante la conexión. Este método se basa en el hecho de que diferentes aplicaciones muestran un comportamiento distinto de su patrón de tráfico y, por tanto, pueden clasificarse mediante técnicas complejas de aprendizaje automático (*Machine Learning*, ML). En la literatura, pueden encontrarse diversos métodos para la clasificación del tráfico cifrado. En [82], se presenta un método de aprendizaje supervisado para la identificación de aplicaciones Android a partir del tráfico cifrado de la red. Trabajos más recientes han probado la eficacia del aprendizaje profundo para la clasificación de este tipo de tráfico cifrado. En [83], los autores presentan un método de clasificación del tráfico basado en redes neuronales profundas. En este trabajo se muestran los buenos resultados que ofrece el uso de redes neuronales frente a otro tipo de algoritmos de clasificación empleando un conjunto de datos, previamente etiquetado, generado por diferentes aplicaciones. Sin embargo, este tipo de técnicas supervisadas requieren de un conjunto de datos previamente clasificado para su entrenamiento, lo cual rara vez está disponible. Otras alternativas usan algoritmos no supervisados que no requieren de un aprendizaje previo. En [84], se presenta un método no supervisado para la clasificación, a posterior, del tráfico cursado en redes celulares de acceso radio. Este método se basa en el hecho de que la identificación del tipo de servicio se puede realizar mediante el uso de un conjunto de descriptores que representan las propiedades la conexión en base a las ráfagas de los datos generados. Sin embargo, las trazas de conexión radio no incluyen explícitamente este tipo de descriptores y, por tanto, deben estimarse a partir de otros parámetros de tráfico disponibles. En ausencia de un conjunto de datos correctamente clasificado, los autores en [84] hacen uso de un conjunto de estadísticas del tráfico móvil, publicados por un conocido proveedor, para validar la clasificación del tráfico obtenida por su método. Los resultados muestran que la distribución del tráfico estimada por dicho método es muy similar a la incluida en dicho estudio.

En ausencia de un conjunto de trazas reales que incluyan el servicio utilizado por el usuario para cada conexión, el método de clasificación descrito en [84] se utiliza como primer paso del método automático presentado en este capítulo. Para este etiquetado, se necesitan para cada conexión los siguientes descriptores del tráfico:

- El tiempo de conexión RRC.
- El volumen total de tráfico PDCP DL.

- El porcentaje de volumen de tráfico UL, η_{UL} [%], calculado como

$$\eta_{UL} = 100 \times \frac{V_{UL}}{V_{UL} + V_{DL}} \ , \tag{3.4}$$

siendo V_{UL} el volumen total de datos UL, y V_{DL} el volumen total de datos DL transmitido en la conexión.

- La proporción del volumen de tráfico DL transmitido en el último TTI, $\eta_{UL}^{ultimoTTI}$, calculado como

$$\eta_{UL}^{ultimoTTI} = \frac{V_{DL}^{ultimoTTI}}{V_{DL}} \ , \tag{3.5}$$

siendo $V_{DL}^{ultimoTTI}$ el volumen de datos DL transmitidos en los últimos TTI de cada ráfaga, y V_{DL} el volumen total transmitido en la conexión.

- El ratio de actividad DL, η_{DL}^{activo}, calculado como la proporción entre los TTIs activos y la duración efectiva de la conexión,

$$\eta_{DL}^{activo} = \frac{T_{DL}^{activo}}{T_{efec}} \ , \tag{3.6}$$

siendo T_{DL}^{activo} el tiempo total de los TTIs activos, y T_{efec} la duración total de la conexión.

- El *throughput* de sesión DL, $TH_{DL}^{sesión}$ [bps].

En base a estos descriptores, se calculan para cada conexión los siguientes parámetros a nivel de ráfaga, en el enlace descendente, requeridos por el método utilizado para la clasificación del tráfico: a) duración media de la ráfaga, b) número de total de ráfagas, y c) tamaño medio de ráfaga. Finalmente, los grupos resultantes de la clasificación se asocian a diferentes aplicaciones genéricas analizando las características de los distintos descriptores de cada grupo. Por ejemplo, el análisis de los parámetros a nivel de ráfaga de las conexiones de uno de los grupos revela que son conexiones con pocas ráfagas de gran tamaño. Esto produce que el ratio de actividad DL y el *throughput* de sesión sean los mayores de todos los grupos. Estas características son típicas de los servicios de datos de búfer completo (p. ej., descarga de grandes ficheros mediante el protocolo de transferencia de ficheros FTP, descarga de aplicaciones ...). El método de clasificación del tráfico no está detallado en este capítulo. Tanto los detalles del método como los resultados que validan su correcto funcionamiento están detallados en [84].

3.3.2. Paso 2: Estimación de umbrales de QoS

La QoE de cada servicio se modela mediante una función de utilidad específica, $f^{(s)}$, dependiente de diferentes indicadores de QoS. Para el método propuesto en este capítulo, el análisis se ha restringido a aquellos grupos de aplicaciones que agrupan el mayor porcentaje de las conexiones en una red móvil actual; concretamente, Voz sobre LTE (*Voice over LTE*, VoLTE), servicios de datos de búfer completo (p. ej., descarga de aplicaciones, actualización de software, descarga de ficheros FTP ...) y servicios de transmisión progresiva (p. ej., audio/vídeo streaming ...).

Para reducir la complejidad del método, únicamente se considera un indicador QoS por servicio, tratando de identificar cuál es el que mayor impacto tiene en la QoE para cada uno de los servicios (es decir, $N = 1 \ \forall \ s$ en (3.3)). Además, este indicador no es siempre el mismo para todos los servicios ($QoS_1(s_1) \neq QoS_1(s_2)$). En algunos casos, como los servicios en tiempos real (p. ej., VoLTE), el indicador QoS seleccionado es el retardo de paquete, al ser el que produce la mayor degradación en la experiencia que el usuario percibe. En el caso de servicios no basados en tiempo real (p. ej., la descarga de aplicaciones), el *throughput* de usuario, y no el retardo, es el que afecta en mayor medida. La literatura ha demostrado que la experiencia de usuario, para la mayoría de los servicios, está directamente relacionada con un solo indicador de QoS. Por ejemplo, en [85] se presenta un modelo analítico para estimar la QoE de servicios de difusión de vídeo basado en diferentes métricas a nivel de red (p. ej., el *throughput* medio de sesión, la tasa de pérdida de paquete o el tiempo de ida y vuelta). En él, se demuestra que la QoE para ese tipo de servicios está altamente relacionada con un único indicador de QoS (el *throughput* medio de sesión). Por otro lado, está extensamente aceptado que las llamadas de voz se ven principalmente afectadas por el retardo de paquete [86].

De ahora en adelante, se asume que la QoE derivada de una conexión k de un servicio s, $MOS^{(s)}(k)$, está condicionada por el valor del indicador i de mayor impacto en el servicio, $QoS_i^{(s)}(k)$. Por tanto, cuando dicho indicador traspasa un determinado umbral ($QoS_{i,min}^{(s)}$ o $QoS_{i,max}^{(s)}$, dependiendo del servicio), los usuarios experimentan su peor QoE, $MOS_{min}^{(s)}$, que se refleja en uno o varios indicadores de la conexión dependiendo del servicio, $QoS_j^{(s)}(k), ...$ Por ejemplo, en el caso de usuarios insatisfechos de VoLTE debido a un retardo excesivo, se espera que tiendan a terminar de forma repentina sus conexiones, y, por tanto, el efecto de

la interrupción se verá reflejado en la duración de la conexión. Por otro lado, los servicios no categorizados como servicios de tiempo real, como por ejemplo aplicaciones en segundo plano, interactivos o de difusión, la degradación sería más perceptible en el volumen del tráfico para cada conexión (un mal servicio impacta en que se demandan menos datos porque la sesión se interrumpe antes). Por tanto, para poder detectar conexiones con baja QoE, se requiere de un análisis de aquel indicador de tráfico, denotado a partir de ahora como $QoS_j^{(s)}(k)$, que para cada servicio refleja mejor el estado insatisfactorio del usuario (p. ej., duración de la conexión para VoLTE o volumen de datos para servicios de difusión). De este modo, la estimación de la QoE de una conexión k del servicio s se basa en el indicador de QoS j, $QoS_j^{(s)}(k)$, usado para inferir el comportamiento del usuario, y el indicador de QoS i, $QoS_i^{(s)}(k)$, como el indicador con el mayor impacto en la QoE.

Como el comportamiento de un usuario no es determinista, este $QoS_i^{(s)}$ tiene un componente aleatorio que hace que conexiones con el mismo $QoS_i^{(s)}$ no se interrumpan con el mismo valor de $QoS_j^{(s)}$. Para corregirlo, se construye una curva de regresión cuantílica que relaciona la $QoS_i^{(s)}$ y $QoS_j^{(s)}$ de la conexión para cada servicio. Para ello, se definen intervalos de los valores de $QoS_i^{(s)}$ y se obtiene el percentil del 50 % (mediana) de la distribución de $QoS_j^{(s)}$ para cada intervalo, $QoS_{j,50th\,tile}^{(s)}(QoS_i^{(s)})$. La Figura 3.2 muestra un ejemplo de este proceso para el servicio de VoLTE utilizando datos ficticios. En el eje x se representa el indicador de mayor impacto en el servicio ($QoS_i^{(s)}$), el retardo de paquete DL, mientras que el eje y representa el valor de la duración de la conexión, asumido como el indicador que mejor refleja el comportamiento de usuario, $QoS_j^{(s)}$. Cada punto representa una conexión obtenida de las trazas y clasificada, tras el paso 1, como servicio de VoLTE. La línea solida representa la curva de regresión del cuantil de orden 0,50 (es decir, mediana) obtenida mediante el proceso descrito anteriormente.

Sobre esta curva de regresión cuantílica, se estima el umbral de QoS para cada servicio, $QoS_{i,th_{min}}^{(s)}$. Dicho umbral de QoS define un límite entre dos estados: un estado degradado, donde el usuario percibe una mala experiencia y tiende a interrumpir la conexión, y un estado normal, donde el servicio es suficientemente bueno como para usarlo sin inconvenientes. En la Figura 3.2, este umbral se estima como el valor de retardo de paquete DL que hace que la curva de regresión cuantílica decrezca. Como este límite depende del servicio, a continuación, se modela el comportamiento de usuario esperado para las principales clases de servicio.

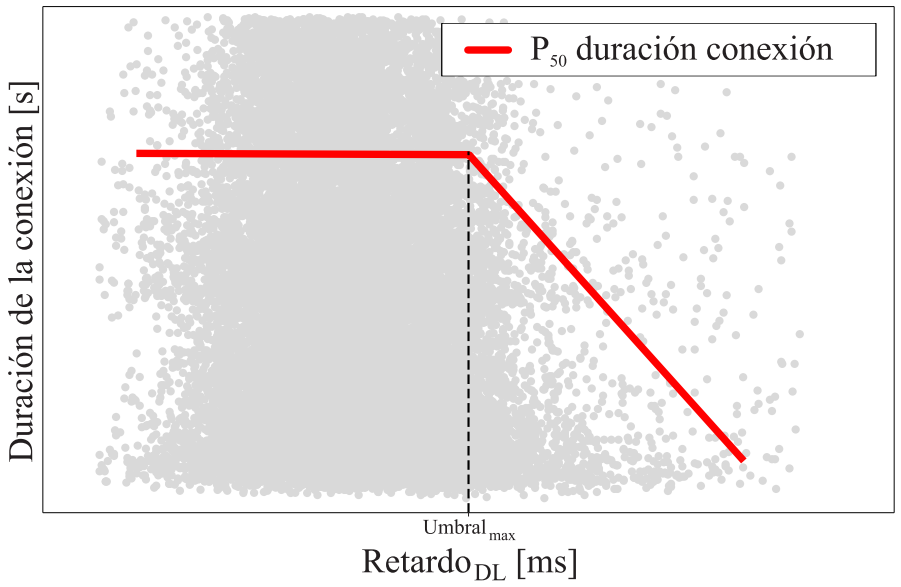
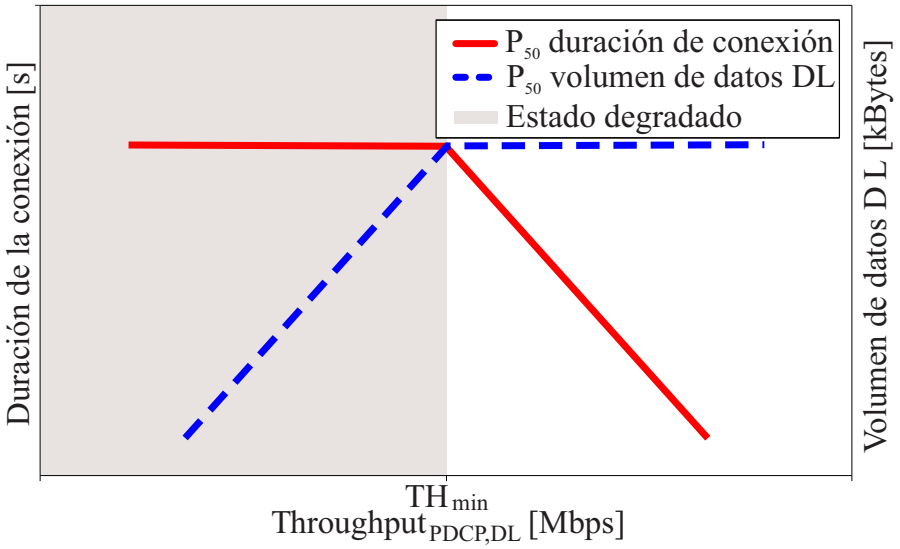


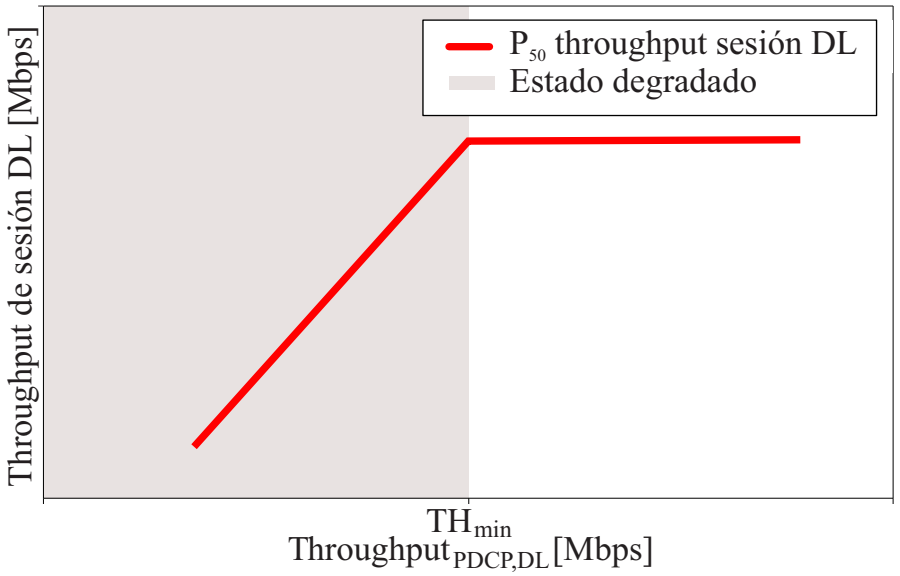
FIGURA 3.2: Modelo del comportamiento esperado de usuario para el servicio de VoLTE.

Este comportamiento se define en base a unas asunciones (es decir, el indicador con mayor impacto en el rendimiento del servicio y el indicador que mejor refleja la satisfacción de usuario) que posteriormente se validan haciendo uso de datos reales. La Figura 3.3 muestra la relación genérica esperada entre el QoS seleccionado y los indicadores de tráfico (es decir, $QoS_i^{(s)}(k)$ y $QoS_j^{(s)}(k)$) para cada clase de servicio.

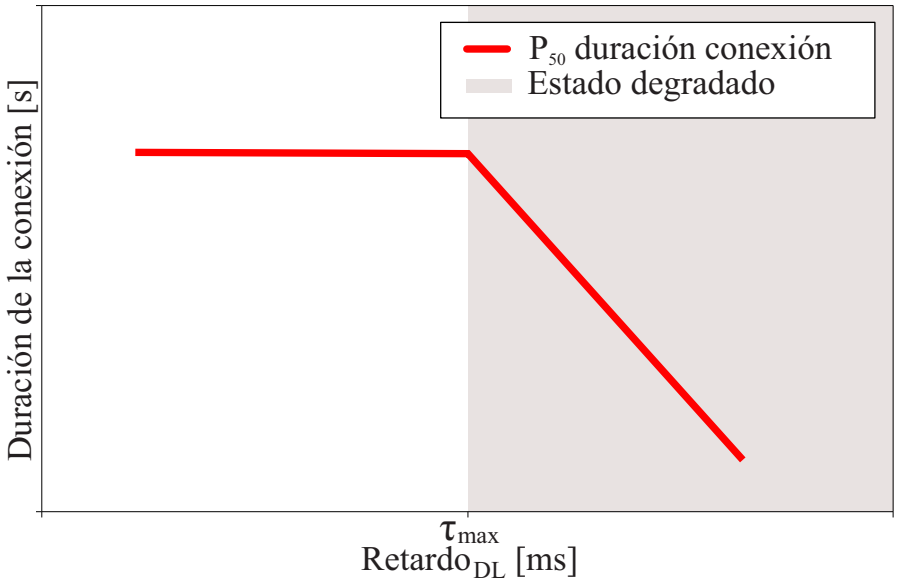
En los servicios de datos de búfer completo, Figura 3.3 (a), los datos están disponibles desde el inicio de la conexión, por lo que el patrón de tráfico asociado se basa en unas pocas ráfagas de gran duración en las que los datos se transmiten a gran velocidad. Por tanto, se requieren tantos recursos como sea posible hasta que todos los datos se han transmitido. En este tipo de servicios, se asume que el usuario tiende a interrumpir la conexión si el tiempo de descarga excede un determinado umbral. Dicho comportamiento debería reflejarse tanto en la duración como en el volumen de tráfico de la conexión, como se muestra en la Figura 3.3 (a). El eje x representa la media del *throughput* PDCP DL de la conexión, medido considerando solo los TTIs activos (excluyendo el último), que se considera el indicador de QoS que mayor impacto tiene en la QoE de estos servicios [76]. El eje y primario representa la duración de la conexión, mientras que el eje y secundario representa el volumen de datos DL por conexión (ambos usados para modelar el comportamiento del usuario). La línea sólida representa la mediana de la distribución de la duración de la conexión, mientras que la línea discontinua



(a) Modelo del comportamiento de usuario para servicios de datos de búfer completo.



(b) Modelo del comportamiento de usuario para servicios streaming.



(c) Modelo del comportamiento de usuario para VoLTE.

FIGURA 3.3: Modelo del comportamiento de usuario esperado para las principales clases de servicios.

representa la mediana de la distribución del volumen total de datos DL. Para mayor claridad, el área sombreada, etiquetada como estado degradado, engloba aquellas conexiones con condiciones de enlace inaceptables para el usuario y, por tanto, tienen una mayor probabilidad de ser interrumpidas. Como se observa en la figura, se espera que los usuarios intenten mantener la conexión hasta que se supera un determinado umbral en la duración de la conexión. Si las condiciones de enlace mejoran, la duración de la conexión se reduce ya que los datos se transmiten más rápido, como se observa en la zona no degradada en la figura. Por el contrario, el volumen de datos por conexión se mantiene constante, ya que no se ve afectado por el estado de la conexión a partir de un determinado punto. Por tanto, el umbral de QoS mínimo, TH_{min}, en servicios de datos de búfer completo se estima como el valor de la media del *throughput* PDCP DL a partir del cual la duración de la conexión se mantiene constante y el volumen de datos descargados decae.

Los servicios de streaming, Figura 3.3 (b), también están afectados por el *throughput* de usuario, por lo que el indicador de QoS seleccionado es de nuevo el *throughput* PDCP DL. Sin embargo, el comportamiento de usuario esperado es totalmente diferente en cuanto a duración y volumen de datos de la conexión. Las sesiones de streaming consisten en largas conexiones con un gran volumen de datos, pero distribuido en muchas ráfagas. A diferencia de los servicios de datos de búfer completo, los servicios de streaming son flexibles, por lo que una buena condición del enlace no resulta necesariamente en una reducción de la duración de la sesión. Por tanto, la duración de la conexión podría no ser un buen indicador de QoS para reflejar el comportamiento del usuario. Por contra, el *throughput* de sesión DL, obtenido mediante la división del volumen total de datos DL por la duración de la conexión (incluyendo los periodos de silencio), podría indicar la calidad de la descarga. La Figura 3.3 (b) muestra el comportamiento de usuario genérico esperado en este tipo de servicios, representando la relación entre el *throughput* PDCP DL y el de sesión DL. La línea solida representa la mediana del *throughput* de sesión y el área sombreada define el estado degradado. Como se observa, en dicho estado, el *throughput* de sesión decrece al igual que el *throughput* PDCP DL. Cuando el *throughput* PDCP DL es suficientemente bueno, el *throughput* de sesión se mantiene constante, reflejando que el segundo no está condicionado por el primero. Por tanto, el umbral mínimo de QoS, TH_{min}, para servicios de streaming es el valor del *throughput* PDCP DL que hace que la mediana del *throughput* de sesión empiece a decaer si dicho valor se reduce.

En un servicio VoLTE, Figura 3.3 (c), la duración de la conexión es el indicador más representativo para modelar el comportamiento de usuario. Sin embargo, a diferencia del servicio de datos de búfer completo, el indicador de QoS que mayor impacto tiene en la QoE, es el retardo de paquete. La Figura 3.3 (c) muestra el modelo del comportamiento genérico de usuario en VoLTE representando la variación de la duración de la conexión causado por cambios en el retardo de paquete DL. Al igual que en las figuras anteriores, la línea sólida representa la mediana de la duración de la conexión y el área sombreada el estado degradado. Se observa que la mediana de la duración debería caer cuando el retardo de paquete DL aumenta por encima de cierto límite. Por tanto, el umbral mínimo de QoS, τ_{max}, para VoLTE es el valor de la media del retardo de paquete DL a partir del cual la media de la duración de la conexión decae cuando dicho valor es superado.

Pese a presentar un modelo de comportamiento que podría englobar a los principales servicios ofrecidos en una red móvil, se espera que, en redes reales, algunos servicios podrían no estar completamente representados por los tres casos explicados anteriormente. Por ejemplo, un servicio web o aplicaciones para el uso de redes sociales podrían mostrar un comportamiento diferente dependiendo del tamaño de los objetos intercambiados durante la conexión. Algo similar puede suceder para servicios de streaming en tiempo real, viéndose afectados por requisitos de latencia mucho más estrictos. Sin embargo, la metodología propuesta para modelar el comportamiento de los tres servicios presentados anteriormente puede extenderse a otros servicios para identificar las relaciones entre el comportamiento del usuario y las condiciones de la red.

3.4. Análisis de rendimiento

En esta sección se valida el método anteriormente descrito para estimar los umbrales de QoS a nivel de servicio haciendo uso de un conjunto de trazas de conexión radio tomadas de diferentes redes LTE reales. Para ello, en primer lugar, se describen los datos usados y su clasificación y, posteriormente, los resultados obtenidos en la obtención de umbrales. Finalmente, se describen algunas consideraciones de implementación.

Tabla 3.1: Clasificación de servicios de las trazas de conexión.

Servicio	**Conjunto 1**	**Conjunto 2**	**Conjuntos combinados**
Datos de búfer completo	1477 (3.04 %)	3228 (31.89 %)	4,705 (8 %)
Streaming	1988 (4,08 %)	952 (9.4 %)	2,940 (5 %)
VoIP	20,582 (42.28 %)	-	20,582 (35 %)
Navegación web (elementos grandes)	1808 (3.71 %)	1132 (11.18 %)	2,940 (5 %)
Navegación web (elementos pequeños)	22,828 (46.89 %)	4811 (47.53 %)	27,639 (47 %)

3.4.1. Metodología experimental

La validación se realiza haciendo uso de dos conjuntos de datos independientes generados a partir de trazas anónimas obtenidas en la interfaz radio de dos sistemas LTE diferentes. Ambos sistemas están lo suficientemente maduros como para cubrir los diferentes valores de QoS que, posteriormente, se utilizarán para derivar los correspondientes umbrales de QoS. El conjunto 1 se obtiene de 1.960 celdas LTE que cubren un área urbana de 3.900 km^2. Específicamente, las trazas se recopilan durante un periodo de dos horas (desde las 10:00 am a las 12:00 pm) produciendo un total de 48.683 conexiones, de las cuales el 42 % se han categorizado como QCI 1 y 58 % en el rango de QCIs 6-9. Por otro lado, el conjunto 2 se obtiene desde las 10:00 hasta las 11:00 am en un total de 145 celdas LTE, cubriendo un área urbana de 125 km^2. Durante ese periodo, se generan un total de 10.123 conexiones, todas ellas etiquetadas en el rango de QCIs entre el 6 y 9.

Las trazas obtenidas se procesan para obtener los descriptores de cada conexión requeridos para la posterior clasificación del tráfico, descritos en la Sección 3.3. Posteriormente, las conexiones son clasificadas haciendo uso del método de aprendizaje no supervisado descrito en la Sección 3.3.1. La Tabla 3.1 recoge el resultado de la clasificación de las trazas de conexión para cada conjunto por separado y el resultado final de combinar ambos. Tras la clasificación, el 8 % de las conexiones son clasificadas como servicios de datos de búfer completo, el 5 % se etiquetan como servicios de streaming, el 35 % se clasifican como VoIP, el 5 % como navegación web de páginas con grandes elementos (p. ej., grandes imágenes, vídeos ...) y el 47 % como navegación web de páginas con pequeños elementos o redes sociales.

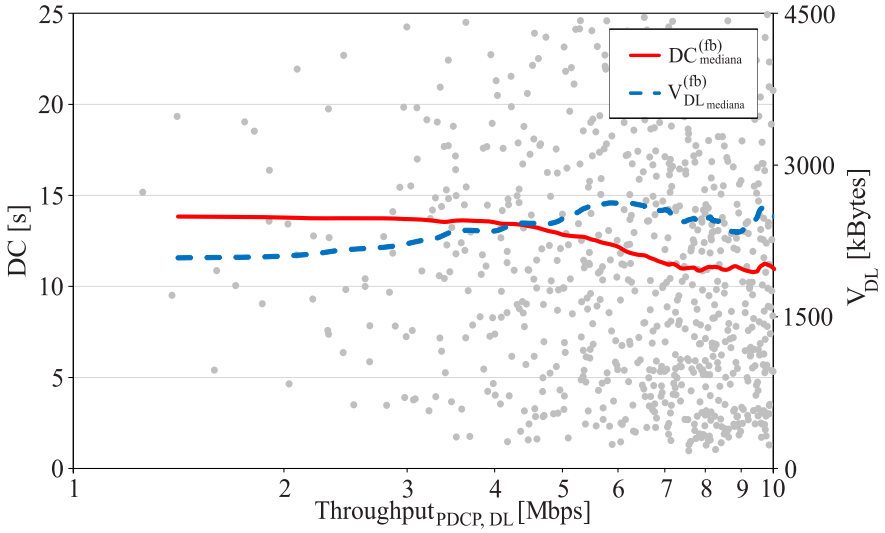


FIGURA 3.4: QoS e indicadores de tráfico para los servicios de datos de búfer completo.

3.4.2. Resultados

La Figura 3.4 muestra el análisis de las conexiones de datos de búfer completo. Cada punto en la figura representa una conexión clasificada como servicio de este tipo. La línea sólida representa la mediana de la duración de la conexión, $DC^{(fb)}_{mediana}$, y la línea discontinua representa la mediana del volumen de datos DL, $V^{(fb)}_{DL_{mediana}}$. El eje x, que representa el *throughput* PDCP DL, está ajustado a valores bajos (menos de 10 Mbps) para identificar el límite entre los dos estados especificados en la Figura 3.3 (a). Los resultados confirman el impacto esperado en el comportamiento de usuario, ya que, para un menor valor de *throughput* PDCP DL, el volumen de datos DL disminuye y la duración de la conexión se mantiene. El mínimo umbral de QoS se puede determinar como el valor $TH^{(fb)}_{PDCP,DL}$ que causa que $DC^{(fb)}_{mediana}$ caiga y $V^{(fb)}_{DL_{mediana}}$ se mantenga constante. De la figura, se estima que el umbral para este servicio específico en la red bajo estudio es $TH^{(fb)}_{PDCP,DL,min} \approx 5$ Mbps.

La Figura 3.5 muestra el análisis para los servicios de streaming. Cada punto en la figura representa una conexión de dicho servicio. La línea sólida indica la mediana del *throughput* de sesión DL, $TH^{(str)}_{sesión,DL_{mediana}}$. Los resultados muestran que $TH^{(str)}_{sesión,DL}$ presenta una tendencia similar al comportamiento esperado en la Figura 3.3 (b). Por tanto, el umbral mínimo de QoS para este servicio se puede fijar como el valor de $TH^{(str)}_{PDCP,DL}$ que produce un máximo en el $TH_{sesión,DL}$. En este caso, $TH^{(str)}_{PDCP,DL,min} \approx 30$ Mbps.

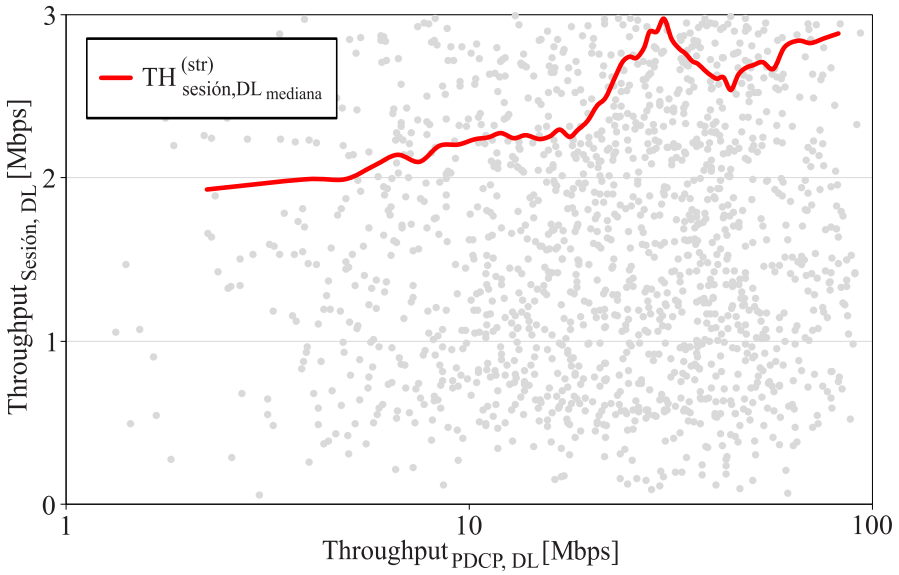


FIGURA 3.5: QoS e indicadores de tráfico para los servicios de streaming.

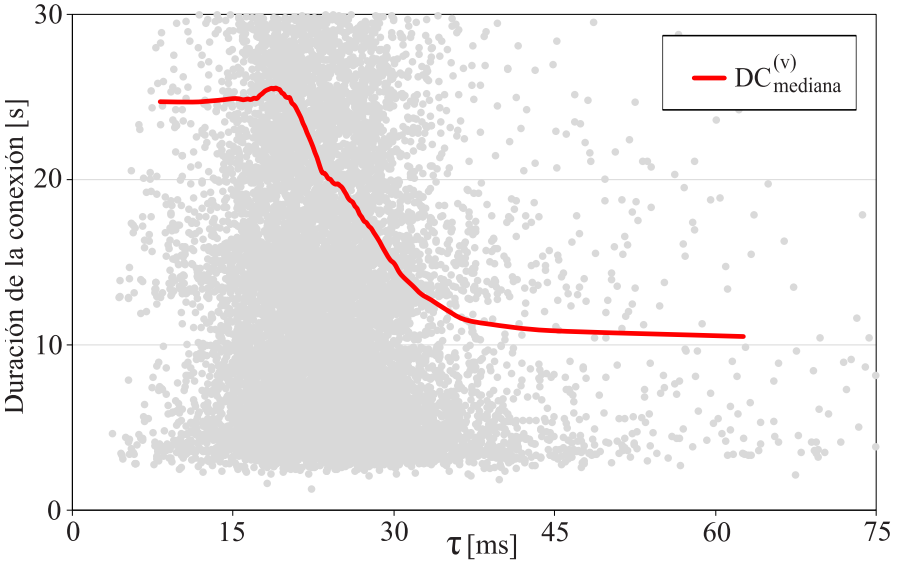


FIGURA 3.6: QoS e indicadores de tráfico para el servicio VoLTE

La Figura 3.6 muestra el análisis de VoLTE. Cada punto en la figura representa una conexión VoLTE. La línea sólida, que define $DC^{(v)}_{mediana}$, confirma el impacto de los usuarios previamente descrito en la Figura 3.3 (c). De la figura, se puede inferir que el umbral de retardo de paquete DL máximo es $\tau^{(v)}_{max} \approx 20$ ms .

La Figura 3.7 muestra el análisis obtenido para servicios de navegación web en páginas con grandes objetos. Cada punto en la figura representa una conexión etiquetada como navegación web con esta particularidad. La línea sólida representa $DC^{(wg)}_{mediana}$, y la línea discontinua $V^{(wg)}_{DL_{mediana}}$. Generalmente, este servicio debería

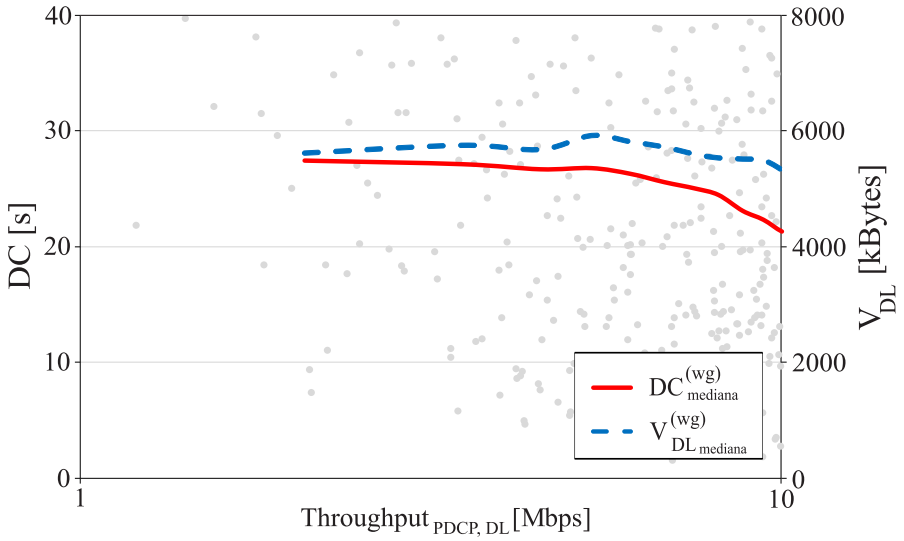


FIGURA 3.7: QoS e indicadores de tráfico para el servicio de navegación web (objectos grandes)

presentar un comportamiento de usuario similar al de servicios de datos de búfer completo. Sin embargo, el volumen de datos DL parece no verse claramente afectado por cambios en el *throughput* PDCP DL. La razón principal se basa en el hecho de que, en una sesión web, la cantidad de datos que son transferidos es menor que en servicios de datos de búfer completo y, por tanto, las condiciones de enlace deben ser mucho peor para que el usuario perciba esta degradación. En base a los datos disponibles para realizar este análisis, el valor del umbral mínimo de QoS para este servicio no se puede obtener.

Por último, la Figura 3.8 muestra el análisis de la navegación web con objetos pequeños y redes sociales. Cada punto en la figura representa una conexión identificada como uno de estos servicios. La línea sólida representa $DC^{(wp)}_{mediana}$ y la línea discontinua $V^{(wp)}_{DL_{mediana}}$. Se observa que $DC^{(wp)}_{mediana}$ y $V^{(wp)}_{DL_{mediana}}$ no presentan cambios independientemente de los valores de *throughput*. Esto se debe al hecho de la cantidad de datos que se genera en este tipo de servicios es muy pequeña para cada conexión. Como consecuencia, la satisfacción de usuario se basa más en una transferencia de datos satisfactoria que en la duración de la conexión. Por tanto, solo una condición de enlace extremadamente mala podría impactar DC. En base a esto, no puede estimarse un valor de $TH^{(wp)}_{PDCP,DL,min}$

La Tabla 3.2 recoge los resultados de la estimación para todos los servicios evaluados.

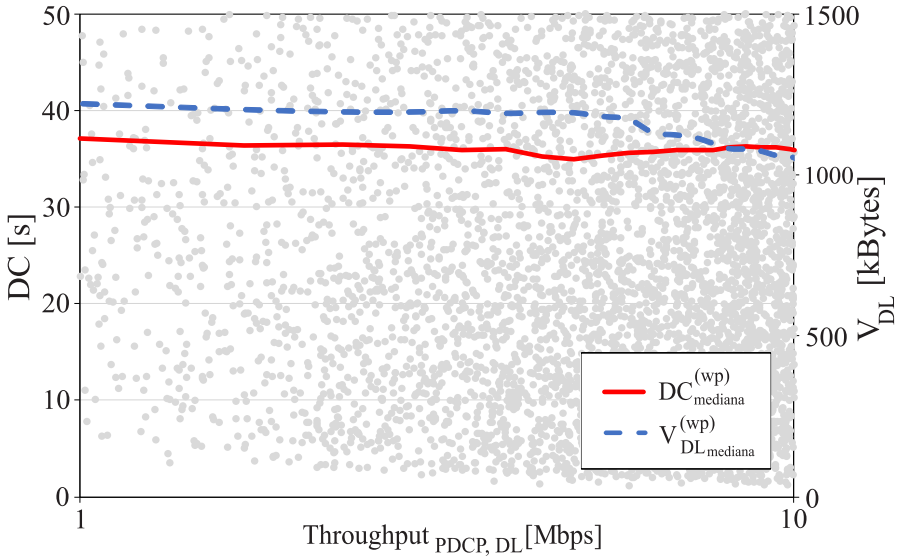


FIGURA 3.8: QoS e indicadores de tráfico para el servicio de navegación web (objectos pequeños)

TABLA 3.2: Umbrales de QoS estimados por servicio.

Servicio	**Indicador de QoS**	**Umbral**
Datos de búfer completo	*Throughput* PDCP DL	5 Mbps
Streaming	*Throughput* de sesión DL	30 Mbps
VoLTE	Retardo de paquete DL	20 ms
Navegación web (elementos grandes)	*Throughput* PDCP DL	-
Navegación web (elementos pequeños)	*Throughput* PDCP DL	-

3.4.3. Consideraciones de implementación

El método de obtención de umbrales se diseña como un esquema centralizado que puede integrarse en un sistema de soporte de operaciones (Operations Support System, OSS) en la red. Debido a su sencillez, el coste computacional es relativamente bajo. La complejidad en tiempo tiene un aumento lineal en función del número de trazas analizadas. En la práctica, el proceso que más tiempo requiere es el preprocesado de las trazas, que puede realizarse con herramientas de procesamiento de trazas (proporcionadas por proveedores de OSS) y el proceso de clasificación, el cual se realiza haciendo uso de un algoritmo no supervisado y que puede implementarse, junto con el resto del método, en cualquier lenguaje de programación (en este caso, Matlab [87]). Específicamente, el tiempo total de

ejecución para el conjunto de datos utilizados en un portátil con un procesador de cuatro núcleos de 2,6 GHz, es menos de 5462 s (92 s por cada 1.000 conexiones).

3.5. Conclusiones

En este capítulo, se ha propuesto un método automático para la estimación de umbrales de QoS usados en funciones de utilidad de usuario a nivel de servicio. El método se basa en trazas de conexión obtenidas de un sistema LTE real. A diferencia de las funciones de utilidad clásicas basados en umbrales de QoS obtenidos en pruebas de laboratorio, los umbrales de QoS obtenidos mediante este método permiten lidiar con una gran variedad de sistemas y factores humanos, reflejando la experiencia de usuario en base a las condiciones reales de la red.

En la primera etapa, las trazas de conexión se clasifican en grupos de aplicaciones en base al QCI y a diferentes descriptores de tráfico registrados por conexión. A continuación, el umbral de QoS mínimo se infiere a nivel de servicio analizando el indicador de QoS que mayor impacto tiene en la experiencia de usuario y el indicador de tráfico que mejor refleja el comportamiento de usuario.

El método se ha probado con trazas recopiladas de dos redes LTE reales, obteniendo un valor mínimo de *throughput* de usuario DL de 5 Mbps para servicios de datos de búfer completo, 30 Mbps para servicios de streaming y un retardo máximo de paquete DL de 20 ms para servicios de VoLTE. Para los servicios de navegación web (tanto páginas con objetos grandes como pequeños) no se ha podido definir un umbral mínimo, ya que estos servicios no requieren de condiciones de red tan exigentes como el resto de los servicios. Por tanto, para poder observar el comportamiento de usuario esperado, las condiciones de red deberían ser mucho peores. El hecho de que las redes seleccionadas garanticen unos estándares de calidad extremadamente buenos impide tener un número suficiente de conexiones degradadas, que permitan derivar los umbrales de satisfacción. Para solventar esta limitación, deben buscarse conjuntos de datos de redes de operadores menos exigentes.

Este método puede automatizarse totalmente, eliminando la necesidad de pruebas subjetivas que, por lo general, requieren de mucho tiempo y un alto coste para realizarse. Por otro lado, gracias a su baja carga computacional, puede ejecutarse

periódicamente para detectar cambios de tendencia en el usuario. Alternativamente, el análisis puede extenderse a redes 5G y sistemas de Internet de banda ancha por satélite para evaluar el impacto de las capacidades de la red en el comportamiento general de usuario.

Capítulo 4

Monitorización y análisis del tráfico para la gestión de QoE en redes móviles

En este capítulo se revisan las carencias y los principales retos del uso de aplicaciones de monitorización y análisis de tráfico (*Traffic Monitoring and Analysis*, TMA) para la gestión automática de la QoE en redes móviles. A diferencia del método presentado en el Capítulo 3, donde se busca ajustar de forma automática los modelos de QoE existentes mediante umbrales de QoS basados en indicadores de rendimiento de recursos de la red (*Resource Key Performance Indicators*, R-KPIs), el uso de soluciones TMA permiten construir de manera flexible nuevos indicadores de rendimiento específicos de servicio (*Service Key Performance Indicators*, S-KPIs), que pueden usarse para operaciones de gestión de QoE más complejas.

Este capítulo se divide en 4 secciones. La Sección 4.1 describe la estructura de una solución TMA genérica para grandes volúmenes de datos en redes móviles. Posteriormente, se propone una posible metodología para la validación de este tipo de aplicaciones tras su despliegue en la Sección 4.2. En la Sección 4.3, se presentan diferentes casos de uso reales utilizando este tipo de aplicaciones. El objetivo es mostrar el potencial de las aplicaciones TMA para el análisis y la gestión de QoE en redes móviles reales. Por último, se exponen conclusiones de este capítulo en la Sección 4.4.

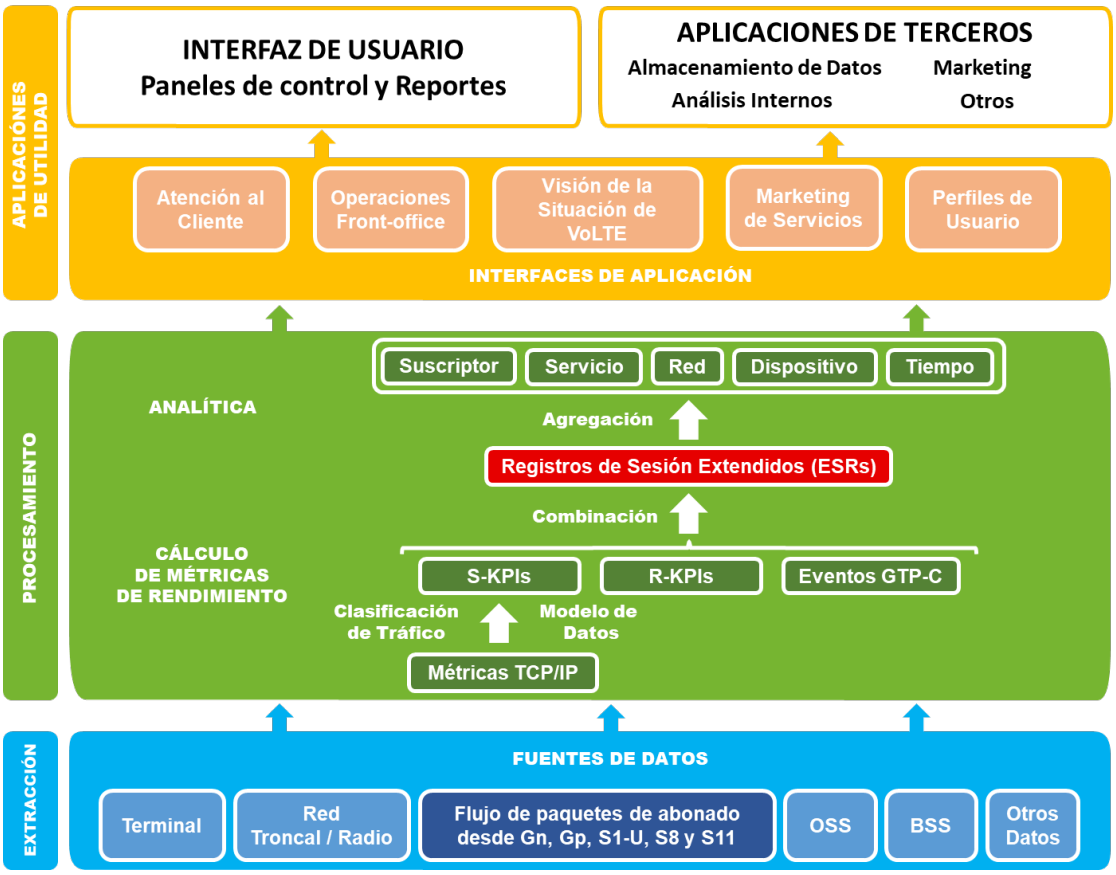


FIGURA 4.1: Esquema básico de una aplicación TMA para la monitorización de la QoE.

4.1. Estructura de una aplicación TMA para la gestión de la QoE

En esta sección, se describe la estructura básica de una aplicación genérica de TMA para la monitorización y gestión de la QoE en redes móviles. Casi todos los componentes presentados en esta estructura pueden encontrarse en la mayoría de soluciones de TMA más actuales (p. ej., [88]). Dicha estructura, se muestra con más detalle en la Figura 4.1, donde se diferencian las tres principales capas que la componen.

a) *Extracción*: La monitorización del tráfico de la red se inicia capturando los paquetes generados en algunas de las principales interfaces de red. Para dicho proceso, a menudo se utiliza una interfaz de aplicación de programación (*Application Programming Interface*, API) conocida como *pcap* (packet capture) [89]. El tráfico capturado se clasifica a nivel de sesión y usuario

usando la dirección de protocolo de Internet (*Internet Protocol*, IP) y de la Identidad Internacional del Abonado Móvil (*International Mobile Subscriber Identity*, IMSI). Esta información se enriquece después con datos generados por los equipos de la red troncal y radio, los Sistemas de Soporte de Operación/Negocio (*Operating/Business Support Systems*, OSS/BSS) e, incluso, terminales móviles controlados por los propios operadores. Si está disponible, cierta información contextual (p. ej., hora del día, localización, información meteorológica o aspectos técnicos del propio dispositivo) podría añadir gran valor en esta etapa, pues permitiría mejorar los procesos de detección de fallos y optimización.

b) *Procesamiento*: Las entradas de la etapa de procesamiento consisten en métricas de rendimiento del Protocolo de Control de Transmisión (*Transmission Control Protocol*, TCP) como, por ejemplo, el *throughput* de la sesión IP, la tasa de pérdida de paquetes o el tiempo de ida y vuelta (*Round-Trip Time*, RTT) del paquete, calculados mediante el análisis de los diferentes paquetes intercambiados a nivel de sesión. Dichas métricas pueden agregarse a nivel de servicio clasificando los paquetes en función de la clase de servicio en uso. Actualmente, esta clasificación puede incluir desde servicios de conversación en tiempo real (p. ej., VoLTE, Skype, juegos online) y servicios de difusión (p. ej., televisión/radio por Internet), servicios de transmisión en diferido (p. ej., Youtube, Spotify), interactivos (p. ej., navegación web, redes sociales) hasta servicios de ejecución en segundo plano (p. ej., mensajería, intercambio de ficheros, servicios en la nube). Estos datos se enriquecen con eventos del Protocolo de tunelización GPRS en el Plano de Control (*GPRS Tunneling Protocol for Control Plane*, GTP-C) que aporta información de señalización entre el Nodo de Soporte del Servicio GPRS (*Serving GPRS Support Node*, SGSN), y el Nodo de Soporte de la Puerta de enlace GPRS (*Gateway GPRS Support Node*, GGSN) en la red troncal.

c) *Aplicaciones de utilidad*: El uso de Interfaces Gráficas de Usuario (*Graphical User Interfaces*, GUIs) con paneles de control y reportes permiten evaluar el estado de la QoE en la red. Por tanto, este tipo de herramientas suelen implementarse en la etapa final de la estructura. Existen múltiples escenarios donde podría hacerse uso de los datos generados en la etapa de procesamiento con fines tales como almacenamiento de dichos datos, análisis de uso

interno, o diferentes procesos de marketing. El objetivo es entender el comportamiento y expectativas del usuario para, así, poder crear promociones y planes de tarificación más efectivos que permitan, en última instancia, mejorar la satisfacción del usuario y, por tanto, evitar una posible pérdida de clientes [90].

En la Figura 4.1 se observa el flujo de procesamiento de los datos. El flujo se inicia monitorizando, de forma pasiva, los enlaces más importantes de la red. Este proceso debe estar totalmente estandarizado para evitar, en la medida de lo posible, la interacción de los distintos equipos que componen la red. Dicha monitorización se realiza mediante el uso sondas conectadas en diferentes puntos de la red. Éstas deben ubicarse, idealmente, en las principales interfaces de la red troncal, ya que son los puntos donde se agrega la mayor parte del tráfico de toda la red. En redes 3G, la conexión se realiza en el enlace *Gn* entre el SGSN y el GGSN, capturando datos tanto en el plano de usuario como de control. Dicha interfaz permite monitorizar, de forma precisa, el rendimiento de la conexión extremo a extremo. En el caso de las redes 4G y las primeras redes radio *5G non-standalone* (es decir, aquellas que comparten la red troncal con sus predecesoras), los puntos clave de monitorización se localizan en la interfaz *S11* entre el elemento principal de gestión del acceso de los usuarios (*Mobility Management Entity*, MME) y la puerta de enlace de servicio *Serving Gateway*, SGW) en el plano de control, y la interfaz *S1-U* entre el eNodeB y SGW en el plano de usuario. El uso de estas interfaces es clave para una correcta clasificación del tráfico a nivel de usuario. Además, esta información se enriquece con datos de otro tipo de interfaces como, *Gp* en 3G y *S8* en 4G para los escenarios de movilidad. Del mismo modo, pueden monitorizarse diferentes interfaces del subsistema multimedia IP (*IP Multimedia Subsystem*, IMS), como por ejemplo el enlace *Mm*, para recopilar mensajes intercambiados entre el núcleo del IMS y redes IP externas. Toda esta información se combina con indicadores de rendimiento obtenidos de distintos equipos que componen la red, así como de indicadores más precisos de las trazas de conexión.

Posteriormente, durante la etapa de procesamiento, se utiliza la gran cantidad de datos recopilada de la etapa anterior para generar nueva información más específica para aplicarse en procesos de gestión de la QoE. Estos datos comprenden la siguiente información a nivel de abonado: a) flujo de paquetes de las principales interfaces de la red, y b) contadores y archivos de trazas con eventos de señalización, generados por los nodos en la red troncal y radio. A la salida de esta etapa,

se construyen los llamados Registros de Sesión Extendidos (*Extended Session Records*, ESRs) [90]. Estos registros periódicos son generados a nivel de servicio para cada abonado/usuario, combinando los S-KPIs con los diferentes elementos de red involucrados en la sesión. Esta etapa se compone de diferentes fases, desde que se captura el tráfico hasta que se generan los ESRs.

En las primeras fases de la etapa de procesamiento, el tráfico capturado se clasifica por servicio y se calculan diferentes métricas TCP/IP. Una vez el tráfico se obtiene a nivel de paquete, este puede clasificarse para determinar el servicio específico para cada sesión. En este punto, las métricas de rendimiento TCP/IP (a partir de ahora TCP) pueden calcularse de los datos extraídos previamente. Dicho cálculo es posible gracias al control de flujo TCP, que hace que la dinámica del tráfico esté relacionada con el rendimiento extremo-a-extremo de la conexión. Esto permite que puedan detectarse problemas en diferentes puntos de la comunicación TCP (p. ej., en la interfaz radio) analizando el tráfico monitorizado a lo largo del camino (p. ej., una sonda conectada en la interfaz de la red troncal). Las métricas TCP pueden dividirse también en función de la dirección del enlace (es decir, enlace descendente o ascendente) mediante el uso de la dirección IP (incluida también en la información capturada). Existe la posibilidad, incluso, de segregar el tráfico en función del segmento de la red en el que este es monitorizado (sonda-a-terminal o sonda-a-Internet) haciendo uso del contenido de los paquetes y de los mensajes de confirmación de recepción (ACK) en las distintas direcciones de la interfaz. Debido a la gran cantidad de información que se obtiene en la etapa de extracción, deben fijarse una serie de mecanismos para reducir la carga computacional en la etapa de procesamiento. Para ello, la clasificación del tráfico y el cálculo de las métricas TCP solo se realiza para un grupo determinado de servicios y ráfagas de datos lo suficientemente grandes.

Tras la clasificación del tráfico, se calculan los diferentes S-KPIs a nivel de usuario, sesión y servicio. Para ello, se buscan los eventos más característicos del servicio. Por ejemplo, en el caso de servicios web, se calcula el tiempo de descarga web como el tiempo entre el primer mensaje *Hypertext Transfer Protocol* (HTTP) GET enviado por el usuario y el último mensaje HTTP de respuesta enviado por el proveedor del servicio. La Figura 4.2 muestra el flujo de eventos originados desde que el usuario accede a la web (evento 1) hasta que la página se descarga completamente (evento 27). En este caso, el tiempo de descarga web se estima por la aplicación TMA como el tiempo entre el evento 8 (primer mensaje

HTTP GET a la página destino) y el evento 24 (último mensaje de respuesta del proveedor del servicio). Como resultado, la aplicación genera un registro con información de la sesión. La Figura 4.3 muestra un ejemplo de un registro generado por la aplicación para una sesión web. Este registro contiene información propia de la sesión (p. ej., inicio de la sesión, tipo de servicio, información del usuario ...), información relevante del tipo del servicio (p. ej., tipo de servicio, proveedor, protocolo ...) y los S-KPIs calculados (p. ej., tiempo de acceso web, volumen de datos descargados, tiempo de descarga ...). Los S-KPIs estimados se combinan con otros KPIs obtenidos de diferentes segmentos de la red (p. ej., radio o troncal). Toda esta información se sincroniza temporalmente y se incluye en uno o más ESRs segregados a diferentes niveles, tales como usuario, servicio, tecnología de acceso radio o proveedor de servicio. La Figura 4.4 muestra un ejemplo de un ESR generado por la aplicación TMA. En este ESR se incluye tanto la información de la sesión web de la Figura 4.3, como de otras sesiones que el mismo usuario ha iniciado durante la duración del ESR. En caso de que el servicio pueda clasificarse, la información de la sesión se recopila en una entrada específica dentro del ESR (p. ej., web o vídeo). Además, se incluyen diferentes métricas de todas las sesiones contenidas en el ESR (p. ej., *throughput* TCP, volumen de datos, pérdida de paquete ...). Como resultado, cada ESR contiene una gran cantidad de datos a nivel de suscriptor y sesión que pueden usarse para estimar el rendimiento de la conexión extremo-a-extremo. Estas estimas permiten identificar sesiones con valores de S-KPI inaceptables que afectan directamente la experiencia de usuario. Para que los datos sean manejables, los ESRs se agregan en periodos de tiempos (generalmente, 1-5 minutos) respetando la granularidad previamente indicada.

El conjunto de datos obtenidos por una aplicación TMA puede utilizarse, en fases posteriores, para generar información significativa que aporte valor a los operadores. La Figura 4.5 muestra un conjunto de técnicas multidisciplinares involucradas en este proceso que van desde una simple visualización para el análisis exploratorio de los datos, a los algoritmos más sofisticados de aprendizaje automático. El objetivo es construir diferentes modelos que permitan clasificar, caracterizar y predecir el comportamiento de cada sesión, seleccionando los factores más importantes en cada caso. La construcción de dichos modelos dependerá del escenario a abordar:

a) *Clasificación*: Estos modelos pueden utilizarse para agrupar las sesiones por servicios. Originalmente, este tipo de clasificaciones se realizaba analizando

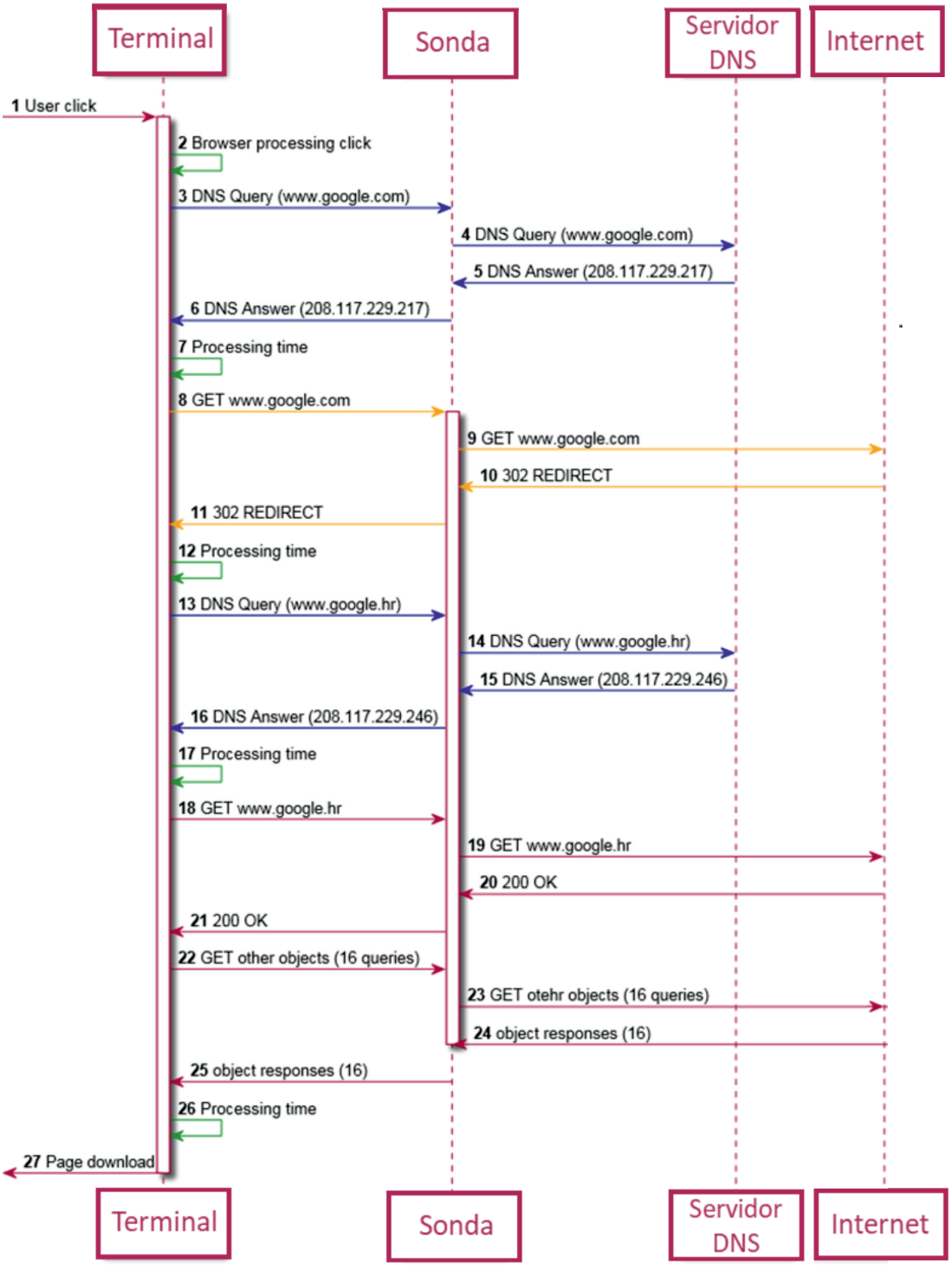


FIGURA 4.2: Estimación del tiempo de descarga web por la aplicación TMA.

```
{
    "pag": {
        "time": 1453108520.698826,
        "type": "wp"
    },
    "record": {
        "id": {
            "start_time": 1453108518656802,
            "imsi": "ANONYMOUS",
            "bearer": 5
        },
        "page_id": 5705374,
        "http": {
            "srv_host": "www.google.hr",
            "agent": "Mozilla\/5.0 (Linux; U; Android 4.1.2; en-gb; GT-I9305 Build\/JZO54K)"
        },
        "perf": {
            "dl_time": 2.042024,
            "resources": 10,
            "cached_resources": 0,
            "up_bytes": 6772,
            "down_bytes": 207497,
            "resources_ok": 10,
            "access_time": 0.129063,
            "max_tput": 5720476,
            "down_bytes_comp": 207299,
            "sessions": 7
        },
        "traffic": [
            {
                "facet": "protocol",
                "value": "HTTP"
            },
            {
                "facet": "functionality",
                "value": "web browsing"
            },
            {
                "facet": "service-provider",
                "value": "Google"
            }
        ]
    }
}
```


FIGURA 4.3: Ejemplo de registro para una sesión web.

el protocolo de los mensajes. Sin embargo, esto ya no es posible debido a que el tráfico está generalmente cifrado. Como alternativa, pueden analizarse las características del tráfico para segregar las sesiones. Esto se realiza mediante reglas heurísticas o algoritmos de agrupación automáticos basados en técnicas de aprendizaje no supervisado (p. ej., k-means) [80].

b) *Regresión*: La construcción de estos modelos permiten la estimación de S-KPIs a partir de métricas TCP. En primer lugar, pueden utilizarse terminales de pruebas en entornos de laboratorio para derivar la relación existente entre estos parámetros. Posteriormente, la construcción del modelo puede realizarse haciendo uso de técnicas de regresión clásicas (p. ej., regresión lineal generalizada) o algoritmos más complejos basados en aprendizaje supervisado (p. ej., *support vector machines*, redes neuronales o algoritmos de

```
{
    "start": 1453108518656,
    "dur": 18986,
    "imsi": "ANONYMOUS",
    "msisdn": "ANONYMOUS",
    "imeitac": "ANONYMOUS",
    "imeisvn": 1,
    "n_event": 28,
    "mcc": 219,
    "mnc": 10,
    "sources": [...],
    "first_event": "T_WEBPAGE",
    "bearers": {...},
    "kpi": {...},
    "kpivec": {
        "web": {
            "vec": [
                [
                    "3G",
                    17,
                    {
                        "service_provider": "Google",
                        "functionality": "web browsing",
                        "protocol": "HTTP"
                    },
                    [
                        0.129063,
                        1,
                        2.04202,
                        1,
                        207497
                    ]
                ],
                [...]
            ],
            "legend": [
                "wp_acc_time",
                "wp_acc_succ",
                "wp_time",
                "wp_succ",
                "wp_size"
            ]
        },
        "traf": {...},
        "tcp": {...},
        "video": {...}
    }
}
```


FIGURA 4.4: Ejemplo de ESR.

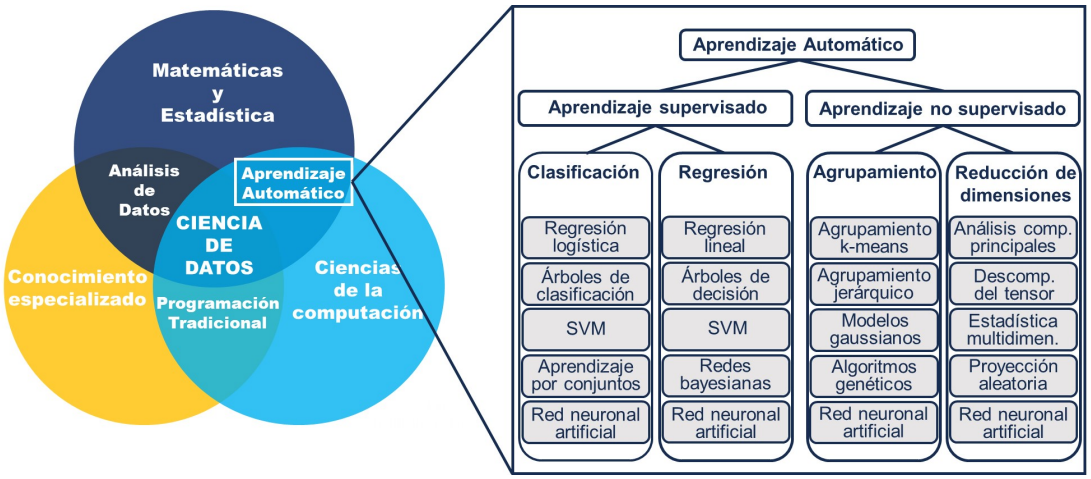


FIGURA 4.5: Taxonomía de algoritmos de aprendizaje automático.

aprendizaje por conjuntos) [91].

c) *Predicción*: Los modelos predictivos permiten anticipar tendencias en la QoE a lo largo del tiempo para poder tomar decisiones proactivas en base a diferentes criterios derivados de la experiencia de usuario. Esto puede realizarse mediante el análisis tradicional de series temporales (p. ej., ARIMA) o modelos de regresión basados en el aprendizaje supervisado (p. ej., *support vector machines* o redes neuronales). Estos últimos permiten hacer uso de datos históricos obtenidos de elementos de red similares para poder mejorar la precisión de la estimación.

d) *Reducción de variables*: Los procesos de reducción de variables permiten identificar aquellas métricas TCP más relevantes para cada S-KPI. De esta forma, la complejidad computacional de los modelos disminuye, acelerando el proceso de aprendizaje, haciéndolos más genéricos y facilitando la interpretación por parte del operador.

En estos procesos, los modelos estadísticos pueden utilizarse para aislar el efecto de una variable importante o asegurar que los modelos son interpretables. Por otro lado, las técnicas de ML (*Machine Learning*) tienen mayor popularidad cuando el número de predictores es mayor o se prima la precisión en la predicción.

4.2. Validación de una aplicación TMA

Durante la etapa de procesamiento de una aplicación TMA se realizan estimas de diferentes S-KPIs que, posteriormente, se combinan con otros parámetros para la construcción de los ESRs. Sin embargo, dichas estimaciones deben validarse para asegurar que reflejan, con precisión, el comportamiento del servicio. Para ello es necesario una metodología para validar dichas estimas en una aplicación TMA desplegada en el núcleo de red, utilizando como referencia las medidas realizadas en el extremo de la comunicación con aplicaciones instaladas en el terminal de usuario. Con ello, se pretenden afrontar los problemas surgidos por estimar indicadores de rendimiento extremo-a-extremo a partir de medidas en puntos intermedios de la conexión. A continuación, se describen los pasos de una metodología genérica que pueda utilizarse independientemente de los fabricantes de sondas, de los servicios a validar, de la tecnología radio, etc.

1. *Equipamiento para los experimentos.* Para realizar una validación precisa, en primer lugar, se requiere del uso de herramientas que permitan evaluar el rendimiento de la red en un extremo de la comunicación. Generalmente, estas herramientas son aplicaciones instaladas en terminales móviles comerciales que incluyen un analizador de S-KPIs (p. ej., SNACK, NPT, NEMO ...) junto con procesos automáticos que permiten simular interacciones humanas. Al estar localizados en un lado del canal de la comunicación, permiten medir, con precisión, la calidad extremo-a-extremo en base a diferentes S-KPIs tal como los percibe el usuario final.

2. *Definición de los experimentos.* Una vez se dispone del equipamiento necesario para realizar la validación, a continuación, se deben definir una serie de pruebas que permitan cubrir una gran variedad de escenarios que engloben los servicios más demandados, diferentes perfiles de usuario, diferentes condiciones de red (es decir, alta/baja interferencia, traspaso, alta/baja eficiencia espectral ...). Además, se deben repetir lo suficiente para obtener medidas en un largo periodo de tiempo (p. ej., un día). En cada ciclo, las pruebas se deberían realizar para cada servicio de forma aislada y secuencial, facilitando la separación de las estimas en la aplicación TMA en diferentes ESRs.

3. *Automatización de los experimentos.* La validación requiere del uso de datos obtenidos de los terminales móviles, así como de la aplicación TMA (es decir,

usuario e interfaz de red). Para ello, los experimentos definidos previamente se deben ejecutar en un proceso automático incluyendo los siguientes pasos. En primer lugar, se ejecutan automáticamente diferentes peticiones desde el terminal móvil simulando el comportamiento de usuario. Posteriormente, la aplicación TMA estima, a partir de los paquetes recopilados en la capa de red, el rendimiento extremo-a-extremo experimentado por el usuario. Finalmente, se evalúan los S-KPIs estimados por la aplicación TMA comparándolos con los valores reales obtenidos por la herramienta instalada en el terminal móvil.

4. *Alineamiento temporal entre componentes.* Una vez se han ejecutado los experimentos y la aplicación TMA ha capturado los datos derivados de dichas pruebas es necesario un proceso de ajuste. Con el fin de comparar las medidas obtenidas en la aplicación TMA y en el terminal móvil, se necesita realizar una corrección temporal para ajustar la diferencia de tiempo existente en los relojes de ambas interfaces. Se recomiendan tres condiciones para facilitar ese proceso. En primer lugar, los servicios deben ejecutarse de forma separada en cada ciclo y con una separación temporal suficiente para que cada prueba se recoja en ESRs diferentes. En segundo lugar, cada servicio solo puede ser evaluado una vez por ciclo con el fin de evitar que la información de dos ejecuciones distintas se incluya en un mismo ESR. De este modo, se evita que se enmascaren sesiones de mala calidad con otras de mejor rendimiento, distorsionando las estimas para el mismo servicio. Por último, para facilitar el alineamiento temporal, se hace uso de una conexión ficticia realizada por un servicio que permita definir claramente el inicio de las pruebas. En este caso, se suele escoger un servicio con un volumen de datos TCP que esté muy por encima del resto de servicios a evaluar (p. ej., una conexión FTP). La Figura 4.6 muestra una propuesta para este paso. El eje x representa un eje temporal en el que se distingue el momento en el que se inicia cada ciclo de pruebas. El eje y representa el volumen TCP medido por la aplicación TMA y reportado en el ESR. La línea sólida azul representa el volumen TCP medido en el enlace descendente, mientras que la línea verde representa el volumen TCP medido en el enlace ascendente. Se distinguen 3 ciclos de pruebas diferentes, separados varios minutos entre ellos. Cada ciclo empieza con un servicio (servicio 1) que genera un volumen TCP UL muy superior al del resto de servicios ejecutados. Posteriormente, se ejecuta un servicio (servicio 2) que genera un volumen TCP DL, de nuevo, destacando sobre el resto. Por último, se ejecutan los servicios (servicios 3 y

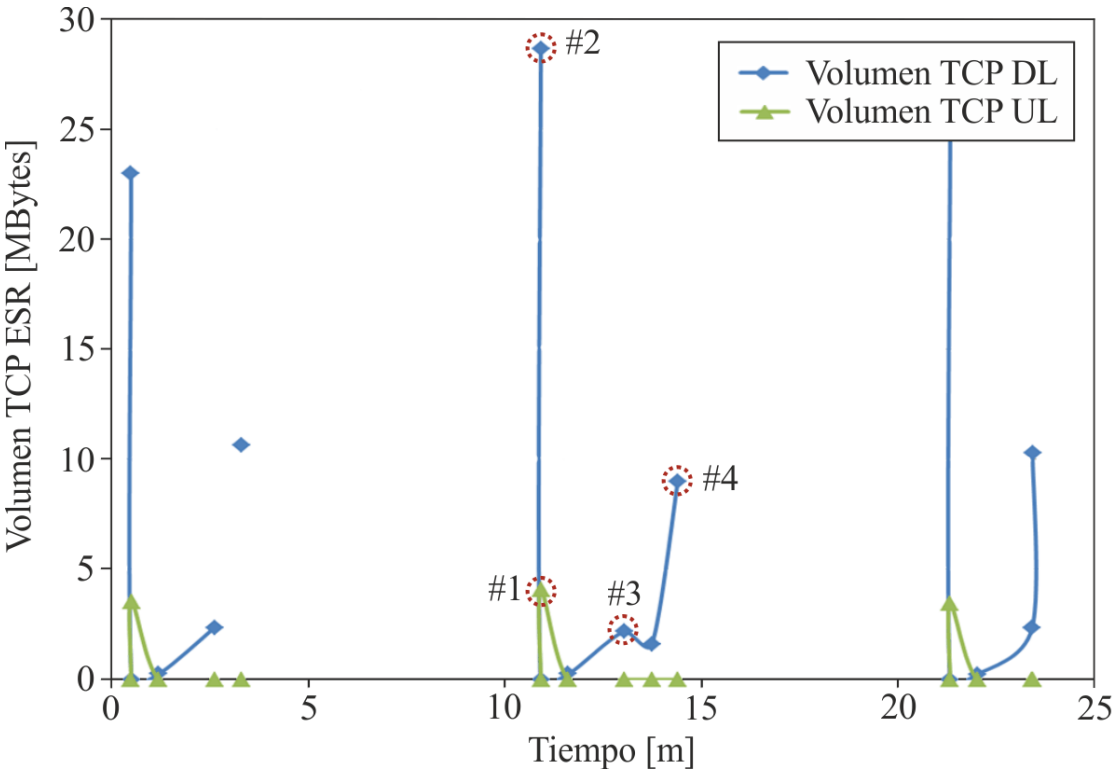


FIGURA 4.6: Alineamiento temporal de la aplicación TMA en el proceso de validación.

4) a evaluar para identificar los S-KPIs estimados para cada servicio y poder validarlos. Por tanto, la finalidad de los servicios 1 y 2 es la de determinar el inicio del ciclo, asegurando que son suficientemente característicos como para diferenciarse del resto de servicios a evaluar.

5. *Calibración de la aplicación TMA*. En el proceso de validación es importante tener en cuenta cómo se definen los S-KPIs calculados por ambas partes. Habitualmente se asume que ambas aplicaciones realizan los mismos cálculos. Sin embargo, esto puede no ser cierto para todos los indicadores. Por ejemplo, la Figura 4.2 muestra la diferencia del tiempo de descarga web estimado por la aplicación TMA con respecto al proceso real percibido por el terminal móvil. Las aplicaciones instaladas en los terminales móviles generalmente consideran el tiempo de resolución del dominio y el inicio de la comunicación TCP en el cálculo del tiempo de descarga web (eventos 2-7 en la figura). Sin embargo, una aplicación TMA no puede considerar estos tiempos ya que solo detecta el tipo de servicio una vez se manda el primer mensaje HTTP, lo que generalmente sucede cientos de milisegundos después. Así mismo, la aplicación TMA es totalmente ajena a los diferentes procesos ejecutados en el terminal móvil y que dependen de múltiples factores (p. ej., el modelo del

dispositivo, sistema operativo, la antigüedad ...). Para poder mitigar estas diferencias, puede analizarse el tráfico generado por los terminales móviles para identificar cómo se calculan los diferentes S-KPIs. Posteriormente, pueden aplicarse correcciones en la estimación de los S-KPIs, por la aplicación TMA, añadiendo cierto tiempo extra que compense esas diferencias.

Otro escenario habitual sucede con el uso de páginas web dinámicas, que incluyen objetos dinámicos (p. ej., anuncios). Uno de estos objetos puede interpretarse, de forma errónea por la aplicación TMA, como parte de la página web haciendo que los valores de los S-KPIs estimados no sean correctos. Estos riesgos pueden mitigarse configurando los parámetros de la aplicación para identificar diferentes sesiones cuando recibe dos ráfagas de datos consecutivas. Generalmente, este tipo de acciones correctivas permite incrementar la precisión en las medidas y estimas.

Finalmente, el proceso de validación no concluye con la evaluación de los valores estimados de los S-KPIs, sino también con el ajuste de los parámetros de configuración internos de la propia aplicación TMA (p. ej., duración máxima de un ESR, valores mínimos de una ráfaga para ser considerada por la aplicación ...). Estas acciones correctivas son clave antes de desplegar la aplicación en cualquier red comercial.

4.3. Descripción de casos de uso de una aplicación TMA

Existen una gran variedad de usos de una aplicación TMA en el contexto de las comunicaciones móviles, evidenciando el gran potencial que este tipo de herramientas tienen en el análisis y la gestión de QoE para los operadores. A continuación, se presentan diferentes casos de usos de dicha aplicación, desplegada en una red real. En primer lugar, se describe el uso de una aplicación TMA para la construcción de modelos para la estimación de S-KPIs con los datos generados por dicha herramienta. El segundo caso de uso cubre la construcción de modelos predictivos con técnicas de ML para anticipar posibles degradaciones del servicio. Posteriormente, se describe un tercer caso de uso en herramientas que proporcionen reportes que permitan conocer el estado general de la red mediante la evaluación de diferentes métricas de rendimiento de la QOE. Por último, se explica cómo

una aplicación TMA puede utilizarse para la identificación y el diagnóstico de conexiones problemáticas en redes celulares.

4.3.1. Construcción de un modelo para la estimación de S-KPIs

Como se ha definido anteriormente, una de las operaciones durante la fase de procesamiento en las aplicaciones TMA es el cálculo de diferentes S-KPIs a partir de las métricas TCP recopiladas en la fase anterior. En muchos casos, estos indicadores pueden inferirse directamente de la información capturada en esta capa de protocolo. Por ejemplo, el tiempo de descarga web es calculado utilizando los mensajes HTTP intercambiados entre el terminal y el proveedor de servicio. En otros casos, se requieren procesos más complejos para realizar dicho cálculo (p. ej., creación de modelos analíticos). Mediante un análisis descriptivo, puede definirse la relación existente entre los S-KPIs medidos por el analizador instalado en el terminal móvil usado en las pruebas de validación de la aplicación TMA y las métricas TCP capturadas por las sondas. Esto permite la creación de modelos para estimar dichos S-KPIs e incluir sus valores en los registros generados a la salida de la aplicación.

Para este caso de uso, se describe el proceso para la creación de un modelo de regresión polinómico multivariable que permite la estimación del tiempo inicial de búfer, uno de los S-KPIs más importantes del servicio de vídeo streaming. Como entrada del modelo, se proporcionan las siguientes métricas TCP recopiladas tras la fase de extracción de la aplicación TMA:

a) La tasa de pérdida de paquetes en el enlace descendente.

b) El tiempo medio de ida y vuelta (RTT) calculado desde el terminal a la sonda.

c) El tiempo medio de ida y vuelta (RTT) calculado desde la sonda a Internet.

d) El tiempo medio total de ida y vuelta (RTT) calculado desde el terminal a Internet.

e) El *throughput* medio en el enlace descendente de la sesión de vídeo.

f) El *throughput* medio en el enlace descendente sin considerar la fase lenta de inicio de la conexión TCP.

g) La tasa media de transmisión de vídeo.

h) El volumen de datos transmitido en el enlace descendente.

A la salida, el modelo proporciona una estimación del tiempo inicial de búfer para cada sesión de vídeo.

La construcción del modelo se divide en dos fases claramente diferenciadas. En una primera fase, se utiliza un método univariante de filtrado de variables basado en la prueba chi-cuadrado ([92]) para seleccionar las métricas TCP más relevantes para la estima del S-KPI. Posteriormente, se emplea un modelo de regresión polinómico multivariable para la construcción del modelo [91]. Finalmente, se realiza un proceso automático de optimización mediante el ajuste de los parámetros de configuración de cada fase, permitiendo seleccionar la combinación de número de variables y grado polinómico que mejor estima del tiempo inicial de búfer produce.

Para la construcción y evaluación del modelo, se usan los datos obtenidos por una aplicación TMA desplegada en una red LTE real. En este proceso, la aplicación captura el tráfico generado por diferentes terminales de pruebas, con diferentes condiciones radio, durante la fase de validación de dicha aplicación. Para ello, cada terminal simula el consumo de un servicio de vídeo streaming y calcula el tiempo inicial de búfer usado para entrenar el modelo. Tras el proceso de validación de la aplicación, se recopilan un total de 1.739 conexiones de vídeo streaming generadas por los diferentes terminales. Este conjunto de datos se divide en dos grupos para el entrenamiento y la prueba del modelo (80 % para entrenamiento y 20 % para prueba). Para la evaluación de la estima del S-KPI se utiliza el error cuadrático medio (MSE) entre el S-KPI obtenido en el terminal de prueba y la estima realizada por el modelo calculado como

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(X_i - \hat{X}_i)^2 \ , \tag{4.1}$$

siendo n el número total de conexiones de vídeo streaming, X_i el valor del tiempo inicial de búfer calculado por el terminal de pruebas para la conexión i, y $\hat{X}_i$ la estima del tiempo inicial de búfer obtenida por el modelo para la conexión i.

El proceso de optimización se realiza evaluando el MSE obtenido tras la combinación de las 8 métricas TCP disponibles a la entrada del modelo, con diferentes grados polinómicos. El rango de grados utilizados comprende desde 2^{o} hasta 6^{o} grado. Tras el proceso, se determina la mejor combinación de ambos parámetros de configuración con la que menor MSE medio se obtiene.

La Figura 4.7 muestra la comparativa de cada modelo combinando el número de predictores y el grado polinómico en base al MSE. En la figura, el eje x representa el número total de predictores (métricas TCP) usado para la estimación del S-KPI, mientras que en el eje y se muestra el MSE entre el S-KPI medido en el terminal y el estimado para cada combinación de métricas seleccionadas. Las diferentes líneas en la figura representan los modelos (grado polinómico) para relacionar los valores del S-KPI y las métricas TCP. El círculo sólido indica la combinación óptima de variables y el grado del polinomio que genera el mínimo MSE con el conjunto de datos de prueba (MSE = 1,18 s^2 haciendo uso de 5 variables y un polinomio de 5^{o} grado). Los resultados muestran valores muy similares haciendo uso de un polinomio de 4^{o} orden y 5 variables, concretamente, el RTT medio desde el terminal móvil a la sonda, el RTT medio desde la sonda a Internet, el *throughput* medio en el enlace descendente de la sesión de vídeo, el *throughput* medio en el enlace descendente sin considerar la fase lenta de inicio de la conexión TCP y la tasa media de transmisión de vídeo.

Una vez construido, el modelo se valida haciendo uso de un terminal de pruebas en el que se emula el consumo de un servicio de vídeo streaming. Para ello, se realizan diferentes conexiones para este tipo de servicio con diferentes condiciones de la red. El tiempo inicial de búfer se calcula tanto en el terminal móvil como en la aplicación TMA, obteniendo un rango de valores del S-KPI objetivo entre 0 y 50 s dependiendo de dichas condiciones. La Figura 4.8 muestra la comparación entre el tiempo inicial de búfer medido en el terminal móvil y el estimado por la aplicación TMA. El eje x representa el tiempo inicial de búfer estimado por la aplicación TMA, mientras que el eje y representa el medido por en el terminal móvil. Cada punto en la figura representa una conexión de video emulada por el terminal móvil y capturada por la aplicación TMA. Para facilitar la comparación, se superpone la línea de regresión. La validación del modelo pone de manifiesto su capacidad de estimar el S-KPI obteniéndose un valor del 80 % percentil del error absoluto de solo 0,16 s entre ambos dispositivos.

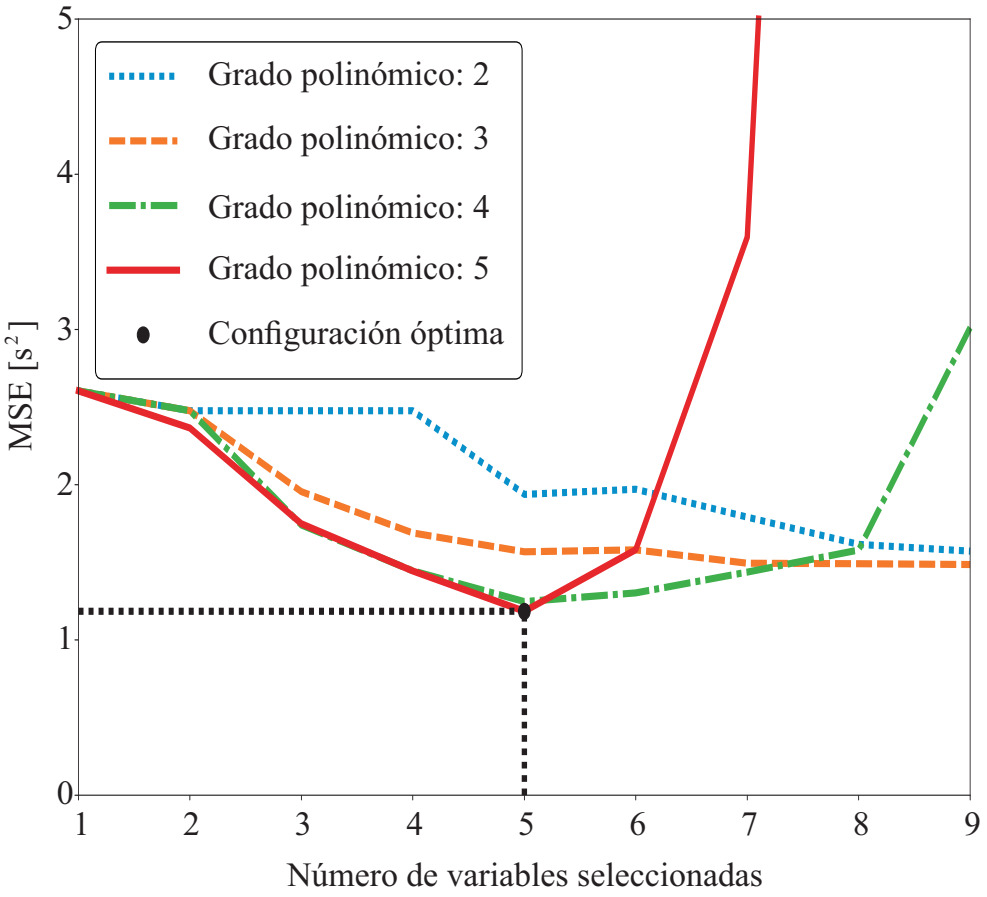


FIGURA 4.7: Comparativa de los modelos para estimar el tiempo inicial de búfer en base al MSE.

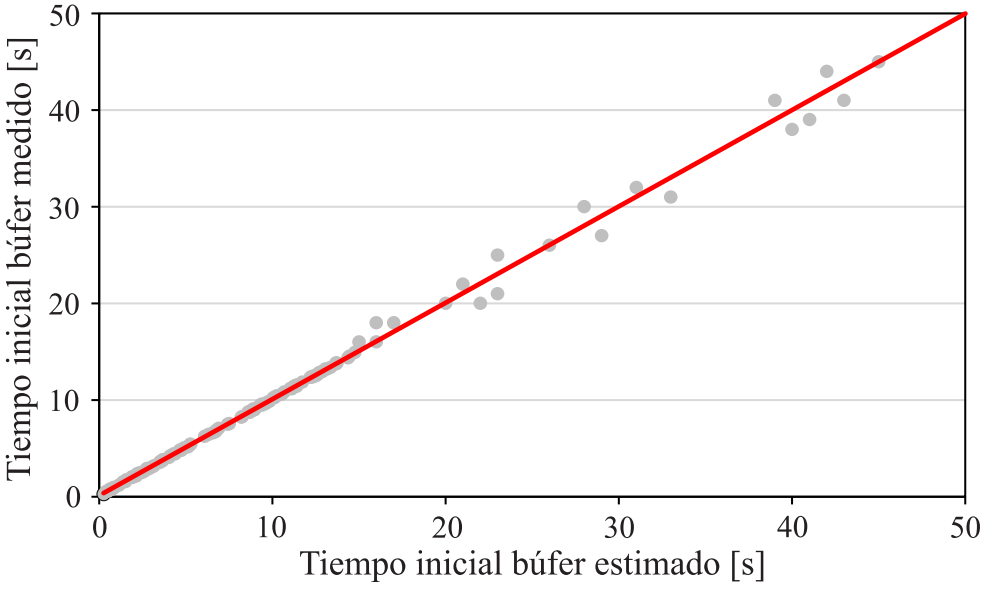


FIGURA 4.8: Comparación del tiempo inicial de búfer estimado y medido.

4.3.2. Construcción de modelo para la predicción de S-KPIs

Alternativamente al análisis descriptivo, las aplicaciones TMA pueden utilizarse también para realizar un análisis predictivo para prever posibles degradaciones del servicio, permitiendo diseñar acciones correctivas que mejoren su funcionamiento. En este contexto, la información generada por la aplicación TMA tras la fase de procesamiento puede utilizarse para la creación de modelos predictivos que permitan estimar la evolución de los diferentes S-KPIs calculados, para poder gestionar la QoE para un determinado servicio a lo largo del tiempo. En el caso de detectarse problemas en el rendimiento de dicho servicio, se podrían ejecutar proactivamente medidas de control en base a criterios de QoE. Estos análisis pueden consistir en modelos clásicos basados en análisis de series temporales o en algoritmos de ML más modernos para construir modelos más robustos. Como ejemplo de este caso de uso, en este trabajo se desarrolla un modelo predictivo para estimar la tendencia del tiempo medio de descarga web a nivel de celda. Este S-KPI es calculado por la aplicación TMA y proporcionado a la salida de la fase de procesamiento. El modelo de predicción hace uso de una red neuronal recurrente (*Recurrent Neural Network*, RNN) [93]. La validación del modelo se realiza utilizando una aplicación TMA desplegada en una red LTE real.

La construcción del modelo se inicia mediante la recopilación de series temporales del tiempo medio de descarga web calculado por la aplicación TMA para las diferentes conexiones de este tipo de servicio. Como ya se ha explicado en este capítulo, el tiempo de descarga web se calcula como el tiempo entre el primer mensaje HTTP GET enviado por el usuario y el último mensaje HTTP de respuesta enviado por el proveedor del servicio. Esta información se añade en los ESRs a la salida de la fase de procesamiento. Posteriormente, estos valores se agregan a nivel de celda y hora utilizando la información disponible en los registros. Estas series temporales se utilizan para la fase de entrenamiento del modelo.

La Tabla 4.1 recoge los principales parámetros utilizados para la creación del modelo. El modelo se construye mediante el entrenamiento de una red LSTM (*Long-Short Term Memory*) [94]. La elección de este tipo de red radica en su capacidad de mitigar el problema de desvanecimiento del gradiente, que puede ocurrir en la RNN cuando se procesan largas secuencias temporales. La red es configurada con 24 unidades (o neuronas), asegurando la capacidad suficiente para aprender los diferentes patrones a su entrada. Las unidades determinan la complejidad de

Tabla 4.1: Principales parámetros del modelo de predicción.

Red Neuronal		
Tipo de red	LSTM	
Número de neuronas	24	
Función de activación	*tanh*	
Compilador		
Función de pérdida	*MSE*	
Optimizador	*Adam*	
Entrenamiento		
Pasos temporales	12	
Número de muestras	Entrenamiento	381,000 (80 %)
	Prueba	95,250 (20 %)
Épocas	1000	
Tamaño de lote	20	

la red y su capacidad de aprender por lo que debe haber un balance entre coste y rendimiento. La función de activación utilizada para el modelo es la tangente hiperbólica (*tanh*), comúnmente utilizada por su capacidad para manejar gradientes y facilitar el aprendizaje. La entrada del modelo es unidimensional con 12 pasos temporales. Los pasos temporales se refieren a la cantidad de muestras temporales anteriores que se utilizan para predecir el siguiente valor en la secuencia. El modelo se compila utilizando el error cuadrático medio como función de pérdida y el optimizador *Adam* [95], que permite ajustar los pesos de la red utilizando técnicas adaptativas de tasa de aprendizaje.

El modelo se entrena durante 1000 épocas utilizando un tamaño de lote de 20. Una época completa significa que todo el conjunto de datos de entrenamiento se ha utilizado una vez para actualizar los pesos de la red. Por otro lado, el tamaño de lote (o número de muestras) define cuándo se actualizan los pesos de la red, ofreciendo mejor eficiencia durante la fase de entrenamiento que si se procesan todas las muestras a la vez. Para la fase de entrenamiento, y su posterior validación, se utilizan los datos recopilados por una aplicación TMA desplegada en una red LTE real durante, aproximadamente, 3 meses. En este tiempo, se captura el tráfico generado en 254 celdas obteniendo un total de 476.250 muestras tras la fase de agregación por hora. Este conjunto de datos se divide posteriormente en dos grupos (80 % para el entrenamiento y 20 % para prueba).

Tras la construcción del modelo, este se evalúa con los datos recopilados de una de las celdas de la red durante una semana completa. Para ello, el modelo se ejecuta 7*24 veces, en el que se toman los valores del S-KPI de las 12 horas anteriores como

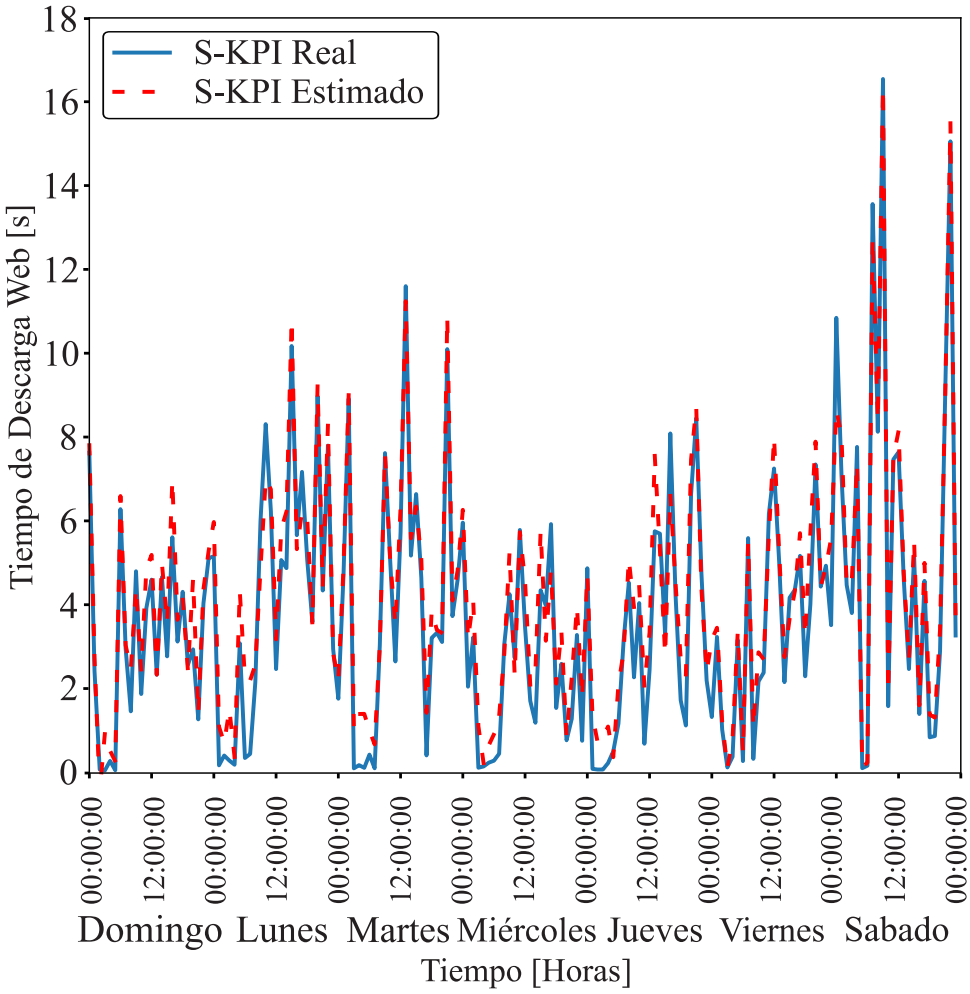


FIGURA 4.9: Predicción de la evolución horaria del tiempo medio de descarga web en una celda.

entrada. La Figura 4.9 muestra los resultados obtenidos por el modelo predictivo en comparación con los valores reales obtenidos por la aplicación TMA. En la figura, la línea solida representa el valor real medido del S-KPI mientras que el valor predicho se representa mediante la línea discontinua. De ambas líneas se puede observar cómo la predicción realizada por el modelo es muy cercana al valor real percibido por la aplicación. Dicha observación se confirma mediante el valor del *MSE* calculado haciendo uso del conjunto de datos para validación ($MSE = 0{,}95\ s^2$).

Los resultados confirman la capacidad del modelo para predecir el comportamiento del S-KPI a lo largo del tiempo. Este tipo de modelos son de gran utilidad para los operadores ya que permiten reducir el tiempo de reacción, aumentando la satisfacción del usuario. Las aplicaciones TMA juegan un papel muy importante ya que proporcionan, con una fina granularidad, la información necesaria

para poder calcular y monitorizar diferentes S-KPIs en tiempo real. Por ello, este mismo proceso se puede replicar para otro tipo de S-KPI o, para la construcción de modelos con una granularidad más fina (p. ej., minutos). Por otro lado, el bajo coste computacional de este tipo de modelos tras su entrenamiento permite su integración en sistemas de monitorización y gestión de la red.

4.3.3. Evaluación del estado general de la QoE de los servicios de una red

En el marco actual, una óptima monitorización y control de la QoE supone un factor diferencial entre los operadores de redes móviles. En este contexto, las aplicaciones TMA son una herramienta idónea para evaluar el comportamiento de la QoE para los diferentes servicios usados por los usuarios. La información disponible en los ESRs, tras la fase de procesamiento, puede utilizarse para generar reportes y paneles gráficos para controlar el estado de las diferentes celdas de la red en base a los valores de los diferentes S-KPIs calculados. Esto proporciona una herramienta de fácil uso y entendimiento para no solo conocer el estado actual de la red sino el impacto que posibles operaciones de optimización pueden tener tras su aplicación.

Para ello, las métricas generadas por la aplicación TMA se recopilan para obtener los S-KPIs más relevantes para los diferentes servicios consumidos en la red. Posteriormente, mediante el procesamiento de esta información y la aplicación de diferentes reglas o umbrales que determinan diferentes niveles de calidad del indicador, se puede determinar el estado de la celda para un servicio específico. En este caso de uso, se recopilan métricas de 3 servicios en base a la demanda del operador: servicio de vídeo streaming, servicio de navegación web y un servicio móvil de banda ancha consistente en una prueba de velocidad basada en una descarga FTP. Para estos servicios, se obtienen los siguientes S-KPIs con el fin de monitorizar la calidad de experiencia percibida: a) tasa de bloqueo de vídeo, calculada como el tiempo que el vídeo está interrumpido para la carga del búfer, divido entre el tiempo total de visualización; y retardo inicial para la transmisión del vídeo, b) tasa de visualización exitosa de la página web y tiempo medio de respuesta de página durante la navegación web, y c) *throughput* medio del enlace descendente para el servicio de banda ancha. Cada S-KPI se calcula agregando todas las sesiones de un servicio a nivel de celda.

TABLA 4.2: Valores de S-KPIs en el escenario.

Servicio	S-KPI	Umbral	Valor medio	95^{th} percentil	5^{th} percentil	Nº celdas válidas
Vídeo streaming	Tasa de bloqueo	4 %	2.85 %	13.34 %		848 (17.15 %)
	Retardo de inicio	2 s	3.51 s	9.96 s		2450 (17.15 %)
Navegación web	Tasa de visualización exitosa	98 %	97.22 %		91.26 %	3578 (74.11 %)
	Tiempo medio de respuesta	2 s	0.29 s	0.64 s		4747 (98.32 %)
Servicio de banda ancha	Throughput DL FTP	5 Mbps	13.83 Mbps		2.59 Mbps	4312 (89.31 %)

Para la recopilación de los datos, se despliega la aplicación en una red LTE real compuesta de 4.828 celdas durante un día completo. La Tabla 4.2 recoge los valores de diferentes métricas para los S-KPIs listados anteriormente en la red bajo estudio. Las Columnas 4-6 muestran el valor medio, el 95^{th} y 5^{th}-percentil, respectivamente, de los S-KPIs calculados a nivel de celda para evaluar el estado de las peores celdas. La Columna 3 muestra el umbral definido por el operador para determinar si una celda es válida o, por el contrario, muestra un rendimiento peor de lo esperado. Los resultados muestran que el servicio ofrecido por la red para la navegación web y el servicio de banda ancha es adecuado, con al menos el 74,11 % de las celdas cumpliendo el umbral definido para cada S-KPI. Sin embargo, el servicio de vídeo streaming presenta valores peores.

Los resultados demuestran que este tipo de análisis son de gran utilidad para identificar celdas con problemas de rendimiento en servicios específicos. De esta forma, se pueden realizar análisis automáticos para identificar la causa de dicho problema evaluando todos los R-KPIs (es decir, indicadores de acceso radio, transporte y troncal) recopilados en los ESRs. En el caso de detectarse un problema, el operador es notificado y debe aplicar las medidas correctivas necesarias para solucionarlo. Además, este tipo de análisis también permite identificar si el mal funcionamiento es debido al proveedor del servicio y, por tanto, no son necesarios cambios en la red. Un caso real detectado al analizar en detalle los ESRs en las pruebas realizadas durante el despliegue de la aplicación, mostró una mala experiencia de usuario percibida en el servicio de vídeo streaming debido a un alto RTT causado por el retardo generado durante un traspaso entre celdas en la misma portadora, así como una alta tasa de pérdida de paquetes en la red troncal originada por un problema en el proveedor de servicio. A partir de los resultados obtenidos tras el análisis, el operador pudo aplicar las medidas correctivas necesarias para mitigar los problemas originados en su red y, finalmente, mejorar la QoE de aquellas celdas y servicios afectados.

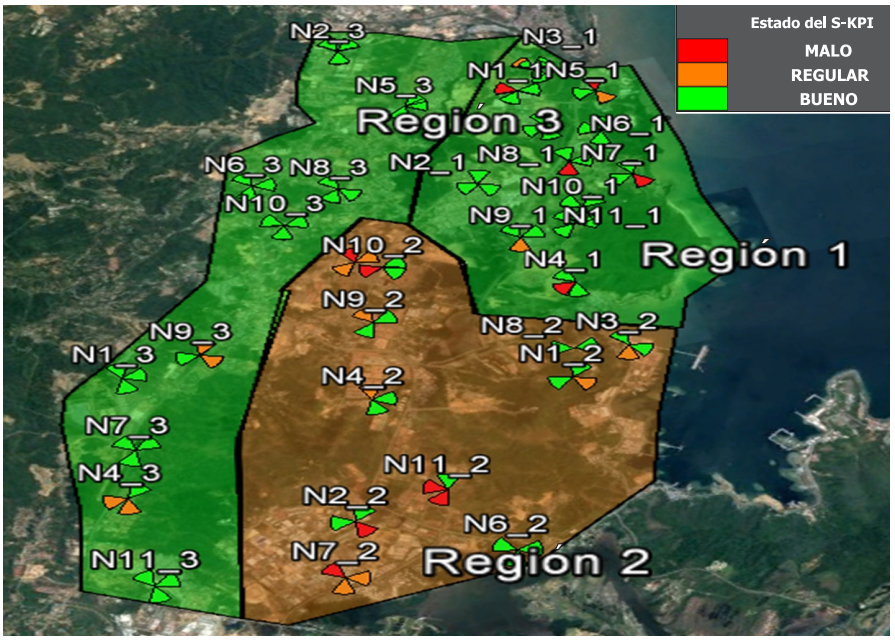


FIGURA 4.10: Evaluación de S-KPIs mediante mapas geográficos.

Por otro lado, los datos proporcionados por una aplicación TMA también se pueden procesar para ser presentados de forma gráfica en reportes que permitan evaluar el estado general de la red en términos de QoE. Las distintas estimas de los S-KPIs pueden representarse en mapas que muestren el grado de calidad por áreas geográficas. Para ello, los valores de los distintos S-KPIs son agregados a nivel de celda y el indicador de calidad se calcula en base a umbrales previamente definidos. La Figura 4.10 muestra un ejemplo de mapa de calidad de S-KPI, donde cada celda se representa en sectores. El estado del sector se representa con un color determinando si cumple con los valores fijados para considerarlo un sector con buena calidad, si está cerca de cumplirlo o si está muy por debajo del valor esperado. En el ejemplo, el área se divide en diferentes regiones en función del valor medio obtenido para el S-KPI.

Alternativamente, se pueden crear directamente mapas de QoE mediante el uso de funciones de utilidad para poder representar los distintos valores de los S-KPIs. Este tipo de mapas son de mayor utilidad para el operador, permitiendo identificar regiones con problemas para el usuario y realizando acciones correctivas.

4.3.4. Diagnóstico de conexiones problemáticas

Aunque al análisis de la QoE a nivel de celda permite evaluar el estado general de la red, el uso de datos agregados dificulta la identificación real del problema detrás de una mala experiencia de usuario. Además, se podría obtener una conclusión errónea de la causa, resultando en acciones innecesarias por parte del operador. Haciendo uso de toda la información generada por una aplicación TMA, pueden realizarse análisis más complejos para diagnosticar la causa específica de cada conexión y planificar medidas correctivas de forma más efectiva.

Para este caso de uso, en este trabajo, se ha desarrollado un método automático para la identificación y diagnóstico de conexiones problemáticas a partir de la información disponible a la salida de una aplicación TMA. El método propuesto se basa en el hecho de que la mayoría del tráfico generado por los servicios consumidos hoy en día en las redes móviles utiliza el protocolo de transporte TCP, precisamente el protocolo capturado por la aplicación. A partir de aquí, es posible identificar conexiones de mala calidad analizando el *throughput* en el enlace descendente de dicha capa. Tras la identificación de aquellas conexiones con mala calidad, el proceso de diagnosis se realiza haciendo uso de los diferentes indicadores de rendimiento disponibles. La entrada del método consiste en el conjunto de datos recopilados por la aplicación de diferentes puntos de la red. La información de salida es un listado de conexiones con problemas de rendimiento y la causa de dicho problema.

La definición del método se divide en varias etapas. En primer lugar, se realiza un proceso de limpieza, transformación y procesado de los datos disponibles a la entrada del método. En esta etapa, se eliminan posibles valores erróneos y/o atípicos en los datos, así como se construyen nuevos indicadores que permitan enriquecer el proceso de diagnosis. Posteriormente, se identifican de forma automática aquellas conexiones con problemas de rendimiento. El método se basa en el hecho de que la calidad de experiencia percibida por el usuario, para la mayoría de los servicios que utilizan la capa de transporte TCP, está altamente relacionada con el *throughput* DL medido en dicha capa. Por tanto, aquellas conexiones cuyo valor medio de *throughput* TCP DL sea inferior a un determinado umbral, $TH_{min}^{TCP,DL}$, serán etiquetadas como conexiones problemáticas. Dicho umbral se ajusta de forma automática con el percentil del 5 % de la distribución del *throughput* de todas las

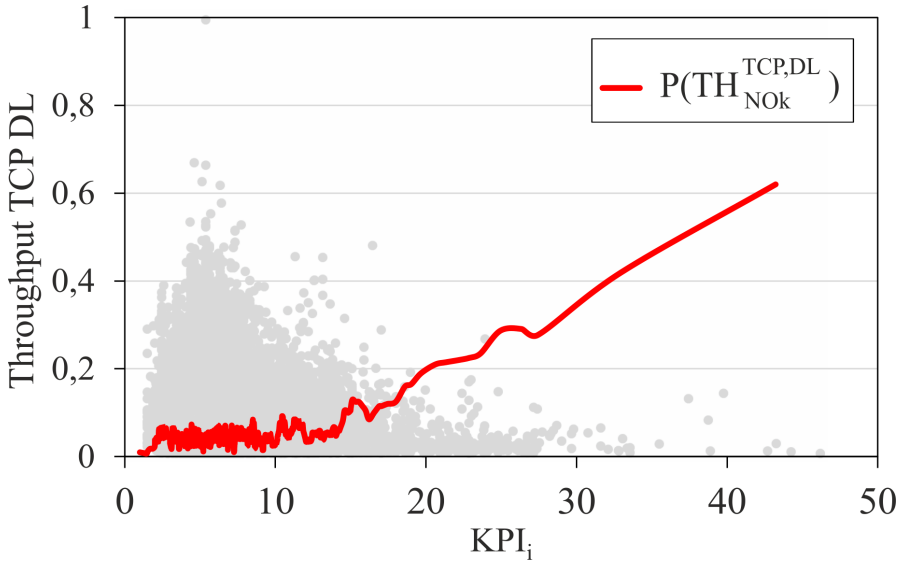


FIGURA 4.11: Probabilidad condicional de bajo *throughput* para el indicador i.

conexiones. Finalmente, se realiza el proceso de diagnosis de aquellas conexiones que no cumplen con el umbral previamente definido.

Para el diagnóstico, el método se basa en el cálculo de la probabilidad condicional. En primer lugar, los valores de los diferentes indicadores son agrupados en intervalos para cada KPI_i. Posteriormente, se estima la probabilidad de bajo *throughput* TCP DL debido al indicador i para cada uno de los intervalos k a partir del ratio de llamadas problemáticas en dicho intervalo. Dicho cálculo se define como

$$P(TH_{NOk}^{TCP,DL}(i,k)) = \frac{N_{TH^{TCP,DL}(i,k)<TH_{min}^{TCP,DL}}}{N_{TH^{TCP,DL}(i,k)}}, \tag{4.2}$$

siendo $N_{TH^{TCP,DL}(i,k)<TH_{min}^{TCP,DL}}$ el número total de conexiones con problemas de *throughput* TCP DL en el intervalo k para el indicador i, y $N_{TH^{TCP,DL}(i,k)}$ el número total de conexiones de dicho intervalo e indicador.

Para cada intervalo k, se calcula tanto la media de los valores del indicador i como la media móvil de $P(TH_{NOk}^{TCP,DL}(i,k))$, utilizando intervalos consecutivos. Finalmente, para aquellos intervalos en los que no hay valores del indicador i en las conexiones, se realiza un proceso de interpolación lineal a partir de los intervalos adyacentes para estimar el valor. De esta forma, se obtiene una función de probabilidad lineal condicional para problemas de *throughput* TCP DL en función del valor de un indicador de rendimiento específico.

La Figura 4.11 muestra un ejemplo gráfico del comportamiento de la probabilidad condicional de bajo *throughput* para un indicador de rendimiento i. El eje x representa los diferentes valores del indicador de rendimiento i analizado como posible causante del bajo rendimiento del *throughput* de la conexión. Por otro lado, el eje y representa el valor normalizado del *throughput* TCP DL. Cada punto representa una conexión, mientras que la línea solida representa la probabilidad condicional de bajo *throughput* en base al indicador i.

De la figura se deduce que, a partir de la función de probabilidad, se puede identificar de forma cualitativa las causas de aquellas conexiones con problemas de rendimiento en base a los diferentes indicadores de rendimiento disponibles. Aquellos indicadores que no son causa del problema mostrarán un valor de probabilidad condicional constante, igual al ratio global de conexiones problemáticas en la red (en este caso el 5 %, como valor del percentil usado para determinar el umbral). Por el contrario, aquellos indicadores que estén altamente relacionados con el problema presentarán un ratio de conexiones problemáticas superior en determinados intervalos. Por tanto, se calcula la probabilidad condicional de todos los indicadores disponibles para cada conexión identificada con un rendimiento inadecuado y, finalmente, se elige aquel indicador cuya probabilidad condicional sea mayor. Con el fin de hacer un diagnóstico más preciso, aquellas conexiones problemáticas cuya probabilidad condicional no sea superior al 5 % para ningún indicador no son etiquetadas con ninguna causa conocida.

Con el fin de validar el método, se hace uso de los datos generados por una aplicación TMA desplegada en una red 3G real compuesta de 1.976 celdas. La aplicación recopila datos durante 2 horas, obteniendo un total de 165.327 conexiones. Posteriormente, se utiliza el método descrito anteriormente para identificar aquellas conexiones problemáticas y su correspondiente causa. Tras una primera ejecución, se identifica un porcentaje considerable de conexiones sin un diagnóstico válido. Con el fin de obtener resultados más precisos, algunas conexiones se diagnostican de forma manual ya que su causa podría estar relacionada con escenarios específicos que no pueden ser identificados de forma automática. El resultado de este análisis puede utilizarse para mejorar el método y considerar escenarios concretos a ser identificados automáticamente en futuras ejecuciones.

Del análisis manual, se identifican 4 casos concretos que no pueden ser diagnosticados de forma automática por el método, pero que representan un porcentaje considerable del total de conexiones problemáticas.

Caso 1: Estrangulamiento del ancho de banda. El estrangulamiento (throttling) es una medida comúnmente utilizada por los operadores móviles para regular el tráfico de la red y reducir los problemas de congestión por falta de recursos. Dicha medida consiste en limitar el ancho de banda máximo disponible de aquellos usuarios que han agotado su cuota de descarga de datos especificada en los planes de tarificación. Generalmente, esta limitación se implementa mediante el descarte de paquetes a nivel de red en algún punto de la red troncal [96]. La información generada por la aplicación permite identificar el tráfico saliente desde la sonda (usada para capturar el tráfico) hacía Internet, y poder calcular las métricas necesarias para identificar aquellas conexiones afectadas por esta causa. De otra forma, el análisis de la tasa de pérdida de paquetes podría llevar a resultados incorrectos al no poder considerarse una medida que nada tiene que ver con el estado de la red. La Figura 4.12, ilustra el proceso de estrangulamiento del ancho de banda. El eje x representa los valores de *throughput* TCP DL de cada conexión, mientras que el eje y muestra la tasa de pérdida de paquetes de nivel de red en el mismo enlace. La línea sólida representa la probabilidad de que la tasa de pérdida de paquetes DL sea superior al valor medio de las conexiones, $P(TPP > 0,1)$ (estimado en 0,1 para el juego de datos usado). Dicha probabilidad se define como

$$P(TPP > 0,1) = \frac{N_{TPP^{TCP,DL}(i,k)>0,1}}{N_{TPP^{TCP,DL}(i,k)}}, \tag{4.3}$$

siendo $N_{TPP^{TCP,DL}(i,k)>0,1}$ el número total de conexiones con una tasa de pérdida de paquetes TCP DL superior a 0,1 en el intervalo k para el indicador i, y $N_{TPP^{TCP,DL}(i,k)}$ el número total de conexiones de dicho intervalo e indicador.

En la figura, se distinguen 4 intervalos de *throughput* en los que la tasa de pérdida de paquetes destaca claramente sobre el resto. Dichos valores se corresponden con los intervalos [63, 128] kbps, [0,8, 1,2] Mbps, [2,8, 3,2] Mbps y [5, 10] Mbps. Dichos valores coinciden con los valores de velocidad de descarga de datos mínima garantizada por el operador en sus diferentes planes de tarificación (64 kbps, 1 Mbps, 3 Mbps y 5 Mbps).

Caso 2: Comunicaciones a través de equipos fuera de la red. De las conexiones problemáticas detectadas, cierto porcentaje se diagnostica como problemas de latencia en sus enlaces. Dichos problemas se ven reflejados en los valores de RTT en el segmento de red de cada conexión. La Figura 4.13 muestra un histograma del RTT medio en el segmento de red entre la herramienta TMA e Internet, junto

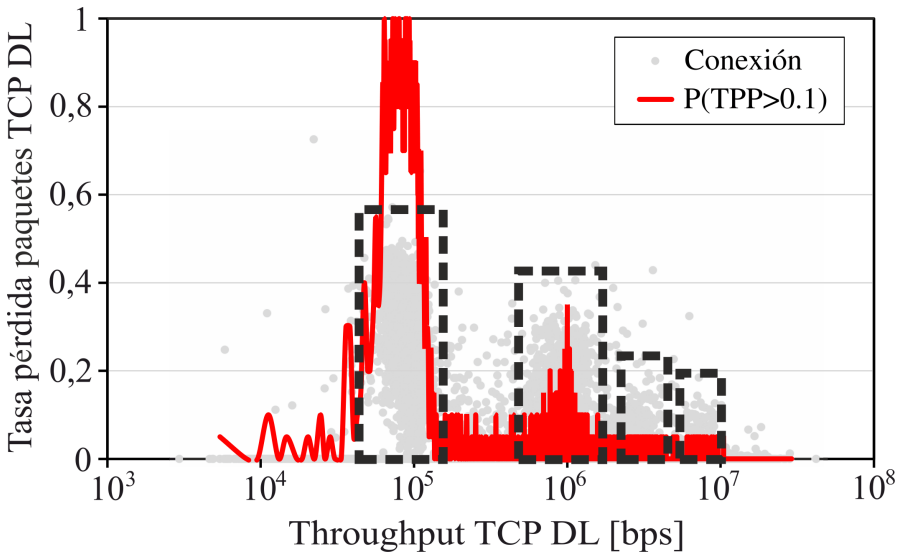


FIGURA 4.12: Análisis de estrangulamiento del throughput TCP DL.

con la función de distribución acumulada de dicho indicador. En el eje x se representan los valores de RTT. El eje y primario muestra el porcentaje de muestras, mientras que el eje y secundario representa la función de distribución acumulada. En dicha figura se distinguen 2 rangos de valores de RTT que destacan sobre el resto en cuanto a porcentaje de muestras se refiere. Tras una evaluación exhaustiva de estos grupos, se identifican 2 valores de RTT que se repiten habitualmente, 2 ms y 11 ms. El análisis detallado de los datos recopilados para estas conexiones muestra que el primer valor de RTT lo representan conexiones que hacen uso del protocolo de comunicación HTTP (p. ej., servicio web). Estas conexiones pasan por un servidor proxy del operador, causante de esta pequeña latencia percibida por el usuario. El segundo valor lo presentan conexiones a servidores concretos en el mismo país donde la red está desplegada. Por otro lado, se detecta un alto porcentaje de conexiones con un retardo superior a 150 ms. Un valor tan alto de latencia suele estar relacionado con comunicaciones de larga distancia. Un ejemplo de este tipo de comunicaciones es la comunicación por satélite. Estas comunicaciones han adquirido una mayor popularidad debido al incremento en la demanda de los servicios multimedia y de disponibilidad en cualquier área de la red [97]. Los valores de retardo superiores a 150 ms son similares a los producidos por conexiones por satélites orbitando en las conocidas como órbitas medias (*Medium Earth Orbit*, MEO) situadas a unos 10.000-20.000 km de la Tierra. Por tanto, se puede deducir que aquellas conexiones problemáticas debido a valores de RTT superior a los 150 ms se corresponden con conexiones que presentan algún tramo por satélite.

Caso 3: Conexiones a través de la Red digital de servicios integrados (RDSI).

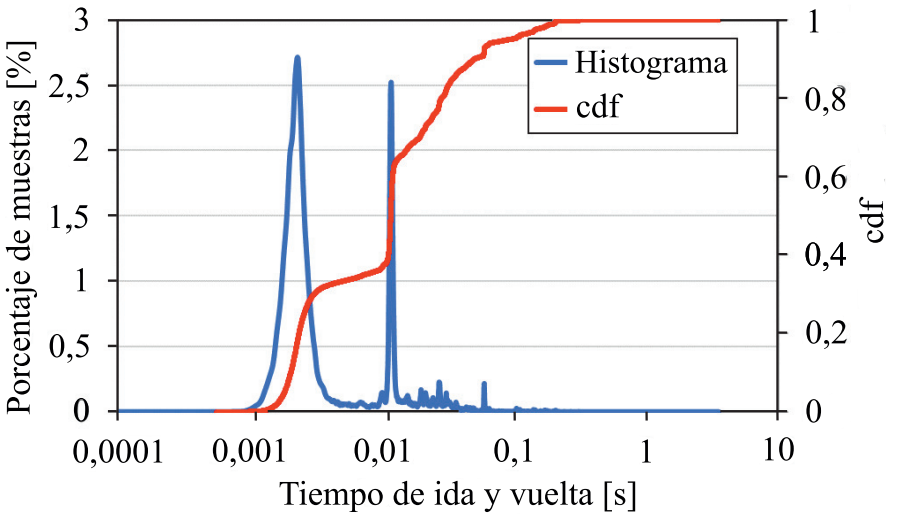


FIGURA 4.13: Análisis del tiempo de ida y vuelta de paquetes TCP.

La RDSI surge como la evolución de la red telefónica clásica, ofreciendo conexiones digitales extremo a extremo. Este tipo de red permite la integración de múltiples servicios con acceso único, a través de canales de transmisión que pueden alcanzar velocidades de hasta 64 kbps [98]. Dentro del juego de datos usado, se identifica un grupo de conexiones donde el volumen total de bytes DL se diagnostica como causa principal de la mala calidad. Este diagnóstico se puede ver en la Figura 4.14. En dicha figura, se muestra la función de probabilidad de bajo throughput TCP DL en función del volumen total de bytes descargados durante la conexión. El eje x representa los valores del total de bytes DL de cada conexión, mientras que en el eje y se representa el *throughput* TCP DL. La línea sólida representa la probabilidad condicional de bajo *throughput* TCP DL. En la figura, se observa que cuando el volumen total es de 481 kBytes, la probabilidad aumenta significativamente. Estas conexiones forman parte de conexiones de mayor duración que la aplicación TMA fragmenta en registros de duración de 1 minuto (duración máxima de los registros en base a la configuración usada en la aplicación TMA), por lo que el *throughput* de estos registros fragmentados es 64 kbps. Por tanto, se pueden considerar como conexiones a través de la RDSI, aquellas sin diagnóstico y cuyo *throughput* TCP DL esté cercano a 64 kbps.

Caso 4: Sesión fragmentada en múltiples conexiones. Tras el proceso inicial de identificación y diagnósticos de conexiones problemáticas, se detectan diferentes celdas con un alto porcentaje de conexiones problemáticas sin diagnosticar. Un análisis detallado de una de estas celdas muestra que un gran número de dichas conexiones pertenecen al mismo usuario para una misma sesión (se observa que las

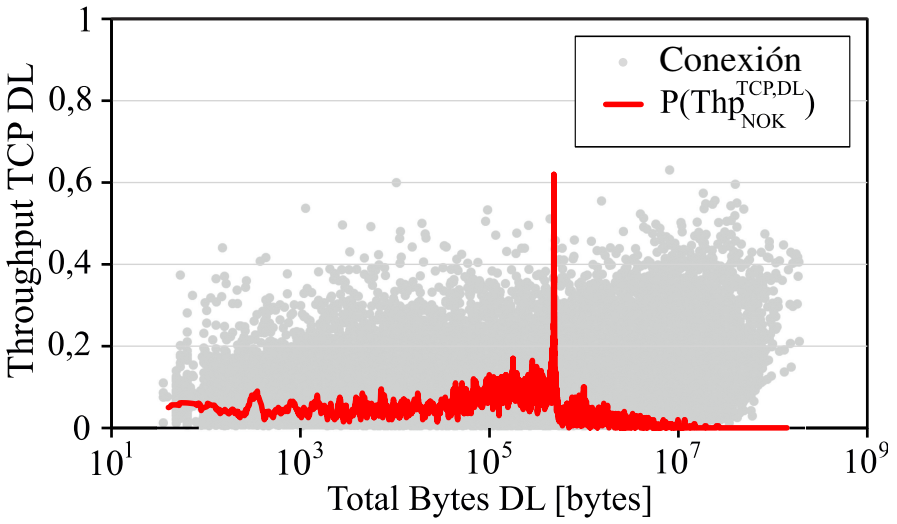


FIGURA 4.14: Análisis de conexiones a través de la RDSI.

conexiones son secuenciales). Este es el comportamiento habitual de la aplicación TMA cuando fragmenta conexiones en diferentes registros de duración fija (1 minuto en la configuración desplegada). Con el fin de identificar si realmente hay un problema de calidad, es necesario realizar un estudio a nivel de sesión. Para ello, se reconstruye dicha sesión a partir de todas las conexiones pertenecientes al mismo usuario, haciendo uso del IMSI. La Figura 4.15 muestra el valor de diferentes medidas de *throughput* para las conexiones de un mismo usuario. El eje x muestra un eje temporal que representa el instante de inicio de cada conexión relativo a la primera conexión disponible. En el eje y se representan los distintos valores de *throughput*. Los círculos representan el *throughput* TCP DL de todas las conexiones del usuario, las aspas representan el *throughput* TCP de sus conexiones problemáticas y los cuadrados representan el *throughput* de sesión DL, calculado como el volumen total de datos descargados por conexión entre la duración (considerando los periodos de silencio entre ráfagas). La figura refleja la hipótesis inicial de que todas las conexiones se corresponden con una única sesión de varios minutos que la aplicación TMA fragmenta en diferentes conexiones. Dicha sesión está asociada a un servicio de bajo régimen binario, característico de servicios de audio streaming (p. ej., Spotify) o vídeo streaming de bajo régimen binario (p. ej., Youtube a baja calidad). Este análisis pone de manifiesto la existencia de aplicaciones que presentan este tipo de comportamiento en el tráfico capturado, permitiendo aislar este tipo de conexiones de aquellas que afectan la calidad de experiencia.

Al margen de los casos que requieren de análisis específicos, el proceso de diagnosis permite identificar, de forma automática, la causa de un alto porcentaje

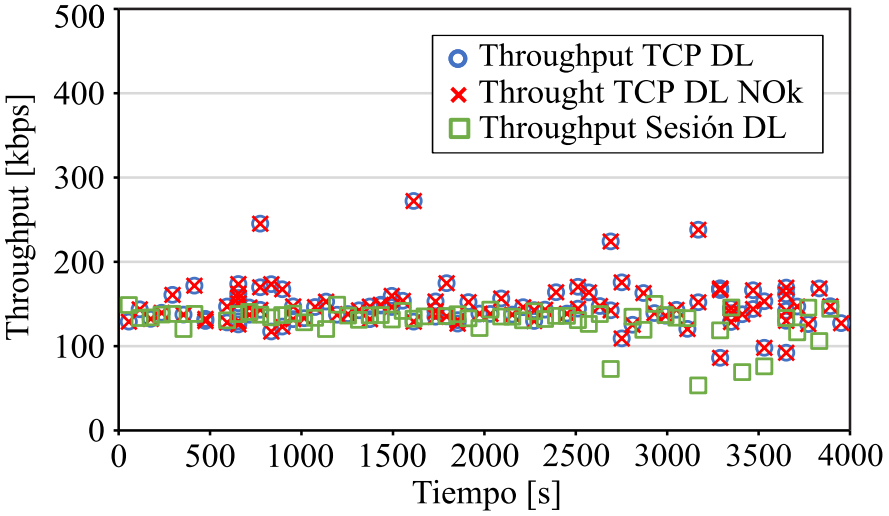


FIGURA 4.15: Análisis de conexiones de un mismo usuario.

TABLA 4.3: Resultado del diagnóstico de las conexiones de bajo throughput TCP DL.

Diagnóstico	**Conexiones [%]**
Throttling	32.37
Sin determinar (con datos radio)	23.41
Sin determinar (sin datos radio)	16.82
Múltiples conexiones	11.19
Otras	11.65
Enlace por satélite	4.56

de conexiones con problemas. Para el juego de datos empleado, se determina un umbral de *throughput* TCP DL mínimo de 443 kbps, dando como resultado un total de 8.266 conexiones problemáticas de las 165.325 recopiladas por la herramienta.

Los resultados del proceso de diagnosis se recopilan en la Tabla 4.3 mostrando el porcentaje de conexiones para cada diagnóstico. Del mismo modo, la Figura 4.16 muestran el resultado en un gráfico de sectores. De los resultados se observa que se han diagnosticado de forma exitosa casi el 60 % de las conexiones con problemas de *throughput* TCP. Del 40 % restante, casi el 17 % se corresponden con conexiones que falta de datos en sus registros por lo que el proceso de diagnosis no se puede realizar de forma eficiente. Más del 11 % de las conexiones han sido automáticamente diagnosticadas en base a los parámetros de entrada del método.

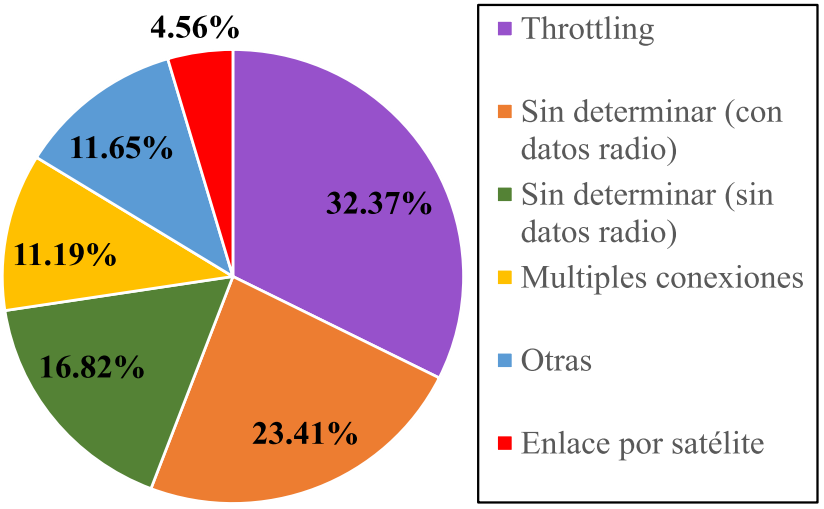


FIGURA 4.16: Diagnóstico de conexiones de bajo throughput TCP DL.

4.4. Conclusiones

En este capítulo, se ha introducido el uso de aplicaciones TMA para la gestión de la QoE en las redes móviles. Este tipo de aplicaciones está adquiriendo una gran popularidad en la industria por lo que su uso serás cada vez más frecuente para procesos de gestión de red. En primer lugar, se ha presentado la estructura genérica de este tipo de aplicaciones indicando las diferentes fases que la componen, las diferentes fuentes de datos que recopilan, el procesamiento realizado sobre los datos obtenidos y las posibles aplicaciones genéricas que se le pueden dar en entornos reales. Posteriormente, se ha expuesto la necesidad de un proceso de validación de una aplicación TMA y se ha presentado una metodología genérica para validar la solución tras su despliegue en una red real. El objetivo de esta fase es el de validar la información que proporciona la aplicación tras su fase de procesamiento en forma de métricas e indicadores de rendimiento. Por último, se han presentado diferentes casos de usos específicos que hacen uso de una aplicación TMA para el análisis y la monitorización de la QoE.

Para el primer caso de uso se ha construido un modelo para la estimación del tiempo inicial de búfer, uno de los indicadores de rendimiento más característicos

del servicio de vídeo streaming. Este modelo hace uso de la información recopilada por la aplicación TMA para construir un modelo de regresión polinómico multivariable. El modelo es ajustado automáticamente para determinar la mejor combinación de métricas usadas como entrada y grados del polinomio utilizado. Durante su construcción, el modelo se valida con datos reales obtenidos de una red LTE real determinando que la configuración óptima del modelo consiste en 5 variables de entrada (el RTT medio desde el terminal móvil a la sonda, el RTT medio desde la sonda a Internet, el *throughput* medio en el enlace descendente de la sesión de vídeo, el *throughput* medio en el enlace descendente sin considerar la fase lenta de inicio de la conexión TCP y la tasa media de transmisión de vídeo) y un polinomio de 5^{9} grado. Con esta configuración, se obtiene un error cuadrático medio de 1,18 s^2. Posteriormente, el modelo se evalúa haciendo uso de un terminal de pruebas simulando el consumo del servicio de vídeo streaming. En este caso, se obtiene un valor de 0,16 s para el percentil del 80 % del error absoluto.

El segundo caso de uso ha consistido en la construcción de un modelo predictivo para prever la tendencia del tiempo de descarga web a nivel de celda (indicador relacionado con el servicio de navegación web). Dicho modelo está basado en una red neuronal LSTM y, de nuevo, se construye con los datos proporcionados por una aplicación TMA desplegada en una red LTE real. Posteriormente, dicho modelo es evaluado haciendo uso de los datos recopilados de una celda de la misma red durante una semana completa. Los resultados muestran un valor del error cuadrático medio de 0,95 s^2.

Para el tercer caso de uso, se utiliza la información disponible a la salida de la fase de procesamiento de la aplicación para la generación de reportes de diferente naturaleza para mostrar el rendimiento general de la red mediante el estado de cada una de sus celdas, definiendo umbrales para los diferentes S-KPIs de los principales servicios de la red. Este tipo de análisis son de gran utilidad para los operadores de red ya que permiten identificar problemas en servicios específicos en las diferentes celdas que componen la red. Para mostrar el beneficio de este caso de uso se utilizan los datos generados por una aplicación TMA desplegada en una red LTE real compuesta de 4.828 celdas durante un día completo. Se generan 2 tipos de reportes, uno en forma de tabla mostrando los diferentes S-KPIs evaluados, el umbral utilizado, distintas métricas para medir el estado del servicio en la red y el porcentaje de celdas que se consideran válidas en base al umbral. Por otro lado, se muestra un ejemplo de representación gráfica del estado de la red en base a un

S-KPI específico. Dicha representación consiste en un mapa geográfico de la red donde se diferencian las distintas celdas distribuidas en regiones donde se identifica el estado mediante diferentes combinaciones de colores.

Para el último caso de uso, se ha construido un método automático para la identificación y diagnóstico de conexiones problemáticas usando las diferentes métricas recopiladas por una aplicación TMA. El método se basa en el hecho de que la mayoría de los servicios de la red utilizan el protocolo de transporte TCP, y su rendimiento está altamente relacionado con el *throughput* TCP DL percibido durante la conexión. En primer lugar, el método identifica como conexiones problemáticas todas aquellas cuyo valor medio de *throughput* TCP DL esté por debajo del valor del percentil del 5 % de la distribución de todas las conexiones. Posteriormente, calcula la probabilidad condicional de bajo *throughput* para todos los indicadores disponibles. Finalmente, se elige aquel indicador cuya probabilidad condicional sea mayor al del resto de indicadores, y superior al 5 %. Tras el proceso de diagnosis automático, se identifica un alto porcentaje de conexiones con mal rendimiento sin una causa identificada. Este tipo de conexiones son analizadas manualmente para un mejor entendimiento. Como resultado de este análisis se identifican 4 casos concretos que permiten explicar la causa de un gran número de dichas conexiones. En concreto, las causas son: a) el estrangulamiento del ancho de banda (*throttling*), b) comunicaciones a través de equipos fuera de la red, c) conexiones a través de la red RDSI, y d) conexiones pertenecientes a una misma sesión fragmentada en diferentes registros por parte de la aplicación TMA. Para la evaluación del método se utiliza un conjunto de datos obtenido de una aplicación TMA desplegada en una red 3G real. En total, se obtienen 165.325 conexiones de las cuales 8.266 tiene un valor de *throughput* TCP DL inferior al umbral mínimo (443 kbps para el conjunto de datos). De todas las conexiones con problemas de rendimiento, el 60 % son diagnosticadas con éxito. El 17 % restante de las conexiones no disponen de todos los indicadores por lo que el proceso de diagnosis no se puede realizar de forma correcta. El otro 23 % se corresponde con conexiones cuya causa no ha podido ser identificada de forma automática ni se corresponde con ninguno de los 4 casos específicos analizados.

Por último, las pruebas realizadas en entornos reales han mostrado la capacidad de estas herramientas para detectar problemas en redes móviles tras su despliegue. Alguno de los modelos se puede integrar dentro del conjunto de herramientas de

las redes autoorganizadas (*Self-Organizing Network*, SON) para la gestión de los diferentes servicios disponibles en la red [99].

Capítulo 5

Priorización automática de alarmas para la gestión de fallos en redes móviles

En los capítulos anteriores, se han explicado diferentes métodos y mecanismos para gestionar la QoE percibida por el usuario mediante su modelado y monitorización, haciendo uso de técnicas de BDA (*Big Data Analytics*). Estos procesos están principalmente enfocados en identificar posibles ajustes en la red para mejorar la calidad de experiencia que el usuario percibe en función del rendimiento de servicios específicos. Sin embargo, otro factor clave para asegurar dicha calidad es mantener un buen estado de la red, reduciendo el impacto de fallos en los elementos que la componen. Una degradación o malfuncionamiento de la red tiene un efecto directo en la satisfacción final del usuario. Por tanto, uno de los procesos más críticos en la gestión de la red y en la percepción de calidad por parte del usuario es, precisamente, la gestión de fallos de sus elementos.

Una de las fases clave en el proceso de gestión de fallos es la detección de dichos fallos, y la identificación de su gravedad. Este proceso se realiza mediante la recolección de alarmas generadas por los diferentes elementos de la red y su posterior análisis por parte del personal especializado. El principal reto de estas fases es el gran número de alarmas que deben ser analizadas diariamente (miles) para identificar aquellas que realmente están relacionadas con problemas críticos que requieren de acciones específicas por parte del personal técnico para la resolución del fallo. Generalmente, cuando una alarma es aislada debido a la gravedad

del incidente, esta requiere de la creación de un ticket de incidencia que debe ser resuelto por un técnico de mayor experiencia decidiendo la acción requerida a implementar. En este contexto, parece crucial identificar aquellas alarmas que finalmente requerirán de la creación de un ticket de incidencia para la restauración del servicio.

En este capítulo, se presenta un método automático para identificar y priorizar las alarmas generadas por los elementos de la red. El método permite identificar aquellas alarmas que, inevitablemente, requieren de la generación de un ticket de incidencia para un análisis posterior por parte del personal especializado, con el fin de resolver el problema en la red. De esta forma, se consigue reducir el número de alarmas que deben analizarse por el equipo a cargo de dicho proceso y, por tanto, reducir el tiempo de resolución.

El capítulo se divide en 5 secciones. La Sección 5.1 presenta una revisión del estado de la técnica para la gestión de fallos en la red móvil por parte de los operadores. En la Sección 5.2 se formula el problema de la gestión de fallos en las redes móviles. Tras formular el problema, en la Sección 5.3 se presenta un método automático para la priorización de alarmas a ser monitorizadas haciendo uso de técnicas de BDA. Posteriormente, la Sección 5.4 recoge los resultados obtenidos tras evaluar el modelo desarrollado en un escenario real. Finalmente, la Sección 5.5 expone las principales conclusiones de este capítulo.

5.1. Revisión del estado de la técnica

En un contexto en el que tanto los servicios como las redes son tan similares entre operadores, una óptima gestión de la red se presenta como un factor diferencial entre competidores. Uno de los procesos de gestión de la red más importante es la gestión de fallos (*Fault Management*, FM), para reducir el impacto que estos pueden tener en la red [68].

Como ya se ha indicado anteriormente, en el proceso de FM se recopilan estadísticas de rendimiento de los diferentes sistemas y enlaces de la red en tiempo real. Una vez se detecta un fallo o degradación de alguno de los servicios, éste es notificado a los administradores de la red mediante diferentes secuencias de alarmas. Una vez se identifica la causa del fallo, se intenta solucionar de forma remota o mediante una acción manual, si fuera necesario. Tradicionalmente, este

tipo de operaciones se ha realizado en el Centro de Operaciones de Red (*Network Operation Center*, NOC), donde un grupo de expertos localiza y resuelve los fallos mediante el análisis de todas las alarmas recopiladas de los diferentes segmentos de la red. Esto hace que el proceso de FM sea una de las tareas que requieren más tiempo y especialización de los procesos de gestión de la red [100]. Esto se ha visto acentuado en los últimos años debido al tamaño y la complejidad de las nuevas redes celulares, donde coexisten elementos de red de diferentes vendedores y tecnologías, lo que resulta en una gran cantidad de alarmas (decenas de miles por día). El principal reto para los administradores de red es identificar ese pequeño porcentaje de alarmas que realmente aportan información útil sobre el problema. Un escenario donde se requiere tanta intervención humana convierte el proceso de FM en una tarea poco fiable que no puede seguir siendo gestionada por los administradores de red de forma manual [101].

Los sistemas de FM son críticos para minimizar las pérdidas económicas y físicas frente a situaciones anormales en la red [19]. Sin embargo, se ha demostrado que, actualmente, estos sistemas no tienen la capacidad para gestionar de forma efectiva la gran cantidad de alarmas generadas en la red [20]. Por tanto, el diseño de nuevos sistemas de FM que puedan solventar ese problema se ha convertido en una tarea crucial para los operadores.

A lo largo de los años, el objetivo de los operadores ha sido el diseño de métodos que permitan reducir el número de alarmas gestionadas por los procesos de FM. El enfoque más extendido es la correlación de alarmas [22]. Éste se basa en el hecho de que un mismo fallo puede generar múltiples alarmas relacionadas, las cuales se notifican al NOC para su posterior inspección. La correlación de alarmas permite agruparlas reduciendo el porcentaje a ser monitorizadas [102]. Para ello, se han presentado una extensa variedad de propuestas de sistemas para la correlación de alarmas. Este tipo de sistemas se podrían agrupar en tres categorías dependiendo del tipo de algoritmo usado:

- Algoritmo de similitud. Se crean grupos de alarmas similares en base a reglas de asociación definidas heurísticamente o técnicas de ML (*Machine Learning*) más sofisticadas [103].
- Algoritmo basado en el conocimiento. Se basa en modelos o grafos causales que hacen uso de redes Bayesianas para relacionar alarmas [104].

- Algoritmo estadístico. Las alarmas son agrupadas en base a similitudes estadísticas [105].

El uso de estos sistemas no es algo reciente en redes de telecomunicaciones. En [49], se describe un modelo para predecir la propagación de alarmas a lo largo de la red tras el uso de herramientas de correlación en redes celulares. Del mismo modo, en [106] se presentan diferentes métodos para relacionar alarmas en episodios. Estos modelos se basan en el hecho de que diferentes alarmas generadas de forma secuencial están relacionadas con el mismo episodio y, por tanto, se pueden generar reglas para describir o predecir el comportamiento de dicha secuencia. Incluso el uso de redes neuronales para la correlación de alarmas en redes celulares ha sido ya discutido varios años atrás, como se recoge en [102]. En años más recientes, se pueden encontrar trabajos relacionados con este tema. En [107], se propone un nuevo método para agrupar alarmas que están relacionadas con causas del problema similares. En [108], se discute el problema de analizar, interpretar y reducir el número de alarmas antes de localizar el fallo en redes celulares. En [109], los autores proponen un sistema de detección de alarmas que aprende de anomalías anteriores en base a las condiciones del error y le permite detectar futuras ocurrencias. Más recientemente, en [110], se presenta un método de agrupación de alarmas en base a 3 grupos de fallos predefinidos.

Es tal la importancia del desarrollo de sistemas para la correlación de alarmas, que no solo se discute en las redes de telecomunicaciones, sino en otros sectores de la industria. Por ejemplo, en [111] se describe un sistema para la eliminación de alarmas irrelevantes, en base a repetitividad en plantas de procesamiento. Del mismo modo, en [112] se propone un algoritmo para identificar las alarmas más frecuentes en plantas de procesamiento, y aquellas que derivan en toda una secuencia de alarmas. Estas investigaciones continúan en la actualidad como demuestra el trabajo presentando en [113], donde se describe un sistema para la integración de información relacionada con procesos y alarmas obtenida de diferentes fuentes para desarrollar metodologías de monitorización que incluyen diagnóstico de fallos, evaluación de alarmas, y otro tipo de herramientas de análisis. Este sistema es validado en diferentes plantas de procesamiento tales como una planta de refinería de aceite o una planta de hidrógeno.

A pesar de los avances en el desarrollo de sistemas que permitan disminuir el número de alarmas, ninguno de ellos ha tenido aún éxito en reducir dicho número

a una sola alarma por incidente [114]. Incluso aunque así fuera el caso, seguiría siendo necesaria la intervención manual de un experto para determinar cuál de esas alarmas requieren un análisis posterior para la resolución del fallo. En este proceso, el equipo del NOC identifica aquellas alarmas que requieren de la creación un ticket de incidencia (a partir de ahora ticket, por simplicidad). Estos tickets contienen la información necesaria para poder iniciar las acciones correctivas. Esto refleja, claramente, la gran importancia que los tickets tienen en el proceso de FM. Es por ello por lo que se han realizado muchos trabajos de investigación recalcando su rol en estos procesos de FM, ya que están directamente relacionados con el fallo real detectado en la red [115][116]. Existen diferentes trabajos cuyo objetivo es relacionar las alarmas y los tickets. En [117], se presenta un método para incorporar, de forma automática, información añadida en los tickets en el proceso de correlación de alarmas. Este método está basado en el alineamiento temporal de los eventos (alarmas y tickets) que afectan a elementos comunes en la red. Sin embargo, el modelo heurístico usado es burdo y requiere de un conocimiento humano previo, ya que se usa la información añadida para enriquecer los datos del ticket. Además, sigue mostrando las mismas limitaciones en cuanto a la reducción del número de alarmas que los trabajos anteriores. Debido a esto, aún existe la necesidad de encontrar un mecanismo que permita aislar alarmas que, irrevocablemente, requieren de la creación de un ticket de incidencia.

Del mismo modo que se ha explicado en capítulos anteriores, el uso de técnicas de BDA es también de gran utilidad para optimizar los procesos de FM. Se pueden utilizar para caracterizar y clasificar alarmas en base a un conjunto limitado de parámetros o características. Con este fin, se han propuesto en la literatura una gran variedad de modelos de clasificación, que van desde simples reglas heurísticas hasta complejos algoritmos automáticos de ML basados en aprendizaje no supervisado (p. ej., k-means) o supervisado (p. ej., SVM o redes neuronales). Sin embargo, seleccionar el algoritmo que mejor se ajuste a los requisitos es aún un reto, ya que no existe un único clasificador que pueda resolver todo tipo de problemas. Con el fin de mitigar este problema, el uso de métodos de ensamble es una técnica comúnmente empleada para combinar la salida de diferentes clasificadores. Por ejemplo, en [118] se propone una metodología para construir clasificadores de ensamble formulando el problema de la selección del clasificador como un problema de optimización. Del mismo modo, en [119] se presenta otro método de ensamble que asegura suficiente diversidad entre los clasificadores base. Para ello, el método

divide los distintos clasificadores haciendo uso de un algoritmo de agrupamiento y genera un ensamble final seleccionando un clasificador por grupo.

En este capítulo, se describe un novedoso modelo para identificar y priorizar alarmas en base a su necesidad de generar un ticket de incidencia. Este modelo permite reducir el número de alarmas que deben ser analizadas manualmente por el equipo del NOC, permitiendo centrarse únicamente en aquellas alarmas que son realmente la causa del problema. Al igual que en [117], tanto la información de las alarmas como la de los tickets se utiliza para mejorar el proceso de resolución de incidentes. Sin embargo, a diferencia del trabajo mencionado, este modelo no hace uso de la información de los tickets para mejorar el proceso de correlación de alarmas con el fin de reducir el conjunto a una sola alarma por incidente. En su lugar, se utiliza para identificar, de forma automática, aquellas alarmas más prioritarias que requieren de la creación de un ticket para iniciar acciones de restauración lo antes posible. La base del modelo consiste en un método de ensamble basado en aprendizaje supervisado que estima la probabilidad de que una alarma requiera la creación de un ticket. El modelo se entrena y valida con datos, tanto de alarmas como de tickets, obtenidos de un NOC en una red real.

Las principales contribuciones de este capítulo son: a) adquirir un mejor entendimiento de cómo funciona el proceso de FM, así como las distintas etapas que lo componen, b) presentar un modelo predictivo que prioriza alarmas en base a la necesidad futura de acciones por parte del personal especializado, y c) la validación de dicho modelo con el uso de un conjunto de alarmas y tickets generados en una red comercial.

5.2. Formulación del problema

El proceso de FM tiene un papel clave en una correcta gestión de las redes móviles con el fin de mitigar el impacto que un fallo puede tener en el servicio. Dicho proceso se puede dividir en diferentes etapas que cubren desde, la generación de la incidencia generada por un fallo en algún elemento de la red, hasta la última acción requerida para la restauración del servicio. La Figura 5.1 muestra una metodología básica de la gestión de fallos en una red celular. Se observa como dicha gestión se puede dividir en 4 pasos: detección del fallo, notificación, diagnóstico de dicho fallo y resolución [120].

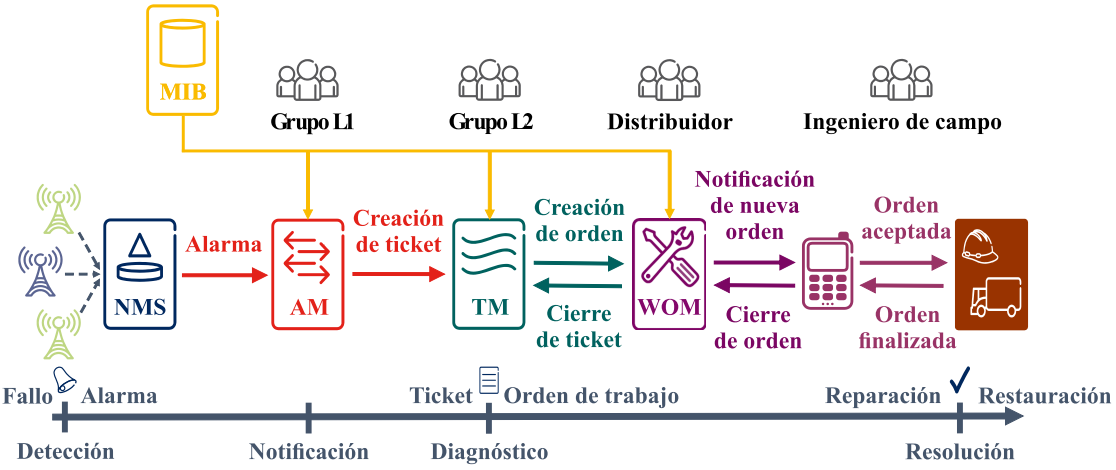


FIGURA 5.1: Flujo de trabajo del proceso de gestión de fallos en una red celular.

1. *Detección del fallo*: La primera etapa de detección proporciona al NMS (*Network Management System*) la capacidad de detectar y notificar los fallos generados en la red. Para ello, todos los elementos de la red deben reportar periódicamente su estado al NOC. Se pueden configurar dos modos de detección en el NMS: pasivo y activo [121]. En el modo pasivo, los agentes son configurados por el equipo de gestión para notificar al gestor cuando se cumple una determinada condición que podría derivar en un fallo en el sistema. Sin embargo, si el agente deja de funcionar, no se emite ninguna notificación y, por tanto, el fallo no se detecta. El modo activo resuelve este problema, ya que es el gestor el que comprueba, de forma periódica, el estado de cada agente enviando mensajes de control. En caso de que un agente no conteste a la petición con la información requerida, el gestor puede reaccionar generando una alarma para una investigación en detalle.

2. *Notificación del fallo*: Tras producirse un fallo, toda la información relacionada se envía al agente por parte del NMS, el cual la evalúa haciendo uso de un conjunto de reglas predefinidas. En el caso de cumplirse alguna de las reglas, el agente notifica al equipo de gestión mediante un mensaje que se visualiza en una consola o es recibido por correo electrónico. Esta notificación, conocida normalmente como *alarma*, incluye una breve descripción del fallo en el formato especificado por el vendedor del equipo [22]. Dicha descripción incluye información tal como el dispositivo/servicio que genera el fallo, un texto describiendo el problema, el tipo de fallo (equipamiento, comunicación, medioambiental ...), la severidad de la alarma (notificación, leve, crítica ...) y metadatos relacionados con el proceso (el identificador del

fallo, el tiempo de generación del fallo, el estado de la resolución ...). Todas estas alarmas son recopiladas y evaluadas por el equipo especializado durante el proceso de gestión de alarmas (*Alarm Management*, AM). En este proceso, las alarmas son etiquetadas en función de la categoría y prioridad con el fin de identificar las más críticas. El grupo L1 comprueba, en tiempo real, todas las alarmas para identificar aquellas que requieren de un análisis más detallado o acciones correctivas. Durante este proceso, la información de la alarma se enriquece con datos sobre el dispositivo afectado obtenidos del MIB (*Manager Information Base*). Para aquellas alarmas más críticas, que requieran de una evaluación más detallada, se genera un ticket de incidencia y se escala al grupo L2. Dichas alarmas han sido aisladas del resto asociadas al mismo fallo para una inspección más en detalle y para la resolución del fallo.

3. *Diagnóstico del fallo*: Tras la generación del ticket, el grupo L2 inicia un proceso de gestión del ticket (*Ticket Management*, TM) donde se realiza un análisis para obtener el diagnóstico de la causa del fallo. Por lo general, aquellos fallos que no derivan en la creación de ticket suelen repararse de forma automática.

4. *Resolución del fallo*: Una vez se ha identificado la causa del problema, se inician las acciones correctivas necesarias. En algunos casos, el fallo se puede arreglar de forma remota sin necesidad de intervención por parte del servicio de reparación. Sin embargo, en ciertos casos, la resolución del problema requiere de una acción física con presencia por parte del ingeniero de campo en el lugar donde el dispositivo se encuentra desplegado. En estos casos, el grupo L2 genera una orden de trabajo gestionada por el proceso de gestión de ordenes de trabajo (*Work Order Management*, WOM). Una vez la orden se crea, el distribuidor envía una notificación al ingeniero de campo correspondiente, quien debe aceptar la orden e iniciar las acciones correctivas (p. ej., reemplazar un componente defectuoso). Tras la resolución del fallo, y la posterior restauración del servicio, la orden de trabajo y el ticket asociado se cierran una vez el ingeniero completa la orden con la información de las acciones realizadas.

En base a lo explicado en el flujo de trabajo del proceso de FM, y al problema que suponen la ingente cantidad de alarmas que se pueden generar debido a un

mismo fallo en la red, la necesidad de desarrollar un método que permita reducir lo máximo posible el tiempo requerido entre la detección del fallo y la restauración final del servicio es crítica. De acuerdo con la Figura 5.1, el objetivo es seleccionar solo aquellas alarmas que están directamente relacionadas con el fallo y que requieren, forzosamente, de la intervención de un experto para la creación de un ticket que permita la resolución de la incidencia.

5.3. Modelo de priorización de alarmas

En esta sección, se presenta un modelo predictivo para la priorización de alarmas basada en la necesidad de la intervención de personal especializado (es decir, cuanto mayor es la prioridad, mayor es la necesidad de un especialista). El modelo utiliza la información de la alarma generada por un fallo en los elementos de la red y genera, a la salida, una predicción de si dicha alama requerirá la creación de un ticket de incidencia, por lo que debe ser priorizada. Para la creación y validación del modelo, dicha salida se compara con información de tickets generados por los técnicos del grupo L1.

La construcción del modelo se basa en la metodología CRISP-DM (*CRoss-Industry Standard Process for Data Mining*) [33]. Como ya se ha explicado anteriormente, esta metodología consiste en 6 etapas: entendimiento del negocio, entendimiento de los datos, preparación de los datos, modelado, evaluación y despliegue. El desarrollo de dicho modelo se ha realizado haciendo uso de una herramienta comercial para minería de datos propiedad de IBM (*International Business Machines*) llamada *IBM SPSS Modeler* [122]. Esta herramienta, basada en bloques de responsabilidad única interconectados, incorpora la metodología CRISP-DM, proporcionando un soporte efectivo para un proyecto de minería de datos. La Figura 5.2 muestra los bloques que componen el modelo de priorización de alarmas, delimitando algunas de las etapas definidas en la metodología. En la figura, los bloques con forma de círculo representan nodos para la importación de los datos usados como entrada en el modelo; los hexágonos son nodos para operaciones básicas (p. ej., mezcla de datos, derivación de campos o balance de datos); las estrellas, nombradas en la herramienta como *super nodos* permiten agrupar nodos básicos que realizan diferentes operaciones, y genera un enlace a un entorno de trabajo diferente donde solo se visualizan dichos nodos (es decir, se utilizan para organización y limpieza del entorno de trabajo principal); los pentágonos permiten

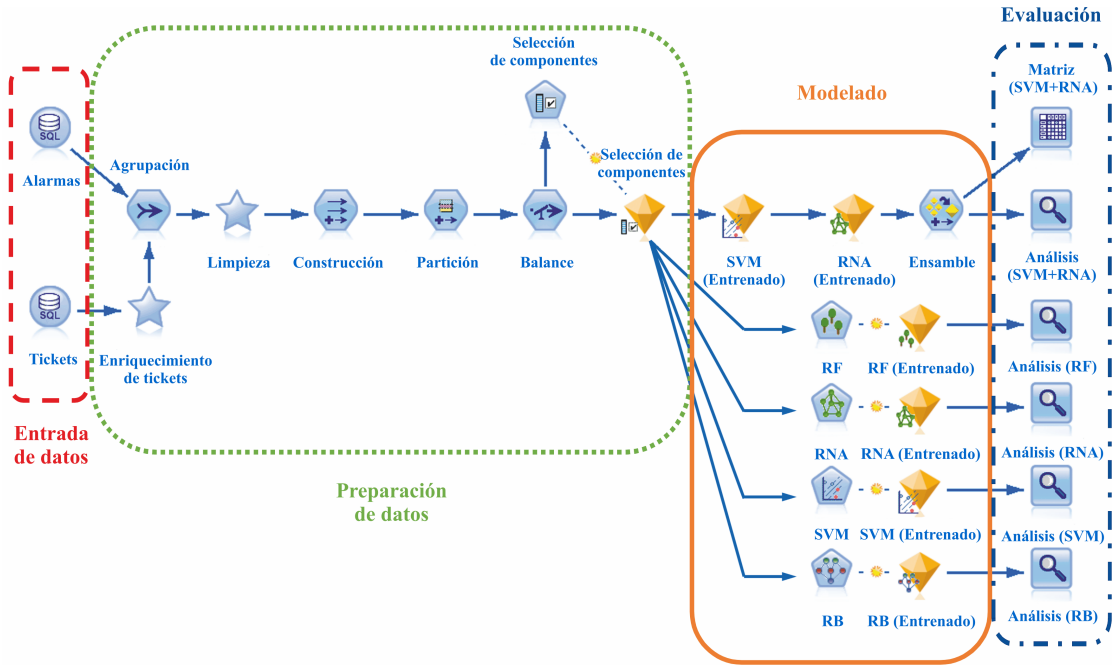


FIGURA 5.2: Etapas de procesamiento en el modelo de priorización de alarmas implementado en IBM SPSS Modeler.

realizar modelos a partir de datos (p. ej., detección de componentes principales, árboles de decisión o redes neuronales); los diamantes dorados, nombrados como *nugget de modelo* en la herramienta, generan contenedores para el resultado de los modelos generados, representando su salida; y, por último, los cuadrados permiten visualizar y/o exportar los datos a un formato adecuado para el uso en otra aplicación. Posteriormente, se detalla en profundidad cada uno de los nodos utilizados en el modelo.

A continuación, se describen cada una de las etapas de la metodología CRISP-DM aplicadas al caso de uso de la priorización de alarmas. Sin embargo, a etapa de entendimiento del negocio se omite ya se ha descrito detalladamente en la Sección 5.2.

5.3.1. Entendimiento de los datos

Adquirir un conocimiento de los datos usados en el proceso de FM es clave para la construcción de un modelo apropiado. Para el modelo descrito en este capítulo, se obtienen datos de dos fuentes de datos principales: alarmas y tickets de incidencia. Ambas fuentes se almacenan generalmente en bases de datos accesibles por el equipo del NOC. En esta etapa, la tarea consiste en explorar cada fuente

para identificar los diferentes datos incluidos y su naturaleza. Este proceso se representa en la Figura 5.2 mediante los nodos de importación de datos agrupados en el área definida por la línea discontinua roja. En general, el esquema de la base de datos de alarmas se compone de tres tipos de datos:

- Datos categóricos para describir el tipo de alarma (p. ej., identificador, tipo de fallo, severidad ...) y su origen (es decir, el identificador del elemento de red que la genera).
- Datos numéricos para indicar, principalmente, el número de ocurrencias de la alarma y códigos numéricos que pueden ser usados posteriormente por el equipo del NOC para diferentes análisis.
- Datos de fecha y hora para registrar los diferentes eventos y estados de la alarma (p. ej., creación de la alarma, notificación, cierre y/o borrado).

Del mismo modo, la información recopilada para los tickets de incidencia muestra un esquema similar en cuanto a tipo de datos, pero estos son enriquecidos con información añadida por los técnicos del grupo L1, como se ha mostrado en la Figura 5.1. En concreto, el tipo de datos que se puede encontrar con más frecuencia en las bases de datos de tickets son:

- Datos categóricos con información de la alarma asociada obtenidos de su base de datos correspondiente e información sobre el ticket (p. ej., descripción del ticket, solución al problema añadida tras la resolución, nombre del especialista del grupo L2 que se hace cargo de la incidencia ...).
- Datos numéricos con códigos añadidos por el equipo del NOC y asociados al identificador de la alarma.
- Datos de fecha y hora para trazar los diferentes estados del ticket.

Tras un análisis preliminar de los datos, se obtienen los siguientes hallazgos sobre el conjunto de datos utilizado:

- Con cada cambio de estado de una alarma o ticket (p. ej., creación, modificación, cierre o borrado), se crea un nuevo registro en la base de datos

correspondiente. Sin embargo, no se hace un mantenimiento de dichos datos para eliminar los registros anteriores. Debido a esto, en la base de datos se almacena la misma información (de ticket y/o alarma) con diferentes estados. Este comportamiento tiene un impacto directo en la creación del modelo ya que puede afectar a su construcción. Por tanto, se requerirá un proceso inicial de filtrado de datos para asegurar siempre un único registro por alarma/ticket con el último estado.

- Hasta que una alarma no se cierra completamente (es decir, se le asigna el estado de cerrado en la base de datos) esta puede generar un ticket de incidencia. Por tanto, utilizar alarmas aún abiertas puede generar un comportamiento inadecuado en el modelo pues consideraría ese tipo de alarmas como alarmas sin necesidad de ticket. Por tanto, es necesario otro proceso de limpieza de datos para filtrar aquellas alarmas que aún no han sido cerradas.

- La creación de los tickets se realiza de forma manual por los técnicos del grupo L1. Puede ocurrir que la información del ticket no contenga la alarma asociada por una mala práctica del técnico responsable o, simplemente, porque no sea un ticket asociado a una alarma sino a una queja de un cliente. Por tanto, este tipo de tickets debe filtrarse previamente a la construcción del modelo.

- No hay un balance entre ambos conjuntos de datos (es decir, existe un alto porcentaje de alarmas sin ticket asociado). Este comportamiento es normal en un proceso de FM ya que, como se ha mencionado anteriormente, no todas las alarmas requieren de la generación de un ticket para realizar acciones posteriores. Algunas de estas alarmas son solo informativas o, incluso, son alarmas asociadas a fallos que se pueden restaurar de forma automática.

5.3.2. Preparación de los datos

Esta etapa incluye la integración, el formateo, la limpieza, la construcción y selección de datos. Los hallazgos listados anteriormente, y considerados como cruciales para una correcta construcción del modelo, son atajados en la fase de preparación. Por tanto, aunque sea una etapa que requiera de mucho tiempo para completarse, es crítica para tener éxito en cualquier proyecto de minería de datos.

En primer lugar, se realiza una limpieza del conjunto de tickets almacenados en la base datos, con el fin de descartar aquellos que no están asociados a ninguna alarma. En la Figura 5.2, este proceso está representando por el nodo etiquetado como *Enriquecimiento de ticket de incidencia*. En el caso de tickets que han sido creados por petición del cliente, se descartan por considerarse fuera del alcance del modelo a construir. Para el resto de los tickets sin un identificador de alarma se inicia un proceso enriquecimiento en el que se cruza con información de alarmas para encontrar alguna coincidencia [117]. Para ello, se usa el identificador del elemento de red que genera la alarma, el emplazamiento donde se encuentra dicho elemento y los tiempos de creación y cierre de la incidencia. Por tanto, se considera que un ticket que fue creado para el mismo elemento de red y emplazamiento y en el mismo rango de tiempo que una alarma, tiene una alta probabilidad de estar relacionado con la misma incidencia y, por tanto, a estar asociado a dicha alarma. En algunos casos, esta asociación se valida mediante un análisis exploratorio de la información relativa a la alarma en el ticket (en el caso de estar disponible) haciendo uso de técnicas de minería de texto. El resultado de este proceso puede dar lugar a diferentes alarmas asociadas a un mismo ticket ya que no se busca filtrar alarmas sino enriquecer los datos de los tickets. Por tanto, cada alarma se considera como independiente, aunque dependan del mismo fallo.

Tras el proceso de enriquecimiento, ambos conjuntos de datos son agrupados en uno solo con el fin de ser utilizados como entrada del modelo para su entrenamiento. Antes, se realiza un proceso de limpieza para eliminar los casos detectados durante la etapa de entendimiento de los datos y que podrían causar un comportamiento inesperado en la respuesta. Tras la limpieza, se construye el indicador objetivo que se utilizará para entrenar el modelo. Para el caso de la priorización de alarmas, la variable utilizada consiste en un campo lógico que identifica aquellas alarmas que generan un ticket (dese ahora, *Con Ticket*) y aquellas otras que se cierran sin la generación de un ticket (desde ahora, *Sin Ticket*). A continuación, el conjunto de datos generado se divide en dos subconjuntos utilizados para el entrenamiento y la validación del modelo.

Posteriormente, se realiza un proceso de balance de datos. Como se ha identificado durante la fase de entendimiento de los datos, el número de casos en los que una alarma no requiere de acciones posteriores y, por tanto, no genera un ticket es mucho mayor que el alternativo (es decir, sí se genera un ticket asociado). La necesidad de este proceso reside en el comportamiento de algunos clasificadores

de ML (p. ej., *random forest* o *support vector machine*) frente a datos de entrenamiento no balanceados. En este tipo de situaciones, estos modelos tienden a favorecer la clase que mayor proporción de observaciones tiene, obteniendo a la salida predicciones que, aunque sean correctas en la mayoría de los casos (la clase con mayor proporción), no clasifica de forma precisa la clase con menor número de observaciones [123]. Por tanto, balancear los datos asegurando un porcentaje similar de muestras para cada tipo de clase permite solucionar este comportamiento [124]. Para ello, en el modelo de priorización de alarmas se utiliza una técnica de balanceo conocida como *undersampling*, que modifica la distribución de datos reduciendo el número de observaciones de la clase mayoritaria [125]. De esta forma se asegura un conjunto de datos con una proporción 50/50 entre las clases (es decir y en este caso, *Con Ticket* y *Sin Ticket*).

Finalmente, el último nodo de la etapa de preparación de datos realiza un proceso automático de selección de componentes. Este proceso tiene como objetivo reducir el número de componentes o variables (también llamadas, predictores) a utilizar en el modelo, identificando aquellas con el mayor impacto en la probabilidad de que una alarma genere un ticket (también llamada, variable dependiente). Para la construcción de este modelo, se ha usado un método de selección basado con el coeficiente de correlación de *Pearson* [126].

Tras concluir la etapa de preparación, los datos generados se pueden utilizar para la construcción del modelo.

5.3.3. Modelado

La primera decisión a la hora de construir el modelo es la selección del algoritmo de ML que mejor se ajuste a las necesidades del caso de uso. Los datos generados por el proceso de FM están claramente clasificados (es decir, se diferencian datos de alarmas y ticket). Esto permite crear un conjunto de datos etiquetado (es decir, alarmas con o sin tickets) para entrenar un modelo. Por tanto, el algoritmo que mejor se ajusta al caso de uso de la priorización de alarma, es un algoritmo supervisado. Para la construcción del modelo, se han evaluado cuatro clasificadores de ML:

- *Random Forest (RF)* [127]: Este algoritmo de aprendizaje construye diferentes modelos basados en arboles de decisión y, finalmente, combina sus

resultados para obtener estimaciones más precisas. En el proceso, se construyen diferentes árboles de decisión seleccionando subconjuntos de datos aleatorios del conjunto de entrenamiento. Finalmente, la salida de los distintos modelos se combina tomando la predicción más frecuente obtenida en todos ellos. Pese a que la salida de este tipo de modelos no se puede interpretar fácilmente, presenta una gran ventaja frente a los algoritmos basados en árboles de decisión clásicos, y es que previene el sobreajuste (es decir, evita que el modelo no ofrezca el mismo rendimiento con un conjunto de datos diferente al usado para su entrenamiento).

- *Red Neuronal Artificial (RNA)* [128]: Una RNA es una colección de unidades de procesamiento (neuronas) interconectadas entre si mediante enlaces. Las neuronas se agrupan en capas que realizan diferentes transformaciones a su entrada, proporcionando una salida específica en función de la capa y la neurona. La información viaja desde la capa de entrada hasta la de salida, pasando por diferentes capas intermedias, realizándose así el aprendizaje de la red. En este proceso, la salida de una neurona es transmitida aplicando diferentes pesos de multiplicación en función de la conexión. Gracias a esto, la RNA es capaz de modelar complejas relaciones no-lineales e, incluso, inferir relaciones que no se conocían inicialmente. Por otro lado, este tipo de modelos se comportan como una caja negra (es decir, no se puede interpretar la lógica definida) y, además, requieren de grandes conjuntos de datos previamente etiquetados y de una gran carga computacional.

- *Support Vector Machine (SVM)* [129]: El proceso de clasificación realizado por el algoritmo de SVM se realiza representando los distintos casos como puntos en un espacio multidimensional. Entonces, se construyen hiperplanos que separan los distintos casos en clases (en este caso, *Con Ticket* y *Sin Ticket*) con una diferencia tan grande como sea posible. Cada nuevo caso se intenta clasificar en función del lado del plano en el que es asignado. SVM se considera un algoritmo con una alta eficiencia ya que se define como un problema de optimización convexa (la función objetivo es cuadrática y las restricciones no son lineales), existiendo una gran variedad de métodos de solución.

- *Red Bayesiana (RB)* [123]: RB es un clasificador probabilístico binario basado en el teorema de Bayes. Calcula la probabilidad de cada una de las clases en el conjunto de datos utilizado, y la probabilidad condicional de cada clase

dado el valor de la clase anterior. Tras el entrenamiento, el algoritmo realiza las predicciones usando dicho teorema. Pese a que este algoritmo se basa en el hecho, poco probable, de que no hay dependencia entre los predictores, ha demostrado ser muy efectivo en una extensa variedad de problemas complejos.

Cada uno de estos modelos se ha utilizado para evaluar su rendimiento en el caso de la priorización de alarmas. En la Figura 5.2, estos modelos se representan con un diamante dorado. Además, se ha construido un quinto modelo haciendo uso del método de ensamble. Como ya se ha mencionado, este tipo de métodos permite combinar diferentes modelos para obtener una precisión mayor. Este tipo de métodos puede utilizarse para obtener diferentes mejoras: a) disminuir la varianza (*bagging*), b) reducir el sesgo (*boosting*), o mejorar las predicciones (*stacking*) [123]. Para la construcción del modelo descrito, se utiliza una técnica de *stacking* [130]. A diferencia de las técnicas de *bagging* y *boosting* (usada, por ejemplo, en RF), *stacking* permite ensamblar modelos de diferentes tipos, identificando la mejor forma de combinar la salida de cada uno de ellos haciendo uso de otra técnica de aprendizaje conocida como meta-learning. En base a esto, se construye un nuevo modelo basado en los modelos de SVM y RNA, que utiliza la salida de estos clasificadores para su entrenamiento. La selección de estos dos clasificadores se basa, principalmente, en la alta precisión de sus predicciones, como se muestra en la sección de resultados.

5.3.4. Evaluación

Tras el entrenamiento de los diferentes modelos descritos anteriormente, en esta etapa se realiza la evaluación de cada uno de ellos con el fin de seleccionar el que mejor rendimiento tiene a la hora de identificar alarmas que tienen un ticket asociado. En todo clasificador binario, siempre se distinguen cuatro categorías básicas:

- Verdadero positivo, denotado como el caso en el que un clasificador binario clasifica correctamente el caso positivo (*Con Ticket*, en este caso).
- Verdadero negativo, en el que se clasifica correctamente un caso negativo (*Sin Ticket*).

- Falso positivo, denotado como el caso en el que la clasificación del caso positivo es incorrecta (*Con Ticket* aunque la alarma realmente no tiene ticket asociado).

- Falso negativo, cuando se clasifica un caso negativo de forma incorrecta (*Sin Ticket* aunque la alarma realmente si tiene ticket asociado).

En base a estas estadísticas, se utilizan las siguientes métricas para evaluar los modelos construidos:

- *Precisión*: Es la proporción de predicciones correctas con respecto al total de muestras. Se define como

$$Precisión = \frac{VP + VN}{P + N}\,, \tag{5.1}$$

  donde VP y VN representan los verdaderos positivos y negativos respectivamente, y P y N representan el número total de valores positivo y negativo, respectivamente. Este tipo de métricas solo refleja de forma efectiva el rendimiento de un clasificador si el conjunto de datos está correctamente balanceado.

- *Sensibilidad*: Esta métrica, también conocida como probabilidad de detección, mide el porcentaje de positivos (en este caso, *Con Ticket*) que son correctamente predichos por el modelo (es decir, el ratio de verdaderos positivos).

- *Especificidad*: Mide el porcentaje de valores negativos (en este caso, *Sin Ticket*) que son correctamente predichos por el modelo (es decir, el ratio de verdaderos negativos).

- *Curva ROC (Receiver Operating Characteristic)* [131]: Representa, gráficamente, la proporción de verdaderos positivos frente a la de falsos positivos (es decir, 1-especificidad) según se varía el umbral de discriminación (umbral que determina cuando una muestra es clasificada como positiva en función de la probabilidad de serlo determinada por el modelo de que una muestra). La Figura 5.3 muestra una curva ROC para un modelo de clasificación de ejemplo. El eje x determina la tasa de falsos positivos, mientras que el eje y la de verdaderos positivos. La línea solida representa la curva ROC del modelo evaluado, y la línea discontinua representa la curva ROC de un modelo

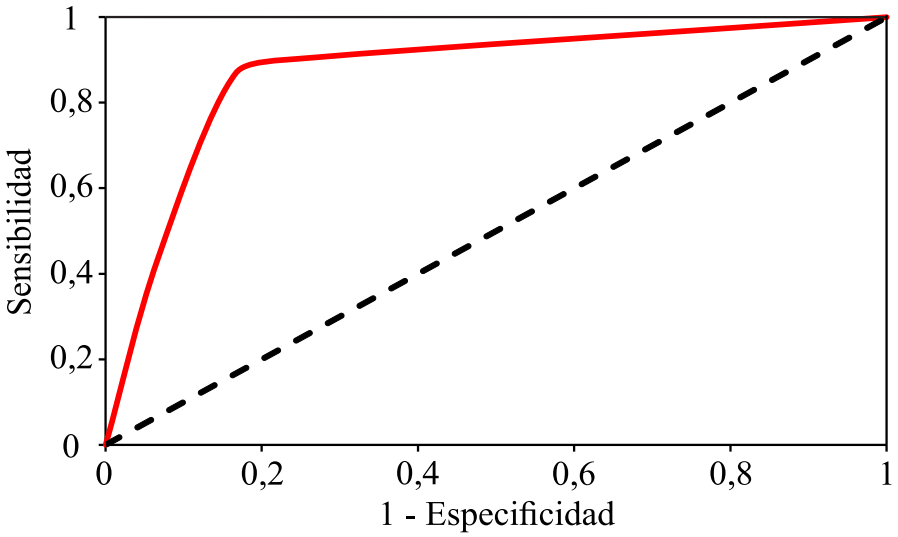


FIGURA 5.3: Evaluación de modelo clasificador mediante la curva ROC.

de clasificación aleatorio (es decir, un modelo sin capacidad de predicción). Para crear este tipo de curvas, se fijan diferentes umbrales de discriminación y se calcula la sensibilidad y especificidad del modelo en base a la probabilidad estimada por el modelo. Un modelo de clasificación perfecto estaría representado con una curva ROC con la forma de una función escalón.

- *Área Bajo la Curva (Area Under Curve, AUC)*: Representa la probabilidad de que el modelo pueda distinguir de forma correcta entre las dos clases (es decir, la positiva y la negativa), y se calcula como el área bajo la curva ROC. En la Figura 5.3, un modelo con una curva ROC con la forma de una función escalón (clasificador perfecto) tendría un $AUC = 1$, es decir, es capaz de distinguir perfectamente entre las dos clases. Por otro lado, si $AUC = 0{,}5$ (es decir, un modelo cuya curva ROC coincide con la línea discontinua de la Figura 5.3), el modelo no es capaz de clasificar de forma correcta las diferentes clases. Por tanto, mientras mayor sea el AUC de un modelo (es decir, lo más cercano a 1), mejor será.

5.4. Análisis de rendimiento

En esta sección se valida el modelo anteriormente descrito para priorizar alarmas en base a la necesidad de generar un ticket. Para ello, en primer lugar, se presenta la metodología experimental y, posteriormente, los resultados obtenidos. Finalmente, se describen algunas consideraciones de implementación.

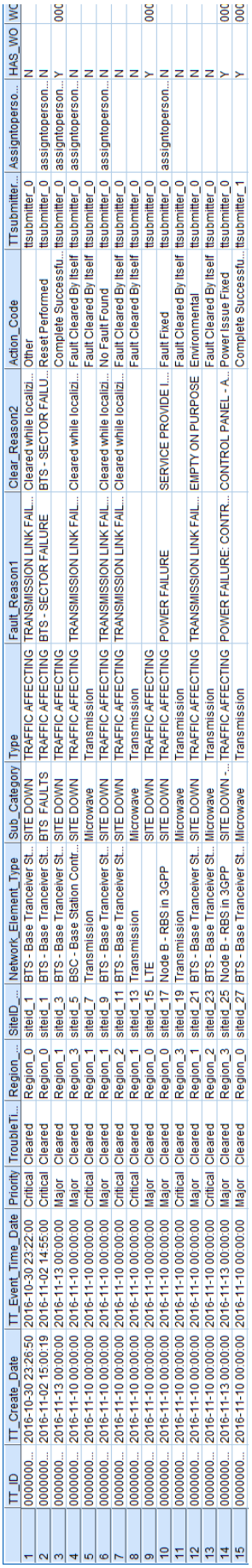
5.4.1. Metodología experimental

La evaluación del modelo se lleva a cabo en un escenario real sobre una red celular compuesta de 7.758 emplazamientos cubriendo un área geográfica de 705.589 km^2 con diferentes tecnologías de acceso radio desplegadas (GSM, UMTS y LTE). El conjunto de datos consiste, principalmente, en: a) datos de alarmas generadas por los diferentes elementos de la red, obtenidas del NMS, y b) datos de tickets generados por el equipo del NOC. En este trabajo, tanto los datos de alarmas como de tickets se han obtenido de una base de datos estructurada donde cada información de la alarma o el ticket está distribuida en diferentes tablas y columnas. La Figura 5.4 muestra una tabla con parte de la información de las alarmas recopiladas y usadas para la fase de entrenamiento y validación. En dicha tabla se muestran algunos de la información disponible para los datos de alarmas. Entre toda la información disponible, se puede diferenciar información contextual de la alarma (p. ej., identificador, información temporal de eventos de la alarma ...), información de localización del elemento de red que genera la alarma (p. ej., región, nodo, emplazamiento ...), información del tipo de alarma (p. ej., severidad, tipo de nodo, categoría, posible causa del fallo ...), e información relacionada con el ticket de incidencia asociada (en caso de que lo hubiera). Por otro lado, la Figura 5.5 muestra una tabla con la información parcial de los tickets recopilados en este trabajo. La información principal recopilada para los tickets se basa en información contextual del ticket (p. ej., identificador, información temporal del ticket, estado, prioridad ...), información de localización del elemento de red que genera la alarma (p. ej., región, nodo, emplazamiento ...), información contextual de la incidencia (p. ej., elemento de red, categoría, tipo, causa del fallo ...), e información relacionada con la gestión del ticket (p. ej., creador del ticket, persona asignada, acción realizada, orden de trabajo asociada ...).

Esta información se recopila durante un total de 3 meses, donde se registran un total de 5.877.444 alarmas y 44.637 tickets. Estos valores muestran, claramente, la diferencia entre las 2 clases (es decir, *Con Ticket* y *Sin Ticket*) que se indicó anteriormente resaltando la necesidad de balancear el conjunto de datos. En el proceso de evaluación, el conjunto de datos se divide en dos grupos, un conjunto de entrenamiento (con datos recopilados los 2 primeros meses) y conjunto de prueba (con datos recopilados el último mes).

	Serverserial	Region_...	Node_anon	Siteid_a...	Nodetype	Category	X733specificprob	Severity	Firstoccurrence	Lastoccurrence	Cleartime	Deletedat	TT_ID	Ttcreationtime	Ttclosuretime
1	96771267....	Region_0	node_0	siteid_0	Node B - RBS in 3GPP -...	MATE actioned	UtranCell_NbapMessageFailure	0.000	42675.009	42675.009	42675.009	42675.010		25569.042	25569.042
2	96805016....	Region_1	node_1	siteid_1	Transmission	MATE actioned	Site Abis control link broken	0.000	42675.019	42675.019	42675.019	42675.023		25569.042	25569.042
3	97085738....	Region_2	node_2	siteid_2	Node B - RBS in 3GPP	MATE actioned	UMTS Cell Unavailable	0.000	42675.132	42675.132	42675.136	42675.142		25569.042	25569.042
4	99233183....	Region_2	node_3	siteid_3	Node B - RBS in 3GPP	MATE actioned	UMTS Cell Unavailable	0.000	42676.081	42676.081	42676.084	42676.089		25569.042	25569.042
5	117401285....	Region_0	node_4	siteid_4	Transmission	Automated	Loss of Comms	0.000	42684.486	42684.486	42684.486	42684.488		25569.042	25569.042
6	117405498....	Region_0	node_5	siteid_5	Node B - RBS in 3GPP	MANUAL 3G ...	DigitalCable_Disconnected	0.000	42684.488	42684.488	42684.492	42684.494		25569.042	25569.042
7	117405525....	Region_2	node_6	siteid_6	Node B - RBS in 3GPP	Automated	BBU CPRI Interface Error	0.000	42684.480	42684.480	42684.480	42684.491		25569.042	25569.042
8	117408654....	Region_3	node_7	siteid_7	BTS - Base Tranceiver ...	MATE	SITE OUTAGE HUA	0.000	42684.489	42684.489	42684.498	42684.499		25569.042	25569.042
9	117414827....	Region_0	node_8	siteid_8	Node B - RBS in 3GPP	MANUAL 3G ...	FanFailure	0.000	42684.491	42684.491	42684.492	42684.493		25569.042	25569.042
10	117414975....	Region_0	node_9	siteid_9	BSC - Base Station Con...	Automated	SYNCHRONOUS DIGITAL PATH F...	0.000	42684.488	42684.488	42684.491	42684.493		25569.042	25569.042
11	117420638....	Region_0	node_10	siteid_10	Transmission	Automated	Loss of Comms	0.000	42684.495	42684.495	42684.495	42684.496		25569.042	25569.042
12	117426315....	Region_0	node_11	siteid_11	Node B - RBS in 3GPP	MANUAL 3G ...	DigitalCable_Disconnected	0.000	42684.495	42684.495	42684.496	42684.499		25569.042	25569.042
13	117426832....	Region_2	node_12	siteid_12	Node B - RBS in 3GPP	MATE	3G: SITE OUTAGE HUA	0.000	42684.496	42684.496	42685.247	42685.248	000000...	42684.497	25569.042
14	117427908....	Region_0	node_13	siteid_13	Node B - RBS in 3GPP	MANUAL 3G ...	DeviceGroup_GeneralHwError	0.000	42684.496	42684.496	42684.532	42684.535		25569.042	25569.042
15	99233875....	Region_4	node_14	siteid_14	BTS - Base Tranceiver ...	MATE actioned	The HDLC link is broken	0.000	42676.083	42676.083	42676.084	42676.090		25569.042	25569.042
16	117429709....	Region_3	node_15	siteid_7		MANUAL 2G ...	RF Unit Maintenance Link Failure	0.000	42684.488	42684.488	42684.515	42684.517		25569.042	25569.042

FIGURA 5.4: Previsualización de alarmas.



	TT_ID	TT_Create_Date	TT_Event_Time_Date	Priority	TroubleTi...	Region_...	SiteID_...	Network_Element_Type	Sub_Category	Type	Fault_Reason1	Clear_Reason2	Action_Code	TTsubmitter...	Assigntoperso...	HAS_WO	WO
1	0000000...	2016-10-30 23:26:50	2016-10-30 23:22:00	Critical	Cleared	Region_0	siteid_1	BTS - Base Tranceiver St...	SITE DOWN	TRAFFIC AFFECTING	TRANSMISSION LINK FAIL...	Cleared while localizi...	Other	ttsubmitter_0		N	
2	0000000...	2016-11-02 15:00:19	2016-11-02 14:55:00	Critical	Cleared	Region_0	siteid_1	BTS - Base Tranceiver St...	BTS FAULTS	TRAFFIC AFFECTING	BTS - SECTOR FAILURE	BTS - SECTOR FAILU...	Reset Performed	ttsubmitter_0	assigntoperson...	N	
3	0000000...	2016-11-13 00:00:00	2016-11-13 00:00:00	Major	Cleared	Region_1	siteid_3	BTS - Base Tranceiver St...	SITE DOWN	TRAFFIC AFFECTING			Complete Successfu...	ttsubmitter_0	assigntoperson...	Y	000
4	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_3	siteid_5	BSC - Base Station Contr...	SITE DOWN	TRAFFIC AFFECTING	TRANSMISSION LINK FAIL...	Cleared while localizi...	Fault Cleared By Itself	ttsubmitter_0	assigntoperson...	N	
5	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Critical	Cleared	Region_1	siteid_7	Transmission	Microwave	Transmission			Fault Cleared By Itself	ttsubmitter_0		N	
6	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_1	siteid_9	BTS - Base Tranceiver St...	SITE DOWN	TRAFFIC AFFECTING	TRANSMISSION LINK FAIL...	Cleared while localizi...	No Fault Found	ttsubmitter_0	assigntoperson...	N	
7	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Critical	Cleared	Region_2	siteid_11	BTS - Base Tranceiver St...	SITE DOWN	TRAFFIC AFFECTING	TRANSMISSION LINK FAIL...	Cleared while localizi...	Fault Cleared By Itself	ttsubmitter_0		N	
8	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Critical	Cleared	Region_1	siteid_13	Transmission	Microwave	Transmission			Fault Cleared By Itself	ttsubmitter_0		N	
9	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_0	siteid_15	LTE	SITE DOWN	TRAFFIC AFFECTING				ttsubmitter_0		Y	000
10	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_0	siteid_17	Node B - RBS in 3GPP	SITE DOWN	TRAFFIC AFFECTING	POWER FAILURE	SERVICE PROVIDE I...	Fault Fixed	ttsubmitter_0	assigntoperson...	N	
11	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Critical	Cleared	Region_3	siteid_19	Transmission	Microwave	Transmission			Fault Cleared By Itself	ttsubmitter_0		N	
12	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_1	siteid_21	BTS - Base Tranceiver St...	SITE DOWN	TRAFFIC AFFECTING	TRANSMISSION LINK FAIL...	EMPTY ON PURPOSE	Environmental	ttsubmitter_0		N	
13	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Critical	Cleared	Region_2	siteid_23	BTS - Base Tranceiver St...	Microwave	Transmission			Fault Cleared By Itself	ttsubmitter_0		N	
14	0000000...	2016-11-13 00:00:00	2016-11-13 00:00:00	Major	Cleared	Region_3	siteid_25	Node B - RBS in 3GPP	SITE DOWN -...	TRAFFIC AFFECTING	POWER FAILURE: CONTR...	CONTROL PANEL - A...	Power Issue Fixed	ttsubmitter_0		Y	000
15	0000000...	2016-11-10 00:00:00	2016-11-10 00:00:00	Major	Cleared	Region_0	siteid_27	BTS - Base Tranceiver St...	Microwave	Transmission			Complete Successfu...	ttsubmitter_1		Y	000

FIGURA 5.5: Previsualización de ticket de incidencias.

De las dos fuentes de datos principales, los datos de alarma se utilizan para la entrada del modelo. Los tickets permiten crear la variable dependiente del modelo. En total, la información de una alarma incluye 32 campos. Tras el proceso automático de selección de componentes, el número de campos se reduce a solo 15, considerados los realmente relevantes para la predicción. Estos campos proporcionan información sobre:

- La alarma. Indicando el tipo, la severidad o el número de ocurrencias, entre otros.
- El elemento afectado. Indicando el tipo de elemento y el vendedor.
- El dominio de red donde la alarma fue generada (p. ej., red de acceso radio, sistema de soporte de operaciones (OSS), núcleo de conmutación de paquetes).

Para la implementación de las diferentes etapas que componen la construcción del modelo (definidas por la metodología CRISP-DM), se usa la herramienta comercial IBM SPSS Modeler.

Como se ha indicado en la sección anterior, se han construido y evaluado cinco clasificadores binarios. Cuatro de ellos haciendo uso de algoritmos específicos (RF, RNA, SVM y RB), y un quinto que ensambla dos (RNA+SVM). Para evaluar los modelos, se hace uso de las métricas descritas en la subsección 5.3.4. La herramienta usada en este trabajo permite ajustar diferentes parámetros de configuración para la construcción de cada clasificador de forma específica. La Tabla 5.1 muestra los valores de los diferentes parámetros usados para cada modelo.

La evaluación de cada modelo se realiza, en primer lugar, comparando el valor de las métricas de precisión, sensibilidad y especificidad. Posteriormente, dicha evaluación se refuerza, de forma gráfica, con el uso de las curvas ROC para cada modelo, así como el valor del área bajo la curva (AUC). La generación de las curvas se realiza de forma automática por parte de la herramienta. En este proceso, para cada modelo, se altera el valor del umbral de confianza (valor mínimo de probabilidad estimada para determinar la clase positiva) y se obtienen tanto la sensibilidad como la especificidad obtenida. Posteriormente, se dibuja la curva ROC donde cada punto de la curva se corresponde con la combinación de sensibilidad y especificidad para cada umbral de confianza.

Tabla 5.1: Configuración de los modelos de clasificación binario.

Clasificador	**Parámetro**	**Configuración**
RNA	Número de unidades por capa	Automático
	Tiempo máximo de entrenamiento	15 min
	Regla combinación destino categóricos	Votación
	Regla combinación destino continuos	Media
	Conjunto de prevención de sobreajuste	30 %
	Estrategia valores perdidos en predictores	Suprimir
SVM	Orden de clasificación para destino categórico	Ascendente
	Precisión de regresión	0,1
	Función de penalización	L2
	Parámetro de penalización (λ)	0,1
RB	Datos en particiones	Entrenamiento
	Tipo de estructura	TAN
	Método de aprendizaje	Máxima verosimilitud
RF	Número de modelos a crear	100
	Tamaño de muestra	1
	Número máximo de nodos	10000
	Profundidad máxima de árbol	10
	Tamaño mínimo del nodo hijo	5
	Porcentaje máx. de valores perdidos	70 %
	Porcentaje máximo de categoría única	95 %
	Número máximo de categorías	49
	Variación mínima	0,05
	Número de intervalos	10
RNA+SVM	Campos generados por modelos	Filtrar
	Método de conjunto	Votación por confianza
	Estrategia de selección del valor	Mayor confianza

5.4.2. Resultados

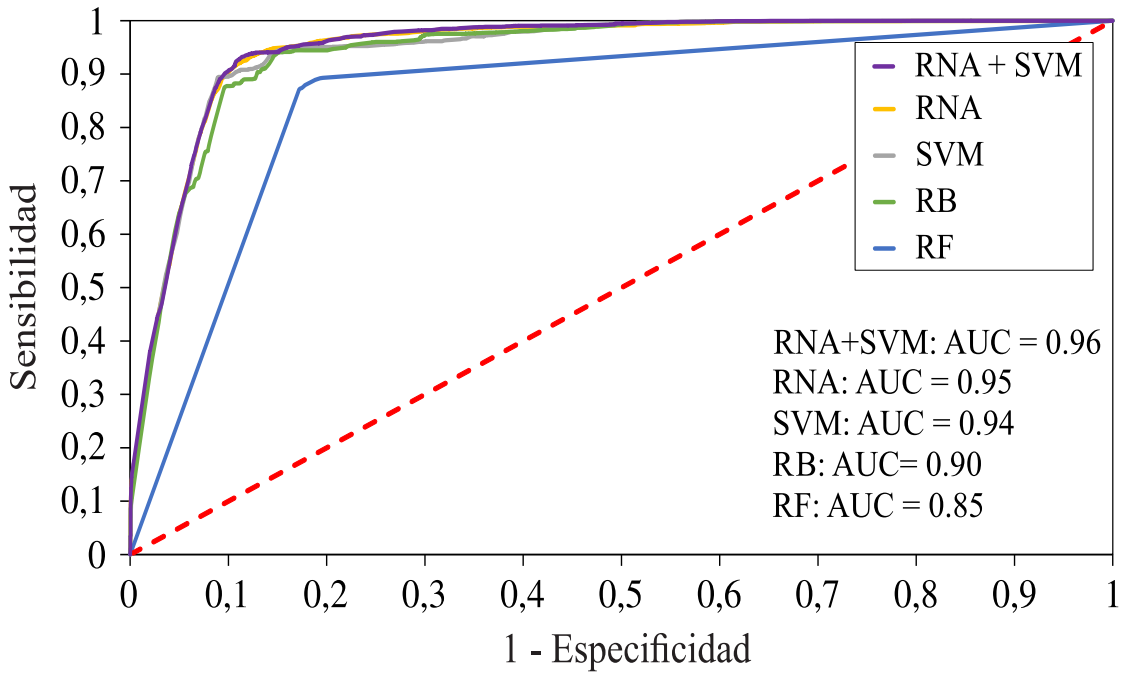
La Tabla 5.2 muestra el valor de las diferentes métricas obtenidas para cada modelo básico, descrito anteriormente, y las obtenidas tras el método de ensamble combinando el modelo RNA y SVM, tanto para el subconjunto de entrenamiento como de pruebas. Los resultados de cada uno de los clasificadores básicos (sin considerar el método de ensamble) muestran que los modelos RNA y SVM destacan con respecto al resto de modelos simples en términos de precisión a la hora de clasificar las alarmas. Específicamente, tanto el modelo RNA como SVM clasifican, de forma correcta, casi el 90 % de los casos. Además, los valores de sensibilidad (*Con Ticket*, correctamente clasificados) y especificidad (*Sin Ticket*, correctamente clasificado) son similares y cercanos al 90 %. Por otro lado, este comportamiento se muestra tanto en el conjunto de datos de entrenamiento como de prueba, lo que demuestra un modelo sin sobreajuste. El rendimiento de ambos modelos aumenta con el método de ensamble que combina ambos clasificadores para obtener un

Tabla 5.2: Comparación de los modelos de clasificación binario.

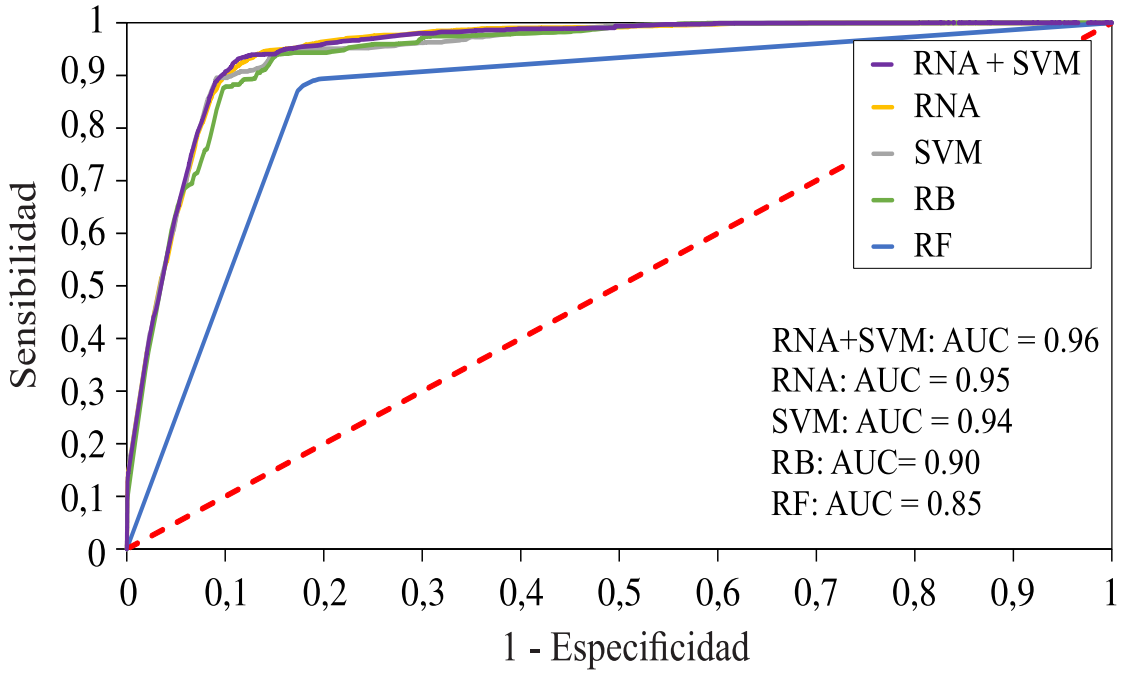
Clasificador	Subconjunto	Precisión	Sensibilidad	Especificidad
RNA	Entrenamiento	89,90 %	92,38 %	88,56 %
	Prueba	89,12 %	92,31 %	88,39 %
SVM	Entrenamiento	88,75 %	93,07 %	85,71 %
	Prueba	85,69 %	94,03 %	85,51 %
RB	Entrenamiento	85,30 %	88,05 %	82,28 %
	Prueba	83,92 %	89,94 %	84,05 %
RF	Entrenamiento	82,84 %	86,12 %	80,51 %
	Prueba	81,15 %	85,66 %	80,01 %
RNA+SVM	Entrenamiento	91,45 %	93,41 %	89,02 %
	Prueba	89,90 %	94,19 %	88,45 %

modelo final con valores de precisión, sensibilidad y especificidad por encima del de cada modelo por separado.

La misma validación se puede realizar gráficamente mediante el análisis de las curvas ROC. La Figura 5.6 muestra dicha curva, siendo la Figura 5.6 a) el conjunto de entrenamiento, y la Figura 5.6 b) el conjunto de prueba. En ambas figuras, el eje x representa la tasa de falso positivo (es decir, $1 - Especifidad$), mientras que el eje y representa la tasa de verdadero positivo (es decir, $Sensibilidad$). Cada una de las líneas solidas representan la curva de ROC de los 4 clasificadores básicos que se han utilizado para la construcción del modelo y el resultado del método de ensamble de los dos modelos con mejor rendimiento (es decir, RNA y SVM). La línea discontinua representa la curva de referencia, utilizada en la curva ROC para representar un clasificador incapaz de discernir de forma correcta entre clases. Por otro lado, en las subfiguras, se incluye el valor de AUC para cada modelo, facilitando su comparación. De estas subfiguras se puede concluir que los clasificadores básicos RNA y SVM ofrecen la mejor precisión en cuanto a estimación, reflejado en sus respectivas curvas de ROC al estar más alejadas de la línea de referencia. Con el método de ensamble de ambos clasificadores se consigue un rendimiento mejor, como refleja su correspondiente curva. De igual modo, sus valores de AUC (0,95 y 0,94, para los clasificadores RNA y SMV, y 0,96 para el resultado del método de ensamble) son superiores al de los otros modelos. Por otro lado, el uso de las curvas de ROC permite evaluar que ambos modelos se comportan de forma similar en ambos subconjuntos, lo que muestra la ausencia de sobreajuste.



(a) Conjunto de entrenamiento.



(b) Conjunto de prueba.

FIGURA 5.6: Comparación de rendimiento de los clasificadores binarios en base a la curva ROC.

5.4.3. Consideraciones de implementación

El modelo de priorización de alarmas se diseña como un esquema centralizado que puede ser integrado en el NOC de una red celular real. A pesar de su complejidad, la carga computacional es relativamente baja. Durante el proceso de construcción del modelo, las etapas que más tiempo consumen son la preparación de los datos y el entrenamiento del modelo, etapas que se podrían realizar previamente. Específicamente, el tiempo total de ejecución para el conjunto de datos utilizados en un portátil con un procesador de cuatro núcleos de 2,6 GHz, es de 7.000 s para la preparación de los datos, 2.035 s para el entrenamiento del modelo RNA, 1.370 s para SVM, 1.820 s para el modelo RF, 1.275 s para RB y 4.055 s para el modelo de ensamble de RNA+SVM. Una vez entrenados, el tiempo de ejecución para nuevas estimas haciendo uso de los datos de prueba no es superior a 15 s para cada modelo.

5.5. Conclusiones

En este capítulo, se ha propuesto un modelo para la priorización de alarmas en redes celulares en base a la necesidad de generar un ticket de incidencia. Dicho modelo está basado en un clasificador de aprendizaje supervisado. El modelo utiliza como datos de entrada las alarmas generadas en la red y reportadas al centro de operaciones. El objetivo del modelo es reducir el número de alarmas que el equipo del NOC debe monitorizar con el fin de restaurar el servicio tras un fallo. A diferencia de otros métodos basados en la correlación de alarmas generadas por el mismo fallo, el modelo descrito en este trabajo prioriza aquellas alarmas que, inevitablemente, requerirán de la creación de un ticket para la resolución del problema.

Para la creación del modelo se han evaluado cuatro clasificadores supervisados diferentes (RNA, RF, RB y SVM), así como un método de ensamble para combinar varios modelos básicos con el fin de mejorar el rendimiento global. Todos ellos han sido entrenados y validados con el mismo conjunto de datos. Para la validación del modelo se ha usado un conjunto de datos de alarmas y tickets obtenidos de una red celular real donde coexisten diferentes tecnologías de acceso radio. Los resultados han demostrado que los modelos de redes neuronales y SVM ofrecen un rendimiento superior al modelo de *Random Forest* y redes Bayesianas (89,12 %

y 86.69 % frente a 83,92 % y 81,15 %, respectivamente). Estos valores mejoran ligeramente cuando se combinan los modelos de RNA y SVM haciendo uso del método de ensamble (89,90 % de precisión).

Gracias a su baja carga computacional, este modelo puede ser totalmente integrado en los procesos de monitorización del NOC, permitiendo al equipo especializado identificar aquellas alarmas más prioritarias reduciendo costes económicos y de tiempo. Además, se aumentaría significativamente la satisfacción de usuario pues los fallos de la red tendrían menores tiempo de resolución. Este modelo se puede mejorar para priorizar no solo aquellas alarmas que requieran de la generación de un ticket de incidencia, sino aquellas alarmas que, finalmente, resultaran en la creación de una orden de trabajo y una acción correctiva por parte de un especialista.

Capítulo 6

Conclusiones finales

En este último capítulo se presentan las principales conclusiones de la tesis. En primer lugar, se destacan las contribuciones más importantes del trabajo. A continuación, se describen las posibles líneas extensión. Por último, se listan las diferentes publicaciones generadas durante el desarrollo de esta tesis.

6.1. Principales contribuciones

La popularidad reciente del uso de técnicas de BDA aplicadas a la gestión de redes celulares se debe, en gran parte, al crecimiento del número de usuarios y servicios de movilidad en la red, y a la gran complejidad de los procesos de gestión debido a la heterogeneidad de elementos y tecnologías que coexisten en dichas redes. Esto ha llevado a los operadores a buscar técnicas de gestión más modernas que, por un lado, se enfoquen más en la gestión de la calidad de experiencia y satisfacción de usuario, y, al mismo tiempo, simplifiquen y optimicen los procesos actuales para un funcionamiento adecuado de la red. En esta tesis se ha realizado un estudio detallado, análisis y experimentación de nuevas técnicas de BDA aplicadas a los procesos de monitorización y optimización de la QoE, así como al proceso de resolución de fallos de la red. Todo ello pretende mejorar los procesos de gestión de red para incrementar la satisfacción final del usuario proporcionando un servicio más individualizado y fiable.

El trabajo se ha iniciado con una descripción de los principales conceptos aplicados en este trabajo para mejorar los procesos actuales de gestión de la red, presentado en el Capítulo 2. Dentro de este capítulo, en primer lugar, se ha evaluado el uso de técnicas de BDA en el ámbito de las telecomunicaciones, identificando algunas de las áreas de aplicación más comunes, así como la revisión de las metodologías más comunes a la hora de realizar un proyecto de BDA, incluyendo la utilizada en este trabajo. En segundo lugar, se han presentado las técnicas SON y sus distintas categorías, siendo la autoopimización, la categoría que enmarca este trabajo. Por último, se han introducido los principales procesos de gestión de redes celulares, resaltando la importancia de la gestión de la QoE en los procesos de gestión de rendimiento y de fallos, así como detallando los pasos principales en los que se divide y que han sido cubiertos durante el desarrollo de este trabajo. Tras ello, y aún en el contexto de la gestión de red, se han explicado con detalle los componentes y roles implicados en el proceso de gestión de fallos.

La primera contribución de este trabajo es el método de ajuste automático de umbrales de QoS de las funciones de utilidad utilizadas para el modelado de la QoE, presentando en el Capítulo 3. En la formulación del problema, se ha puesto de manifiesto la importancia de actualizar los modelos actuales de QoE para los distintos servicios de movilidad mediante procesos automáticos que eviten la necesidad de ensayos y que tengan el menor impacto posible en la red. Para abordar este problema, se propone un método automático que hace uso de trazas de conexión para ajustar los valores de los diferentes umbrales de calidad de servicio que definen las funciones de utilidad. El método propuesto se compone de dos etapas. En la primera etapa, se realiza un proceso de clasificación de las trazas de conexión en diferentes servicios haciendo uso de, en un primer paso, del valor de QCI de cada conexión para, finalmente, utilizar un algoritmo de aprendizaje no supervisado que segrega los diferentes servicios en los grupos de QCI entre el 6 y el 9. Tras el proceso de clasificación, se realiza un proceso de estimación de los umbrales de QoS para los servicios más consumidos. Este nuevo método permite obtener los umbrales de QoS en entornos reales, permitiendo lidiar con una gran variedad de sistemas y factores humanos que no pueden considerarse con los métodos clásicos basados en pruebas en entornos de laboratorio. El método se ha evaluado haciendo uso de trazas de conexión obtenidas de una red LTE real. Los resultados validan el método tras estimar los nuevos umbrales de QoS para los servicios de datos de búfer completo, vídeo streaming y servicios de VoLTE.

La segunda contribución de esta tesis, presentada en el Capítulo 4, valida el uso de aplicaciones de monitorización y análisis del tráfico para la gestión de la QoE en redes celulares. En primer lugar, se ha presentado un esquema básico de este tipo de aplicaciones describiendo las diferentes capas que lo componen, y su funcionamiento dentro de la red. A continuación, se ha descrito una metodología genérica para poder validar el correcto funcionamiento de una aplicación TMA tras su despliegue en una red celular comercial. Esta validación permite ajustar los parámetros de configuración y los procesos definidos en la aplicación para obtener unas estimas, lo más precisas posible, de QoE. Los beneficios de la aplicación se han puesto de manifiesto con la presentación de diferentes casos de uso resolviendo diferentes procesos de gestión de la QoE. El proceso de validación de cada caso de uso se ha realizado utilizando los datos generados por una aplicación TMA desplegada en una red móvil comercial.

En la tercera contribución, presentada en el Capítulo 5, se propone el modelo para la priorización de alarmas en redes celulares en base a la necesidad de creación de un ticket de incidencia para la resolución del fallo detectado en la red. Durante la formulación del problema se ha identificado el principal reto en el proceso de gestión de fallos que debe afrontar el equipo especializado debido a la gran cantidad de alarmas que son reportadas y analizadas en los NOCs. El objetivo del modelo propuesto en este trabajo es reducir el número de alarmas que el equipo del NOC debe monitorizar reduciendo así los tiempos de restauración del servicio. A diferencia de los métodos clásicos para la reducción de alarmas mediante procesos de correlación, el modelo descrito en este trabajo prioriza aquellas alarmas que, inevitablemente, requerirán de la creación de un ticket de incidencia para la realización de acciones posteriores para la resolución del fallo. El modelo se ha construido utilizando diferentes algoritmos de aprendizaje supervisado (RNA, RF, RB y SVM), así como un método de ensamble para combinar el resultado de varios modelos. La evaluación del modelo se ha realizado haciendo uso de alarmas obtenidas de una red móvil real. Los resultados confirman que el modelo basado en la combinación de los algoritmos de RNA y SVM ofrece las mejores prestaciones para la estimación de alarmas que necesitan crear un ticket posteriormente.

Tanto el primer método desarrollado para el modelado de la QoE, como el modelo de priorización de alarmas emplean la información utilizada en los procesos de gestión de red y, por tanto, se pueden considerar como soluciones centralizadas. Por otro lado, gracias a su baja carga computacional, pueden ser totalmente

integrados en herramientas de optimización y monitorización automática para los respectivos procesos de gestión (es decir, QoE y fallos). Además, la validación de la aplicación de TMA se ha realizado utilizando la información generada de una aplicación desplegada en una red móvil real, por lo que demuestra claramente su utilidad y fiabilidad en este tipo de entornos.

El uso de diferentes herramientas para el desarrollo de los modelos y métodos de este trabajo ha permitido adquirir un mejor entendimiento de la diferencia en el uso de herramientas de propietario (p. ej., SPSS Modeler) de herramientas de código abierto (p. ej., TensorFlow, Keras, Pandas ...) para proyectos de BDA. Las principales diferencias son:

a) Licencia y coste: Las herramientas de propietario requieren una licencia para su uso. El coste puede ser significativo, especialmente para empresas que requieren de una licencia para cada trabajador. Las herramientas de código abierto están disponibles de forma gratuita para su uso, lo que lo hace más accesibles para usuarios individuales y organizaciones con presupuestos limitados.

b) Flexibilidad: Las herramientas de propietario suelen ofrecer una interfaz gráfica fácil de usar por los usuarios. En el caso de SPSS Modeler, usado en esta tesis, este permite crear y modificar modelos de forma visual, lo que lo hace idóneo para usuarios sin experiencia en programación. Sin embargo, la implementación y el ajuste de algoritmos específicos puede verse limitada en comparación con herramientas de código abierto, que permiten un alto grado de flexibilidad y personalización. En este tipo de herramientas, los usuarios tienen control sobre la arquitectura y el entrenamiento del modelo.

c) Soporte: Las herramientas de propietario tienen un soporte sólido proporcionado por el propietario, pero la comunidad de usuarios suele ser más reducida debido al inconveniente de la licencia. Sin embargo, las herramientas de código abierto suelen beneficiarse de una gran comunidad de usuarios y desarrolladores lo que permite aprender nuevas técnicas y mantenerse actualizado con las últimas innovaciones.

d) Integración: Las herramientas de propietario pueden integrarse fácilmente con otros productos del ecosistema del mismo propietario, pero pueden tener limitaciones para integrarse con sistemas de terceros. Por otro lado, las

herramientas de código abierto son altamente compatibles con una amplia gama de bibliotecas y herramientas, lo que facilita su integración con otros sistemas y flujos de datos.

El modelo de priorización de alarmas es el resultado de una iniciativa para crear un grupo expertos en ciencia de datos impuesta por el marco de financiación a través de la empresa Ericsson. El objetivo de este grupo de expertos era evidenciar las ventajas del uso de herramientas de propietario, como SPSS Modeler, para la creación de nuevos modelos basados en algoritmos de aprendizaje automático sin la necesidad de tener una experiencia técnica previa. Las principales responsabilidades del grupo eran: a) adquirir el conocimiento como experto de la herramienta SPSS Modeler, b) desarrollar modelos de optimización para la gestión de redes celulares, y c) guiar a nuevos científicos de datos en el uso de la herramienta.

6.2. Líneas futuras

En esta sección se proponen algunas de las posibles líneas de investigación derivadas del trabajo realizado en esta tesis.

- *Extensión del método de ajuste de umbrales a los nuevos tipos de servicios propios de los sistemas 5G y 6G.* La aparición de los sistemas 5G trae consigo la presencia de nuevos servicios que coexisten con los ya disponibles en las redes móviles 4G y anteriores (p. ej., servicios de comunicaciones entre máquinas, servicios de misión crítica ...). Estos servicios tendrán nuevas peculiaridades en cuando al uso de recursos y, del mismo modo, generarán comportamientos diferentes en su consumo por parte de los usuarios. Esto requerirá evolucionar el actual proceso de clasificación de servicios del modelo de ajuste, así como realizar un nuevo análisis para identificar los patrones de comportamiento del usuario y determinar los grados de insatisfacción por un mal rendimiento del servicio.

- *Mejora en el cálculo de las métricas realizadas por la aplicación de TMA.* Con el fin de reducir la carga computacional, algunas de las métricas calculadas por la aplicación TMA para la estima de los diferentes S-KPIs solo se

obtienen en momentos específicos durante la sesión del servicio. Por tanto, el número de muestras disponibles en los registros generados por la aplicación podrían no ser suficientes para asegurar una fiabilidad estadística. Por ejemplo, el tiempo de ida y vuelta (RTT) se estima generalmente durante el establecimiento de la comunicación del protocolo TCP. Este proceso solo se realiza al principio de la conexión, y por tanto solo existe un valor de RTT disponible en el registro. Para ser usado en la gestión de la QoE, el RTT debe ser medido periódicamente para detectar posibles fluctuaciones en la red. Por tanto, es necesario que la aplicación TMA también monitorice el tráfico de otras capas de protocolo.

- *Uso de modelos estadísticos para la estima de la QoE en caso de cifrado del tráfico.* En los últimos años, la mayoría de los proveedores de servicio han incluido cifrado en el tráfico por motivos de privacidad, lo que dificulta el proceso de clasificación del tráfico capturado por la aplicación TMA. Incluso si el proceso de clasificación se puede realizar, los mensajes de protocolo claves para la estima de ciertos S-KPIs (p. ej., RTT) no están disponibles cuando el tráfico está cifrado, por lo que debe realizarse por otros medios. Una posible solución consiste en el uso de modelos estadísticos que relacionan métricas TCP y S-KPIs sin la necesidad de estos mensajes específicos. Este tipo de modelos se realizan en entornos de laboratorio por lo que su inclusión en aplicaciones TMA podría facilitar su uso en entornos reales para estimar la experiencia final de usuario a partir de métricas TCP [132].

- *Resolución de los principales problemas en la monitorización del tráfico en redes 5G.* La aparición de los sistemas 5G generan nuevos problemas para el correcto funcionamiento de las aplicaciones TMA. Uno de esos problemas es la aparición de nuevos mecanismos de trasporte. Actualmente, la mayoría de las aplicaciones TMA se basan en el protocolo de comunicación TCP, cuyo control de flujo asegura que la dinámica del tráfico se relaciona con el rendimiento extremo a extremo. Este no es el caso del protocolo UDP (*User Datagram Protocol*). Del mismo modo, la existencia de múltiples flujos de datos en la misma conexión hace más difícil aislar el rendimiento de cada uno. Este es el caso del protocolo QUIC (Quick UDP Internet Connection) [133]. Por otro lado, la aparición de *proxies* para la mejora del rendimiento extremo a extremo (*Performance Enhancing Proxies*, PEPs) tiene un efecto negativo en la precisión de las estimas obtenidas por la aplicación TMA.

En redes celulares, los PEPs se configuran para algunos servicios (p. ej., navegación web) para mitigar problemas de retardo separando la conexión TCP en múltiples conexiones [134]. Sin embargo, estos sistemas modifican algunas métricas TCP obtenidas en el proceso de monitorización del tráfico (p. ej., el RTT), haciendo que estas medidas no puedan ser utilizadas para la estimación del rendimiento extremo a extremo. Otro reto introducido con la tecnología 5G es la segmentación de la red. La segmentación de la red es una de las tecnologías de virtualización clave en 5G, permitiendo a los operadores de red proporcionar redes virtuales dedicadas con funcionalidades específicas a un servicio o cliente sobre una infraestructura física común. Esto requiere almacenar nuevos parámetros de configuración de red en los registros generados por la aplicación TMA con el fin de realizar procesos de análisis precisos si se produce una degradación del servicio. Con el fin de poder utilizar, de forma apropiada, las aplicaciones TMA para la gestión de la QoE en las futuras redes 5G, este tipo de problemas deben ser resueltos.

- *Desarrollo de un modelo para la priorización de alarmas en base a la necesidad de acciones sobre el terreno.* El proceso de gestión de fallos termina habitualmente con la generación de una orden de trabajo y la actuación por parte de un ingeniero de campo para la reparación o reemplazo del elemento de red averiado. Identificar y priorizar aquellas alarmas que, inevitablemente, requerirán de acciones correctivas en el terreno mejorará ostensiblemente el proceso de gestión de fallos, reduciendo drásticamente los tiempos de resolución. Sin embargo, la ventaja fundamental de este modelo es el de proporcionar información relevante en la orden de trabajo que puede ser utilizada por el ingeniero para conocer de antemano las acciones correctivas a realizar. Esto permitiría seleccionar al personal especializado más adecuado en cada caso, además de facilitar la actuación evitando problemas ocasionados por no disponer del equipamiento y/o repuesto requerido en cada situación. Sin embargo, a diferencia de los tickets de incidencia, los datos contenidos en las ordenes de trabajo se corresponden con información no estructurada (p. ej., texto plano en lenguaje local), por lo que se requiere de técnicas de procesamiento de lenguaje natural para poder hacer uso de dicha información en el desarrollo del modelo.

6.3. Lista de publicaciones

El trabajo presentado en esta tesis ha generado los siguientes resultados de investigación:

Artículos

[I] A. García, V. Buenestado, S. Luna-Ramírez, M. Toril, J. M. Ruiz, "A geometric method for estimating the nominal cell range in cellular networks," *Mobile Information Systems (Hindawi)*, vol. 2018, Mayo 2018.

[II] A. García, M. Toril, P. Oliver, S. Luna-Ramírez, R. García, "Big Data Analytics for Automated QoE Management in Mobile Networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 91-96, Agosto 2019.

[III] A. García, M. Toril, P. Oliver, S. Luna-Ramírez, M. Ortiz, "Automatic alarm prioritization by data mining for fault management in cellular networks," *Expert Systems with Applications (Elsevier)*, vol. 158, 113526, Noviembre 2020.

[IV] A. García, C. Gijón, M. Toril, S. Luna-Ramírez, "Data-driven construction of user utility functions from radio connection traces in LTE," *Electronics (MDPI), Special Issue on Radio Access Network Planning and Management*, vol. 10, no. 7, pp. 829, Marzo 2021.

Aportaciones a congresos y reuniones científicas

[V] A. García, V. Buenestado, S. Luna-Ramírez, M. Toril, A. Mendo, "Cálculo del radio de celda basado en el diagrama de Voronoi para redes móviles LTE," *XXX Simposium de la Unión Científica Internacional de Radio (URSI)*, Pamplona (España), Septiembre 2015.

[VI] A. García, M. Toril, P. Oliver, V. Buenestado, S. Luna-Ramírez, "Estimación de umbrales de calidad de servicio para servicios móviles mediante trazas de conexión en LTE," *XXXI Simposium de la Unión Científica Internacional de Radio (URSI)*, Madrid (España), Septiembre 2016.

[VII] A. García, C. Gijón, M. Toril, S. Luna-Ramírez, "Data-driven construction of user utility functions from connection traces in LTE," TD(20)12040 *CA15104 IRACON*, Louvain-la-Neuve (Bélgica), Enero 2020.

[VIII] A. García, M. Toril, S. Luna-Ramírez, V. Buenestado, "Diagnóstico de conexiones problemáticas en redes celulares mediante herramientas de monitorización de tráfico," *XXXV Simposium de la Unión Científica Internacional de Radio (URSI)*, Málaga (España), Septiembre 2020.

El modelo de ajuste automático de los umbrales de QoS de las funciones de utilidad presentado en el Capítulo 3, se describe en [[VI], [VII], [IV]]. En [[VI]] se propone un primer método de ajuste que utiliza un modelo analítico para la clasificación de los diferentes servicios disponibles en la red. En [[VII]] y [[IV]] se mejora el método utilizando técnicas de aprendizaje automático no supervisado para el proceso de clasificación y extendiendo el proceso de ajuste a los servicios más demandados en la red móvil. En [[II]] se presenta la revisión del uso de aplicaciones de monitorización y análisis del tráfico para la gestión de la QoE, detallada en el Capítulo 4. En [[VIII]] se presenta uno de los casos de uso de las aplicaciones TMA incluidos en el Capítulo 4 para el diagnóstico de conexiones problemáticas. Finalmente, en [[III]], se presenta el modelo de priorización de alarmas descrito en el Capítulo 5.

Estas contribuciones se han desarrollado en el marco de los siguientes proyectos y contratos de investigación:

- *Desarrollo de funciones SON para la planificación y optimización de redes* (ref. 59288), contrato de colaboración entre Optimi-Ericsson y la Universidad de Málaga, financiado a través de la agencia IDEA de la Consejería de Ciencia, Innovación y Empresa de la Junta de Andalucía y cofinanciado con fondos FEDER de la Unión Europea [[I], [V], [VI]].
- *Optimización de la calidad de experiencia del servicio de videostreaming 3D en redes 4G* (TIN2012-36455), financiado por el Ministerio de Economía y Competitividad [[I], [V]].
- *Métodos de planificación y optimización de la calidad de experiencia en redes B4G* (TEC2015-69982-R), financiado por el Ministerio de Economía y Competitividad [[II], [VI]].

- *Métodos de planificación automática de redes 5g virtualizadas* (RTI2018-099148-B-I00), financiado por el Ministerio de Ciencia, Innovación y Universidades [[III], [IV], [VII], [VIII]].
- *Predicción de la calidad de experiencia en redes 5G* (UMA18-FEDERJA-256), financiado por la Consejería de Economía, Conocimiento, Empresas y Universidad de la Junta de Andalucía [[IV]].

Otras aportaciones

Otras publicaciones relacionadas con el tema central de esta tesis en las que se ha colaborado como coautor son:

[IX] A.B. Vallejo, M. Toril, A. García, P. Oliver, A. Mendo, S. Pedraza, 'Análisis de problemas de congestión en celdas del metro en una red LTE real," *XXIX Simposium de la Unión Científica Internacional de Radio (URSI)*, Valencia (España), Septiembre 2014.

[X] J.A. Fernández, I. de la Bandera, A. García, M. Toril, S. Luna, "Simulador estático de nivel de sistema del canal ascendente de datos en redes LTE," *XXIX Simposium de la Unión Científica Internacional de Radio (URSI)*, Valencia (España), Septiembre 2014.

[XI] J. L. Bejarano, M. Toril, M. Fernández, S. Luna, A. García, "A Context-Aware Data-Driven Algorithm for Small Cell Site Selection in Cellular Networks," *IEEE Access*, vol. 4, no. 4, pp. 417-420, Agosto 2015.

[XII] J.L. Bejarano, M. Toril, M. Fernández, R. Acedo, A. García, "Algoritmo de detección de huecos de cobertura usando información de redes sociales," *XXXIV Simposium de la Unión Científica Internacional de Radio (URSI)*, Sevilla (España), Septiembre 2019.

Bibliografía

[1] Cisco Systems, Inc. (2018) Cisco Annual Internet Report (2018-2023). Disponible en: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[2] R. Dangi, P. Lalwani, G. Choudhary, I. You, y G. Pau, "Study and investigation on 5G technology: A systematic review," *Sensors*, vol. 22, no. 1, p. 26, 2021.

[3] E. Liotou, D. Tsolkas, N. Passas, y L. Merakos, "Quality of experience management in mobile cellular networks: key issues and design challenges," *IEEE Communications Magazine*, vol. 53, no. 7, pp. 145–153, 2015.

[4] A. Kaloxylos, A. Gavras, y R. De Peppe, "Empowering vertical industries through 5G networks-current status and future trends," *White Paper*, 2020.

[5] A. Sufyan, K. B. Khan, O. A. Khashan, T. Mir, y U. Mir, "From 5G to beyond 5G: A Comprehensive Survey of Wireless Network Evolution, Challenges, and Promising Technologies," *Electronics*, vol. 12, no. 10, p. 2200, 2023.

[6] H. Fourati, R. Maaloul, L. Chaari, y M. Jmaiel, "Comprehensive survey on self-organizing cellular network approaches applied to 5G networks," *Computer Networks*, vol. 199, p. 108435, 2021.

[7] H. Fourati, R. Maaloul, y L. Chaari, "A survey of 5G network systems: challenges and machine learning approaches," *International Journal of Machine Learning and Cybernetics*, vol. 12, pp. 385–431, 2021.

[8] A. Bayazeed, K. Khorzom, y M. Aljnidi, "A survey of self-coordination in self-organizing network," *Computer Networks*, vol. 196, p. 108222, 2021.

[9] N. Barman y M. G. Martini, "QoE modeling for HTTP adaptive video streaming-a survey and open challenges," *IEEE Access*, vol. 7, pp. 30 831–30 859, 2019.

[10] T. Hori y T. Ohtsuki, "QoE and throughput aware radio resource allocation algorithm in LTE network with users using different applications," en *27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2016.

[11] N. Zabetian y B. H. Khalaj, "Quality of Experience (QoE)-based joint admission control and power allocation with guaranteed data rate," *Transactions on Emerging Telecommunications Technologies*, vol. 34, no. 10, p. e4837, 2023.

[12] M. Fiedler, T. Hossfeld, y P. Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, 2010.

[13] ITU-T, "One-Way Transmission Time," en *Recommendation G.114*, 2003.

[14] P. Oliver-Balsalobre, M. Toril, S. Luna-Ramírez, y J. M. R. Avilés, "Self-tuning of scheduling parameters for balancing the quality of experience among services in LTE," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, pp. 1–12, 2016.

[15] J. Navarro-Ortiz, J. M. Lopez-Soler, y G. Stea, "Quality of experience based resource sharing in IEEE 802.11 e HCCA," en *2010 European Wireless Conference (EW)*. IEEE, 2010.

[16] L. R. Jiménez, M. Solera, M. Toril, S. Luna-Ramírez, y J. L. Bejarano-Luque, "The upstream matters: impact of uplink performance on YouTube 360 live video streaming in LTE," *IEEE Access*, vol. 9, pp. 123 245–123 259, 2021.

[17] A. D'Alconzo, I. Drago, A. Morichetta, M. Mellia, y P. Casas, "A survey on big data for network traffic monitoring and analysis," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 800–813, 2019.

[18] M. Abbasi, A. Shahraki, y A. Taherkordi, "Deep learning for network traffic monitoring and analysis (NTMA): A survey," *Computer Communications*, vol. 170, pp. 19–41, 2021.

[19] B. R. Mehta y Y. J. Reddy, *Industrial process automation systems: design and implementation.* Butterworth-Heinemann, 2014.

[20] K. VanCamp, "Alarm management by the numbers," *Chem. Eng. Essentials CPI Prof., Autom. Control*, 2016.

[21] M. Li, M. Yang, y P. Chen, "Alarm reduction and root cause inference based on association mining in communication network," *Frontiers in Computer Science*, vol. 5, p. 1211739, 2023.

[22] G. Jakobson y M. Weissman, "Alarm correlation," *IEEE network*, vol. 7, no. 6, pp. 52–59, 1993.

[23] S. B. Das, S. K. Mishra, y A. kumar Sahu, "Alarm Coloring and Grouping algorithm for root cause analysis," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 5, pp. 572–579, 2021.

[24] M. A. Benatia, A. Louis, y D. Baudry, "Alarm correlation to improve industrial fault management," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 10 485–10 492, 2020.

[25] I. Khan, X. Zhang, M. Rehman, y R. Ali, "A literature survey and empirical study of meta-learning for classifier selection," *IEEE Access*, vol. 8, pp. 10 262–10 281, 2020.

[26] A. K. Sandhu, "Big data with cloud computing: Discussions and challenges," *Big Data Mining and Analytics*, vol. 5, no. 1, pp. 32–40, 2021.

[27] B. Lee, J. Oh, W. Shon, y J. Moon, "A Literature Review on AWS-Based Cloud Computing: A Case in South Korea," en *2023 IEEE International Conference on Big Data and Smart Computing (BigComp).* IEEE, 2023.

[28] M. Mahmoudian, S. M. Zanjani, H. Shahinzadeh, Y. Kabalci, E. Kabalci, y F. Ebrahimi, "An Overview of Big Data Concepts, Methods, and Analytics: Challenges, Issues, and Opportunities," en *2023 5th Global Power, Energy and Communication Conference (GPECOM).* IEEE, 2023.

[29] S. Sarker, M. S. Arefin, M. Kowsher, T. Bhuiyan, P. K. Dhar, y O. J. Kwon, "A Comprehensive Review on Big Data for Industries: Challenges and Opportunities," *IEEE Access*, 2022.

[30] M. Z. Kastouni y A. A. Lahcen, "Big data analytics in telecommunications: Governance, architecture and use cases," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 6, pp. 2758–2770, 2022.

[31] W. J. Frawley, G. Piatetsky-Shapiro, y C. J. Matheus, "Knowledge discovery in databases: An overview," *AI magazine*, vol. 13, no. 3, pp. 57–57, 1992.

[32] A. Azevedo y M. F. Santos, "KDD, SEMMA and CRISP-DM: a parallel overview," *Proc. Intl Assoc. Development of the Information Soc. European Conf. Data Mining*, pp. 182–185, 2008.

[33] R. Wirth y J. Hipp, "CRISP-DM: Towards a standard process model for data mining," en *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*. Manchester, 2000.

[34] K. Okokpujie, G. C. Kennedy, S. Oluwaleye, S. N. John, y I. P. Okokpujie, "An overview of self-organizing network (SON) as network management system in mobile telecommunication system," *Information Systems for Intelligent Systems: Proceedings of ISBM 2022*, pp. 309–318, 2023.

[35] A. Imran, A. Zoha, y A. Abu-Dayya, "Challenges in 5G: how to empower SON with big data for enabling 5G," *IEEE network*, vol. 28, no. 6, pp. 27–33, 2014.

[36] O. Østerbø, O. Grøndalen *y otros.*, "Benefits of self-organizing networks (SON) for mobile operators," *Journal of Computer Networks and Communications*, vol. 2012, 2012.

[37] J. L. Bejarano-Luque, M. Toril, M. Fernández-Navarro, A. J. García, y S. Luna-Ramírez, "A context-aware data-driven algorithm for small cell site selection in cellular networks," *IEEE Access*, vol. 8, pp. 105 335–105 350, 2020.

[38] R. Acedo-Hernández, M. Toril, S. Luna-Ramírez, y C. Ubeda, "A PCI planning algorithm for jointly reducing reference signal collisions in LTE uplink and downlink," *Computer Networks*, vol. 119, pp. 112–123, 2017.

[39] A. Kuurne *y otros.*, "Mobile measurement based frequency planning in GSM networks," *Helsinki University of Technology*, 2001.

[40] M. Toril, S. Luna-Ramírez, y V. Wille, "Automatic replanning of tracking areas in cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2005–2013, 2013.

[41] G. Americas, "The benefits of SON in LTE: Self-optimizing and self-organizing networks," *White paper*, 2009.

[42] N. Baldo, L. Giupponi, y J. Mangues-Bafalluy, "Big data empowered self organized networks," en *European Wireless 2014; 20th European Wireless Conference*, 2014.

[43] V. Buenestado, M. Toril, S. Luna-Ramírez, J. M. Ruiz-Avilés, y A. Mendo, "Self-tuning of remote electrical tilts based on call traces for coverage and capacity optimization in LTE," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 4315–4326, 2016.

[44] S. S. Mwanje y A. Mitschele-Thiel, "Distributed cooperative Q-learning for mobility-sensitive handover optimization in LTE SON," en *2014 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2014.

[45] A. B. Vallejo-Mora, M. Toril, S. Luna-Ramírez, M. Regueira, y S. Pedraza, "Analytical model for estimating the impact of changing the nominal power parameter in LTE," *Mobile Information Systems*, vol. 2018, 2018.

[46] P. Oliver-Balsalobre, M. Toril, S. Luna-Ramirez, y R. G. Garaluz, "Self-tuning of service priority parameters for optimizing quality of experience in LTE," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3534–3544, 2017.

[47] M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, y C. Gijón, "A QoE-driven traffic steering algorithm for LTE networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11 271–11 282, 2019.

[48] S. Hämäläinen, H. Sanneck, y C. Sartori, *LTE self-organising networks (SON): network management automation for operational efficiency*. John Wiley & Sons, 2012.

[49] P. Frohlich, W. Nejdl, K. Jobmann, y H. Wietgrefe, "Model-based alarm correlation in cellular phone networks," en *Proceedings Fifth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, 1997.

[50] M. Amirijoo, L. Jorguseski, R. Litjens, y L. C. Schmelz, "Cell outage compensation in LTE networks: algorithms and performance assessment," en *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*. IEEE, 2011.

[51] A. Gomez-Andrades, R. Barco, I. Serrano, P. Delgado, P. Caro-Oliver, y P. Munoz, "Automatic root cause analysis based on traces for LTE self-organizing networks," *IEEE Wireless Communications*, vol. 23, no. 3, pp. 20–28, 2016.

[52] A. Clemm, *Network Management Fundamentals*. Cisco Press, 2006.

[53] V. Dinesh Chandra, *Principles of computer systems and network management*. Springer, 2009.

[54] R. Schatz, M. Fiedler, y L. Skorin-Kapov, "QoE-based network and application management," en *Quality of Experience: Advanced Concepts, Applications and Methods*. Springer, 2014.

[55] L. Ji, J. Qian, y H. Wang, "E2E-QoE based 6G Sustainability: Challenges and Designing Aspects," en *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*. IEEE, 2023.

[56] ITU-T, "Vocabulary for performance and quality of service. Amendment 2: New definitions for inclusion," en *Recommendation ITU-T P.10/G.100*, 2008.

[57] G. Zheng y L. Yuan, "A review of QoE research progress in metaverse," *Displays*, p. 102389, 2023.

[58] S. Baraković y L. Skorin-Kapov, "Survey and challenges of QoE management issues in wireless networks," *Journal of Computer Networks and Communications*, vol. 2013, 2013.

[59] A. A. Barakabitze, N. Barman, A. Ahmad, S. Zadtootaghaj, L. Sun, M. G. Martini, y L. Atzori, "QoE management of multimedia streaming services in future networks: A tutorial and survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 526–565, 2019.

[60] A. Diaz, P. Merino, y F. J. Rivas, "Customer-centric measurements on mobile phones," en *2008 IEEE International Symposium on Consumer Electronics*. IEEE, 2008.

[61] C. Tselios y G. Tsolis, "On QoE-awareness through virtualized probes in 5G networks," en *2016 IEEE 21st international workshop on computer aided modelling and design of communication links and networks (CAMAD)*. IEEE, 2016.

[62] T. Hobfeld, R. Schatz, M. Varela, y C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 28–36, 2012.

[63] 3GPP, "Transparent end-to-end packet-switched streaming service (PSS); Protocols and codecs," en *Technical Specification 26.234 v10.1.0*, 2011.

[64] 3GPP, "IP multimedia subsystem (IMS); Multimedia telephony; Media handling and interaction," en *Technical Specification 26.114 v11.0.0*, 2011.

[65] L. Song, A. Mohammed, y A. Striegel, "A passive client side control packet-based WiFi traffic characterization mechanism," en *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–7.

[66] M. L. Marí-Altozano, M. Toril, S. Luna-Ramírez, y C. Gijón, "A self-tuning algorithm for optimal QoE-driven traffic steering in LTE," *IEEE Access*, vol. 8, pp. 156 707–156 717, 2020.

[67] M. L. Marí-Altozano, S. S. Mwanje, S. Luna-Ramírez, M. Toril, H. Sanneck, y C. Gijón, "A service-centric Q-learning algorithm for mobility robustness optimization in LTE," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 3541–3555, 2021.

[68] Cisco Systems, Inc., "Network Management System: Best Practices," White Paper, 2017.

[69] S. El Brak, M. Bouhorma, y A. A. Boudhir, "Network management architecture approaches designed for mobile ad hoc networks," *International Journal of Computer Applications*, vol. 15, no. 6, pp. 14–18, 2011.

[70] M. Ersue y B. Claise, "An Overview of the IETF Network Management Standards," *Internet Eng. Task Force, RFC 6632*, 2012.

[71] J. D. Case, M. Fedor, M. L. Schoffstall, y J. Davin, "Simple network management protocol (SNMP)," Reporte técnico, 1989.

[72] S. Waldbusser, "Remote network monitoring management information base version 2 using SMIv2," Reporte técnico, 1997.

[73] U. Warrier, L. Besaw, L. LaBarre, y B. Handspicker, "RFC1189: Common Management Information Services and Protocols for the Internet (CMOT and CMIP)," 1990.

[74] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, y W. Xiang, "Big data-driven optimization for mobile networks toward 5G," *IEEE network*, vol. 30, no. 1, pp. 44–51, 2016.

[75] M. Fiedler, S. Chevul, O. Radtke, K. Tutschku, y A. Binzenhöfer, "The network utility function: a practicable concept for assessing network impact on distributed services," en *19th International Teletraffic Congress (ITC19)*. Publishing House, BUPT, 2005.

[76] P. A. Sánchez, S. Luna-Ramírez, M. Toril, C. Gijón, y J. L. Bejarano-Luque, "A data-driven scheduler performance model for QoE assessment in a LTE radio network planning tool," *Computer Networks*, vol. 173, p. 107186, 2020.

[77] H. Ekstrom, "QoS control in the 3GPP evolved packet system," *IEEE Communications Magazine*, vol. 47, no. 2, pp. 76–83, 2009.

[78] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Policy and charging control architecture," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.203, 2015, version 13.7.0.

[79] F. Palmieri y U. Fiore, "A nonlinear, recurrence-based approach to traffic classification," *Computer Networks*, vol. 53, no. 6, pp. 761–773, 2009.

[80] T. T. Nguyen y G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE communications surveys & tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

[81] A. J. García, M. Toril, P. Oliver, S. Luna-Ramírez, y R. García, "Big data analytics for automated QoE management in mobile networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 91–97, 2019.

[82] V. F. Taylor, R. Spolaor, M. Conti, y I. Martinovic, "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic," en *2016*

IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, 2016, pp. 439–454.

[83] S. B. Banihashemi y E. Akhtarkavan, "Encrypted network traffic classification using deep learning method," en *2022 8th International Conference on Web Research (ICWR)*. IEEE, 2022, pp. 1–8.

[84] C. Gijón, M. Toril, M. Solera, S. Luna-Ramírez, y L. R. Jiménez, "Encrypted traffic classification based on unsupervised learning in cellular radio access networks," *IEEE Access*, vol. 8, pp. 167 252–167 263, 2020.

[85] L. R. Jiménez, M. Solera, y M. Toril, "A network-layer QoE model for YouTube live in wireless networks," *IEEE Access*, vol. 7, pp. 70 237–70 252, 2019.

[86] S. Na y S. Yoo, "Allowable propagation delay for VoIP calls of acceptable quality," en *International Workshop on Advanced Internet Services and Applications*. Springer, 2002, pp. 47–55.

[87] MathWorks. (2022, Septiembre) Matlab. Disponible en: https://www.mathworks.com/products/matlab.html

[88] A. Baer, P. Casas, A. D'Alconzo, P. Fiadino, L. Golab, M. Mellia, y E. Schikuta, "DBStream: A holistic approach to large-scale network traffic monitoring and analysis," *Computer Networks*, vol. 107, pp. 5–19, 2016.

[89] M. Tuexen, F. Risso, J. Bongertz, G. Combs, G. Harris, E. Chaudron, y M. Richardson, "PCAP Next Generation (pcapng) Capture File Format," 2016.

[90] D. Šipuš, "Big data analytics for communication service providers," en *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 2016, pp. 513–517.

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

[92] R. J. Tallarida, R. B. Murray, R. J. Tallarida, y R. B. Murray, "Chi-square test," *Manual of pharmacologic calculations: With computer programs*, pp. 140–142, 1987.

[93] Y. LeCun, Y. Bengio, y G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[94] S. Hochreiter y J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[95] D. P. Kingma y J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[96] S. Floyd y K. Fall, "Promoting the use of end-to-end congestion control in the Internet," *IEEE/ACM Transactions on networking*, vol. 7, no. 4, pp. 458–472, 1999.

[97] P. Chini, G. Giambene, y S. Kota, "A survey on mobile satellite systems," *International Journal of Satellite Communications and Networking*, vol. 28, no. 1, pp. 29–57, 2010.

[98] P. Bocker, *ISDN the integrated services digital network: Concept, methods, systems*. Springer Science & Business Media, 2012.

[99] P. V. Klaine, M. A. Imran, O. Onireti, y R. D. Souza, "A survey of machine learning techniques applied to self-organizing cellular networks," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.

[100] M. Awad y H. Hamdoun, "A framework for modelling mobile radio access networks for intelligent fault management," en *2016 Conference of Basic Sciences and Engineering Studies (SGCAC)*. IEEE, 2016, pp. 43–49.

[101] A. Bouillard, A. Junier, y B. Ronot, "Impact of rare alarms on event correlation," en *Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013)*. IEEE, 2013, pp. 126–129.

[102] H. Wietgrefe, K.-D. Tuchs, K. Jobmann, G. Carls, P. Fröhlich, W. Nejdl, y S. Steinfeld, "Using neural networks for alarm correlation in cellular phone networks," en *International Workshop on Applications of Neural Networks to Telecommunications (IWANNT)*. Citeseer Stockholm, Sweeden, 1997, pp. 248–255.

[103] R. Smith, N. Japkowicz, y M. G. Dondo, "Clustering using an Autoassociator: A Case Study in Network Event Correlation," en *IASTED PDCS*. Citeseer, 2005, pp. 613–618.

[104] Z. Zali, M. R. Hashemi, y H. Saidi, "Real-time intrusion detection alert correlation and attack scenario extraction based on the prerequisite-consequence approach," 2012.

[105] H. Ren, N. Stakhanova, y A. A. Ghorbani, "An online adaptive approach to alert correlation," en *Detection of Intrusions and Malware, and Vulnerability Assessment: 7th International Conference, DIMVA 2010, Bonn, Germany, July 8-9, 2010. Proceedings 7*. Springer, 2010, pp. 153–172.

[106] H. Mannila, H. Toivonen, y A. Inkeri Verkamo, "Discovery of frequent episodes in event sequences," *Data mining and knowledge discovery*, vol. 1, pp. 259–289, 1997.

[107] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis," *ACM transactions on information and system security (TISSEC)*, vol. 6, no. 4, pp. 443–471, 2003.

[108] J.-H. Bellec y M.-T. Kechadi, "Feck: A new efficient clustering algorithm for the events correlation problem in telecommunication networks," en *Future Generation Communication and Networking (FGCN 2007)*, vol. 1. IEEE, 2007, pp. 469–475.

[109] A. Makanju, A. N. Zincir-Heywood, y E. E. Milios, "Interactive learning of alert signatures in high performance cluster system logs," en *2012 IEEE Network Operations and Management Symposium*. IEEE, 2012, pp. 52–60.

[110] Y. Man, Z. Chen, J. Chuan, M. Song, N. Liu, y Y. Liu, "The study of cross networks alarm correlation based on big data technology," en *Human Centered Computing: Second International Conference, HCC 2016, Colombo, Sri Lanka, January 7-9, 2016, Revised Selected Papers 2*. Springer, 2016, pp. 739–745.

[111] J. Liu, K. W. Lim, W. K. Ho, K. C. Tan, R. Srinivasan, y A. Tay, "The intelligent alarm management system," *IEEE software*, vol. 20, no. 2, pp. 66–71, 2003.

[112] J. Folmer y B. Vogel-Heuser, "Computing dependent industrial alarms for alarm flood reduction," en *International Multi-Conference on Systems, Signals & Devices*. IEEE, 2012, pp. 1–6.

[113] W. Hu, S. L. Shah, y T. Chen, "Framework for a smart data analytics platform towards process monitoring and alarm management," *Computers & Chemical Engineering*, vol. 114, pp. 225–244, 2018.

[114] S. A. Mirheidari, S. Arshad, y R. Jalili, "Alert correlation algorithms: A survey and taxonomy," en *Cyberspace Safety and Security: 5th International Symposium, CSS 2013, Zhangjiajie, China, November 13-15, 2013, Proceedings 5*. Springer, 2013, pp. 183–197.

[115] G. Dreo y R. Valta, "Using master tickets as a storage for problem-solving," en *Integrated Network Management IV: Proceedings of the fourth international symposium on integrated network management, 1995*. Springer, 2013, p. 328.

[116] A. Medem, R. Teixeira, N. Feamster, y M. Meulle, "Determining the causes of intradomain routing changes," University Pierre and Marie Curie, Reporte técnico, 2009.

[117] S. Salah, G. Maciá-Fernández, y J. E. Díaz-Verdejo, "Fusing information from tickets and alerts to improve the incident resolution process," *Information Fusion*, vol. 45, pp. 38–52, 2019.

[118] H. Parvin, H. Alinejad-Rokny, B. Minaei-Bidgoli, y S. Parvin, "A new classifier ensemble methodology based on subspace learning," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 25, no. 2, pp. 227–250, 2013.

[119] H. Parvin, M. MirnabiBaboli, y H. Alinejad-Rokny, "Proposing a classifier ensemble framework based on classifier selection and decision tree," *Engineering Applications of Artificial Intelligence*, vol. 37, pp. 34–42, 2015.

[120] J. Ramiro y K. Hamied, *Self-organizing networks: self-planning, self-optimization and self-healing for GSM, UMTS and LTE*. John Wiley & Sons, 2011.

[121] G. Shields, *The Shortcut Guide to Network Management for the Mid-Market*. Realtimepublishers.com, 2007.

[122] International Business Machines Corporation (IBM). (2023) IBM SPSS software. Disponible en: https://www.ibm.com/products/spss-modeler

[123] I. H. Witten y E. Frank, "Data mining: practical machine learning tools and techniques with Java implementations," *Acm Sigmod Record*, vol. 31, no. 1, pp. 76–77, 2002.

[124] H. He y E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[125] X. Y. Liu, J. Wu, y Z. H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2008.

[126] G. Chandrashekar y F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[127] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[128] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[129] C. Cortes y V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, pp. 273–297, 1995.

[130] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[131] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[132] P. Fiadino, P. Casas, A. D'Alconzo, M. Schiavone, y A. Baer, "Grasping popular applications in cellular networks with big data analytics platforms," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 681–695, 2016.

[133] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar *y otros.*, "The quic transport protocol: Design and internet-scale deployment," en *Proceedings of the conference of the ACM special interest group on data communication*, 2017, pp. 183–196.

[134] X. Xu, Y. Jiang, T. Flach, E. Katz-Bassett, D. Choffnes, y R. Govindan, "Investigating transparent web proxies in cellular networks," en *Passive and Active Measurement: 16th International Conference, PAM 2015, New York, NY, USA, March 19-20, 2015, Proceedings 16*. Springer, 2015, pp. 262–276.