



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación
y Desarrollo Tecnológico

Tesis de Maestría

Identificación de patrones de congestión
vehicular

presentado por
Ing. Jorge Luis De Jesus Camacho

como requisito para la obtención del grado de
Maestro en Ciencias de la Computación

Director de tesis
Dr. Hugo Estrada Esquivel

Codirectora de tesis
Dra. Alicia Martínez Rebollar

Cuernavaca, Morelos, México. Enero de 2025.



Centro Nacional de Investigación y Desarrollo Tecnológico
Subdirección Académica

Cuernavaca, Mor.,

09/diciembre/2024

No. De Oficio:

SAC/379/2024

Asunto:

Autorización de impresión de tesis

JORGE LUIS DE JESUS CAMACHO
CANDIDATO AL GRADO DE MAESTRO
CIENCIAS DE LA COMPUTACIÓN
PRESENTE

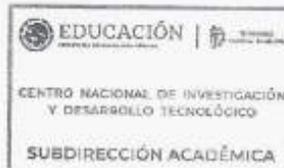
Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "IDENTIFICACIÓN DE PATRONES DE CONGESTIÓN VEHICULAR", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

Excellencia en Educación Tecnológica-
"Conocimiento y Tecnología al Servicio de México"

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO



C. c. p. Departamento de Ciencias Computacionales
Departamento de Servicios Escolares

CMAZ/fmz



Av. Inmersa Intermado Palmira S/N, Col. Palmira, C. P. 62400, Cuernavaca, Morelos
Tel. 01 (777) 3677770, ext. 4304, E-mail: acad_corridos@cenidet.tecnm.mx | acad@cenidet.tecnm.mx



2024
Felipe Carrillo
PUERTO



Centro Nacional de Investigación y Desarrollo Tecnológico
Departamento de Ciencias Computacionales

Cuernavaca, Mor., 29/noviembre/2024
OFICIO/DCC/284/2024

Asunto: Aceptación de documento de tesis
CENIDET-AC-MI4-OFICIO

CARLOS MANUEL ASTORGA ZAGOZA
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes del Comité Tutorial de **JORGE LUIS DE JESUS CAMACHO**, con número de control M22CE048, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis de grado titulado "IDENTIFICACIÓN DE PATRONES DE CONGESTIÓN VEHICULAR", y hemos encontrado que se han atendido todas las observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

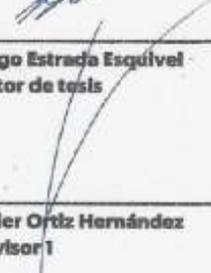
ATENTAMENTE
Excelencia en Educación Tecnológica
"Conocimiento y Tecnología al Servicio de México"



Dr. Hugo Estrada Esquivel
Director de tesis



Dra. Alicia Martínez Reboljar
Codirectora de tesis



Dr. Javier Ortiz Hernández
Revisor 1



Dr. José Jassón Flores Prieto
Revisor 2

C.c.p. Jefe (a) del Depto. de Servicios Escolares



Av. Interior Inter... N. Col. Palmira, C. P. 62490, Cuernavaca, Morelos
Tel. 01 (777) 3627770, ext. 3281, E-mail: doc@cenidet.tecnm.mx | cenidet.tecnm.mx



Dedicatoria

Con cariño dedico este trabajo a:

Mis padres.

Agradecimientos

Agradezco al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el apoyo económico brindado durante mis estudios de maestría.

A mi director de tesis, el Dr. Hugo Estrada, deseo expresar mi más sincera gratitud por su apoyo constante y su guía experta a lo largo de esta etapa crucial de mi formación académica. Su paciencia y la confianza que ha depositado en mi trabajo han sido pilares fundamentales en el desarrollo de esta investigación, y su liderazgo me ha inspirado a dar lo mejor de mí en cada paso del proceso.

A mi codirectora de tesis, la Dra. Alicia Martínez, le debo un agradecimiento especial por su incalculable contribución y compromiso. Su presencia atenta y sus consejos oportunos han sido de un valor inestimable para alcanzar este logro, y su apoyo ha representado una fuente constante de motivación.

A los distinguidos miembros de mi comité revisor, el Dr. Javier Ortiz y el Dr. José Jasson, les extiendo mi más profundo reconocimiento y agradecimiento por su dedicación, su tiempo y sus valiosos conocimientos. Sus observaciones y críticas constructivas no solo han enriquecido esta tesis, sino que también han contribuido significativamente a mi crecimiento como investigador. Su esmero y profesionalismo quedarán grabados en este proyecto y en mi desarrollo académico.

Quiero extender mi agradecimiento al Centro Nacional de Investigación y Desarrollo Tecnológico por brindarme el entorno propicio para llevar a cabo esta investigación.

Queridos padres, familia y seres queridos quiero agradecerles a por todo el apoyo que me han brindado.

Resumen

En esta tesis se abordan los problemas de movilidad urbana, específicamente en el área de congestión vehicular, buscando mejorar la comprensión y gestión de estas condiciones en entornos urbanos. El objetivo principal es identificar patrones de congestión a través del uso de algoritmos de aprendizaje automático, en particular, Random Forest.

El proceso inicia con la adquisición de datos de tráfico en tiempo real y su posterior preprocesamiento, asegurando su calidad y relevancia para el modelo. Durante el desarrollo, se aplicaron técnicas de balanceo de clases para manejar desbalances en los datos de congestión, utilizando SMOTE para optimizar el desempeño del modelo.

Además del desarrollo del modelo de identificación de congestión, se creó un prototipo web que permite visualizar los patrones de tráfico durante los siete días de la semana, las 24 horas del día. Este prototipo está diseñado para facilitar el análisis continuo del comportamiento de la congestión en distintos intervalos de tiempo, proporcionando una herramienta interactiva que permite a los usuarios observar las variaciones de tráfico en tiempo real o en función de horarios específicos. Así, se logra una representación visual de los niveles de congestión, lo que apoya en la toma de decisiones informadas para mejorar la movilidad urbana y gestionar los recursos de tráfico de manera más eficiente.

Finalmente, el modelo Random Forest implementado logra una exactitud del 80%, lo que permite una clasificación efectiva de los niveles de congestión en diversos segmentos e intervalos temporales. Este enfoque ofrece una herramienta robusta para la identificación de patrones de tráfico, apoyando el desarrollo de estrategias de mitigación de congestión en zonas urbanas.

Abstract

This thesis addresses the problems of urban mobility, specifically in the area of vehicular congestion, seeking to improve the understanding and management of these conditions in urban environments. The main objective is to identify congestion patterns through the use of machine learning algorithms, in particular, Random Forest.

The process begins with the acquisition of traffic data in real time and its subsequent preprocessing, ensuring its quality and relevance to the model. During development, class balancing techniques were applied to manage imbalances in congestion data, using SMOTE to optimize model performance.

In addition to the development of the congestion identification model, a web prototype was created that allows traffic patterns to be visualized seven days a week, 24 hours a day. This prototype is designed to facilitate the continuous analysis of congestion behavior at different time intervals, providing an interactive tool that allows users to observe traffic variations in real time or based on specific schedules. Thus, a visual representation of congestion levels is achieved, which supports informed decision-making to improve urban mobility and manage traffic resources more efficiently.

Finally, the implemented Random Forest model achieves an accuracy of 80%, which allows an effective classification of congestion levels in various segments and time intervals. This approach offers a robust tool for the identification of traffic patterns, supporting the development of congestion mitigation strategies in urban areas.

Contenido

Capítulo 1 Introducción	1
1.1 Introducción	2
1.2 Planteamiento del problema.....	2
1.3 Objetivo general.....	3
1.3.1 Objetivos específicos	4
1.4 Alcances.....	4
1.5 Limitaciones.....	4
Capítulo 2 Marco teórico	5
Capítulo 3 Estado del arte	9
3.1 Trabajos del estado del arte.....	10
3.2 Aspectos descriptivos de los trabajos del estado del arte	10
3.3 Identificación de patrones de congestión vehicular en ciudades	10
3.3.1 Infiriendo patrones de tráfico en cascada.....	10
3.3.2 Análisis de patrones de tráfico urbano mediante Clustering	12
3.3.3 Análisis de técnicas inteligentes para la predicción del flujo de tráfico urbano	12
3.3.4 Congestiónometro	14
.....	14
3.3.5 Metodología para obtener datos de tráfico e incidentes de tráfico a través de aplicaciones de mapas.....	15
Capítulo 4 Metodología de solución.....	16
4.1 Metodología de solución.....	17
4.1.1 Fase de modelado de datos	17
4.1.2 Fase de identificación de patrones	18

4.1.3 Fase de desarrollo de la interfaz de usuario	19
4.1.4 Fase de comunicación entre el algoritmo y la interfaz	19
4.2 Arquitectura del prototipo.....	19
4.2.1 Backend del prototipo	21
4.2.2 Frontend del prototipo.....	22
4.2.3 Integración del prototipo de visualización.....	22
Capítulo 5 Desarrollo identificación de patrones.....	23
5.1 Identificación de patrones de congestión vehicular	24
5.2 Implementación del modelado de datos.....	24
5.3 Implementación del algoritmo Random Forest.....	28
5.4 Identificación de patrones	30
5.4.1 Patrón de congestión libre.....	30
5.4.2 Patrón de congestión moderada	33
5.4.3 Patrón de congestión severa.....	36
5.4 Desarrollo de interfaz.....	38
5.5 Comunicación entre componentes	40
Capítulo 6 Pruebas y resultados	44
6.1 Experimentación	45
6.2 Procedimiento de la experimentación.....	45
6.2.1 Prueba de modelo de clasificación.....	45
6.2.2 Prueba de comparación	46
6.3 Resultados obtenidos	46
6.3.1 Prueba de modelo de clasificación.....	46
6.3.2 Resultados de las pruebas de funcionalidad.....	49

.....	52
.....	52
Capítulo 7 Conclusiones y trabajos futuros	55
7.1 Conclusiones.....	56
7.2 Trabajos futuros	57
Referencias.....	58
Bibliografía.....	59

Índice de figuras

Fig. 1 Patrones de congestión en cascada	11
Fig. 2 Metodología para la identificación de patrones en cascada	11
Fig. 3 Identificación de Patrones por Clustering	12
Fig. 4 Técnicas de predicción de flujo de tráfico.....	13
Fig. 5 Sistema de software Congestiónmetro.....	14
Fig. 6 Metodología para la adquisición de datos de tráfico	15
Fig. 7 Metodología de solución	17
Fig. 8 Arquitectura de prototipo.....	20
Fig. 9 Comunicación backend de prototipo	21
Fig. 10 frontend de prototipo	22
Fig. 11 Integración de backend y frontend	22
Fig. 12 Bounding box alcaldía Tlalpan.....	24
Fig. 13 Mapa alcaldía Tlalpan monitoreada por bounding boxes.....	25
Fig. 14 Dataset de datos obtenidos de Here maps	26
Fig. 15 Mapa de monitoreo para la identificación de patrones.....	26
Fig. 16 Dataset para el modelo random forest.....	28
Fig. 17 Prototipo para la visualización de patrones de congestión	39
Fig. 18 Visualización de niveles de congestión.....	40
Fig. 19 Código de backend para la comunicación con frontend.....	43
Fig. 20 Análisis de patrones de congestión nivel calle	50

Índice de tablas

Tabla 1 Patrón de congestión libre.....	32
Tabla 2 Patrón de congestión moderado.....	35
Tabla 3 Patrón de congestión severa.....	38
Tabla 4 Métricas de evaluación	47
Tabla 5 Exactitud del modelo Random Forest.....	49
Tabla 6 Curvas ROC.....	49
Tabla 7 Comparación de valores reales vs predichos para el día Lunes.....	51
Tabla 8 Comparación de valores reales vs predichos para el día Martes	51
Tabla 9 Comparación de valores reales vs predichos para el día Miercoles	52
Tabla 10 Comparación de valores reales vs predichos para el día Jueves.....	52
Tabla 11 Comparación de valores reales vs predichos para el día Viernes	52
Tabla 12 Comparación de valores reales vs predichos para el día Sábado.....	53
Tabla 13 Comparación de valores reales vs predichos para el día Domingo	53
Tabla 14 Evaluación de las predicciones del modelo versus datos adquiridos durante 5 semanas	54

Capítulo 1 Introducción

En este capítulo se presenta la problemática involucrada en el proyecto de tesis, el objetivo general y objetivos específicos de la tesis, así como los alcances y limitaciones del prototipo propuesto en este proyecto de investigación.

1.1 Introducción

La movilidad urbana es una condición necesaria para el crecimiento económico y el desarrollo social. Las personas se desplazan para buscar mejores oportunidades laborales o realizar diferentes actividades como compras, ocio, educación, recreación, etc. (Ian Thomson, 2001). Este escenario se presenta de forma más acentuada en condiciones urbanas donde la movilidad es un factor detonante para el crecimiento económico de la ciudad.

El aumento de la demanda de transporte y del tránsito vial en las ciudades modernas han traído como consecuencia la congestión vehicular, accidentes viales constantes y problemas ambientales como consecuencia de las emisiones contaminantes ambientales de los autos. México es la ciudad más congestionada de Norteamérica y la número 28 entre las metrópolis con el peor tráfico del mundo (TomTom, s.f.).

La medición del impacto de la congestión vehicular en una ciudad no es un tema simple, ya que implican la consideración de diversos parámetros como son la velocidad de circulación, el flujo de tráfico en una cierta dirección, el origen-destino de los viajes, el clima, entre otros (Michael A.P. Taylor, 2016). El análisis de cada uno de estos aspectos es complejo porque depende de factores como la hora y fecha de monitoreo, la dirección de circulación, etc.

Actualmente existe una tendencia en las ciudades para compartir datos en forma abierta y sea de pronóstico climático, calidad del aire, calidad del agua, información sísmológica, datos de transporte público, etc. Las fuentes pueden ser tanto servicios públicos como privados. En el caso de la movilidad en ciudades, los datos están transformando la forma de tomar decisiones en la elección de rutas de viaje, la utilización de servicios de transporte público o privado o las rutas más cortas o rápidas para alcanzar un destino en tiempo real.

Las aplicaciones de monitoreo de tráfico comerciales han comenzado a abrir sus datos, de tal forma que los usuarios pueden conocer información en tiempo real sobre las ubicaciones de los congestionamientos viales. Esto ha permitido que se haya mejorado mucho las estimaciones del tiempo que le tomará a una persona alcanzar un destino en la ciudad tomando en cuenta el nivel de congestión de las calles.

Las plataformas de monitoreo de tránsito proporcionan información sobre la capacidad disponible en la infraestructura vial en diferentes horarios y ubicaciones y, por lo tanto, se puede utilizar para analizar las futuras mejoras en infraestructura vial y aplicar una gestión dinámica del tráfico (Weijermars, 2007).

1.2 Planteamiento del problema

La movilidad es un desafío para las ciudades ya que, de acuerdo con la 18ª edición del 2022 el informe *Demographia World Urban Areas*, la población mundial que habita en ciudades es del 57%. Por lo tanto, las necesidades de la población requieren infraestructura vial eficiente y que permita el desplazamiento masivo de personas en la ciudad (DEMOGRAPHIA, 2022).

Las deficiencias en la infraestructura vial traen consigo problemas de congestión, las cuales tiene múltiples afectaciones en los habitantes de las ciudades: los tiempos muy grandes invertidos en los traslados, el alto costo económico, así como la inseguridad que se vive en los trayectos. Estos factores están deteriorando las condiciones de vida de la población que habita en ciudades.

El estudio de la plataforma *TomTom Traffic Index* señala que, en 2021 México ocupó el lugar número 28 de las ciudades con mayor tráfico y un tiempo extra en cada viaje del 38%. En 2018 México tenía el puesto número 9 y un tiempo de viaje extra del 52%. Por lo tanto, es importante tomar medidas adecuadas en la gestión del tráfico para facilitar la movilidad (TomTom, s.f.).

Una de las tendencias de las ciudades es poner a disposición de los ciudadanos los datos de movilidad urbana utilizando *API's*. Los datos generados se pueden utilizar para obtener más información sobre el sistema de tráfico urbano y la congestión que una incidencia vial puede provocar.

Las grandes ciudades pueden mejorar su movilidad urbana implementando políticas públicas, construcción de vialidades o mejoramiento de la infraestructura. Sin embargo, antes de implementar medidas es importante entender y analizar el comportamiento del tráfico de las ciudades, sobre todo el análisis de aquellos eventos relacionados con incidencias de movilidad, como son embotellamientos de tráfico, accidentes, eventos causados por clima, manifestaciones que obstruyen calle, tiempo en semáforos, embotellamientos, entre otros.

El sentido común nos indica que estos fenómenos antes mencionados tienen efecto directo en el tráfico que se produce en una ciudad, sin embargo, no tenemos certeza de cuál es el impacto que cada uno de estos eventos en el tráfico específico de una zona de la ciudad y tampoco conocemos si estos incidentes de tráfico se producen de forma repetitiva en ciertos días de la semana dependiendo la hora de medición y la localización, es decir no conocemos si existen un patrón que pueda caracterizar las incidencias de diversas zonas de la ciudad.

El desconocimiento de un patrón de movilidad urbana deja al ciudadano sin herramientas para conocer, por anticipado, el tráfico que podrá haber en una zona de la ciudad específica a una cierta hora y por lo tanto el usuario solo cuenta con los datos que pueda darse en tiempo real, lo cuales si bien reflejan el tráfico que se presenta en la ciudad en ese momento, no son representativos del tráfico que podría presentarse en días posteriores y a horas diferentes.

Para dar una solución a esta problemática, en este trabajo de investigación se propone utilizar los datos que producen las aplicaciones de movilidad *Here Maps* para identificar los patrones de congestión que se producen como resultado de incidencias viales en días laborales y días festivos en la Ciudad de México.

1.3 Objetivo general

El objetivo de este trabajo de investigación es utilizar los datos que produce la aplicación de movilidad *Here Maps* para identificar los patrones de congestión que se generan por movilidad vehicular en días laborales y fines de semana en la Ciudad de México.

1.3.1 Objetivos específicos

- Consolidar un dataset con datos históricos de congestión vehicular de la CDMX.
- Implementación de algoritmo de aprendizaje automático para identificación de patrones de congestión vehicular.
- Desarrollo de un sistema web para la visualización de los patrones detectados por el algoritmo de aprendizaje automático.

1.4 Alcances

- La tesis se enfocará en el análisis e identificación de patrones asociados a congestión vehicular de la alcaldía Tlalpan de CDMX.
- La identificación de patrones se enfocará a las calles que se ubiquen dentro de la alcaldía Tlalpan.
- La visualización de patrones de congestión será mostrada en un sistema web.

1.5 Limitaciones

- Los patrones identificados en esta investigación no consideran futuros cambios en infraestructura vial de la alcaldía Tlalpan.
- Se usarán las aplicaciones *Here Maps* como fuente de datos para la creación del dataset de movilidad.
- La implementación visual de los patrones se realizará únicamente en prototipo desarrollado para ejecutarse en localhost.
- Esta investigación descarta el análisis de más alcaldías o ciudades, aunque la metodología puede replicarse en otras alcaldías mientras se cumpla con los requisitos de entrada del sistema de identificación de patrones.

Capítulo 2 Marco teórico

En este capítulo se presentan los fundamentos teóricos más relevantes para el proyecto de tesis.

Ciudades Inteligentes

Una *Smart City* es un centro urbano que, sirviéndose de las nuevas tecnologías, pretende potenciar y volver más interconectada la vida de una ciudad. Su objetivo principal es mejorar los núcleos de movilidad y vivienda, para así fomentar un manejo más eficiente de energía, agua y residuos (BBVA, s.f.).

Una *Smart City* implica reorganizar la manera en que vive los ciudadanos, teniendo como prioridad la convivencia e interacción de las personas al crear edificaciones más ecológicas, y de esa forma se pueda ser parte de una ciudad sustentable (BBVA, s.f.).

Movilidad Urbana

La movilidad se conceptualiza en los desplazamientos origen-destino que tienen lugar en las ciudades, ya sea por medios de transporte motorizados o no motorizados, particulares o colectivos, haciendo referencia a la clasificación general de los modos de transporte que una persona puede utilizar para trasladarse de un lugar a otro (Quezada, 2017).

Gran parte del éxito de la movilidad en las ciudades tiene relación con la infraestructura, tanto por su capacidad como por su funcionamiento (Quezada, 2017).

La movilidad urbana presenta diferentes retos como la extensión y calidad del transporte público, infraestructura vial, retos ambientales, calidad en la salud pública, regulaciones, entre otros. Por tanto, es importante la participación ciudadana a fin de fortalecer aspectos culturales que permitan una movilidad ordenada, tal como respeto a los límites de velocidad, respeto a reglamentos o uso compartido de vehículo (Quezada, 2017).

Para mejorar el tema de la movilidad se requiere realizar inversiones que permitan que el desplazamiento constante genere beneficios económicos, situación que es difícil en entornos de crisis económica (Quezada, 2017).

En todas las ciudades, la cuestión de la movilidad es esencial, ya que es la movilidad la que permite la interacción de las personas y el comercio de bienes, los dos elementos definitorios de las razones de la existencia misma de las ciudades. Y, por supuesto, siendo la movilidad tanto en el corazón de la ciudad, también hay ejemplos muy diversos de patrones, intensidades y formas de movilidad (Quezada, 2017).

Actualmente las ciudades inteligentes están permitiendo transitar hacia el concepto de movilidad urbana inteligente, la cual se orienta a ofrecer servicios relacionados a la movilidad usando los datos que la ciudad produce en tiempo real. Lo que se busca es contar con una movilidad más conectada y sustentable utilizando tecnologías de información (Systems, s.f.).

Reconocimiento de Patrones

Es un área interdisciplinaria que se ocupa de los procesos sobre ingeniería, computación y matemáticas relacionados con objetos físicos o abstractos, con el propósito de extraer información que permita establecer propiedades entre conjuntos de dichos objetos, los cuales nos permitan interpretar el mundo que nos rodea (Jesús Ariel Carrasco Ochoa, 2011).

El reconocimiento de patrones se clasifica en cuatro enfoques, estadístico, sintáctico, lógico combinatorio y redes neuronales; El reconocimiento estadístico de patrones se basa en la teoría de probabilidad y estadística y supone que se tiene un conjunto de medidas numéricas con distribuciones de probabilidad conocidas y a partir de ellas se hace el reconocimiento. (Ochoa, 2018) El reconocimiento sintáctico de patrones, este enfoque se basa en encontrar las relaciones estructurales que guardan los objetos de estudio, utilizando la teoría de lenguajes formales. El objetivo es construir una gramática que describa la estructura del universo de objetos (Ochoa, 2018).

El reconocimiento de patrones lógico combinatorio se basa en la idea de que la modelación del problema debe ser lo más cercana posible a la realidad del mismo, sin hacer suposiciones que no estén fundamentadas. Uno de los aspectos esenciales del enfoque es que las características utilizadas para describir a los objetos de estudio deben ser tratadas cuidadosamente. (Ochoa, 2018). El siguiente enfoque son las redes neuronales. Este enfoque supone que tiene una estructura de neuronas interconectadas que se estimulan unas a otras, las cuales pueden ser “entrenadas” para dar una cierta respuesta cuando se le presentan determinados valores (Ochoa, 2018).

Minería de Datos

La minería de datos es el proceso de descubrir patrones y conocimientos interesantes a partir de grandes cantidades de datos. Las fuentes de datos pueden incluir bases de datos, almacenes de datos, la Web, otros repositorios de información o datos que se transmiten al sistema de forma dinámica (Jiawei Han, 2012).

Para resolver problemas de reconocimiento de patrones se han seguido diferentes enfoques, entre los cuales encontramos:

El aprendizaje automático investiga cómo los ordenadores pueden aprender (o mejorar su rendimiento) en función de los datos. Un área de investigación principal es que los programas informáticos aprendan automáticamente a reconocer patrones complejos y a tomar decisiones inteligentes basadas en datos. Por ejemplo, un problema típico de aprendizaje automático es programar una computadora para que pueda reconocer automáticamente los códigos postales escritos a mano en el correo después de aprender de un conjunto de ejemplos (Jiawei Han, 2012).

El aprendizaje supervisado es básicamente sinónimo de clasificación. La supervisión en el aprendizaje proviene de los ejemplos etiquetados en el conjunto de datos de entrenamiento (Jiawei Han, 2012).

El aprendizaje no supervisado es esencialmente sinónimo de agrupación. El proceso de aprendizaje no está supervisado, ya que los ejemplos de entrada no están etiquetados como clase. Normalmente, podemos usar la agrupación para descubrir clases dentro de los datos (Jiawei Han, 2012).

El aprendizaje semi-supervisado es una clase de técnicas de aprendizaje automático que hacen uso de ejemplos etiquetados y no etiquetados al aprender un modelo. En un enfoque, se utilizan ejemplos etiquetados para aprender modelos de clase y se utilizan ejemplos sin etiquetar para refinar los límites entre clases (Jiawei Han, 2012).

Congestión Vehicular

La causa fundamental de la congestión es la fricción entre los vehículos en el flujo de tránsito. La congestión es la condición que prevalece si la introducción de un vehículo en un flujo de tránsito aumenta el tiempo de circulación de los demás (Ian Thomson, 2001).

A medida que aumenta el tránsito, se reducen cada vez más fuertemente las velocidades de circulación. Algunos vehículos generan más congestión que otros. En la ingeniería de tránsito, se expresa cada tipo de vehículo en equivalencias pcu, que significa *passenger car unit*, o unidades de coches para personas. Un auto tiene una equivalencia de 1, y los demás una equivalencia correspondiente a su influencia perturbadora sobre el flujo de tránsito, o el espacio vial que efectivamente ocupan, en comparación con un auto (Ian Thomson, 2001).

Capítulo 3 Estado del arte

En este capítulo se presenta la revisión del estado del arte con el objetivo de presentar un panorama general en los trabajos de investigación que se consideraron relevantes para el desarrollo de este trabajo de investigación.

3.1 Trabajos del estado del arte

En esta sección se describen los artículos relacionados a la identificación de patrones de congestión vehicular.

3.2 Aspectos descriptivos de los trabajos del estado del arte

Los trabajos del estado del arte han sido descritos utilizando un conjunto de criterios con la finalidad de realizar una descripción uniforme y consistente de todos y cada uno de los trabajos relacionados. Los criterios utilizados son los siguientes:

- Descripción: en este apartado se muestra una breve descripción del trabajo analizado.
- Objetivo: describe el objetivo principal que persigue el autor con su trabajo de investigación.
- Arquitectura o metodología de solución: describe el conjunto de procedimientos utilizados en el trabajo de investigación analizado.
- Pruebas realizadas: describe el conjunto de pruebas empleadas en el trabajo de investigación.
- Conclusiones: se describe las conclusiones del trabajo analizado

3.3 Identificación de patrones de congestión vehicular en ciudades

3.3.1 Infiriendo patrones de tráfico en cascada

El artículo *Inferring Traffic Cascading Patterns* (Yuxuan Liang, 2017) tiene como objetivo proporcionar información de la manera que se comporta el tráfico en cascada. En la (*fig. 1(a)*) podemos observar un posible comportamiento del tráfico, siendo R_1, R_2, R_3 hasta R_n segmentos de calles de la red urbana, en la (*fig. 1 (b)*) podemos ver una serie de combinaciones o posibles afectaciones de las calles, los patrones pueden tener variaciones de afectación dependiendo del día y hora. En la (*Fig. 1 (c)*) nos muestra el patrón en cascada algo más parecido a un grafo dirigido mostrando que si existe una afectación en R_1 afectara a R_3 y a su vez afectara a R_4 y R_5 . La identificación de patrones en cascada nos puede ayudar a predecir las condiciones del tráfico e identificar los cuellos de botella.

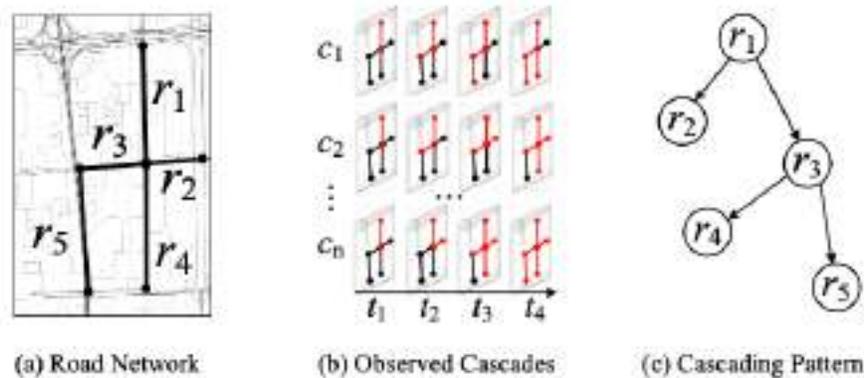


Fig. 1 Patrones de congestión en cascada

La identificación de patrones en cascada es difícil por tres factores, la interacción con los usuarios mediante tweets, las múltiples fuentes de datos y la correlación geoespacial. Al no tener un método para unir toda la información simultáneamente en tiempo real hace difícil identificar los patrones.

La metodología propuesta por los autores es un enfoque probabilístico para la inferencia de patrones en cascada como se muestra en la fig. 2 mediante el CasInf que es un modelo de Probabilidad de Transmisión Individual (ITL), un modelo de intensidad Ambiental (EMT) y un algoritmo de Construcción de Patrones en Cascada.

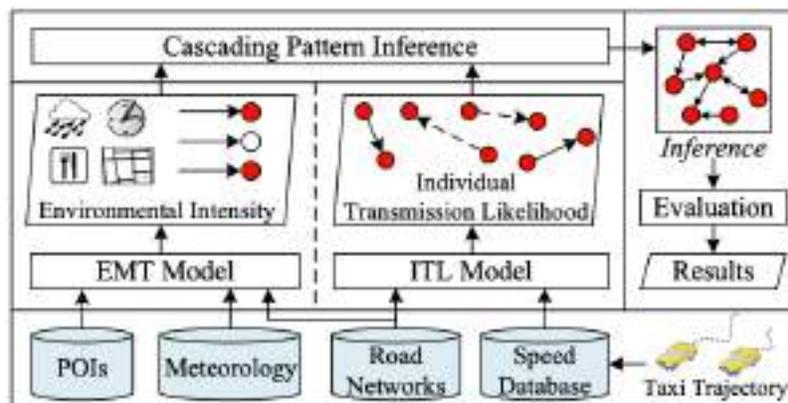


Fig. 2 Metodología para la identificación de patrones en cascada

El artículo considera un caso de estudio para el distrito de Wangjing donde denotan los segmentos de la carretera contenidos en el patrón. Se explica el por qué y cómo se propaga la congestión del tráfico en esa área con base en los siguientes factores: las horas punta de la mañana y las horas punta de la tarde.

Los resultados del artículo nos indican que el modelo CasInf tiene mejores resultados en la predicción del comportamiento del tráfico en comparación con los modelos NetInf y MultiTree, en un futuro los autores planean implementar su modelo en movilidad humana para entender los patrones de movilidad humana en las ciudades.

3.3.2 Análisis de patrones de tráfico urbano mediante Clustering

El objetivo de esta tesis doctoral *Analysis of urban traffic patterns using clustering* (Weijermars, 2007) es obtener más información sobre el tráfico urbano analizando las variaciones en los volúmenes de tráfico.

La Dra. Weijermars desarrolló un método para el análisis de las variaciones temporales y espaciales en los volúmenes de tráfico urbano utilizando datos de los centros de información de tráfico urbano. El preprocesamiento de datos es la tarea más importante en el trabajo de investigación de la Dra. Weijermars, ya que se tuvieron que validar los datos del centro de información de tráfico urbano.

La metodología utilizada en la investigación es el análisis por clúster, se aplica para la determinación de patrones de tráfico urbano que estos sirven para la previsión de tráfico.

La investigación fue un caso de estudio para la ciudad de Almelo, una ciudad de tamaño mediano en los Países Bajos.

Como resultado de la investigación se identificaron patrones de tráfico urbano típicos como se muestra en la fig. 3, así como los factores temporales, circunstanciales y espaciales en los que se basan estos patrones. El trabajo de investigación también se centra en la forma de explicar estos patrones por las variaciones en la demanda y la oferta de viajes.

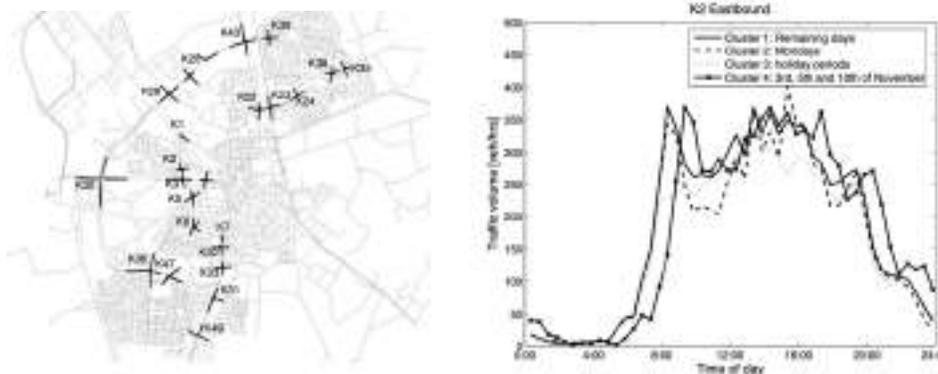


Fig. 3 Identificación de Patrones por Clustering

3.3.3 Análisis de técnicas inteligentes para la predicción del flujo de tráfico urbano

El objetivo del artículo Análisis de técnicas inteligentes para la predicción del flujo de tráfico urbano (Boris Medina Salgado, 2021) es brindarnos un estado del arte sobre las técnicas empleadas

para el tratamiento de datos de tráfico urbano y muestra una taxonomía general para el entendimiento computacional del flujo vehicular.

Las ciudades están evolucionando, implementando sistemas de transporte inteligente (ITS) y se perfilan como alternativa para una eficiencia en los servicios de transporte, beneficiando a los conductores y pasajeros del sistema.

El Internet de las Cosas (IoT) se emplea en sistemas de tecnologías de información y comunicación. Esta tecnología conecta sensores inteligentes y dispositivos a los que se pueden acceder a través de Internet.

Con el crecimiento en desarrollo de poder computacional y el volumen de datos de tráfico, la aplicación de diversos métodos de aprendizaje automático y muy estrechamente relacionado con la inteligencia artificial es posible realizar tareas de clasificación, agrupamiento, predicciones y reconocimiento de patrones.

En el artículo se tiene una revisión de técnicas de aprendizaje automático para el uso de algoritmos para la predicción de flujo de tráfico en zonas urbanas. En la fig. 4 muestra una taxonomía de las técnicas en la predicción de flujo de tráfico.

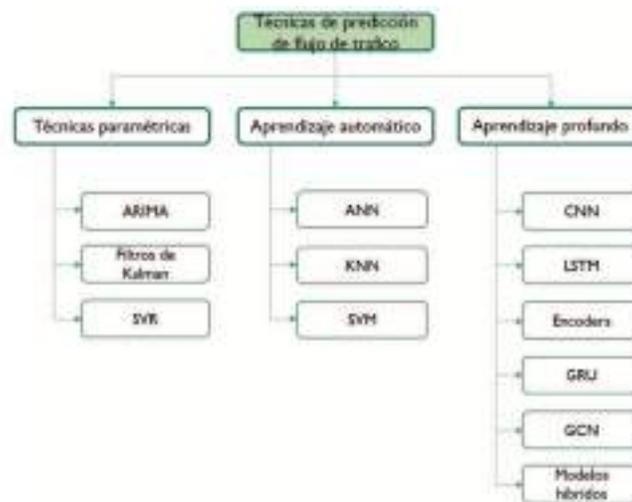


Fig. 4 Técnicas de predicción de flujo de tráfico

Los autores nos muestran las técnicas paramétricas, aprendizaje automático y aprendizaje profundo que se usan en diversos métodos para resolver las problemáticas de predicción de flujo de tráfico.

Las conclusiones del análisis de las diferentes técnicas implementadas para la predicción de flujo de tráfico en un conjunto de datos Caltrans PeMS que son proporcionados por el Departamento de

Transportes de California, EEUU. Muestra resultados sobresalientes de algoritmos utilizados para tener una mejor aproximación en la predicción de flujo de tráfico.

3.3.4 Congestiónmetro

El Congestiónmetro es un proyecto que ejecutó el Gobierno de la Ciudad de Buenos Aires, cuyo el objetivo fue generar una herramienta para la planificación y gestión del tránsito que emplee datos masivos de la aplicación Waze. Para su desarrollo se utilizó tecnología de código abierto y se diseñó de forma que pudiera ser adaptado y replicado en otras ciudades de la región sin que se requieran hardware especializado. Permite complementar los tradicionales Sistemas Inteligentes de Transporte (ITS) y ampliar la información disponible sobre el tránsito a toda la ciudad.

Los desarrolladores del proyecto se enfocan en la gran problemática que vive Buenos Aires en temas de congestión vial, como en muchas otras grandes ciudades de América Latina, la congestión de las vías urbanas genera significativos incrementos en costos y tiempos de transporte de personas y bienes, y aumenta las emisiones de gases y ruidos que afectan la calidad de vida y contribuyen al calentamiento global.

Para hacer frente a este desafío, los autores del proyecto comentan que es necesario contar con datos fiables y actualizados que permitan mejorar la planificación y evaluación de las intervenciones urbanas, y la gestión del tránsito en tiempo real. En la fig. 5 podemos observar el sistema en ejecución, de un lado muestra la congestión de la ciudad gráficamente en un mapa y de lado derecho muestra un gráfico en barras de la ocupación de la ciudad.

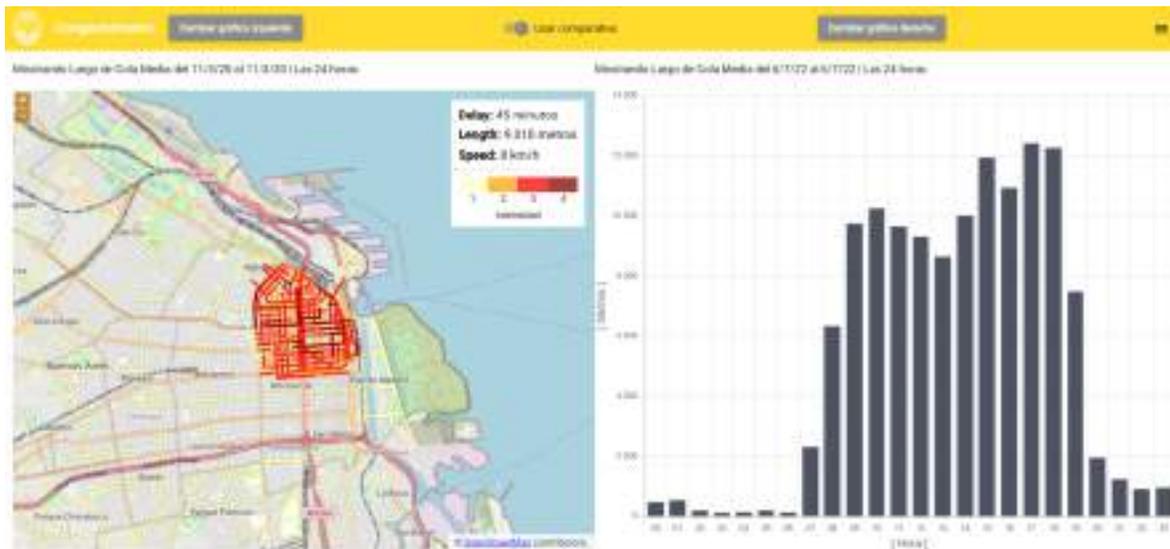


Fig. 5 Sistema de software Congestiónmetro

3.3.5 Metodología para obtener datos de tráfico e incidentes de tráfico a través de aplicaciones de mapas

En el artículo metodología para obtener datos de tráfico e incidentes de tráfico propone una metodología para la obtención de datos de tráfico e incidencias en una área geográfica que consta de 5 etapas.

El artículo tiene como objetivo adquirir los datos de tráfico e incidentes para implementar técnicas estadísticas o la identificación de patrones de movilidad.

La metodología propuesta en este artículo consta de 5 etapas las cuales son: elección de aplicación para la obtención de datos de tráfico e incidencias, obtención de datos mediante API KEY o KEY, identificación de área de donde se obtendrán datos, adquisición de datos a través de request a la aplicación y almacenamiento de los datos en una base de datos.



Fig. 6 Metodología para la adquisición de datos de tráfico

La metodología propuesta cuenta con 5 fases como se observa en la fig. 6, para obtener datos de tráfico e incidencias, el artículo describe que esta metodología es muy flexible para adquirir los datos de aplicaciones de mapas en diferentes dimensiones de área, las características que proporcionan las aplicaciones permiten identificar el tráfico e incidencias para su análisis para diversos objetivos de investigación. El artículo destaca que varias aplicaciones comparten las mismas características que describen el tráfico o incidencias esto ayuda a los investigadores a trabajar con varias aplicaciones para comparar resultados en las investigaciones.

El artículo concluye que los datos de tráfico e incidencias es la materia prima para el desarrollo de nuevas investigaciones, la metodología propuesta ayuda a crear bases de datos historias para futuros análisis de movilidad, desarrollo de índices de movilidad vehicular y análisis de impacto de tráfico e incidencias.

Capítulo 4 Metodología de solución

En este capítulo se presenta la metodología propuesta para el desarrollo de esta tesis. Este capítulo se divide en dos apartados importantes: metodología de solución y arquitectura del prototipo.

4.1 Metodología de solución

En esta tesis se desarrolló un prototipo que permite la identificación de patrones de congestión vehicular en la alcaldía Tlalpan de la CDMX, los datos de tráfico se utilizan como entrada para la identificación de patrones. En este contexto, se propone el desarrollo de un sistema donde el usuario podrá visualizar el comportamiento de la congestión en calles de la alcaldía Tlalpan utilizando tecnologías *open source*, de esta manera permite que sea utilizable en diversos proyectos de movilidad.

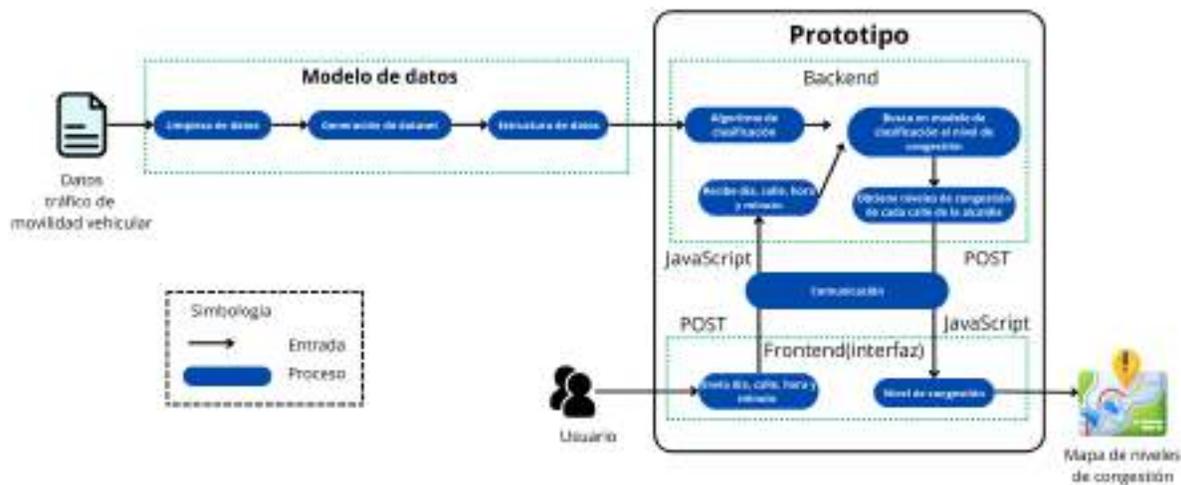


Fig. 7 Metodología de solución

La fig. 7 muestra la metodología propuesta para el desarrollo del prototipo. La metodología de solución consta de 4 fases principales, la fase de modelado de datos, la fase de identificación de patrones, la fase del desarrollo de la interfaz de usuario y la fase de la comunicación entre el algoritmo y la interfaz.

A continuación, se describe cada una de las fases que comprende la metodología de solución propuesta para generar los patrones de congestión vehicular y el uso de las tecnologías open source.

4.1.1 Fase de modelado de datos

Un modelo de datos es la forma de organizar y almacenar nuestros datos de tráfico. La importancia de tener un modelo de datos es eficiente para la identificación de patrones de congestión vehicular.

Un modelo de datos diseñado y optimizado permite identificar los niveles de congestión de una manera rápida y realizar análisis por fechas específicas.

El modelo de datos es importante para esta investigación, son la principal fuente para el modelo de clasificación de niveles de congestión. La creación de un modelo de datos permitió la identificación de las características relevantes, identificar los segmentos de cada calle y sus niveles de tráfico. Tener un modelo de datos ayuda a crear el histórico de datos de tráfico para otras investigaciones.

El modelo de datos desarrollado en esta investigación considera diversas variables clave relacionadas con el tráfico vehicular, fundamentales para la clasificación y análisis de los niveles de congestión. Entre los datos incorporados se incluyen el día de la semana, el intervalo de tiempo (inicio y fin), el segmento de la calle analizado y el factor de atasco, que mide la densidad vehicular en función de rangos previamente definidos. Estas variables permiten identificar patrones específicos de flujo vehicular, segmentar las condiciones de tráfico en distintas categorías (como flujo libre, moderado o congestión severa) y analizar la evolución de estos patrones a lo largo del tiempo. Además, el modelo contempla la generación de un histórico de datos, lo que facilita el análisis comparativo entre diferentes periodos y la reutilización de la información en futuras investigaciones relacionadas con la movilidad urbana. Este enfoque asegura un procesamiento eficiente y preciso para la detección de patrones y la planificación estratégica del tráfico.

4.1.2 Fase de identificación de patrones

En esta fase propuesta en la metodología de solución se implementó el algoritmo *Random Forest* para la clasificación del nivel de congestión. El algoritmo utiliza los datos de tráfico para clasificar el nivel de congestión, *Here Maps* utiliza una escala de 0 a 10 para determinar el nivel de congestión, tener estas escalas permite etiquetar los datos y usar el algoritmo supervisado. Los datos tienen la información del día de la semana, el minuto del día y el segmento de calle que pertenece, como etiqueta meta tiene el nivel de congestión registrada en ese momento por *Here Maps*. Con esta información el algoritmo infiere el posible comportamiento que tendrá.

El modelo recibe los datos de del día, la hora y minuto que está en intervalos de 5 minutos y la calle que se desea saber el comportamiento, con estos datos se realiza la petición al modelo y este devuelve el nivel de congestión. Esta información se pide al usuario a través de una interfaz (frontend) que se comunica con nuestro backend el cual es independiente, su principal función es cargar el modelo entrenado, recibir los datos de usuario y enviar los niveles de congestión para ser graficados en la interface. Este desarrollo se realizó con tecnologías open source favorece la libertad de uso y distribución del software.

4.1.3 Fase de desarrollo de la interfaz de usuario

En esta fase de la metodología de solución se desarrolló la interfaz de usuario. La interfaz consiste en la visualización de un mapa donde se muestran los niveles de congestión, el usuario selecciona la calle que desea analizar, el día de la semana, la hora, el minuto y el nivel de congestión en el cual puede elegir ver todos los niveles o alguno en específico. La interfaz se desarrolló separado del modelo con el objetivo que pueda usarse o modificarse en proyectos futuros.

El desarrollo es mediante HTML y JavaScript, tecnologías de código abierto que reducen los costos en adquisición de licencias y cualquiera que acceda puede realizar modificaciones, mejoras y contribuir en una mejora continua para la visualización de patrones de congestión.

4.1.4 Fase de comunicación entre el algoritmo y la interfaz

En esta fase de la metodología de solución, se establece la comunicación entre el algoritmo de identificación de patrones congestión y la interfaz de usuario. La comunicación se lleva a cabo a través del protocolo HTTP y mediante peticiones cliente-servidor. En estas peticiones, el usuario solicita información sobre los niveles de congestión de calles de la alcaldía Tlalpan enviando una petición al algoritmo.

Esta petición contiene los datos de la calle de la alcaldía Tlalpan y el horario por analizar, para el cual se desean obtener los patrones de congestión. El servidor, a su vez, recibe esta solicitud y la procesa, analizando el modelo para identificar los patrones de congestión correspondientes. Finalmente, el servidor devuelve al cliente los niveles de congestión marcados en las calles de la alcaldía,

4.2 Arquitectura del prototipo

La arquitectura del prototipo se basó en aplicaciones de código abierto para la visualización de los niveles de congestión acorde a los patrones identificados en el modelo de clasificación. El desarrollo en código abierto permite que esta investigación pueda ser utilizada para futuras mejoras o ser base para modificar el código y realizar las modificaciones adecuadas para implementarse en la identificación de patrones de congestión de otra ciudad. El prototipo tiene una arquitectura que separa el *backend* y *fronted* permitiendo que sea una aplicación exportable y adaptable a nuevos proyectos.

La arquitectura desarrollada consta de la parte visual (*frontend*) y la parte del modelo de clasificación (*backend*). El backend es donde se carga el modelo de clasificación, fue desarrollado en el lenguaje *Python* y el *Framework Flask*. El uso de estas dos tecnologías de desarrollo brinda una sólida y eficiente base para el desarrollo de la investigación. Python, como lenguaje de programación, es reconocido por su legibilidad, facilidad de uso y versatilidad, lo que permitió un desarrollo útil y eficiente para la identificación de patrones de congestión.

El *frontend* facilita la visualización de los niveles de congestión. Esta sección se ha programado utilizando *HTML* y *JavaScript*, lo que permite crear una interfaz interactiva e intuitiva. *HTML* se encarga de la estructura básica del contenido, organizando y presentando la información de manera clara y ordenada. Por otro lado, *JavaScript* agrega dinamismo y funcionalidad a la página, permitiendo actualizar y manejar los patrones de congestión, ofrecer niveles de congestión interactivos, y mejorar significativamente la experiencia del usuario. Gracias a la combinación de estos lenguajes, el sistema no solo resulta visualmente atractivo sino también altamente funcional y responsivo.

La interacción entre los dos componentes se lleva a cabo mediante el protocolo *HTTP* y utiliza el formato *JavaScript* como se observa en la fig. 8, lo cual asegura una interoperabilidad fluida y eficaz entre el frontend y el backend. Con el uso del protocolo *HTTP*, se establece una comunicación basada en solicitudes y respuestas, permitiendo que el frontend envíe peticiones al backend para obtener la información requerida, como los niveles de congestión, y reciba las respuestas respectivas.

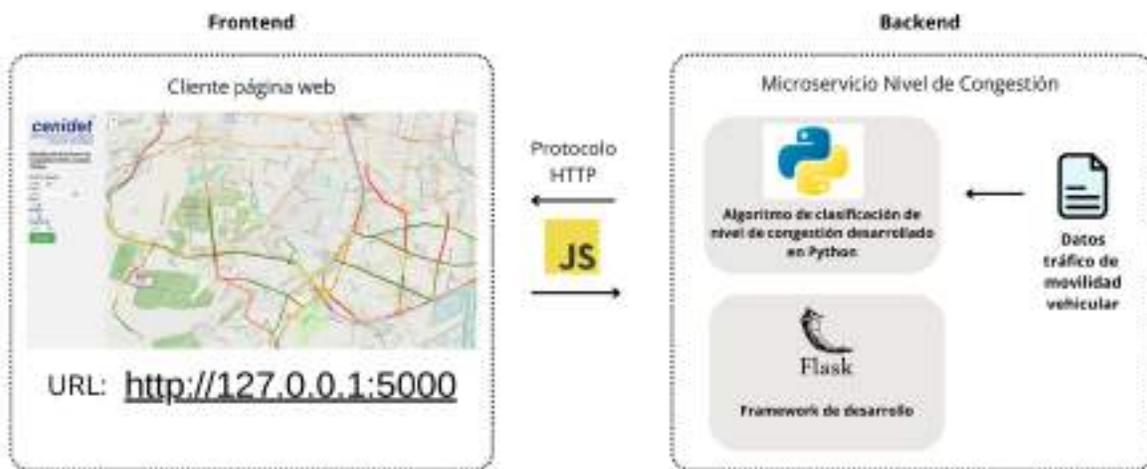


Fig. 8 Arquitectura de prototipo

El formato JSON (JavaScript Object Notation) se utiliza para estructurar y transmitir datos entre el frontend y el backend de manera ligera y fácil de interpretar tanto para los sistemas como para los desarrolladores. JSON se ha convertido en un estándar para el intercambio de datos en aplicaciones web debido a su facilidad de lectura y escritura, así como su compatibilidad con múltiples lenguajes de programación. Gracias al uso de HTTP y JSON, la comunicación entre el frontend y el backend es independiente del lenguaje o tecnología empleada en cada componente, lo que brinda mayor flexibilidad y facilita la integración con otros sistemas o servicios web.

A continuación, se detalla cada una de las partes que comprende la arquitectura utilizada para este proyecto de investigación.

4.2.1 Backend del prototipo

En este apartado se describe el desarrollo del *backend* para el prototipo de identificación de patrones de movilidad en congestión vehicular. El objetivo principal de este proyecto es proporcionar a los usuarios una herramienta que les permita visualizar el comportamiento de la congestión en diferentes puntos de la alcaldía Tlalpan, horarios de afectación y por día de la semana. El *backend* se encarga de procesar y analizar estos datos para generar la visualización de los niveles de congestión acorde a los patrones identificados como se muestra en la fig. 9. El *backend* del prototipo se basa en una arquitectura de servicio web que permite la comunicación con el *frontend* y el acceso al modelo de clasificación de niveles de congestión.

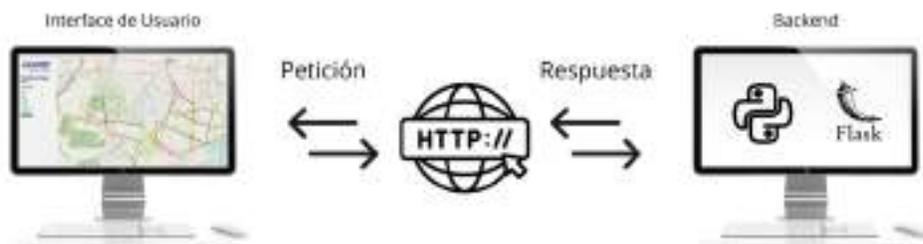


Fig. 9 Comunicación backend de prototipo

4.2.2 Frontend del prototipo

Para el desarrollo del *frontend* del prototipo, como se muestra en la fig. 10, se tiene como objetivo principal proporcionar a los usuarios una interfaz intuitiva y amigable que les permita interactuar con el sistema de patrones de congestión vehicular. El *frontend* fue desarrollado utilizando *flask*, una biblioteca de JavaScript para construir interfaces de usuario, y se comunica con el *backend* a través del protocolo HTTP para obtener los datos necesarios y mostrar las recomendaciones de rutas al usuario.

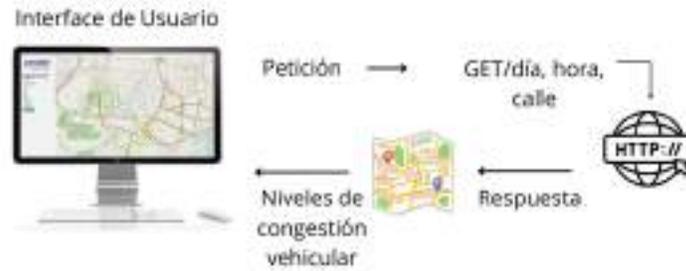


Fig. 10 frontend de prototipo

4.2.3 Integración del prototipo de visualización

El sistema de visualización de patrones de congestión vehicular se despliega de manera local para realizar las peticiones y mostrar los patrones de congestión, se realiza de esta manera para que puedan realizarse modificaciones al código o cambiar el modelo de clasificación utilizado como futuras mejoras. En la fig. 11 se observa la arquitectura que se utilizó para la integración del backend y frontend.

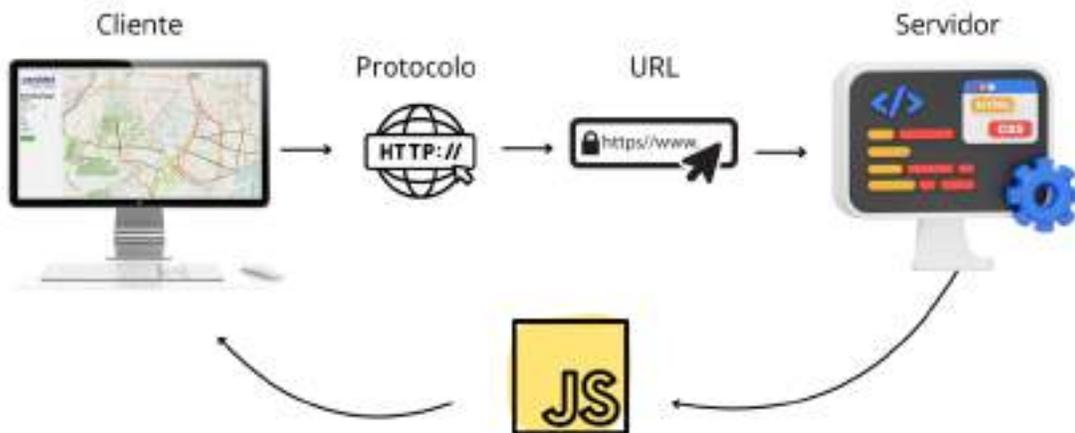


Fig. 11 Integración de backend y frontend

Capítulo 5 Desarrollo identificación de patrones

En este capítulo se detallan las actividades realizadas durante el desarrollo de la identificación de patrones de congestión vehicular.

5.1 Identificación de patrones de congestión vehicular

El prototipo desarrollado en esta tesis, denominado identificador de patrones de congestión vehicular fue desarrollado en cuatro fases. La fase 1 es el **modelado de datos** que fue de importancia para identificar las variables de mayor importancia, organizar, almacenar y modelar los datos. La fase 2 es el **desarrollo e implementación del algoritmo** donde se implementó el algoritmo Random Forest y haciendo uso del modelo de datos se clasificará acorde a nivel de congestión. La fase 3 es el **desarrollo de la interfaz de usuario** que consistió en generar una interfaz con un mapa donde se visualizan los niveles de congestión predichos por el modelo de clasificación. Por último, para la fase 4 se realizó la **comunicación entre componentes** del algoritmo y la interfaz para completar el prototipo de esta tesis.

5.2 Implementación del modelado de datos

Para conformar el modelo de datos se realizaron las siguientes tareas: a) obtención de los datos de tráfico de la alcaldía Tlalpan, b) limpieza de los datos seleccionando información de utilidad, c) conformación de un nuevo *dataset* con los datos de congestión.

a) Obtención de los datos

Los datos de congestión de tráfico se obtuvieron a través de la tesis “Desarrollo de un índice dinámico de movilidad urbana basado en minería de datos para determinar el grado de competitividad entre localidades”, para poder obtener los datos se limitó la alcaldía Tlalpan mediante Bounding Box como se muestra en la fig. 12 y se recopilieron datos que comprenden del 20 de septiembre del 2023 al 18 de octubre del 2023 y del 27 de octubre del 2023 al 26 de noviembre del 2023.

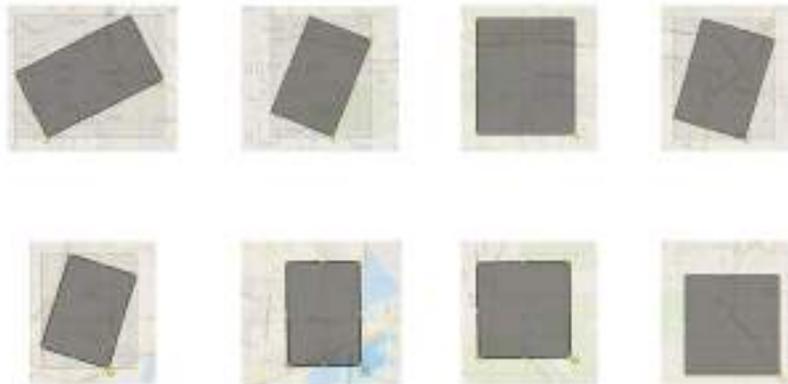


Fig. 12 Bounding box alcaldía Tlalpan

Los datos de tráfico obtenidos son la velocidad, coordenadas, fecha, hora, minuto, mes, año, color de tráfico, longitud de segmento, día, nombre del día, nombre del mes, versión, unidad, id, nombre de calle, velocidad sin límite, bounding box.

b) Limpieza de los datos

Para la limpieza de los datos se analizó el conjunto de archivos excel (.xlsx) que previamente se obtuvieron de la plataforma Here Maps y se visualizaron en el mapa como se observa en la fig. 13.



Fig. 13 Mapa alcaldía Tlalpan monitoreada por bounding boxes

Los datos obtenidos contienen la información de tráfico, como el nombre de la calle, coordenadas, horario de registro, nombre del día, mes, año entre otros campos descritos anteriormente como se observa en la fig. 14. Los datos requieren de limpieza para obtener los mejores resultados en nuestro modelo de Random Forest.

Como primer paso fue la unificación de los dataset, la petición se realizaba cada 5 minutos a los servidores de Here Maps de los cuales se generaban 276 archivos con los datos de tráfico de cada día, en cada archivo viene la información de las lecturas tomadas para cada calle de la alcaldía Tlalpan. Los datos obtenidos son una representación de las lecturas de tráfico tomadas por Here Maps mediante cámaras de tráfico, satélites, usuarios y datos de terceros.

El dataset obtenido de la unificación es de 5,238,730 registros de tráfico y 17 variables.

ID	FechaHora	Speed	latFactor	Type	geometry_WKT	traffic_how	length	traffic_speedkmh	traffic_howdir	max	coll	streetname	traffic_status	tempo_completo	Fecha	minmax_miles	
0	07	2003-10-0	15820034	01	segmento	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	0	507	1	0	270	Calle de Francisco Alamo	sur	7	05:00:00	2003-10-01	0
1	183	2003-10-0	20000000	00	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	0	578	1	0	111	Carretera Tlalpan Sur	norte	1	05:00:00	2003-10-01	0
2	374	2003-10-0	10000000	00	segmento	LINESTRING (98.21717 19.32036, 98.21717 19.32036)	0	581	1	0	273	Carretera Tlalpan Sur	norte	1	05:00:00	2003-10-01	0
3	48	2003-10-0	20000034	00	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	0	604	1	0	111	Carretera Tlalpan Sur	norte	1	05:00:00	2003-10-01	0
4	58	2003-10-0	21200033	00	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	0	672	1	0	48	Carretera Tlalpan Sur	norte	1	05:00:00	2003-10-01	0
1028720	184	2003-11-0	11000034	00	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	23	681	0	0	83	Calle de Francisco Alamo	sur	4	21:00:00	2003-11-09	1489
1028720	186	2003-11-0	8480046	00	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	21	781	0	0	84	Calle de Francisco Alamo	sur	6	21:00:00	2003-11-08	1489
1028720	187	2003-11-0	11000034	01	FMCI	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	23	848	0	0	80	Calle de Francisco Alamo	sur	4	21:00:00	2003-11-09	1489
1028720	271	2003-11-0	03200033	00	segmento	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	22	884	0	0	100	Carretera Tlalpan Sur	norte	6	21:00:00	2003-11-09	1423
1028720	449	2003-11-0	11000034	00	segmento	LINESTRING (98.20878 19.32036, 98.20878 19.32036)	23	907	0	0	80	Carretera Tlalpan Sur	norte	6	21:00:00	2003-11-09	1489

Fig. 14 Dataset de datos obtenidos de Here maps

Se realizó un análisis exploratorio de datos se encontraron registros repetidos, esto se debe a que un segmento puede estar dentro de dos bounding box cercanos y es por lo que se registra dos veces, para eliminar esto se realizó un filtro por fecha y hora, si había un segmento igual registrado se eliminaban y se quedaba el primer registro.

Los datos se reducen a 2,501,058, esta reducción también se da por la selección de la muestra de datos que cada segmento debe tener al menos 10 mil registros para identificar un patrón de congestión vehicular. En la fig. 15 podemos observar la reducción de calles, sin embargo, las calles con mayor monitoreo son calles que conectan la alcaldía Tlalpan.



Fig. 15 Mapa de monitoreo para la identificación de patrones

c) Conformación del dataset

Una vez realizada la limpieza de los datos se conformó un nuevo *dataset* que contiene los datos necesarios para la identificación de patrones de congestión vehicular. El nuevo conjunto de datos, ver fig.16, contiene la información necesaria para identificar los segmentos como velocidad, id, fecha, minuto del día, bloque, factor de atasco, día de la semana y nombre de la calle.

El análisis de patrones de tráfico vehicular se basa en un conjunto de datos estructurados que incluyen múltiples variables relevantes. A continuación, se describen los atributos utilizados:

Velocidad: Representa la velocidad promedio de los vehículos en el segmento de la calle registrado, expresada en kilómetros por hora (km/h). Este atributo es esencial para evaluar la fluidez del tráfico en distintos intervalos de tiempo.

ID: Identificador único asociado al segmento de la calle o al punto donde se recopila la información. Este valor permite diferenciar y localizar los registros dentro del sistema de monitoreo. Se expresa como un número entero.

Fecha: Fecha en la que se tomó cada registro, expresada en formato estándar de fecha (YYYY-MM-DD). Este atributo facilita la clasificación temporal de los datos y su agrupación por días específicos.

Minuto del día: Indica el minuto exacto del día en que se generó el registro, con un valor entre 0 (correspondiente a las 00:00 horas) y 1439 (equivalente a las 23:59 horas). Este formato numérico entero permite un análisis más granular del comportamiento del tráfico a lo largo del día.

Bloque: Segmento o tramo específico de la calle donde se recopila la información. Este atributo, es numérico, permite relacionar la información con la ubicación exacta dentro del área de estudio.

Factor de atasco: Métrica relativa que refleja el nivel de congestión vehicular en un segmento dado. Se expresa como un valor decimal entre 0 y 10, donde 0 representa flujo libre y 10 indica máxima congestión. Este atributo es clave para identificar los niveles de congestión en diferentes condiciones.

Día de la semana: Indica el día específico en que se registra el dato, representado como texto (por ejemplo, “lunes”, “martes”) o como un valor numérico (0 para domingo, 6 para sábado). Este atributo permite identificar patrones recurrentes asociados a días laborales o fines de semana.

Nombre de la calle: Denominación oficial del segmento vial donde se registra el dato, expresada como una cadena de texto. Este atributo facilita la identificación y el análisis espacial de las condiciones del tráfico.

	speed	ID	Fecha	minutos_del_dia	trafico_tipobloque	jamFactor	trafico_numdia	calle
0	4.654747	46	2023-09-20	0	1	0.0	3	Autopista Urbana Sur
1	2.886751	62	2023-09-20	0	1	0.8	3	Calzada Mexico Xochimilco
2	4.440971	54	2023-09-20	0	1	0.0	3	Calzada Viaducto Tlalpan
3	2.357023	135	2023-09-20	0	1	0.2	3	Calzada Acoepa
4	4.594663	150	2023-09-20	0	1	0.0	3	Av Viaducto Tlalpan
...
2501053	2.838231	101	2023-11-26	1435	6	0.2	7	Calzada de Tlalpan
2501054	2.357023	91	2023-11-26	1435	6	0.0	7	Calzada Mexico Xochimilco
2501055	4.714045	176	2023-11-26	1435	6	0.0	7	Autopista Urbana Sur
2501056	2.357023	112	2023-11-26	1435	6	6.1	7	Calzada de Tlalpan
2501057	2.738613	34	2023-11-26	1435	6	0.2	7	Calzada de Tlalpan

2501058 rows x 8 columns

Fig. 16 Dataset para el modelo random forest

Con esta información se conformó un solo conjunto de datos como se observa en la fig. 16, donde contiene la información necesaria para trabajar con el modelo de aprendizaje automático para la identificación de patrones.

5.3 Implementación del algoritmo Random Forest

En esta sección se describe el módulo de Python que se implementó para la identificación de patrones. El algoritmo utilizado para los patrones es Random Forest, un algoritmo que se utiliza para el aprendizaje supervisado y nos ayudará a identificar el comportamiento de los segmentos en todas las horas del día.

Para llevar a cabo la identificación de patrones de congestión vehicular, se utilizó un modelo de clasificación supervisada basado en el algoritmo Random Forest. Este modelo fue elegido por su capacidad para manejar grandes volúmenes de datos, su robustez ante el sobreajuste y su habilidad para interpretar la importancia de las características. Random Forest es un ensamblado de múltiples árboles de decisión, donde cada árbol contribuye a la predicción final mediante un sistema de votación. Este enfoque es particularmente útil en el contexto de la congestión vehicular, ya que permite identificar patrones complejos y no lineales en los datos de tráfico.

Antes de aplicar el modelo, fue necesario abordar el desbalance de clases presente en el conjunto de datos. Las clases de tráfico (verde, amarillo y rojo) se encontraban distribuidas de la siguiente manera: **green** con 2,175,037 observaciones, **red** con 215,672 observaciones y **yellow** con 110,349 observaciones. Esta distribución desigual podría llevar al modelo a tener un sesgo hacia la clase mayoritaria. Para mitigar este problema, se utilizó el método SMOTE (Synthetic Minority Over-sampling Technique), el cual genera nuevas observaciones sintéticas para las clases minoritarias. Este procedimiento equilibra el conjunto de datos, permitiendo que el modelo aprenda patrones de todas las clases de manera más equitativa.

Una vez balanceado el conjunto de datos, se dividieron las muestras en variables predictoras (X) y la variable objetivo (y). Las características elegidas fueron `trafico_numdia`, `minutos_del_dia` y `length`, que representan variables relacionadas con el día de la semana, el momento específico del día y la longitud del tramo de tráfico, respectivamente. Estas características fueron seleccionadas debido a su correlación potencial con los niveles de congestión y su capacidad para capturar las variaciones temporales y espaciales del tráfico.

Para evaluar el rendimiento del modelo, se dividió el conjunto de datos en entrenamiento y prueba, asignando el 70% de las observaciones al entrenamiento y el 30% a la prueba. Esta división asegura que el modelo pueda generalizar correctamente a datos no vistos.

El modelo Random Forest fue configurado con los siguientes hiperparámetros, seleccionados para optimizar su rendimiento y prevenir el sobreajuste:

- **criterion:** “entropy” - se utiliza la ganancia de información para la división en los nodos.
- **max_depth:** 20 - establece la profundidad máxima de los árboles para limitar la complejidad del modelo.
- **max_features:** 3 - selecciona tres características en cada división, lo cual introduce variedad en los árboles y evita el sobreajuste.
- **min_samples_leaf:** 4 - asegura que cada hoja tenga un mínimo de cuatro muestras, lo que ayuda a suavizar las predicciones.
- **min_samples_split:** 20 - establece un umbral mínimo para dividir nodos, proporcionando mayor estabilidad en los árboles.

- **n_estimators:** 100 - indica la cantidad de árboles en el bosque, asegurando que el modelo tenga una base robusta para la votación.

Tras inicializar el modelo con los parámetros mencionados, se procedió al entrenamiento del clasificador utilizando el conjunto de datos de entrenamiento. La fase de entrenamiento permite al modelo aprender los patrones subyacentes de las características que influyen en los niveles de congestión. Posteriormente, el modelo fue evaluado en el conjunto de prueba para medir su precisión y rendimiento general.

5.4 Identificación de patrones

En esta etapa se describe cada patrón identificado con el algoritmo Random Forest, dentro de nuestro conjunto de datos tenemos 3 tipos de congestión, libre, moderado y Severa.

5.4.1 Patrón de congestión libre

El patrón de Flujo Libre en Tráfico Vehicular demuestra cómo una adecuada infraestructura y una gestión planificada pueden mantener condiciones de tráfico óptimas, incluso ante posibles aumentos en el volumen vehicular. La implementación de este patrón en áreas con baja densidad de tráfico, como carreteras rurales o autopistas bien diseñadas, fomenta tiempos de viaje más cortos, menor estrés para los conductores y una reducción en emisiones y consumo de combustible.

Desde el punto de vista computacional, el modelo Random Forest facilita la predicción de condiciones de flujo libre en función de segmentos específicos y franjas horarias, permitiendo una planificación proactiva.

Este análisis subraya la importancia de políticas urbanas sostenibles y estrategias de movilidad compartida para mantener el flujo libre a largo plazo, destacando su papel crucial en la eficiencia y sostenibilidad del tráfico vehicular.

Nombre del patrón: Flujo Libre en Tráfico Vehicular.

Contexto: Carreteras o áreas urbanas con un bajo volumen de tráfico y una infraestructura vial adecuada para manejar la demanda de vehículos sin congestión significativa.

Problema: Aunque el flujo libre no presenta problemas de congestión, puede ser necesario mantener esta condición en áreas donde se espera un aumento en el volumen de tráfico en el futuro.

Fuerzas:

- Infraestructura vial diseñada para un flujo eficiente de vehículos.

- Volumen de tráfico actual bajo.
- Deseo de mantener una movilidad fluida y segura.

Estructura:

- Diseño adecuado de la infraestructura vial.
- Planificación urbana que favorezca el flujo vehicular.
- Monitoreo y gestión del crecimiento del volumen de tráfico.

Solución:

- Mantenimiento adecuado de la infraestructura vial.
- Planificación urbana que considere el crecimiento futuro del tráfico.
- Implementación de políticas para fomentar el uso compartido de vehículos y formas alternativas de movilidad.

Consecuencias:

- Tiempos de viaje cortos y consistentes.
- Menor estrés y riesgo de accidentes para los conductores.
- Eficiencia en el consumo de combustible y reducción de emisiones.

Ejemplos de uso del patrón:

- Aplicación en áreas rurales o en zonas urbanas con baja densidad de población.
- Implementación en carreteras diseñadas específicamente para mantener un flujo libre, como autopistas y vías expresas.

Patrón relacionado:

- Este patrón puede estar relacionado con la Planificación Urbana Eficiente y la Gestión Sostenible de la Movilidad.

El patrón de **flujo libre** en el tráfico vehicular se caracteriza por condiciones en las que los vehículos pueden desplazarse sin interrupciones significativas, reflejado en un **factor de atasco menor o igual a 3.99 (≤ 3.99)**. Refleja un patrón de fluidez en diversos horarios a lo largo de la semana, como se detalla en la tabla 1. Durante los días hábiles (lunes a viernes), las condiciones de flujo libre predominan en las primeras horas de la madrugada (12:00 a.m. a 6:35 a.m.), en periodos intermedios de la mañana y tarde (entre 9:40 a.m. y 2:45 p.m., dependiendo del día), y en las horas nocturnas (desde las 5:00 p.m. hasta casi la medianoche). Este comportamiento evidencia intervalos de menor demanda vehicular, lo que permite desplazamientos óptimos.

Los días sábado y domingo presentan una distribución diferente. El sábado se identifican periodos de flujo libre a lo largo de casi toda la jornada, con interrupciones mínimas y horarios predominantemente continuos, mientras que el domingo mantiene flujo libre de manera ininterrumpida durante todo el día (12:00 a.m. a 11:55 p.m.). Estos resultados sugieren que los fines de semana tienen menor densidad vehicular, favoreciendo velocidades constantes y mínimas interrupciones en el tránsito.

Tabla 1 Patrón de congestión libre

Día	Inicio	Fin	Segmento	Factor de Atasco
Lunes	12:00 a.m.	6:35 a.m.	90	≤ 3.99
Lunes	9:50 a.m.	2:45 p.m.	90	≤ 3.99
Lunes	3:40 p.m.	3:45 p.m.	90	≤ 3.99
Lunes	5:00 p.m.	11:55 p.m.	90	≤ 3.99
Martes	12:00 a.m.	6:35 a.m.	90	≤ 3.99
Martes	9:55 a.m.	2:45 p.m.	90	≤ 3.99
Martes	5:25 p.m.	6:30 p.m.	90	≤ 3.99
Martes	6:40 pm	11:55 pm	90	≤ 3.99
Miércoles	12:00 a.m.	6:35 a.m.	90	≤ 3.99
Miércoles	10:00 a.m.	2:45 p.m.	90	≤ 3.99
Miércoles	4:30 p.m.	4:40 p.m.	90	≤ 3.99
Miércoles	5:25 p.m.	6:30 p.m.	90	≤ 3.99
Miércoles	6:40 p.m.	6:50 p.m.	90	≤ 3.99
Miércoles	7:25 p.m.	11:55 p.m.	90	≤ 3.99
Jueves	12:00 a.m.	6:35 a.m.	90	≤ 3.99
Jueves	10:00 a.m.	2:45 p.m.	90	≤ 3.99
Jueves	4:35 p.m.	4:40 p.m.	90	≤ 3.99
Jueves	5:20 p.m.	5:50 p.m.	90	≤ 3.99
Jueves	6:25 p.m.	6:30 p.m.	90	≤ 3.99
Jueves	6:40 p.m.	6:50 p.m.	90	≤ 3.99
Jueves	7:25 p.m.	11:55 p.m.	90	≤ 3.99
Viernes	12:00 a.m.	6:35 a.m.	90	≤ 3.99

Viernes	6:45 a.m.	7:15 a.m.	90	≤ 3.99
Viernes	7:30 a.m.	7:40 a.m.	90	≤ 3.99
Viernes	8:30 a.m.	8:55 a.m.	90	≤ 3.99
Viernes	9:05 a.m.	12:15 p.m.	90	≤ 3.99
Viernes	12:30 p.m.	2:45 p.m.	90	≤ 3.99
Viernes	5:05 pm	6:55 pm	90	≤ 3.99
Viernes	7:15 pm	11:55 pm	90	≤ 3.99
Sábado	12:00 a.m.	2:15 p.m.	90	≤ 3.99
Sábado	2:25 p.m.	3:10 p.m.	90	≤ 3.99
Sábado	3:20 p.m.	9:20 p.m.	90	≤ 3.99
Sábado	9:30 p.m.	11:55 p.m.	90	≤ 3.99
Domingo	12:00 a.m.	11:55 p.m.	90	≤ 3.99

5.4.2 Patrón de congestión moderada

El patrón de Congestión Moderada en Tráfico Vehicular refleja cómo la infraestructura urbana puede enfrentar desafíos de capacidad en momentos críticos del día sin una congestión severa pero con efectos notorios en la movilidad y la eficiencia.

La implementación de estrategias de gestión adaptativa, como el uso de semáforos inteligentes y la promoción de movilidad compartida, puede mitigar los impactos negativos y mejorar la experiencia de viaje, la seguridad y la calidad del aire.

Desde una perspectiva computacional, el uso del clasificador Random Forest permite identificar segmentos específicos y horarios en los que se presenta una congestión moderada, facilitando así una intervención focalizada.

Este análisis subraya la importancia de políticas de gestión de tráfico que optimicen el uso de la infraestructura sin requerir grandes inversiones, promoviendo una movilidad urbana más sostenible y eficiente.

Nombre del patrón: Congestión Moderada en Tráfico Vehicular.

Contexto: Áreas urbanas o tramos de carreteras con un flujo constante de vehículos que experimentan períodos de congestión moderada durante ciertas horas del día o en situaciones específicas.

Problema: La congestión moderada puede causar retrasos menores, pero aún así impacta negativamente en los tiempos de viaje, la eficiencia del transporte y la calidad del aire.

Fuerzas:

- Variabilidad en la demanda de viajes a lo largo del día.
- Infraestructura vial que no se adapta completamente a las necesidades de flujo de tráfico.
- Deseo de minimizar los impactos negativos de la congestión sin inversiones significativas en infraestructura.

Estructura:

- Identificación de los puntos y momentos críticos de congestión.
- Análisis de la demanda y capacidad de la infraestructura vial.
- Implementación de medidas de gestión de tráfico adaptativas.
- Promoción de la movilidad inteligente y el uso eficiente de la infraestructura existente.

Solución:

- Implementación de sistemas de control de tráfico adaptativos.
- Promoción del uso compartido de vehículos y de formas alternativas de movilidad.
- Fomento del teletrabajo y la flexibilidad de horarios laborales.
- Mejora de la coordinación entre servicios de transporte público y privado.

Consecuencias:

- Reducción de los tiempos de viaje y la frustración de los conductores.
- Mejora de la eficiencia del transporte y la calidad del aire.
- Aumento de la seguridad vial.
- Optimización de la infraestructura vial existente.

Ejemplos de uso del patrón:

- Aplicación en áreas urbanas con congestión moderada durante horas pico, mediante el uso de semáforos inteligentes, sistemas de información al conductor, etc.

Patrón relacionado:

- Puede relacionarse con otros patrones de gestión de tráfico, como Control de Flujo de Tráfico y Gestión de Intersecciones Viales.

El patrón de **flujo moderado** en el tráfico vehicular, definido por un **factor de atasco en el rango de 4 a 7.99 ($\geq 4 \leq 7.99$)**, se caracteriza por una reducción en la velocidad de desplazamiento debido a un aumento en la densidad vehicular, sin llegar a niveles críticos de congestión. Como se observa en la tabla 2, este comportamiento predomina en horarios de alta demanda durante los días hábiles, particularmente en las horas pico matutinas (entre las 6:40 a.m. y las 9:55 a.m.) y vespertinas (de 2:50 p.m. a 7:20 p.m.), asociados con actividades laborales y escolares. Los viernes presentan un patrón similar, pero con intervalos más breves y distribuidos, lo que refleja una dinámica vehicular distinta al final de la semana laboral. En contraste, los fines de semana muestran una menor incidencia de flujo moderado, concentrándose en periodos cortos durante la tarde, como se evidencia el sábado entre las 2:20 p.m. y las 3:20 p.m. Este análisis destaca la influencia de los horarios laborales y escolares en la dinámica vehicular del segmento analizado, así como la reducción de la demanda durante los fines de semana. La identificación de estos patrones es clave para implementar estrategias que optimicen el flujo vehicular en los periodos críticos, mejorando la movilidad en el segmento estudiado.

Tabla 2 Patrón de congestión moderado

Día	Inicio	Fin	Segmento	Factor de Atasco
Lunes	6:40 a.m.	6:50 a.m.	90	$\geq 4 \leq 7.99$
Lunes	7:05 a.m.	7:50 a.m.	90	$\geq 4 \leq 7.99$
Lunes	8:15 a.m.	9:45 a.m.	90	$\geq 4 \leq 7.99$
Lunes	2:50 p.m.	3:35 p.m.	90	$\geq 4 \leq 7.99$
Lunes	3:50 p.m.	4:55 p.m.	90	$\geq 4 \leq 7.99$
Martes	6:40 a.m.	6:50 a.m.	90	$\geq 4 \leq 7.99$
Martes	7:05 a.m.	7:50 a.m.	90	$\geq 4 \leq 7.99$
Martes	8:15 a.m.	9:50 a.m.	90	$\geq 4 \leq 7.99$
Martes	2:50 p.m.	3:20 p.m.	90	$\geq 4 \leq 7.99$
Martes	4:20 p.m.	5:20 p.m.	90	$\geq 4 \leq 7.99$
Martes	6:35 p.m.	6:40 p.m.	90	$\geq 4 \leq 7.99$
Miércoles	6:40 a.m.	7:50 a.m.	90	$\geq 4 \leq 7.99$
Miércoles	8:05 a.m.	9:55 a.m.	90	$\geq 4 \leq 7.99$
Miércoles	2:50 p.m.	4:25 p.m.	90	$\geq 4 \leq 7.99$
Miércoles	4:45 p.m.	5:20 p.m.	90	$\geq 4 \leq 7.99$
Miércoles	6:35 p.m.	6:40 p.m.	90	$\geq 4 \leq 7.99$
Miércoles	6:55 p.m.	7:20 p.m.	90	$\geq 4 \leq 7.99$
Jueves	6:40 a.m.	7:50 a.m.	90	$\geq 4 \leq 7.99$
Jueves	8:15 a.m.	9:55 a.m.	90	$\geq 4 \leq 7.99$

Jueves	2:50 p.m.	4:30 p.m.	90	$\geq 4 \leq 7.99$
Jueves	4:45 p.m.	5:15 p.m.	90	$\geq 4 \leq 7.99$
Jueves	5:55 p.m.	6:20 p.m.	90	$\geq 4 \leq 7.99$
Jueves	6:35 p.m.	6:40 p.m.	90	$\geq 4 \leq 7.99$
Jueves	6:55 p.m.	7:20 p.m.	90	$\geq 4 \leq 7.99$
Viernes	6:40 a.m.	6:45 a.m.	90	$\geq 4 \leq 7.99$
Viernes	7:20 a.m.	7:25 a.m.	90	$\geq 4 \leq 7.99$
Viernes	7:45 a.m.	8:00 a.m.	90	$\geq 4 \leq 7.99$
Viernes	8:15 a.m.	8:25 a.m.	90	$\geq 4 \leq 7.99$
Viernes	9:00 a.m.	9:05 a.m.	90	$\geq 4 \leq 7.99$
Viernes	12:20 p.m.	12:25 p.m.	90	$\geq 4 \leq 7.99$
Viernes	2:50 p.m.	3:50 p.m.	90	$\geq 4 \leq 7.99$
Sábado	2:20 p.m.	2:25 p.m.	90	$\geq 4 \leq 7.99$
Sábado	3:15 p.m.	3:20 p.m.	90	$\geq 4 \leq 7.99$

5.4.3 Patrón de congestión severa

El patrón de Congestión Severa en Tráfico Vehicular evidencia la necesidad urgente de soluciones estructurales y de gestión en áreas con alta densidad de tráfico y limitada infraestructura. La congestión severa afecta negativamente no solo los tiempos de viaje y la experiencia del usuario, sino también el consumo de combustible, la calidad del aire y el desarrollo económico urbano. Abordar esta problemática requiere una combinación de medidas, incluyendo la mejora de la infraestructura vial, la expansión y eficiencia del transporte público, y la adopción de políticas de movilidad sostenible. La implementación de sistemas de gestión de tráfico inteligente y el fomento de alternativas de transporte compartido son esenciales para reducir la congestión y sus efectos en la calidad de vida y el ambiente urbano. En resumen, enfrentar la congestión severa es clave para promover ciudades más habitables, seguras y sostenibles a largo plazo.

Nombre del patrón: Congestión Severa en Tráfico Vehicular.

Contexto: Áreas urbanas o tramos de carreteras con un alto volumen de vehículos y una infraestructura vial insuficiente para manejarlos eficientemente.

Problema: La congestión severa causa retrasos significativos, aumenta el consumo de combustible, empeora la calidad del aire y reduce la calidad de vida de los residentes y la eficiencia económica.

Fuerzas:

- Incremento en el número de vehículos.
- Infraestructura vial insuficiente.
- Falta de sistemas de transporte público eficientes.
- Necesidad de movilidad urbana sostenible y segura.

Estructura:

- Identificación de los puntos críticos de congestión.
- Análisis de flujos de tráfico y patrones de viaje.
- Implementación de medidas de gestión de tráfico y transporte.
- Promoción de formas alternativas de movilidad.

Solución:

- Mejora de la infraestructura vial y de transporte público.
- Implementación de sistemas de gestión de tráfico inteligente.
- Fomento del uso compartido de vehículos y del transporte público.
- Diseño de políticas de planificación urbana que favorezcan la movilidad sostenible.

Consecuencias:

- Reducción de los tiempos de viaje y de la congestión vehicular.
- Mejora de la calidad del aire y del ambiente urbano.
- Estímulo de la economía local y regional.
- Mayor seguridad vial.

Ejemplos de uso del patrón:

- Aplicación en ciudades con problemas de congestión vehicular, como implementación de peajes urbanos, restricciones de circulación, carriles exclusivos para vehículos compartidos, etc.

El **patrón de congestión severa** en el tráfico vehicular, identificado por un **factor de atasco en el rango de 8 a 10 ($\geq 8 \leq 10$)**, evidencia condiciones críticas en las que el flujo vehicular se ve significativamente restringido, con velocidades extremadamente reducidas y tiempos de espera prolongados. Este comportamiento se concentra en horarios específicos durante los días hábiles, principalmente en las horas pico matutinas (entre las 6:55 a.m. y las 8:10 a.m.) y en algunos periodos vespertinos. Los martes y viernes presentan intervalos adicionales de congestión severa en la tarde, destacando especialmente los viernes con un periodo prolongado de 3:55 p.m. a 5:00 p.m. Este comportamiento refleja una alta demanda vehicular asociada con traslados hacia actividades laborales o escolares por la mañana, así como con el retorno a los hogares en ciertos días por la tarde. La distribución de estos periodos pone de manifiesto la necesidad de estrategias

específicas de mitigación para estos horarios críticos, con el objetivo de mejorar la movilidad y reducir los impactos negativos de la congestión en el segmento analizado.

Tabla 3 Patrón de congestión severa

Día	Inicio	Fin	Segmento	Factor de Atasco
Lunes	6:55 a.m.	7:00 a.m.	90	$\geq 8 \leq 10$
Lunes	7:55 a.m.	8:10 a.m.	90	$\geq 8 \leq 10$
Martes	6:55 a.m.	7:00 a.m.	90	$\geq 8 \leq 10$
Martes	7:55 a.m.	8:10 a.m.	90	$\geq 8 \leq 10$
Martes	3:25 p.m.	4:15 p.m.	90	$\geq 8 \leq 10$
Miércoles	7:55 a.m.	8:00 a.m.	90	$\geq 8 \leq 10$
Jueves	7:55 a.m.	8:10 a.m.	90	$\geq 8 \leq 10$
Viernes	8:05 a.m.	8:10 a.m.	90	$\geq 8 \leq 10$
Viernes	3:55 p.m.	5:00 p.m.	90	$\geq 8 \leq 10$

5.4 Desarrollo de interfaz

Para el desarrollo de la interfaz de usuario se priorizó la utilización de tecnologías de código abierto para la visualización de los patrones de congestión. Para el desarrollo de la interface se utilizó html y JavaScript.

La interfaz de usuario está conformada por los siguientes elementos para la visualización de los patrones de congestión, en la parte izquierda se tiene el menú para seleccionar el día de la semana, calle, hora, minuto del día y el nivel de congestión. Estos menús contextuales tienen preestablecidos los valores que el usuario puede elegir para mostrar los patrones de congestión vehicular, este diseño es intuitivo y fácil de usar. El botón de mostrar despliega en el mapa que se encuentra en la parte derecha los niveles de congestión de las calles monitoreadas. Este mapa es un desarrollo *open source* de OpenStreetMap que cuenta con la cartografía de la mayoría de las ciudades y es de código abierto. La interfaz completa y niveles de congestión se muestra en la fig. 17.



Fig. 17 Prototipo para la visualización de patrones de congestión

El prototipo desarrollado tiene como objetivo visualizar el comportamiento de la congestión las 24 horas del día, los 7 días de la semana. El funcionamiento es fácil e intuitivo para que pueda visualizarse de una manera sencilla. Este prototipo al ser código abierto tiene la facilidad de que pueda implementarse en otros proyectos, una característica de el prototipo es que puede visualizarse por una calle en específico lo cual ayuda a identificar el comportamiento de la misma para una mejor visualización del comportamiento.

El funcionamiento del prototipo consta de seleccionar el día, la calle, la hora, el minuto y el nivel de congestión que desea visualizar. En el mapa se visualizará el color del nivel de congestión previamente definidos, verde para un flujo libre, amarillo para flujo moderado y rojo para una

congestión severa. En la fig. 18 se muestra el mapa con los diferentes niveles de congestión que pueden visualizarse.

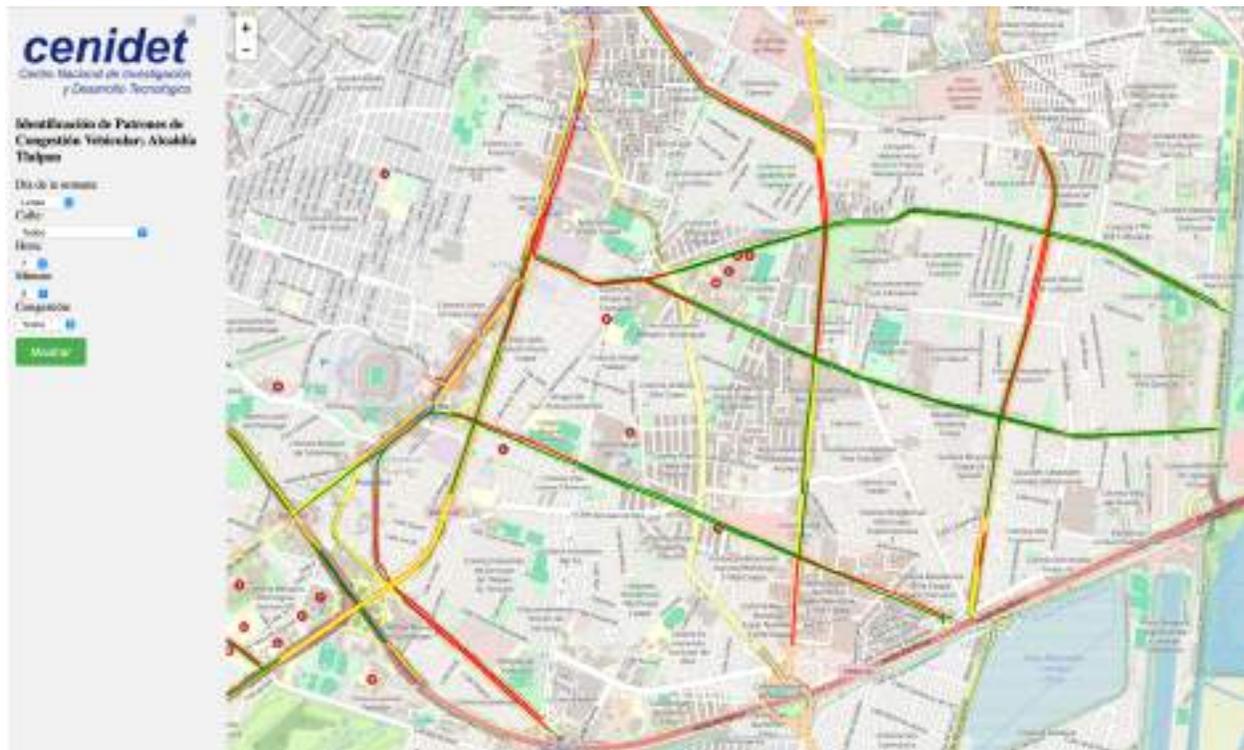


Fig. 18 Visualización de niveles de congestión

5.5 Comunicación entre componentes

La interacción entre componentes se lleva a cabo mediante el protocolo HTTP usando solicitudes de tipo POST. Una solicitud POST envía datos desde el frontend al backend para que sean procesados (por ejemplo, día, hora y nombre de la calle). El backend procesa estos datos, ejecuta un modelo de predicción de tráfico y devuelve los resultados.

La conexión al backend se realiza con la asignación de una URL a la que se apuntan las peticiones realizadas. Para el prototipo se configuró el puerto 5000, el cual es direccionado en nuestra URL que tiene asignada a la IP 127.0.0.1:5000, esta IP es asignada por default. Otro elemento que tiene nuestra URL es la asignación de web html, en este caso es un index.html. El botón *Mostrar* es el elemento que conecta nuestro *frontend* y el *backend* el cual ayuda a mostrar los patrones de congestión.

Este servidor utiliza el framework Flask para recibir solicitudes HTTP y hacer predicciones basadas en un modelo de Machine Learning. El servidor está configurado para aceptar solicitudes POST a través del endpoint /predecir, el cual recibe datos como día, hora, minuto y nombre de calle. Este modelo predictivo, almacenado en un archivo .pkl y cargado con joblib, generará el patrón de comportamiento de la congestión en una calle determinada y devuelve el color correspondiente al estado de tráfico en diferentes segmentos de la calle.

El frontend es una interfaz de usuario que recopila los datos necesarios desde un formulario, como el día, la calle, la hora y los minutos, junto con una opción de nivel de congestión que filtra los resultados según el estado de tráfico deseado. Al hacer clic en el botón Mostrar, el frontend envía estos datos al servidor mediante una solicitud HTTP POST y espera la respuesta del backend. Una vez recibida, la interfaz procesa y muestra el resultado de las predicciones de tráfico en un mapa interactivo, utilizando colores para diferenciar los estados de tráfico.

La comunicación entre el frontend y el backend se realiza mediante el protocolo HTTP, donde el frontend envía una solicitud POST y el backend responde con un JSON que contiene los datos predichos. Este intercambio ocurre de la siguiente manera:

1. Inicio de la Interacción (Frontend):

- Cuando el usuario hace clic en el botón mostrar, se capturan los valores del formulario (día, calle, hora, minuto y filtro de tráfico jf).
- Estos valores se formatean como un objeto JSON y se envían al servidor Flask mediante fetch, configurado con el método POST y encabezado Content-Type: application/json.

2. Recepción y Procesamiento de la Solicitud (Backend):

- El servidor Flask recibe la solicitud POST en el endpoint /predecir.
- Extrae los valores de día, hora, minuto y nombre de calle del JSON recibido. Luego, convierte estos valores para ser compatibles con el modelo predictivo.
- El backend usa un diccionario para mapear el nombre de la calle ingresado a uno o más segmentos identificados por ID. Por cada ID, se crea una entrada que se pasa al modelo de Machine Learning.
- El modelo genera una predicción para cada segmento de la calle, y el backend organiza estos resultados en una lista JSON que se devuelve al frontend.

3. Procesamiento de la Respuesta en el Frontend:

- El frontend recibe la lista de predicciones, que incluye los segmentos de la calle con sus respectivos colores de tráfico.
- La interfaz limpia el mapa interactivo de segmentos previos para evitar superposiciones.
- Dependiendo del valor de `jf` seleccionado por el usuario, la interfaz filtra y colorea solo aquellos segmentos de calle que cumplen el criterio deseado (green, yellow, red, o todos los colores).
- Finalmente, muestra en el mapa cada segmento en el color correspondiente al estado de tráfico predicho.

En la fig. 19 se muestra la función de backend donde la función `predecir` envía los datos al modelo en una aplicación Flask recibe datos en formato JSON para estimar el nivel de congestión en una calle específica a una hora dada. Los datos enviados por el cliente incluyen el día, la hora, el minuto y el nombre de la calle. El día se convierte en un número, y la hora y minuto se convierten en el “minuto del día” para facilitar su uso en el modelo de predicción.

```

from flask import Flask, request, jsonify
from flask_cors import CORS
import pickle
import json
import pandas as pd

app = Flask(__name__)
CORS(app, resources={r"/api/*": {"origins": "http://localhost:3000"}})

# Cargando el modelo
modelo = joblib.load('Aeros/lorpedotomas/lorpedotomas.pkl')

# Diccionario para convertir días a minutos
dias_a_minutos = {
    "Lunes": 1,
    "Martes": 2,
    "Miércoles": 3,
    "Jueves": 4,
    "Viernes": 5,
    "Sábado": 6,
    "Domingo": 7
}

# Diccionario para mapear nombres de calles a IDs
nombres_a_ids = {
    nombres: id for nombres, id in [
        ("Avenida Gabriela Mistral", 149, 400),
        ("Avenida O'Higgins Sur", 546, 348, 45, 508, 37, 38, 35, 23, 568, 153, 44, 47, 176, 340),
        ("Av. Amada de Riquelme", 1118, 119, 1340),
        ("Av. Gabriela", 185, 68, 141, 94, 148, 179, 171, 880),
        ("Av. Desamparados Sur", 105, 38, 26, 184, 68, 1200),
        ("Av. San Fernando", 179, 77, 181, 38, 89, 184, 880),
        ("Av. San Jerónimo", 132, 1550),
        ("Av. Vicente Fiquet", 1158, 44, 151, 300),
        ("Av. del Amor", 580, 128, 1370),
        ("Avenida Canal de Miraflores", 115, 108, 140, 128, 128, 118, 117, 188, 104, 111, 1400),
        ("Avenida Aconcagua", 1125, 26, 78, 14, 79, 12, 3, 79, 98, 93, 8, 1340),
        ("Avenida Acandú", 1120, 138, 83, 1360),
        ("Avenida Santa", 113, 330, 51, 38, 137, 136, 88, 87, 77, 74, 78, 170),
        ("Avenida Manuel Rodríguez", 182, 81, 81),
        ("Avenida Sebastián Fiquet", 154, 53, 81, 186, 207, 265, 96, 108, 183, 88, 98),
        ("Avenida de Tiligano", 1138, 131, 113, 22, 34, 53, 388, 112, 77, 78, 98, 1800),
        ("Avenida de los Baños", 189, 84, 78, 75, 15, 85),
        ("Avenida Presidente Balmaceda", 188, 83, 83, 1, 3, 87, 73, 83, 8, 72),
        ("Avenida Estación Alameda", 1148, 8, 7),
        ("Avenida Estación Naval Militar", 1158, 66),
        ("Parícuta Sur", 1341, 582, 26, 3520),
        ("Troncos", 189, 48, 85, 146, 45, 186, 37, 58, 77, 25, 189, 158, 48, 47, 178, 34, 98, 178, 128, 174, 99, 68, 143, 94, 188, 178, 177, 18, 56, 58, 88, 184,
            88, 128, 78, 77, 288, 78, 88, 184, 88, 32, 188, 188, 48, 188, 28, 88, 128, 127, 138, 108, 142, 128, 124, 118, 117, 188, 108, 111, 188, 188, 28,
            78, 14, 73, 17, 3, 78, 88, 93, 8, 134, 133, 138, 85, 138, 13, 133, 31, 18, 132, 138, 68, 67, 21, 78, 18, 13, 62, 92, 81, 54, 53, 97, 188, 18),
            585, 93, 186, 585, 88, 138, 137, 131, 32, 34, 51, 188, 112, 27, 28, 182, 180, 68, 64, 78, 75, 15, 68, 68, 64, 85, 3, 2, 87, 71, 83, 3, 72, 148, 8, 1, 158, 88, 147, 582, 26, 3520]
    ]
}

# Función para convertir minutos a hora y minutos
def minutos_a_hora_minutos(minutos):
    hora = minutos // 60
    minutos_restantes = minutos % 60
    return hora, minutos_restantes

# Función para mapear un nombre de calle a su ID
def obtener_id_calle(nombre_calle):
    return nombres_a_ids.get(nombre_calle, -1)

# Función para predecir el color de la casa
def predecir_color(nombre_calle, dia, hora, minutos):
    # Convertir día a minutos
    dia_minutos = dias_a_minutos.get(dia, 0)
    # Convertir hora y minutos a minutos totales
    minutos_totales = hora * 60 + minutos
    # Obtener el ID de la calle
    id_calle = obtener_id_calle(nombre_calle)
    # Predecir el color
    resultado = modelo.predict([id_calle, dia_minutos, minutos_totales])
    # Mapear el resultado a un color
    color = resultado[0]
    return color

# Ruta para predecir el color de la casa
@app.route('/api/prediccion', methods=['POST'])
def prediccion():
    datos = request.json
    dia = datos['dia']
    hora = datos['hora']
    minutos = datos['minutos']
    nombre_calle = datos['calle']

    # Convertir día a minutos
    dia_minutos = dias_a_minutos.get(dia, 0)
    # Convertir hora y minutos a minutos totales
    minutos_totales = hora * 60 + minutos
    # Obtener el ID de la calle
    id_calle = obtener_id_calle(nombre_calle)
    # Predecir el color
    resultado = modelo.predict([id_calle, dia_minutos, minutos_totales])
    # Mapear el resultado a un color
    color = resultado[0]

    return jsonify(resultado)

if __name__ == '__main__':
    app.run(debug=True)

```

Fig. 19 Código de backend para la comunicación con frontend

Capítulo 6 Pruebas y resultados

En este capítulo se muestran las pruebas realizadas al prototipo con el objetivo de validar el correcto funcionamiento del planificador.

6.1 Experimentación

La experimentación tiene como objetivo realizar pruebas al prototipo desarrollado donde se desea evaluar los patrones de congestión vehicular encontrados mediante una predicción del nivel de congestión vehicular, esto es posible al contar con datos que pueden categorizarse con un nivel de congestión libre, moderada y severa. Los datos de nivel de congestión serán comparados versus niveles de congestión reportadas de Here Maps que es la plataforma donde se adquirieron los datos de tráfico. Las pruebas evalúan la calidad y congruencia del nivel de congestión vehicular para la alcaldía Tlalpan de la CDMX.

Para lograr este objetivo, se realizan las evaluaciones del algoritmo random forest propuesto para la predicción del nivel de congestión. Las pruebas contemplan también realizar una comparación entre el nivel de congestión identificado en los patrones versus niveles de congestión de Here Maps, se tiene el objetivo de demostrar el porcentaje de similitud entre lo que se predice contra el color de congestión vehicular registrado en Here Maps.

La experimentación se realizó con dos tipos de pruebas:

- I. Prueba de modelo de clasificación; el objetivo de esta prueba es realizar una evaluación de métricas al modelo Random Forest, en esta fase se evaluará mediante las métricas accuracy, recall, F1-score y ROC-AUC.
- II. Prueba de comparación; el objetivo de esta prueba es evaluar el porcentaje de similitud; nivel de congestión vs nivel de congestión de Here Maps, esta prueba tiene el objetivo de evaluar en que porcentaje nuestra predicción es similar a los datos reales obtenidos una semana que comprende del 27 de Nov al 3 de Dic del 2023.

6.2 Procedimiento de la experimentación

En esta sección se describen brevemente las pruebas realizadas para evaluar la clasificación del nivel de congestión y la similitud con los niveles de congestión registrados en Here Maps. El objetivo principal de las pruebas es verificar la identificación de patrones de congestión vehicular. Las pruebas se aplican en dos ámbitos diferentes: prueba del modelo de clasificación de nivel de congestión vehicular el cual queremos evaluar el modelo implementado utilizando los datos de la identificación de patrones y prueba de comparación de similitud de nivel de congestión de los patrones versus los niveles de congestión de Here Maps.

6.2.1 Prueba de modelo de clasificación

Una vez identificado los patrones de congestión vehicular, los datos serán utilizados para implementar el algoritmo random forest para la clasificación de nivel de congestión vehicular, esto nos ayuda a predecir como podría ser el comportamiento e identificar si el comportamiento es similar a la de nuestros patrones identificados. Para obtener los mejores parámetros se implementa

una búsqueda por rejilla ya que usa la validación cruzada y así se obtiene una comparación con diferentes parámetros, para evaluar el rendimiento de este modelo se usarán las métricas accuracy, recall, F1-score y ROC-AUC. El objetivo es determinar su rendimiento en la tarea de clasificar los niveles de congestión, al evaluar el modelo con estas métricas, obtenemos una imagen completa de su rendimiento en términos de precisión, capacidad para detectar casos positivos, balance entre precisión y Recall, y capacidad de discriminación entre clases. Esto te ayudará a comprender cómo se está desempeñando el modelo para clasificar el nivel de congestión vehicular.

6.2.2 Prueba de comparación

Para esta prueba de comparación, se utilizarán las clasificaciones del modelo random forest para comparar con los niveles de congestión obtenidos de Here Maps, el objetivo de esta prueba es conocer el porcentaje de similitud de la clasificación del nivel de congestión obtenido del modelo random forest versus los niveles de congestión reportados por Here Maps, para obtener el porcentaje de similitud se compara el nivel de congestión de nuestros patrones vs nivel de congestión de Here Maps, en nuestro sistema contamos con 3 niveles de congestión, que son verde es un flujo libre, amarillo que es una congestión moderada y rojo que es una congestión severa. La evaluación se asigna de la siguiente manera, cuando los valores son iguales se asigna el valor de uno (1) y cuando el valor es diferente se asigna cero(0), posteriormente se suman los valores y se dividen entre el número de segmentos con los que cuente la calle. Esta prueba pretende experimentar con datos obtenidos de 1 semana y con un historico de datos de 5 semanas para obtener el porcentaje general de similitud del nivel de congestión.

6.3 Resultados obtenidos

En esta sección se muestran los resultados obtenidos tras someter al planificador de rutas a pruebas. Como evidencia de las pruebas se obtuvieron los siguientes resultados.

6.3.1 Prueba de modelo de clasificación

En esta prueba tiene como objetivo evaluar las métricas a nuestro modelo Random Forest, las pruebas que pretenden aplicarse son las siguientes precisión, recall y F1-score. El conjunto de datos cuenta con 3 clases de nivel de congestión que es verde para un flujo libre, amarillo para un flujo moderado y rojo que es una congestión severa.

La precisión mide la exactitud de las predicciones positivas. Se calcula como el número de verdaderos positivos dividido por el número de todos los positivos predichos (verdaderos positivos más falsos positivos).

El recall mide la capacidad del modelo para encontrar todas las instancias relevantes dentro de una clase. Se calcula como el número de verdaderos positivos dividido por el número de todos los positivos reales (verdaderos positivos más falsos negativos).

La puntuación F1 es una medida que combina la precisión y el recall en una sola métrica. Es el promedio armónico de la precisión y el recall, dando igual peso a ambos. Se calcula como el doble del producto de la precisión y el recall dividido por la suma de la precisión y el recall.

Como se comentó en la etapa 5.3 se realizó un balanceo de datos, para tener una mejor predicción en el nivel de congestión, se aplicó la técnica SMOTE para balancear los datos, las clases ahora tienen cada una 2,175,037 registros para un total de 6,525,111.

Para evaluar nuestro modelo, una vez entrenado se tomó la parte predictiva de nivel de congestión y se aplicó las métricas de evaluación, teniendo como resultado lo siguiente.

Una vez que se identificaron los mejores parámetros para el algoritmo Random Forest, se implementó a nuestro conjunto de datos balanceado, las características utilizadas fueron minuto del día (0-1435), día de la semana (1-7) y Factor de atasco (0-10), nuestra etiqueta meta era la clase de nivel de congestión, para este conjunto de datos se tomó el 70 por ciento de los datos para entrenamiento y el 30 por ciento restante como nuevas instancias para la clasificación del nivel de congestión.

En la tabla 1 se tiene los resultados que se obtuvieron para cada una de las clases y a continuación se describen los resultados obtenidos y las fórmulas utilizadas para la evaluación.

Tabla 4 Métricas de evaluación

Clases	Precision	Recall	F1-Score
verde	0.88	0.8	0.84
roja	0.8	0.81	0.81
amarillo	0.78	0.83	0.8

$$precisión = \frac{verdaderos\ positivos(VP)}{verdaderos\ positivos(VP) + falsos\ positivos(FP)} \quad (1)$$

$$Recall = \frac{\text{verdaderos positivos}(VP)}{\text{verdaderos positivos}(VP) + \text{falsos negativos}(FN)} \quad (2)$$

$$F1 - Score = \frac{\text{precisión} * \text{recall}}{\text{precisión} + \text{recall}} \quad (3)$$

Clase verde: Esta clase, que representa situaciones con flujo libre de tráfico, muestra un buen equilibrio entre precisión (0.88) y recall (0.8), resultando en la puntuación F1 más alta entre las tres clases (0.84). Esto indica que el modelo es bastante efectivo al identificar correctamente las instancias de la clase verde, con una baja tasa de falsos positivos y falsos negativos.

Clase amarilla: Esta clase, representa situaciones de flujo moderado de tráfico, aunque la precisión es la más baja de las tres clases (0.78), el recall es el más alto (0.83), lo que resulta en una puntuación F1 de 0.8. Esto puede indicar que el modelo tiende a identificar más instancias como pertenecientes a la clase amarilla, incluyendo algunas que no lo son (falsos positivos), pero también logra no omitir muchas instancias reales de esta clase.

Clase roja: Esta clase, representa situaciones de flujo intenso de tráfico, la precisión para esta clase es relativamente alta (0.8), y su recall es casi idéntico (0.81), lo que lleva a una puntuación F1 de 0.81. Esto sugiere que el modelo tiene un desempeño equilibrado en la identificación de esta clase, con una tasa de acierto y error moderadamente uniforme.

La exactitud o accuracy es otra métrica fundamental para evaluar modelos de clasificación. Mide la proporción de predicciones correctas (tanto positivas como negativas) respecto al total de casos examinados. Esta métrica es muy intuitiva y proporciona una visión general rápida de la efectividad del modelo, especialmente en contextos donde las clases están equilibradas. Sin embargo, en situaciones de desbalanceo significativo entre clases, puede dar una visión engañosa del rendimiento del modelo, ya que podría estar sesgada hacia la clase mayoritaria. La fórmula para calcular el Accuracy es la siguiente:

$$accuracy = \frac{\text{verdaderos positivos}(VP) + \text{verdaderos negativos}(VN)}{\text{Total de casos}} \quad (4)$$

En la tabla 2 muestra el accuracy que tiene nuestro modelo que es de 0.80, esto quiere decir que el 80% de las predicciones realizadas por el modelo son correctas. Esto sugiere que el modelo es bastante competente en general para predecir las clases correctamente en el conjunto de datos utilizado. Aunque un 80% de exactitud es notable, aún hay un margen del 20% de error. Analizar los errores (mediante la revisión de los falsos positivos y falsos negativos) podría proporcionar perspectivas sobre cómo mejorar el modelo.

Tabla 5 Exactitud del modelo Random Forest

Acurracy	0.80
----------	------

El ROC-AUC Score, representa el área bajo la curva del gráfico de Característica Operativa del Receptor (Receiver Operating Characteristic-ROC), es una métrica importante para evaluar el rendimiento de un modelo de clasificación, especialmente en problemas de clasificación binaria y multiclase.

One-vs-Rest: Calcula una curva ROC por clase, tratando cada clase como positiva una vez contra todas las demás como negativas, y luego promedia los AUC de todas estas curvas.

One-vs-One: Calcula una curva ROC para cada par de clases, promediando luego todos los AUC obtenidos.

En la tabla 3 nos muestra los resultados de ROC-AUC Score que mide la capacidad del modelo para distinguir entre clases, y un valor cercano a 1 indica un excelente rendimiento. Con un score de aproximadamente 0.943 en ambos métodos, el modelo demuestra una capacidad muy alta para clasificar correctamente entre las diferentes clases. Este alto score sugiere que el modelo tiene un excelente rendimiento en términos de sensibilidad (capacidad para detectar verdaderos positivos) y especificidad (capacidad para evitar falsos positivos).

Tabla 6 Curvas ROC

ROC-AUC Score (One-vs-Rest):	0.943173134
ROC-AUC Score (One-vs-One):	0.943173428

6.3.2 Resultados de las pruebas de funcionalidad

La Calzada Acoxpa se compone de 12 segmentos como se muestran en la fig. 20, cada segmento se identifica con un ID único, para esta prueba se tomaron los segmentos de toda la Calzada

Acoxa. Con esta prueba se desea conocer el porcentaje de similitud entre los niveles de congestión de los patrones identificados versus los niveles de congestión obtenidas por Here Maps.

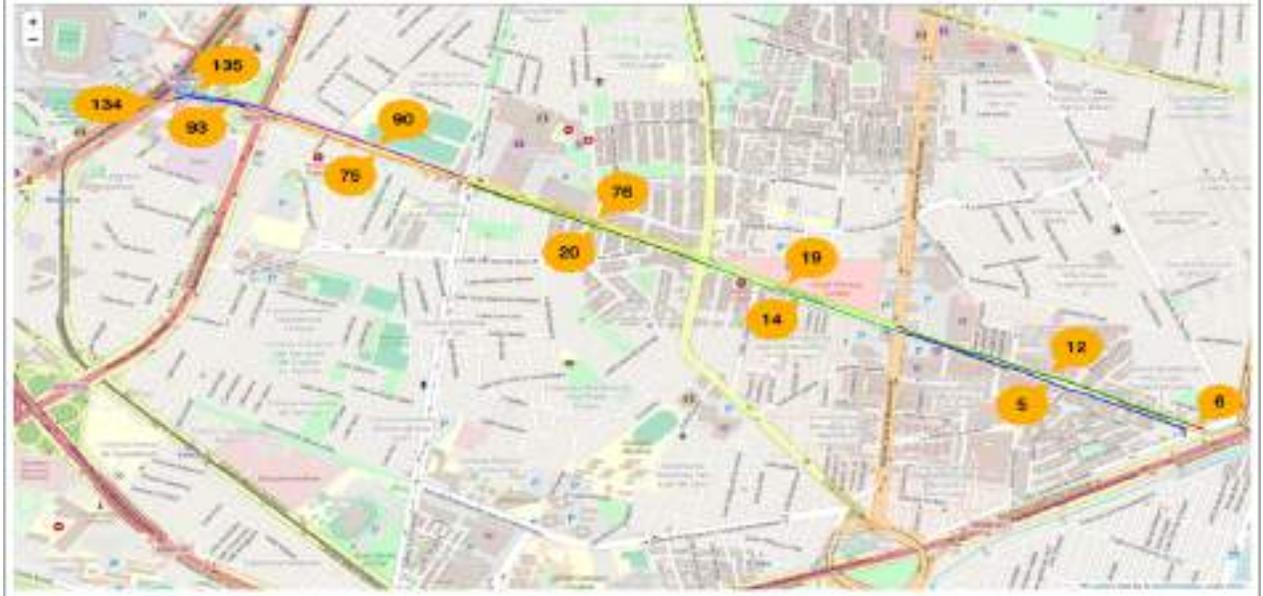


Fig. 20 Análisis de patrones de congestión nivel calle

Para la prueba de comparación de similitud se realizó lo siguiente, en una tabla se colocan los 12 segmentos de la Calzada Acoxa, para cada ID se compara el nivel de congestión de nuestro algoritmo vs el nivel de congestión de Here Maps, comparando estos datos las 24 horas del día, en la tabla 5 podemos observar la descripción de los 12 segmentos con su respectivo nivel de congestión, para calcular el porcentaje de coincidencia se realiza de la siguiente manera

$$\text{porcentaje de coincidencia} = \frac{\text{Número de coincidencias}}{\text{Total de elementos comparados}} * 100 \quad (5)$$

En la tabla 4 observamos el nivel de congestión de nuestra predicción vs el nivel de congestión de Here Maps, en la columna de porcentaje de coincidencia aplicamos la fórmula (5) y obtenemos el valor correspondiente. Con este resultado podemos conocer el porcentaje que se parecen nuestros patrones de predicción contra los niveles de congestión de Here Maps. Las celdas marcadas en

En la tabla 11 realizamos una comparación de todas las instancias que se tienen por cada día, los datos fueron recolectados cada 5 min y estas instancias son las que se compararon. En las tablas anteriores se comparaba por hora, en la tabla 12 estamos comparando el porcentaje de similitud de todo el día para cada segmento de la Calzada Acoxa, y así obtenemos con que nivel predcimos comparado con los registros de Here Maps.

Tabla 14 Evaluación de las predicciones del modelo versus datos adquiridos durante 5 semanas

ID de Segmento	Porcentaje coincidencia Lunes	Porcentaje coincidencia Martes	Porcentaje coincidencia	Porcentaje coincidencia Jueves	Porcentaje coincidencia Viernes	Porcentaje coincidencia Sábado	Porcentaje coincidencia Domingo
135	81.8541667	82.2337963	81.53935185	78.7037037	82.80280741	81.77083333	81.51935185
134	89.18981481	98.99833333	97.38583333	98.17962963	98.14814815	99.47916667	99.47916667
93	97.56944444	96.46990741	92.30324074	92.47685185	93.46064815	96.76138889	97.91666667
90	79.16666667	76.27314815	76.79398148	73.09027778	74.47916667	73.32175926	74.16981481
76	57.87037037	67.32962963	68.05555556	79.3125	75.9837963	71.41203704	48.17962963
75	69.7337963	69.61805556	74.18081481	71.81018519	74.51703704	63.34605556	48.71681481
20	70.8912037	68.51851852	71.70136889	70.83333333	76.5625	68.43518519	63.86868889
19	74.88423528	63.77314815	63.39531852	64.40572222	61.22685185	66.99937037	63.25231481
14	71.23842593	82.00231481	83.3912637	80.63342593	83.33333333	83.50694444	69.15509259
12	71.41203704	70.42824074	67.24537037	65.74074074	63.77314815	67.70833333	72.64861111
6	99.24768519	99.24768519	98.50046296	98.66898148	99.0162037	99.34342593	99.33703704
5	89.07407407	68.68898148	99.07407407	98.61111111	88.72685185	96.11194444	99.0162037
Porcentaje por día	81.17768204	81.1101466	81.33198302	80.03733838	81.80459105	80.2662037	76.41746934

Como conclusión se observa que el porcentaje de coincidencia de la Calzada Acoxa es muy aceptable, de acuerdo a las evaluaciones nos encontramos dentro de ese 80% de precisión, aun se tiene un margen para mejorar, con esto podemos observar que los patrones identificados si coinciden con los datos obtenidos de niveles de congestión recolectados posteriormente en Here Maps.

Capítulo 7 Conclusiones y trabajos futuros

En este capítulo se presentan las conclusiones derivadas de la identificación de patrones de congestión vehicular. Además, se exploran oportunidades para el avance futuro de la movilidad urbana e identificamos áreas clave de investigación y desarrollo.

7.1 Conclusiones

Esta tesis tuvo como objetivo identificar patrones de congestión vehicular en la alcaldía Tlalpan de la Ciudad de México. A lo largo del desarrollo del estudio, se lograron determinar los horarios en los que el tráfico experimenta una transición de flujo libre a congestión severa. Estos hallazgos permiten identificar las calles y periodos críticos, proporcionando una base para mejorar la infraestructura vial o profundizar en el análisis de las causas que originan la congestión severa.

Esta tesis aporta la identificación de los atributos para construir un conjunto de datos que ayuda a modelar los patrones de congestión vehicular. Esta aportación ayuda a que no se adquieran y almacenen datos que no son relevantes para analizar y determinar patrones de congestión vehicular.

La estructura de datos y la metodología utilizada ayudan a que pueda implementarse en la identificación de patrones en otra ciudad, cada ciudad tiene un comportamiento diferente por lo cual concluimos que los patrones de comportamiento no pueden ser iguales, sin embargo, esta tesis aporta una estructura para poder adaptarse a los datos de cada ciudad e identificar sus patrones.

Realizar un prototipo de código abierto destaca una gran ventaja para el uso y mejoras futuras para la visualización de los patrones de congestión vehicular, el uso de recursos mínimos para la implementación de un modelo de aprendizaje automático y este pueda desplegarse en un servidor local ayuda a dar flexibilidad de análisis y toma de decisiones. El prototipo desarrollado puede ser reutilizable en proyectos con un enfoque a movilidad urbana para mejorar las características de la visualización de patrones de congestión vehicular.

Los patrones de congestión vehicular tienen un 80% de exactitud, lo cual es aceptable para este primer acercamiento a un modelo de identificación de patrones de congestión vehicular en la alcaldía Tlalpan de la CDMX, al tener un tiempo de adquisición de datos cada 5 minutos puede influir en la exactitud de estos cambios de congestión.

Se puede concluir que este trabajo de tesis tiene una aportación académica en el área de la investigación para una metodología de solución para la identificación de patrones y en el área de la computación al poder implementar aprendizaje automático para identificar patrones en un gran conjunto de datos.

7.2 Trabajos futuros

Uno de los principales trabajos futuros es implementar la capacidad de adquisición de datos de tráfico en intervalos de un minuto. Actualmente, los datos suelen recolectarse en períodos de 5 minutos, lo que puede provocar la pérdida de detalles significativos en el comportamiento del tráfico. La captura de datos cada minuto permitiría un análisis más granular y preciso, reflejando variaciones sutiles y temporales en la congestión que pueden ser esenciales para un modelo altamente efectivo. La posibilidad de trabajar con datos a este nivel de detalle permitiría no solo modelar los patrones de tráfico más consistentes, sino también identificar fenómenos transitorios que tienen un gran impacto en la congestión. Esto resultaría en un modelo de comportamiento del tráfico que puede ajustarse y replicarse en ciudades con características y problemáticas de movilidad similares a la Ciudad de México (CDMX), proporcionando una herramienta que podría adaptarse a otras metrópolis con una infraestructura y densidad urbana comparable.

Otro trabajo futuro fundamental es la implementación de técnicas avanzadas de selección y almacenamiento de datos basadas en patrones de comportamiento de la congestión. Este enfoque se centraría en identificar aquellos datos que presentan un comportamiento estable o repetitivo y, en consecuencia, archivar únicamente los datos que muestren variaciones o anomalías en la congestión. La idea detrás de este enfoque es optimizar el almacenamiento de datos y reducir la carga de procesamiento, priorizando solo aquellos puntos de datos que ofrezcan información nueva o relevante para el modelo. Esto no solo disminuye la carga de almacenamiento y los costos asociados, sino que también mejora la eficiencia y el rendimiento del sistema, permitiendo el análisis en tiempo real de forma más efectiva y accesible. Además, la identificación de patrones recurrentes podría facilitar la creación de una base de datos de “eventos críticos” de tráfico, permitiendo a los desarrolladores y autoridades tener una visión más clara de las causas detrás de la congestión.

Finalmente, se contempla la creación de una herramienta de navegación avanzada que permita al usuario seleccionar un punto de origen y un destino, calculando y mostrando la ruta óptima para reducir el tiempo de viaje, considerando patrones de congestión históricos y actuales. Este sistema no solo ayudaría a los conductores a reducir tiempos de traslado, sino que también podría informar a los usuarios sobre las condiciones y cambios del tráfico en tiempo real, sugiriendo rutas alternativas que se ajusten al comportamiento de congestión en ese momento. Este trabajo futuro sería un paso importante hacia una movilidad más inteligente y sostenible, permitiendo una mayor flexibilidad y capacidad de respuesta a las condiciones cambiantes de tráfico. Además, la capacidad de predecir rutas óptimas basadas en patrones de congestión específicos sería un recurso valioso para aplicaciones de planificación urbana y para las políticas de gestión de tráfico en áreas de alta densidad.

Referencias

A continuación, se muestran las referencias utilizadas en la realización de esta tesis. Las referencias fueron consultadas en diferentes fuentes de información.

Bibliografía

- La Vanguardia. (9 de abril de 2018). *www.lavanguardia.com*. Obtenido de <https://www.lavanguardia.com/ocio/viajes/20180409/442244212659/aplicaciones-planificar-viaje.html>
- Catala, M., Downing, S., & Hayward, D. (2011, noviembre 15). Expanding the Google Transit Feed Specification to Support Operation and Planning. *National Center for Transit Research*. Retrieved from <https://rosap.ntl.bts.gov/view/dot/39780>
- Celi, S. F. (20 de enero de 2018). Análisis del transporte público a nivel mundial. *Espacios*, Vol. 39, 10. Obtenido de <https://www.revistaespacios.com/a18v39n18/a18v39n18p10.pdf>
- Li, W., Chen, X., & Yang, B. (2010, febrero 2). Bus Travel Transit Path Query Algorithm Based on Ant Algorithm. *Third International Conference on Genetic and Evolutionary Computing*, 665-669. doi:10.1109/WGEC.2009.13
- Colque, A., Valdivia, R., Navarrete, M., & Aracena, S. (2019, noviembre 29). Un sistema de información geográfico para el transporte público basado en el estándar GTFS realtime. *Scielo*, 29(1), 51-62. Retrieved from https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052021000100051
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to Algorithms. En F. Edition (Ed.), *Introduction to Algorithms* (págs. 5-12). Cambridge, Massachusetts: The MIT Press. doi:ISBN 9780262046305
- López, D., Lozano, A., González, H., Guzmán, A., & Maldonado, F. (2018). Hiperpuma: Sistema Multimodal de Información al Viajero. En P. Fernández, M. Suárez, & H. Quiroz, *La movilidad en la Ciudad de México Impactos, conflictos y oportunidades* (págs. 119-151). Ciudad de México: IG.
- Alzaidi, M., & Vagner, A. (31 de mayo de 2021). Trip planning algorithm for GTFS data with NoSQL structure to improve the performance. *Jatit*, Vol. 99, 10. Obtenido de <http://www.jatit.org/volumes/Vol99No10/10Vol99No10.pdf>
- BBVA. (s.f.). *BBVA*. Obtenido de <https://www.bbva.mx/educacion-financiera/blog/que-es-una-smart-city.html>
- Beneyto, R. (19 de Mayo de 2022). *www.imbric.com*. Recuperado el 24 de Abril de 2023, de <https://www.imbric.com/la-importancia-del-transporte-publico-en-las-ciudades/>

- Boris Medina Salgado, E. S.-D.-P. (2021). Análisis de técnicas inteligentes para la predicción del flujo de tráfico urbano. *Komputer Sapiens, XIII(III)*, 54-73.
- Bozyiğit, A., Alankuş, G., & Nasiboğlu, E. (2017, octubre 1). Public transport route planning: Modified dijkstra's algorithm. *International Conference on Computer Science and Engineering (UBMK)*, 502-505. doi:10.1109/UBMK.2017.8093444.
- Brotton, J. (2014). *Historia del mundo en 12 mapas*. España: Penguin Random House.
- Bull, A. (2003). *Congestión de tránsito el problema y cómo enfrentarlo*. CEPAL.
- DEMOGRAPHIA. (2022). DEMOGRAPHIA WORLD URBAN AREAS (Built Up Urban Areas or World Agglomerations). *DEMOGRAPHIA WORLD URBAN AREAS*, 93.
- Fundación Telefónica. (19 de diciembre de 2011). Smart Cities: un primer paso hacia la internet de las cosas. 13-16.
- Google. (6 de octubre de 2021). *API de Google Transit*. Obtenido de <https://developers.google.com/transit/gtfs?hl=es>
- Google support. (22 de octubre de 2021). *support.google.com*. Obtenido de <https://support.google.com/maps/answer/144339?hl=es&co=GENIE.Platform%3DDesktop>
- Google transit. (10 de septiembre de 2021). *developers.google.com*. (Google) Obtenido de <https://developers.google.com/transit/gtfs/guides>
- Ian Thomson, A. B. (2001). *La congestión del tránsito urbano: causas y consecuencias económicas y sociales*. Santiago de Chile.
- Ibáñez, M. O. (29 de Septiembre de 2022). *Banco mundial*. Recuperado el 24 de Abril de 2023, de <https://www.bancomundial.org/es/topic/transport/overview#:~:text=El%20transporte%20es%20fundamental%20para,estos%20beneficios%20no%20se%20materializan>.
- IBM. (1 de junio de 2023). <https://cloud.ibm.com>. Recuperado el 20 de junio de 2023, de <https://cloud.ibm.com/docs/assistant?topic=assistant-dialog-responses-json&locale=es#:~:text=El%20formato%20JSON%20gen%C3%A9rico%20de,por%20una%20aplicaci%C3%B3n%20cliente%20personalizada>.
- INEGI. (29 de enero de 2021). En la Ciudad de México somos 9 209 944 habitantes: Censo de población y vivienda 2020. *censo 2020*, 1-6. Obtenido de https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2021/EstSociodemo/ResultCenso2020_CdMx.pdf
- Jesús Ariel Carrasco Ochoa, J. F. (2011). Reconocimiento de patrones. *Komputer Sapiens, III(II)*, 5-32.

- Jiawei Han, M. K. (2012). *Data Mining Concepts and Techniques*. ELSEVIER.
- Kong, X., Li, M., Tang, T., Tian, K., Moreira-Matias, L., & Xia, F. (2018, septiembre 11). Shared Subway Shuttle Bus Route Planning Based on Transport Data Analytics. *IEEE Transactions on Automation Science and Engineering*, Vol. 15(4), 1507-1520. doi:10.1109/TASE.2018.2865494
- McGlone, D. (13 de agosto de 2013). *Using GTFS Data to Generate Bus Routes with Travel Time*. Obtenido de azavea.com: <https://www.azavea.com/blog/2013/08/13/using-gtfs-data-to-generate-bus-routes-with-travel-time/>
- Michael A.P. Taylor, P. W. (2016). *Understanding Traffic Systems: Data, Analysis and Presentation*. Routledge.
- Negrete, M. E. (1 de marzo de 2018). El desafío de la movilidad y el transporte urbano y metropolitano. *Foro Consultivo Científico y Tecnológico, AC*, 9 - 11. Recuperado el 24 de Abril de 2023, de http://www.foroconsultivo.org.mx/proyectos_estrategicos/img/8/23.pdf
- Notimex. (19 de febrero de 2018). <https://unamglobal.unam.mx>. Obtenido de <https://unamglobal.unam.mx/microbus-o-combi-transporte-publico-mas-utilizado-en-el-valle-de-mexico/>
- Ochoa, J. A. (2018). *Reconocimiento de Patrones*. (INAOEP) Obtenido de <https://ccc.inaoep.mx/~ariel/>
- Ortega, F. A., Marseglia, G., Mesa, J. A., & Piedra de la Cuadra, R. (2021). Un algoritmo para planificar rutas más rápidas con arcos dependientes del tiempo en redes urbanas. En R.-E. e. transporte. Sevilla, España: Universidad de Burgos. Servicio de Publicaciones e Imagen Institucional. doi:10.36443/10259/6927
- Peña, J. A. (3 de septiembre de 2012). Sistemas de transporte en México: un análisis de centralidad en teoría de redes. *EMALCA*, Vol. 3, 3. Obtenido de <https://rde.inegi.org.mx/index.php/2012/09/10/sistemas-de-transporte-en-mexico-un-analisis-de-centralidad-en-teoria-de-redes/>
- Pérez, A. (17 de Agosto de 2017). *bixpe.com*. Recuperado el 25 de Abril de 2023, de <https://www.bixpe.com/blog/beneficios-de-la-planificacion-de-rutas/>
- Queiroz, A. R., Santos, V., Nascimento, D., & Pires, C. E. (2019, noviembre 26). Conformity Analysis of GTFS Routes and Bus Trajectories. *UFCEG*, 1-6. Retrieved from <https://sol.sbc.org.br/index.php/sbbd/article/view/8823/8724>
- Quezada, D. J. (2017). *Movilidad urbana en México*. México: Senado de la República instituto beliSaRio domínguez.

- Sánchez, G., & Lozano, V. M. (16 de julio de 2001). Algoritmo de Dijkstra. Un Tutorial Interactivo. *VII Jornadas de Enseñanza Universitaria de la Informática, JENUI*, 1-5. Obtenido de <http://rua.ua.es/dspace/handle/10045/128198>
- Systems, B. (s.f.). *Blue Systems*. Recuperado el 30 de Noviembre de 2022, de <https://www.bluesystems.io>
- Szincsák, T., & Vágner, A. (29 de enero de 2015). Public transit schedule and route planner application for mobile devices. *Eger, Vol. 2*, 153-161. doi:10.14794/ICAI.9.2014.2.153
- TomTom. (s.f.). *TomTom*. Obtenido de <https://www.tomtom.com/traffic-index/ranking/>
- Vágner, A. (10 de julio de 2021). Route planning on GTFS using Neo4j*. *Annales Mathematicae et Informaticae*, 1-17. doi:10.33039/ami.2021.07.001
- Waze. (22 de abril de 2021). *medium.com*. Obtenido de <https://medium.com/waze/how-to-improve-your-mobility-habits-for-a-greener-future-e07c01a8082e>
- Weijermars, W. (2007). *Analysis of urban traffic patterns using clustering*. Netherlands.
- Wong, J., Landon, R., Kari, W., & Regan, H. (2013, enero 13). Open Transit Data: State of the Practice and Experiences from Participating Agencies in the United States. *Transportation Research Broad*. Retrieved from <https://www.semanticscholar.org/paper/Open-Transit-Data%3A-State-of-the-Practice-and-from-Wong-Reed/a2b17e6f4fe7b738b4c3b8982cff9f446df1c398>
- Yuxuan Liang, Z. J. (2017). Inferring Traffic Cascading Patterns. *SIGSPATIAL*.