



UNIVERSIDAD ESAN  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE TECNOLOGÍAS DE INFORMACIÓN Y SISTEMAS

**Desarrollo de un Algoritmo de aprendizaje profundo para detectar rutas seguras de las  
calles de los Ángeles**

Trabajo de investigación para el curso de Trabajo de Tesis I

Nombre Guillermo Díaz Beltrán

Asesor: Marks Calderón

Lima, 2 de junio de 2024

# Índice general

<b>Índice de Figuras</b>	<b>4</b>
<b>Índice de Tablas</b>	<b>6</b>
<b>1. PLANTEAMIENTO DEL PROBLEMA</b>	<b>7</b>
1.1. Descripción de la Realidad Problemática . . . . .	7
1.2. Formulación del Problema . . . . .	9
1.2.1. Problema General . . . . .	9
1.2.2. Problemas Específicos . . . . .	10
1.3. Objetivos de la Investigación . . . . .	10
1.3.1. Objetivo General . . . . .	10
1.3.2. Objetivos Específicos . . . . .	10
1.4. Justificación de la Investigación . . . . .	11
1.4.1. Teórica . . . . .	11
1.4.2. Práctica . . . . .	11
1.4.3. Metodológica . . . . .	12
1.5. Delimitación del Estudio . . . . .	12
1.5.1. Espacial . . . . .	12
1.5.2. Temporal . . . . .	12
1.5.3. Conceptual . . . . .	12

1.6. Hipótesis . . . . .	13
1.6.1. Hipótesis General . . . . .	13
1.6.2. Hipótesis Específicas . . . . .	13
1.6.3. Matriz de Consistencia . . . . .	13
<b>2. MARCO TEÓRICO</b>	<b>14</b>
2.1. Antecedentes de la investigación . . . . .	14
2.1.1. Real-time deep reinforcement learning based vehicle navigation ( <b>pr'koh</b> )	14
2.1.2. SafeRoute: Learning to navigate streets safely in an urban environment ( <b>pr'saferoute</b> ) . . . . .	16
2.1.3. Predicting motor vehicle theft in Santiago de Chile using graph-convolutional LSTM ( <b>pr'esquivel</b> ) . . . . .	19
2.1.4. Algoritmo para calcular la ruta más segura y óptima ( <b>pr'areiza</b> ) . . . . .	22
2.1.5. Route-the safe: A robust model for safest route prediction using crime and accidental data ( <b>pr'Soni</b> ) . . . . .	24
2.1.6. Real-time safest route identification: Examining the trade-off between safest and fastest routes ( <b>pr'Ghoul</b> ) . . . . .	28
2.1.7. A reinforcement learning-based routing algorithm for large street net- works. ( <b>pr'diya</b> ) . . . . .	29
2.1.8. An Enhanced Motif Graph Clustering-Based Deep Learning Approach for Traffic Forecasting ( <b>pr'zhang</b> ) . . . . .	32
2.2. Bases Teóricas . . . . .	34
2.2.1. Machine Learning . . . . .	34
2.2.2. Natural Language Processing (NLP) . . . . .	34
2.3. Marco Conceptual . . . . .	34
<b>Anexos</b>	<b>36</b>
<b>A. Anexo I: Matriz de Consistencia</b>	<b>37</b>

A.1. Árbol de Problemas . . . . .	37
A.2. Árbol de Objetivos . . . . .	39
<b>B. Anexo II: Resumen de Papers investigados</b>	<b>40</b>
<b>BIBLIOGRAFÍA</b>	<b>42</b>

# Índice de Figuras

1.1. Índice de criminalidad de los países de América 2023 . . . . .	8
1.2. Días con más robos de celulares registrados en el 2023 Osiptel . . . . .	9
2.1. Framework propuesto . . . . .	15
2.2. Comparación de rendimiento de los modelos . . . . .	16
2.3. Análisis de significancia de los modelos con la prueba de Wilcoxon . . . . .	16
2.4. Metodología propuesta por los autores . . . . .	17
2.5. Estructura de la red neuronal del modelo propuesto . . . . .	18
2.6. Comparación de rendimiento de los modelos . . . . .	19
2.7. Metodología propuesta por los autores . . . . .	20
2.8. Estructura de la red convolucional LSTM basada en grafos . . . . .	21
2.9. Riesgo de acoso sexual mediante combinación lineal . . . . .	23
2.10. Metodología propuesta por los autores . . . . .	23
2.11. Metodología Propuesta . . . . .	25
2.12. Flujo de condición para detectar rutas seguras . . . . .	27
2.13. Resultados obtenidos . . . . .	27
2.14. Metodología propuesta por los autores . . . . .	28
2.15. Puntuaciones de evaluación . . . . .	31
2.16. Puntuaciones de evaluación . . . . .	31
2.17. Puntuaciones de evaluación . . . . .	31

2.18. Metodología EMGC-STGCN propuesta por los autores . . . . .	33
2.19. Resultados de modelos propuestos con el conjunto de datos Nav-BJ . . . . .	33
2.20. Resultados de modelos propuestos con el conjunto de datos PeMS . . . . .	34

# Índice de Tablas

2.1. Tabla de resultados. . . . .	24
A.1. Matriz de consistencia. Fuente: Elaboración propia . . . . .	38
B.1. Cuadro Resumen de Papers investigados. Fuente: Elaboración propia . . . . .	41

# Capítulo 1

## PLANTEAMIENTO DEL PROBLEMA

### 1.1. Descripción de la Realidad Problemática

De acuerdo al análisis del crecimiento y expansión urbana del Centro Nacional de Planeamiento Estratégico (CEPLAN) del 2023, menciona que con respecto al censo 2017 del INEI, ha crecido las zonas urbanas a un 1.6 % en comparación al censo de 2007, lo que quiere decir que la población urbana ha crecido en aproximadamente 3 millones de habitantes (**cu'ceplan**). Este crecimiento de la urbanización también a traído consigo un aumento significativo a las acciones criminales que enfrenta Perú, y también en todo el mundo; como es el caso de California (Estados Unidos), ya que se revela que los crímenes violentos, robos y asaltos agravados ha aumentado en un 5.7 % en el año 2023 (**tec'PPIC**). Esto refleja una gran preocupación de la seguridad ciudadana en la que enfrenta la población y las autoridades que buscan tomar medidas cruciales para proteger a los residentes de los entornos urbanos.

El índice de criminalidad en Lima representa un 70.9 %, que se sitúa en el puesto número 17, por debajo de ciudades de Latinoamérica como Caracas (Venezuela), San Pedro Sula (Honduras) y Río de Janeiro (Brasil) que también enfrentan varios desafíos para reducir las acciones delictivas y mejorar la seguridad ciudadana (**cu'numbeo2023**). A pesar de eso, el índice de representatividad de crímenes en Lima no resta importancia a la clara necesidad de encontrar alternativas efectivas que permita reducir dicho índice de criminalidad en otras áreas urbanas del Perú.

Según datos recopilados en la plataforma de Statista entre Julio y diciembre del 2023, por cada 100 habitantes en las zonas urbanas de Perú, el 11.9 % de esta población ha sido víctimas por el robo de pertenencias personales, como dinero, carteras o celulares, siendo esta, la más representativa y la que refleja la parte vulnerable de los ciudadanos al no hacer frente a

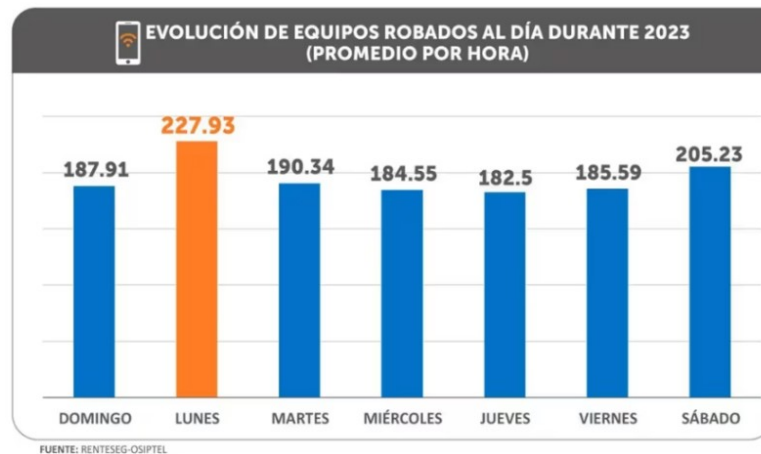


Puesto	Ciudad	Índice de Criminalidad	Índice de Seguridad
1	Caracas, Venezuela	83,59	16,41
2	San Pedro Sula, Honduras	80,55	19,45
3	Río de Janeiro, Brasil	77,63	22,37
4	Fortaleza, Brasil	77,34	22,66
5	Natal, Brasil	77,18	22,82
6	Salvador de Bahía, Brasil	76,65	23,35
7	Recife, Brasil	76,36	23,64
8	Puerto España, Trinidad y Tobago	76,21	23,79
9	Baltimore, Maryland, Estados Unidos	75,49	24,51
10	Rosario, Argentina	75,18	24,82
11	Memphis, Tennessee, Estados Unidos	74,79	25,21
12	Detroit, Michigan, Estados Unidos	74,12	25,88
13	Porto Alegre, Brasil	72,39	27,61
14	Guayaquil, Ecuador	71,58	28,42
15	Albuquerque, Nuevo México, Estados Unidos	71,50	28,50
16	Tijuana, México	71,44	28,56
17	Lima, Perú	70,90	29,10
18	Kingston, Jamaica	70,90	29,10
19	Saint Louis, Missouri, Estados Unidos	70,64	29,36
20	Sao Paulo, Brasil	70,49	29,51
21	Cali, Colombia	70,23	29,77
22	Ciudad de México, México	68,49	31,51

**Figura 1.1:** Índice de criminalidad de los países de América 2023

los robos que se cometen en las calles. Si se toma en cuenta el tipo de acto delictivo que puede atender a la vida humana en las calles, se tiene que, el robo de vehículos representa un 2 %, mientras que el secuestro y extorsión el 0.2 %; que, si bien son menos frecuentes, representan una amenaza al bienestar y seguridad de los ciudadanos de las zonas urbanas (**cu'statista2023**). El último informe del Organismo Supervisor de Inversión Privada en Telecomunicaciones (Osiptel) de 2023, se registraron un total de 1,706,643 casos de robo de celulares en el año 2023, el cual el mayor número de robo ha sido los días lunes (**cu'infobae**). Asimismo, se informó que, entre el periodo de enero y mayo del año 2023, por cada hora, se roban en promedio, 200 celulares a nivel nacional (**cu'comercio**). Estos resultados no solo representan una pérdida material y económica por parte de los afectados, genera desconfianza y preocupación sobre su seguridad personal y la posible vulneración en su privacidad de información que se contenía en el dispositivo del agraviado.

Con lo que respecta al robo de vehículos, que también atenta a la vida y seguridad de la población de las zonas urbanas, según la Asociación Automotriz del Perú (APP) en el año 2023, la venta de vehículos ha representado un aumento del 2.4 % (181,000 autos vendidos) con respecto al año 2022 (**cu'amcha**). Este incremento en la venta de autos, también lleva consigo el aumento a los robos de vehículos, que es un problema que se está volviendo cada vez más persistente en Perú, ya que, según la Superintendencia Nacional de Registros Públicos,



**Figura 1.2:** Días con más robos de celulares registrados en el 2023 Osipitel

en promedio se roban 46 autos al día y que en el primer semestre del año 2022 se ha registrado un total de 9,758 autos robados en el Perú (**cu'track**).

Estas cifras demuestran la necesidad de desarrollar herramientas y/o estrategias que permitan mitigar los riesgos relacionados con los actos delictivos en entornos rurales, donde se destaca el robo de celulares y vehículos, que, a su vez, pueden estar relacionadas entre sí. Las aplicaciones móviles para ayudar al usuario a elegir rutas más cortas para llegar a su destino en el menor tiempo posible mediante un motor de crowdsourcing (recopilación de tráfico), son cada vez más comunes y fundamentales de utilizar; ya que, en 2022, más de 150 millones de personas utilizan Waze; mientras que Google maps ha sido instalada por más de 10 mil millones de usuarios (**cu'luishernan**). Sin embargo, estas herramientas de ayuda no tienen una funcionalidad que les permita detectar entre qué calles son seguras o no tomando en cuenta información o denuncias de delitos cometidos en esas zonas, ocasionando que los usuarios queden expuestos a situaciones peligrosas como el robo de su celular, de su auto o ambas.

## 1.2. Formulación del Problema

formulación de los problemas de la presente investigación, se elaboró un «árbol de problemas» (véase Anexo [A.1](#)).

### 1.2.1. Problema General

¿Es posible utilizar un algoritmo de aprendizaje profundo para detectar rutas seguras en las calles de los Ángeles?

### 1.2.2. Problemas Específicos

- ¿Qué algoritmo de aprendizaje profundo es el más representativos de los antecedentes?
- ¿Qué alternativas de procesamiento de datos se deben aplicar a la base de datos para entrenar el modelo?
- ¿Qué métricas propuestas en los antecedentes existen para ayudar en la eficiencia y eficacia del modelo propuesto?
- ¿El modelo propuesto evita las zonas con alto nivel de delincuencia?

## 1.3. Objetivos de la Investigación

Para la formulación de los objetivos de la presente investigación se elaboró un «árbol de objetivos» (véase Anexo [A.2](#)).

### 1.3.1. Objetivo General

Desarrollar un algoritmo de aprendizaje profundo para detectar rutas seguras en las calles de los Ángeles.

### 1.3.2. Objetivos Específicos

- Determinar qué algoritmo de aprendizaje profundo es el más representativo de cada antecedente para la detección de rutas seguras.
- Seleccionar una alternativa de procesamiento de datos que permita entrenar un modelo de detección de rutas seguras.
- Evaluar la factibilidad del modelo con las métricas propuestas en los antecedentes para mejorar el tiempo de procesamiento con los menores recursos.
- Demostrar que el modelo propuesto evita las zonas con alto nivel de delincuencia.

## 1.4. Justificación de la Investigación

### 1.4.1. Teórica

Esta investigación se realiza con el propósito de desarrollar un algoritmo que pueda realizar el procesamiento de direcciones y el cálculo de riesgo en los diferentes nodos que existen en un grafo a fin de poder generar métodos de compensación para llegar a un destino de manera segura y rápida. Además, dicho algoritmo será realizado bajo Aprendizaje profundo, para crear un modelo robusto de detección con menos tiempo de procesamiento y recursos para detectar rutas seguras que eviten zonas de alto nivel de crímenes registrados en una ciudad; asimismo, de ser un modelo escalable que no dependa de volver a pasar por el proceso de entrenamiento para cuando se quiera detectar una ruta distinta a otra ciudad.

Actualmente, existen investigaciones que utilizan modelos tradicionales como el algoritmo de Dijkstra, pero que carecen de escalabilidad y su complejidad computacional aumenta a medida que el tamaño de los datos y nuevos escenarios aumenten (Diya Lia y e.al, 2023). Por último, en el Perú, estos tipos de proyectos nos son muy comunes, por lo que el modelo propuesto puede aportar a las nuevas investigaciones en el uso de algoritmos de Aprendizaje Profundo utilizando grafos.

### 1.4.2. Práctica

Al culminar la investigación, se podrá utilizar el algoritmo propuesto, el cuál puede recibir por parte del usuario la dirección a la cuál quiere ir y la dirección en la que se encuentra, para mostrarle en un tiempo de ejecución menor la ruta más segura a la cuál puede ir. Además, podrá tener la capacidad de identificar rutas seguras a partir nuevos datos de crímenes registrados en distintas ciudades.

La presente investigación demostrará que un modelo de aprendizaje profundo puede detectar rutas seguras en grafos cada vez más complejos y ser escalables en comparación a los métodos tradicionales, mejorando de esta manera la calidad de vida de las personas, disminuir el índice de criminalidad, informar a los usuarios que priorizan su seguridad o que no tienen conocimiento de las calles de una ciudad a la que visitan a decidir el camino que debe tomar para sentir más seguro.

### 1.4.3. Metodológica

. La implementación de este algoritmo puede ayudar a las personas a conocer qué rutas son seguras para transportarse y evitar ser víctima de algún crimen, ya que, si bien se cuenta con aplicaciones de enrutamiento en tiempo real como Waze o Google Maps, estos no cuentan con la capacidad de detectar qué rutas son las más seguras, debido a que no trabajan con datos de registros de delitos. Asimismo, esta investigación puede aportar a la disminución del índice de criminalidad de alguna ciudad e informar a las personas qué rutas no deben tomar.

Para ello, se utilizaron técnicas de Aprendizaje Profundo aplicados en grafos que son entrenados con un conjunto de datos de crímenes registrados en las calles de una ciudad.

## 1.5. Delimitación del Estudio

### 1.5.1. Espacial

Para la presente investigación se realizará con los registros de actos delictivos recolectados y almacenados en la web oficial del Departamento de Policía de los Ángeles (APD) para uso público, la cual consiste en denuncias o actos registrados por personas, junto con el tipo de delincuencia y la dirección dónde ocurrió el hecho. Estos datos están registrados en formato “csv”. Además, se utilizará librerías que permitan cartografiar las calles de la ciudad de los Ángeles y poder utilizarlos como grafos.

### 1.5.2. Temporal

La presente investigación analizará un conjunto de datos de registros delictivos de la ya mencionada base de la web oficial del APD, que empieza desde el año 2020 hasta los últimos registros descargados a la fecha que se inicia este trabajo.

### 1.5.3. Conceptual

Esta investigación se centrará en el desarrollo de un algoritmo que logre detectar qué rutas de la ciudad de los Ángeles serán seguras para evitar algún caso de crimen basado en un indicador de riesgo. Para lograrlo, se necesitará herramientas de Aprendizaje Profundo y métodos de procesamiento de datos para desarrollar un modelo que se adapte a los diferentes estilos de grafos de forma óptima. El modelo desarrollado busca poder dar una contribución

a la comunidad con el fin de evitar situaciones peligrosas y apoyar a los distintos métodos de reducción en la tasa de criminalidad que se presenta en las zonas urbanas; asimismo aumentar la participación en técnicas de Aprendizaje Profundo utilizando grafos.

## 1.6. Hipótesis

### 1.6.1. Hipótesis General

Mediante el desarrollo de un algoritmo de aprendizaje profundo se logra detectar rutas seguras en las calles de los Ángeles.

### 1.6.2. Hipótesis Específicas

- La selección del algoritmo de aprendizaje profundo más representativos de cada antecedente mejora significativamente la precisión y eficiencia al detectar rutas seguras.
- La identificación de una alternativa de procesamiento de datos influye positivamente para poder entrenar un modelo de detección de rutas seguras.
- La utilización de métricas propuestas en los antecedentes es factible para optimizar el tiempo de procesamiento del modelo.
- Se demuestra que el modelo propuesto evita las zonas con alto nivel de delincuencia.

### 1.6.3. Matriz de Consistencia

A continuación se presenta la matriz de consistencia elaborada para la presente investigación (véase Anexo [A.1](#)).

## Capítulo 2

# MARCO TEÓRICO

### 2.1. Antecedentes de la investigación

En esta sección se presentarán diversos artículos de investigación o tesis las cuales abordarán diversas técnicas y enfoques que se emplearon para afrontar problemas similares al de esta tesis. Asimismo, a continuación se presenta un cuadro resumen (véase Anexo [B.1](#)) de lo que se presenta en esta sección.

#### 2.1.1. Real-time deep reinforcement learning based vehicle navigation (pr'koh)

**pr'koh** realizaron un artículo de investigación el cual fue publicado en la revista «Applied Soft Computing» en el año 2020. Este fue titulado **pr'koh** la cual traducida al español significa «Navegación de vehículos basada en aprendizaje de refuerzo profundo en tiempo real».

##### 2.1.1.1. Planteamiento del Problema y objetivo

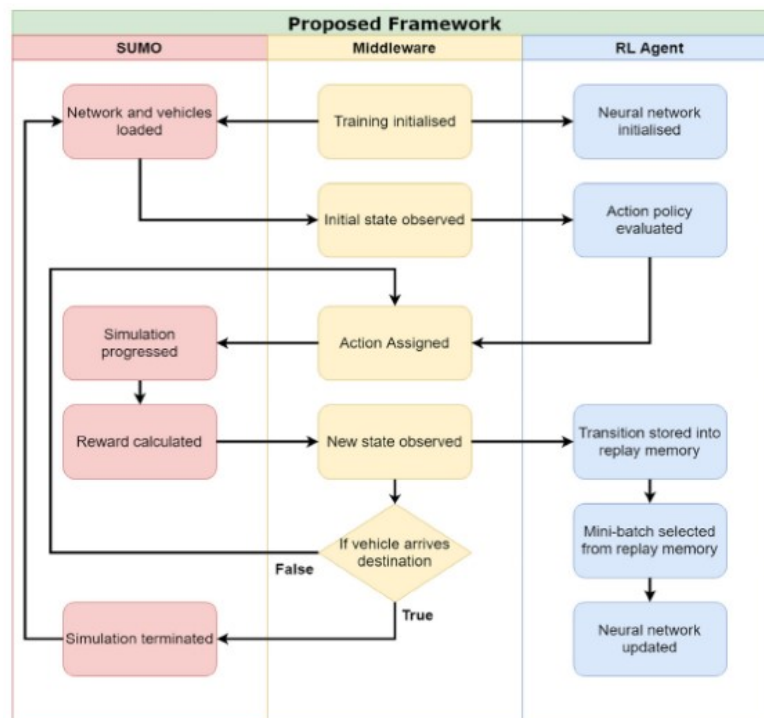
Los autores del siguiente artículo mencionan la gravedad del problema de la congestión vehicular en las ciudades contemporáneas, teniendo como consecuencia el alto consumo innecesario de energía, la contaminación y el tiempo invertido en viaje. Además, menciona la dificultad que tienen los algoritmos existentes para la optimización de las carreteras con el fin de afrontar dicho problema, siendo esta la complejidad que tienen los vehículos al interactuar en un entorno dinámico en tiempo real. Por lo que el objetivo de los autores es implementar un modelo de aprendizaje por refuerzo profundo (DRL) con el fin de construir un sistema de

navegación en tiempo real a través de una secuencia de decisiones.

### 2.1.1.2. Metodología empleada por los autores

Este artículo propone un framework el cual primero construye una simulación en SUMO, que contiene una interfaz de Control de Tráfico (TraCI) y que utiliza mapas urbanos reales en OpenStreetMap mediante el método NETCONVERT. Luego en la misma simulación de SUMO, se puede extraer y pre procesar los datos a través del método TraCI que recupera el número de vehículos, el tiempo de viaje esperado, y la posición actual y el destino.

La segunda parte del framework propuesto es el middleware, que conecta el entorno de SUMO con las redes neuronales DRL bajo un método mejorado de Deep Q-Learning Network (DQN) para mantener y actualizar las políticas de navegación y proporciona comandos para la navegación. El modelo propuesto contiene un esquema de exploración de dos etapas que buscan mejorar la convergencia de la red así como su velocidad. La primera etapa se utiliza la política convencional y la segunda etapa es un método basado en la distancia para reemplazar la selección de una ruta. La estructura de la red neuronal profunda contiene 5 capas conectadas, la primera capa de entrada contiene 150 neuronas, las dos capas ocultas tienen 100 neuronas bajo la función de activación RELU, una estructura de duelo y su capa de salida.



**Figura 2.1:** Framework propuesto



### 2.1.1.3. Resultados obtenidos

Para demostrar la eficiencia del modelo propuesto, se compara con diferentes modelos por ciudad, los resultados de una ciudad demuestran que el modelo reduce como máximo el tiempo de viaje en 5.3 %, 5.1 % y 16.4 % dependiendo de la demanda del tráfico. Otro indicador que se utiliza para medir su eficacia es la prueba de Wilcoxon, demostrando que el modelo propuesto es superior a los demás modelos.

The objective performance comparisons under a various of traffic conditions.

Methods	Map 1			Map 2			Map 3		
	20 vehicles	30 vehicles	50 vehicles	30 vehicles	50 vehicles	80 vehicles	50 vehicles	80 vehicles	120 vehicles
GDUE	81.43	81.78	92.66	109.48	118.23	131.26	162.94	180.04	206.82
Dijkstra	( $\pm 3.31$ )	( $\pm 3.61$ )	( $\pm 14.47$ )	( $\pm 5.74$ )	( $\pm 11.59$ )	( $\pm 19.03$ )	( $\pm 12.75$ )	( $\pm 16.72$ )	( $\pm 25.37$ )
GDUE A*	82.48	82.96	95.54	110.78	119.14	131.77	163.86	181.70	208.51
	( $\pm 4.34$ )	( $\pm 3.87$ )	( $\pm 13.82$ )	( $\pm 8.62$ )	( $\pm 11.76$ )	( $\pm 19.48$ )	( $\pm 13.06$ )	( $\pm 17.44$ )	( $\pm 28.26$ )
Dynamic	79.26	79.70	90.84	105.74	110.96	126.42	148.98	161.34	194.32
Dijkstra	( $\pm 4.94$ )	( $\pm 2.98$ )	( $\pm 11.99$ )	( $\pm 4.68$ )	( $\pm 8.82$ )	( $\pm 9.46$ )	( $\pm 11.50$ )	( $\pm 11.07$ )	( $\pm 9.80$ )
Dynamic A*	81.25	80.26	91.18	107.88	111.36	127.76	149.16	162.12	195.96
	( $\pm 6.40$ )	( $\pm 4.01$ )	( $\pm 13.36$ )	( $\pm 5.90$ )	( $\pm 8.62$ )	( $\pm 9.82$ )	( $\pm 11.65$ )	( $\pm 9.91$ )	( $\pm 8.92$ )
Ant Colony	79.56	79.84	90.98	106.90	113.36	127.98	151.08	176.80	205.12
	( $\pm 4.79$ )	( $\pm 4.72$ )	( $\pm 12.64$ )	( $\pm 4.63$ )	( $\pm 11.82$ )	( $\pm 12.44$ )	( $\pm 9.07$ )	( $\pm 18.00$ )	( $\pm 30.67$ )
RL Agent	78.10	78.72	84.06	103.44	107.66	117.00	143.64	147.82	159.02
	( $\pm 2.66$ )	( $\pm 4.19$ )	( $\pm 5.29$ )	( $\pm 2.48$ )	( $\pm 4.43$ )	( $\pm 6.85$ )	( $\pm 4.84$ )	( $\pm 5.60$ )	( $\pm 8.73$ )

**Figura 2.2:** Comparación de rendimiento de los modelos

The significance analysis by using Wilcoxon test.

RL agent vs previous works	Map 1			Map 2			Map 3		
	20 vehicles	30 vehicles	50 vehicles	30 vehicles	50 vehicles	80 vehicles	50 vehicles	80 vehicles	120 vehicles
GDUE-Dijkstra	$7.11e^{-3}$	$4.55e^{-7}$	$3.34e^{-8}$	$2.48e^{-7}$	$1.56e^{-8}$	$3.27e^{-10}$	$1.26e^{-8}$	$3.73e^{-11}$	$7.58e^{-11}$
GDUE-A*	$3.33e^{-7}$	$5.12e^{-8}$	$6.27e^{-11}$	$4.65e^{-7}$	$6.77e^{-8}$	$3.55e^{-10}$	$2.31e^{-8}$	$4.12e^{-11}$	$7.79e^{-11}$
Dynamic-Dijkstra	$1.90e^{-3}$	$2.86e^{-2}$	$4.70e^{-7}$	$2.52e^{-4}$	$9.71e^{-5}$	$8.20e^{-7}$	$2.15e^{-5}$	$3.25e^{-8}$	$5.98e^{-10}$
Dynamic-A*	$7.21e^{-7}$	$3.90e^{-7}$	$2.63e^{-8}$	$1.23e^{-6}$	$5.74e^{-6}$	$1.02e^{-8}$	$2.21e^{-5}$	$7.06e^{-10}$	$6.86e^{-10}$
Ant-Colony	$1.87e^{-3}$	$2.30e^{-3}$	$3.24e^{-7}$	$2.71e^{-5}$	$4.53e^{-6}$	$3.24e^{-8}$	$2.01e^{-7}$	$8.24e^{-10}$	$6.99e^{-11}$

**Figura 2.3:** Análisis de significancia de los modelos con la prueba de Wilcoxon

## 2.1.2. SafeRoute: Learning to navigate streets safely in an urban environment (pr'saferoute)

**pr'saferoute** realizaron un artículo de investigación el cual fue publicado en la revista «ACM Transactions on Intelligent Systems and Technology» en el año 2020. Este fue titulado **pr'saferoute** la cual traducida al español significa «SafeRoute: aprender a navegar por las calles de forma segura en un entorno urbano».

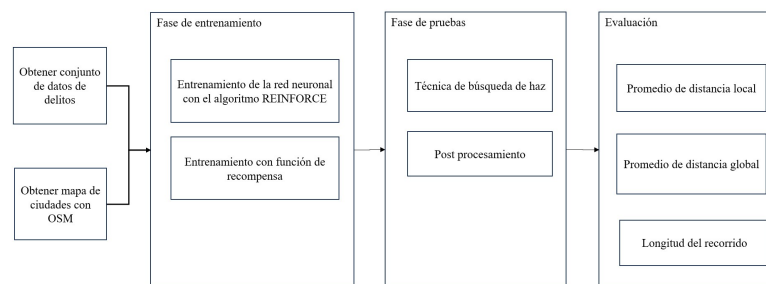
### 2.1.2.1. Planteamiento del Problema y objetivo

Ante los resultados de estudios realizados en la Universidad de Cornell y Hollaback en 2014, se muestra que el 85 % de mujeres tomaron rutas diferentes para ir a su destino con el fin de evitar ser víctimas de acoso o agresión. También se menciona la necesidad de las

personas, como los turistas, de contar con una aplicación que pueda detectar rutas seguras. Con esto en mente, se propone SafeRoute, que es una solución de estos problemas utilizando algoritmos de aprendizaje por refuerzo profundo, que tiene ventaja bajo los algoritmos clásicos como Dijkstra, ya que es una opción a problemas que requieren tomar decisiones incrementales por cada intersección que se presenta entre el origen y el destino, además que tiene la ventaja de adaptarse a nuevos datos sin requerir a ser entrenado nuevamente.

### 2.1.2.2. Metodología empleada por los autores

Se emplea información pública de delitos de la ciudad de Nueva York, San Francisco y Boston, y se filtra los tipos de delitos relacionados con acoso callejero y asalto, los cuales son tiroteo, asalto y robo. Asimismo, para la obtención cartográfica de dichas ciudades, se utilizó OpenStreetMap (OSM).



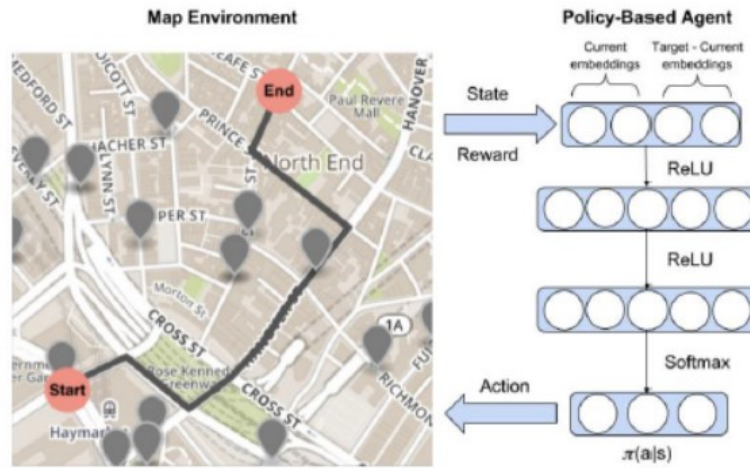
**Figura 2.4:** Metodología propuesta por los autores

De acuerdo a la metodología propuesta por los autores, la arquitectura se divide en dos partes, la primera es donde interviene la red neuronal de aprendizaje por refuerzo, y el segundo es la red de políticas definidas para que la red neuronal utiliza al momento de tomar decisiones. Este entorno se presenta como proceso de decisión de Markov con tuplas que contienen los estados continuos del mapa, las acciones disponibles para el algoritmo y la probabilidad de pasar de un estado a otro.

Se establece una función de recompensa para optimizar las múltiples preferencias del modelo, en este caso es evitar las áreas de crimen al crear una ruta. La definición de esta función se establece de la siguiente manera:

Donde  $n$  es el número de bordes de cada camino,  $m$  es el número de delitos de cada nodo,  $x$  es la lista de puntos de cada camino,  $c$  es la lista de delitos de cada radio,  $p$  es el recorrido y  $k$  es un hiper parámetro.

En la fase de entrenamiento, consta de dos partes, la primera es el pre entrenamiento de



**Figura 2.5:** Estructura de la red neuronal del modelo propuesto

$$r_{CRIME} = \begin{cases} \frac{\sum_{i=1}^n \sum_{j=1}^m distance(x_i, c_{ij})}{\frac{\sum_{i=1}^n number(c_i)}{length(p)}}, & \text{if } c \neq \emptyset \\ \kappa, & \end{cases}$$

la red neuronal para detectar rutas más cortas, en este proceso se utilizan técnicas de aprendizaje por refuerzo, específicamente el algoritmo Monte Carlo Policy Gradient (REINFORCE) para que actualice los parámetros de la red neuronal. La segunda es el entrenamiento con recompensas, donde se entrena nuevamente a la red neuronal considerando la seguridad en cada ruta, premiándolo por evitar áreas con alto índice de crimen, el cuál es la función de recompensa, y de esta manera actualizando la política de la red neuronal.

En la fase de pruebas, se utiliza una técnica de búsqueda de haz para encontrar varias rutas potenciales, luego se realiza un post procesamiento que evite bucles externos que se hayan creado durante la navegación, con el fin de que la ruta generada sea coherente y eficiente. Para la evaluación del modelo propuesto con respecto a otros modelos, se aplicaron métricas como el promedio de la distancia local, el promedio de la distancia global y la longitud del recorrido.

### 2.1.2.3. Resultados obtenidos

El modelo propuesto se comparó con el algoritmo de Dijkstra y SafePath, el cual crea produce caminos no dominados tomando en cuanto la seguridad. Se obtuvieron los mejores resultados con una tasa de aprendizaje de 0.0005 y realizando 5 saltos para un primer entrenamiento y luego 10 saltos.

Se observa que las distancias entre un ruta a la zona de crimen, el que presenta un valor

City	Model	5-hops			10-hops		
		Local	Global	Length	Local	Global	Length
Boston	Dijkstra	0.0554	0.8620	0.2023	0.0484	0.8479	0.4200
	SafePath(Median)	0.0566	0.8648	0.2044	0.0498	0.8639	0.4372
	SafePath(Safest)	0.0568	0.8651	0.2050	0.04817	0.8768	0.4619
	SafeRoute	<b>0.0630</b>	0.8627	0.2361	<b>0.0571</b>	0.8510	0.4903
San Francisco	Dijkstra	0.0882	0.9268	0.4704	0.0964	0.8966	1.0121
	SafePath(Median)	0.0934	0.9423	0.4821	0.1046	0.9344	1.1063
	SafePath(Safest)	0.0937	0.9489	0.5001	<b>0.1084</b>	0.9650	1.2402
	SafeRoute	<b>0.0990</b>	0.9880	0.5468	0.1036	0.9611	1.1659
New York	Dijkstra	0.1344	1.1978	0.4222	0.1016	1.1842	0.7907
	SafePath(Median)	0.1341	1.2125	0.4515	0.0987	1.2267	0.8748
	SafePath(Safest)	0.1344	1.2242	0.4935	0.0978	1.2537	0.9731
	SafeRoute	<b>0.1454</b>	1.2004	0.47267	<b>0.1179</b>	1.1992	0.8915

**Figura 2.6:** Comparación de rendimiento de los modelos

más bajo es Boston, debido a su densidad de criminalidad, mientras que las rutas generadas en Nueva York son las que más se alejan. Sin embargo, se demuestra que tanto en la época 5 como la época 10, el modelo propuesto SafeRoute supera al modelo Dijkstra y los modelos de SafePath, a excepción de la ciudad de San Francisco, el cuál el número que más se aleja de los riesgos es el modelo más seguro de SafePath en la época 10.

### 2.1.3. Predicting motor vehicle theft in Santiago de Chile using graph-convolutional LSTM (pr'esquível)

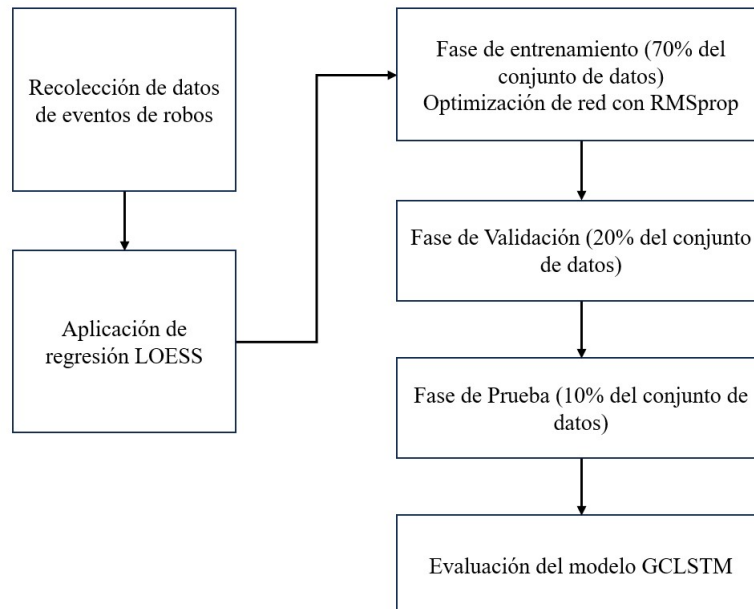
**pr'esquível** realizaron un artículo de investigación el cual fue presentado en la «39th International Conference of the Chilean Computer Science Society» en el año 2020. Este fue titulado **pr'esquível** la cual traducida al español significa «Predicción del robo de vehículos de motor en Santiago de Chile utilizando LSTM gráfico convolucional».

#### 2.1.3.1. Planteamiento del Problema y objetivo

Ante el problema del incremento de autos robados en Chile en el año 2017, se menciona que este tipo de delitos, que, si bien es de bajo riesgo, puede desencadenar otros delitos más graves como el contrabando de armas o drogas, o crímenes internacionales. Es por esto que a través de un modelo de predicción de robo de vehículos puede contribuir a la mitigación y mayor seguridad por parte de la fuerza policial en las zonas con mayor probabilidad de robos. En este trabajo de investigación busca crear una nueva metodología para la predicción de robos de vehículos en la ciudad de Santiago de Chile desarrollando una red neuronal convolucional LSTM basada en grafos (GCLSTM) utilizando la técnica de preprocesamiento de datos, la cuál es la regresión LOESS.

### 2.1.3.2. Metodología empleada por los autores

Los autores proponen la siguiente metodología:

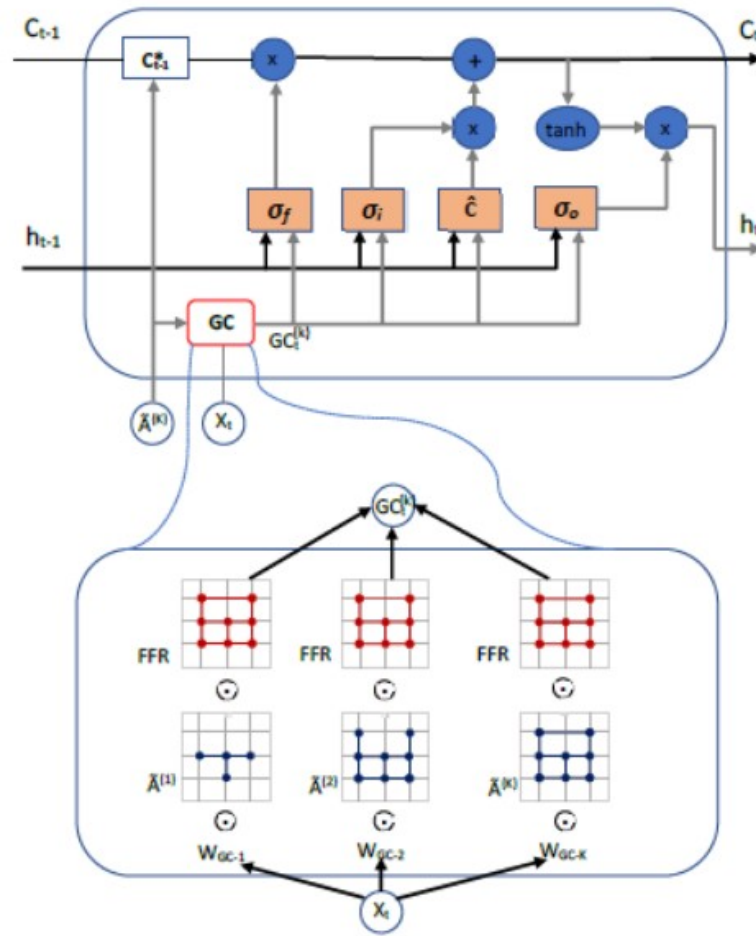


**Figura 2.7:** Metodología propuesta por los autores

El primer paso de esta metodología empieza con la recolección de los datos, que en este caso son los eventos de robos de vehículos en las comunas de la Región de Chile del año 2015 al 2019, que contiene un total de 129.127 registros. Luego se procedió a realizar la suma de los eventos de robo por día de cada comuna para la creación de una serie de tiempo y posterior a esto, aplicar la regresión LOESS, el cual es una técnica no paramétrica que utiliza los mínimos cuadrados ponderados para asignar un mayor peso a los más cercanos y menos a los que están más lejos. La aplicación de dicha técnica permite la reducción de ruido de la serie de tiempo. De este paso, se obtuvo una matriz suavizada de número de eventos de robos, el cuál sirve como input para la red GCLSTM.

Dicha red suma las matrices de adyacencia e identidad para realizar la convolución con parámetros  $K$ , luego se le asigna a cada vecino un peso por iteración y estas van a las compuertas LSTM para la extracción de la última celda para el desarrollo de una capa de salida lineal con la función tanh. Para la optimización de la red GCLSTM, se aplicó el algoritmo de RMSprop para entrenar el modelo hasta que haya un sobreajuste en los datos.

Para la fase de entrenamiento, se utilizó el 70 % del conjunto de datos, 20 % de validación y 10 % para la fase de prueba.



**Figura 2.8:** Estructura de la red convolucional LSTM basada en grafos

### 2.1.3.3. Resultados obtenidos

Como parte de los resultados, en la fase de entrenamiento, se entrenó en el modelo tradicional LSTM y el modelo GCLSTM, dando como resultado que a partir de la iteración 31, la curva MSE del modelo GCLSTM desciende más que el tradicional. Mientras que, en la fase de pruebas, se demuestra, con la métrica R2, que el modelo propuesto obtuvo 0.86, y el modelo LSTM tuvo 0.70, teniendo una mayor similitud la red GCLSTM entre lo esperado con lo observado. Por otra parte, el MSE del modelo propuesto fue 0.014 y el modelo LSTM es de 0.031; demostrando que el modelo puede obtener patrones temporales y espaciales para la predicción de crimen en distintas zonas de una ciudad.

## 2.1.4. Algoritmo para calcular la ruta más segura y óptima (pr'areiza)

pr'areiza realizaron un artículo de investigación el cual fue publicado en el año 2022. Este fue titulado pr'areiza.

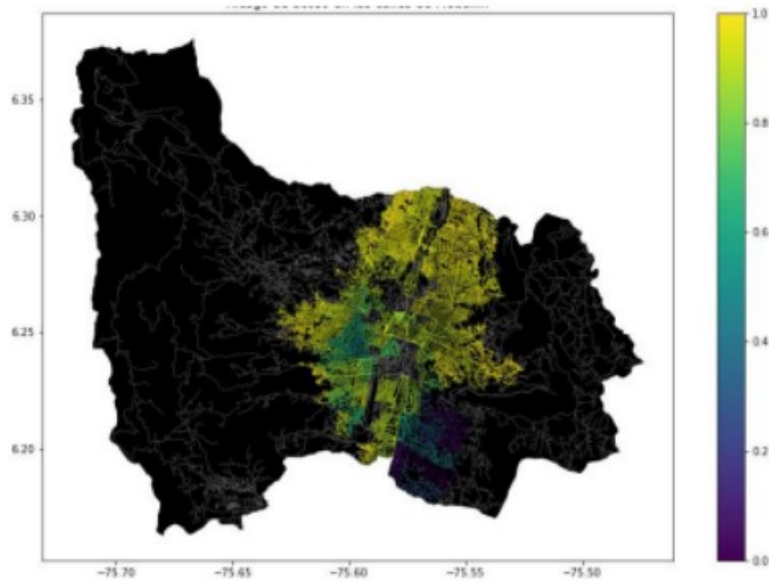
### 2.1.4.1. Planteamiento del Problema y objetivo

La creciente cifra de casos de acoso callejero en las calles de Medellín, así como casos de asaltos y violación, ha tenido como consecuencia que el miedo de las mujeres aumente y por ende, se sientan inseguras por cuál ruta tomar. A pesar de los distintos métodos de prevención, se menciona que no siempre son efectivos para evitar ser víctima de acoso en espacios públicos. Además, aplicaciones de generación de rutas para el transporte, como Google Maps o Waze, no cuentan con la capacidad de detectar rutas seguras, ya que es propio sistema de estas sólo calculan sus rutas en base a la distancia más corta. Este trabajo de investigación busca solucionar este problema implementando el algoritmo de Dijkstra, para calcular tres caminos diferentes tomando en cuenta la distancia y el índice de riesgo.

### 2.1.4.2. Metodología empleada por los autores

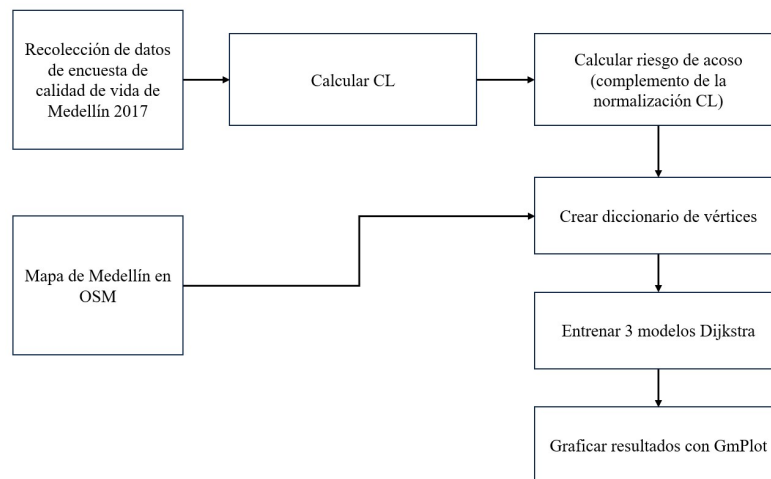
Para la obtención del mapa de Medellín, se empleó Open Street Maps (OSM) en Python, el cual contiene en cada segmento su longitud en metros, en qué sentidos va, y sus representaciones binarias proporcionadas por la propia librería OSM. Para el cálculo del riesgo, se empleó un conjunto de datos de la encuesta de calidad de vida de Medellín del año 2017; luego se procedió a realizar una combinación lineal (CL) mediante el análisis de componentes principales, el cual es la máxima varianza de la proporción de hogares que se sienten inseguros entre la proporción de hogares que tienen ingresos inferiores al salario mínimo; después se normaliza la combinación lineal y la proporción de esta es considera como el riesgo de acoso.

Se crea un diccionario que almacena el peso de cada vértice mediante el producto entre la longitud y riesgo de acoso de cada relación entre origen y destino. Se implementa el algoritmo de Dijkstra que requiere de tres parámetros los cuáles son los grafos creados, y los vértices de origen y destino. Este algoritmo calcula cada vecino del vértice actual a través de sus pesos que están almacenados en el diccionario para calcular el camino con menos peso. Además del cálculo tradicional que emplea, el cuál es  $r*d$ , donde  $r$  es el riesgo de acoso, y  $d$  es la distancia en metros, se calculó dos caminos más. El primero es la suma entre la distancia y el riesgo, mientras que el segundo eleva la distancia al riesgo de acoso. Para graficar estos tres resultados, se empleó la librería GmPlot, une el camino del origen y destino mediante las coordenadas



**Figura 2.9:** Riesgo de acoso sexual mediante combinación lineal

ubicadas en Google maps, siendo más fácil de visualizar para el usuario.



**Figura 2.10:** Metodología propuesta por los autores

#### 2.1.4.3. Resultados obtenidos

Se presenta los resultados que son obtenidos por el algoritmo, estableciendo el origen en la Universidad EAFIT hasta la Universidad Nacional, el cual se puede visualizar en la siguiente tabla:

Se observa que en términos de evitar el mayor riesgo, la mejor opción es el tercer camino (d<sup>3</sup>); sin embargo, la distancia que puede recorrer una persona es amplia a comparación



Camino	Distancia	Riesgo	Tiempo de ejecución
r*d	9061.75m	0.58	1.01033s
r+d	8574m	0.69	1.01030s
d̂	16642m	0.35	1.00640s

**Tabla 2.1:** Tabla de resultados.

de los otros caminos; asimismo, la opción más balanceada en cuanto al índice de riesgo y la distancia recorrida es el primer camino (r\*d). También se demuestra que el tiempo de ejecución es bastante bajo, por lo que abre la oportunidad de ser empleado en una aplicación móvil para el fácil uso de los usuario que lo emplearán en su vida cotidiana.

### 2.1.5. Route-the safe: A robust model for safest route prediction using crime and accidental data (pr'Soni)

**pr'Soni** realizaron un artículo de investigación el cual fue publicado la entrevista en el año 2019. Este fue titulado **pr'Soni** la cual traducida al español significa «Ruta segura: un modelo sólido para la predicción de rutas más seguras utilizando datos sobre delitos y accidentes».

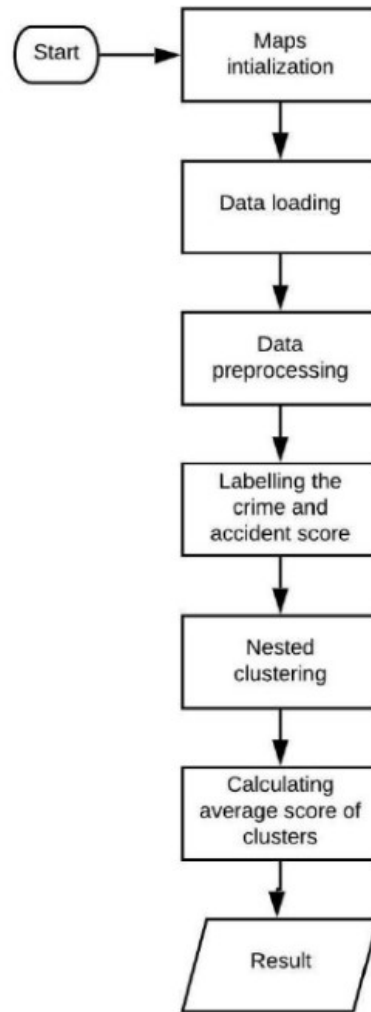
#### 2.1.5.1. Planteamiento del Problema y objetivo

La falta de conocimiento de la zona de una ciudad para un turista, así como invertir su tiempo en investigar sobre esta, puede crear una dependencia a sus conductores o pueden verse expuestas a ciertos peligros, como el robo de sus pertenencias. Asimismo, también se menciona la falta de capacidad que tienen las aplicaciones de rutas al momento de elegir un camino seguro. Con el objetivo de ser una herramienta de ayuda a la hora de elegir la ruta más segura para viajar, este proyecto utiliza algoritmos de aprendizaje automático para la predicción de rutas seguras mediante el cálculo de puntuación de riesgo de cada ruta de las ciudad de Nueva York utilizando datos sobre delitos y accidentes.

#### 2.1.5.2. Metodología empleada por los autores

Este trabajo de investigación presenta la siguiente metodología utilizada para la implementación de su modelo:

Como primer paso de la metodología propuesta, se inicializa el mapa con el API de Google Maps, para luego proceder a configurar su función gmaps. Luego se cargan los datos,



**Figura 2.11:** Metodología Propuesta

los cuáles contienen datos de delitos y accidentes desde el sitio web de datos abiertos de Nueva York, que brindan información del tipo de delito o accidente, así como su latitud y longitud. El siguiente paso se aplica al preprocesamiento de los datos para limpiar los datos atípicos o faltantes. Una vez realizado el preprocesamiento de los datos, se procede a etiquetar la puntuación de accidentes de un punto (AS) y la puntuación de delitos de un punto (CS). Para la puntuación de delitos se asigna ponderaciones que van desde 1 a 15 dependiendo del delito y tipo de castigo que se le impone al sospechoso. Mientras que la puntuación de accidentes se calcula de la siguiente manera:

$$AS = PK*2 + CK*2 + MK*2 + PI + CI + MI$$

Donde:

PK: Recuento de peatones fallecidos

CK: Recuento de ciclistas fallecidos

MK: Recuento de automovilistas fallecidos

PI: Recuento de peatones heridos

CI: Recuento de ciclistas heridos

MI: Recuento de automovilistas heridos

El siguiente paso es la formación de los grupos utilizando K Means, que es un algoritmo de agrupación de aprendizaje automático no supervisado, basado en la latitud y longitud, así como los conjuntos de datos se concatenan para agruparlos y determinar las regiones de riesgo. Específicamente, primero se agrupa en función a la longitud y latitud donde ocurrieron los hechos, para dividir el mapa de Nueva York en regiones más pequeñas. Asimismo, para generar las regiones de riesgo, se agrupa bajo los grupos ya formados en un distrito, como por ejemplo, el distrito de Manhattan.

Después de hacer las agrupaciones de los grupos formados de un distrito, se realiza la puntuación promedio de cada agrupación, los cuáles se calculan de la siguiente manera: Donde

$$C = \sum_{i=1}^N C_S / N$$

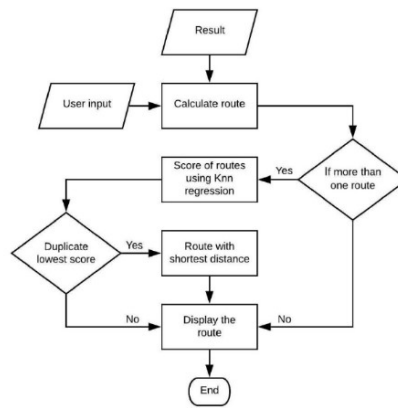
$$A = \sum_{i=1}^N A_S / N$$

N es el número de puntos de un grupo. Después se procede a calcular la puntuación del riesgo (RS) que toma como entrada el origen y destino de la ruta a calcular gracias al API de google maps llamado Direction Service, que proporciona una serie de pasos al usuario para seguir la ruta. Luego se utiliza el modelo KNN Regresor para hallar el vecino más cercano según el grado de riesgo de cada punto. La puntuación de riesgo (RS) se calcula de la siguiente manera:

$$RS = \sum_{i=1}^W C_S + \sum_{i=1}^W A_S$$

Donde W es el número de puntos de rutas que hay entre el origen y el destino.

Como último punto, el modelo debe mostrar la ruta más segura bajo las siguientes dos condiciones: el primero es que la ruta debe presentar la puntuación de riesgo más baja, y la segunda es que si hay más de una ruta segura, se debe elegir la distancia más corta.



**Figura 2.12:** Flujo de condición para detectar rutas seguras

### 2.1.5.3. Resultados obtenidos

Se establece un punto de origen A y un punto de destino B en el distrito de Manhattan, dando como resultado que bajo el modelo de KNN Regressor, el R cuadrado de la puntuación de accidentes sea de 0.91 y la puntuación de criminalidad sea de 0.974; concluyendo que el modelo se ajusta mejor a sus datos.



Figura 6. Ruta sugerida por google maps

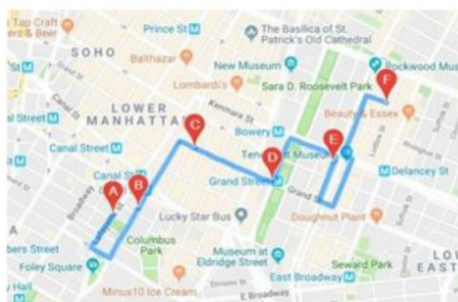


Figura 7. Ruta sugerida a través del modelo propuesto

**Figura 2.13:** Resultados obtenidos

## 2.1.6. Real-time safest route identification: Examining the trade-off between safest and fastest routes (pr`Ghoul)

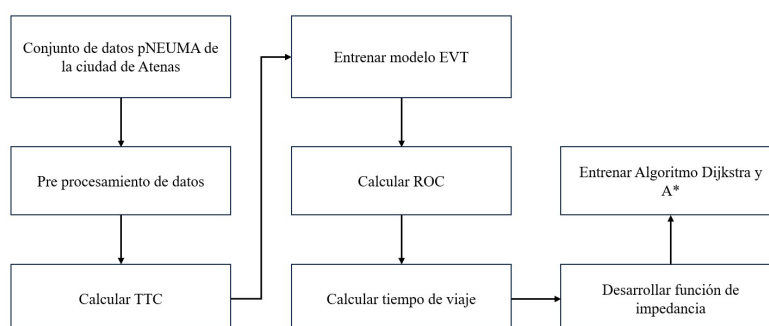
**pr`Ghoul** realizaron un artículo de investigación el cual fue publicado en la revista «Analytic methods in accident research» en el año 2023. Este fue titulado **pr`Ghoul** la cual traducida al español significa «Identificación de la ruta más segura en tiempo real: examen de la relación entre las rutas más seguras y las más rápidas»..

### 2.1.6.1. Planteamiento del Problema y objetivo

Este trabajo de investigación propone un nuevo enfoque de predicción de rutas seguras debido a la creciente congestión vehicular y accidentes, esto con el fin de mejorar el flujo vehicular de un determinado sector utilizando datos en tiempo real para que se sincronice con el modelo y proporcionar avisos para que el usuario evite regiones congestionadas o que están expuestos a peligros en las carreteras. Tiene como objetivo desarrollar un algoritmo de enrutamiento en tiempo real bajo una métrica propuesta que combina el riesgo de accidentes con el tiempo de viaje a lo largo de diferentes puntos de una ruta.

### 2.1.6.2. Metodología empleada por los autores

Se utiliza un conjunto de datos de código abierto llamado pNEUMA, de la ciudad de Atenas, Grecia. Este conjunto de datos utiliza drones para recopilar los datos sobre el tráfico en horas pico, asimismo, bajo esta recopilación, se extrae los datos de la posición, velocidad, aceleración y tipo de vehículo. Luego se pre procesaron los datos mediante la comparación por pares de trayectorias. Después se calculó el tiempo de coalición (TTC) de cada ubicación para aplicarlo al modelo propuesto.



**Figura 2.14:** Metodología propuesta por los autores

El modelo propuesto es un modelo de teoría de valores extremos (EVT), específicamente el modelo bayesiano jerárquico para estimar el riesgo de accidente de cada ubicación en tiempo real y se incorporan covariables para optimizar el rendimiento del modelo propuesto. Después de desarrollar el modelo bayesiano, se calcula el riesgo de colisión (ROC), que estima el número total de accidentes que se observan en una ubicación, también se calcula una métrica de tiempo de viaje para desarrollar una función de impedancia para realizar comparaciones entre los distintos nodos de una red de carreteras bajo un algoritmo de enrutamiento. Para el cálculo de la función de impedancia (S) es de la siguiente manera:

$$S_t = \sum_{i=1}^n ROC_{i,t} \times \bar{TT}_{i,t}$$

Posterior al cálculo de la función de impedancia, se utiliza en el algoritmo Dijkstra o el algoritmo A\*. Después se desarrolla un algoritmo que elige la mejor ruta, el cuál identifica distintas rutas según la preferencia del usuario, tomando en cuenta tanto la seguridad como la movilidad.

### 2.1.6.3. Resultados obtenidos

Bajo la metodología propuesta, se encontró que el 23 % de las rutas más rápidas eran similares a las rutas más seguras. Además, la ruta más segura comparte en un 54 % los mismos enlaces que la ruta más rápida, existiendo un equilibrio entre seguridad y movilidad en distintos escenarios. Este trabajo de investigación aporta con un modelo de recomendación de rutas que presenta una buena eficacia en cuanto a cambios dinámicos en tiempo real en diferentes tipos de condiciones; sin embargo, para el cálculo de riesgo, depende de una percepción de seguridad más que la seguridad real, por lo que se recomienda que se establezca bajo su propia preferencia y prioridades, ya que es difícil cuantificar el riesgo real de un sector.

### 2.1.7. A reinforcement learning-based routing algorithm for large street networks. (pr'diya)

**pr'diya** realizaron un artículo de investigación el cual fue publicado en la « International Journal of Geographical Information Science» en el año 2024. Este fue titulado **pr'diya** la cual traducida al español significa «Un algoritmo de enrutamiento basado en aprendizaje de refuerzo para grandes redes de calles».

### 2.1.7.1. Planteamiento del Problema y objetivo

Ante la problemática de contar con sistemas de evacuación y rutas de emergencia para minimizar los daños ocasionados durante los desastres naturales, los algoritmos tradicionales de enrutamiento cuentan con problemas de escalabilidad cuando estas se enfrentan a evacuaciones a gran escala que involucran a diferentes rutas, asimismo, la complejidad computacional de estos algoritmos puede aumentar a medida que crece y se vuelve más complejo un escenario de evacuación, llevando a tener largos tiempos de cálculos, además que trabajan con pocos parámetros que capturan la complejidad y dinamismo de las emergencias por inundaciones. Ante estas limitaciones y con las contribuciones recientes de algoritmos de aprendizaje por refuerzo profundo (DRL), se desarrolló un algoritmo de aprendizaje por refuerzo para mejorar la eficiencia de enrutamiento en una gran red de carreteras en tiempo real, para lograr dicho objetivo, se entrenó dicho algoritmo en una supercomputadora FASTER, que maneja grandes redes de carreteras de una base de datos espacial.

### 2.1.7.2. Metodología empleada por los autores

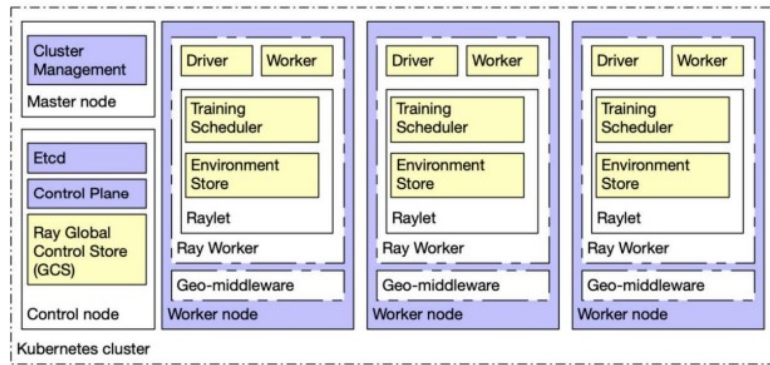
Se recopilan los datos de redes de transporte de la ciudad de Nueva York y Houston, que contienen la longitud y latitud de cada punto de interés, asimismo para los datos de transporte se exporta información cartográfica del área central de Houston y la ciudad de Nueva York.

Para el algoritmo de aprendizaje por refuerzo, el sistema de enrutamiento puede modelarse como una tupla de proceso de decisión de Markov (MDP). Cada tupla contiene el conjunto de estados de cada vértice, conjunto de acciones que considera la red neuronal durante el entrenamiento, la función de recompensa para optimizar la preferencia del usuario, donde se incorpora el factor de seguridad. Además, se establecen políticas con el método de gradientes de políticas, el cual se conecta con el algoritmo de ascenso de gradiente estocástico. Para disminuir la varianza del estimado de gradientes del algoritmo, se utiliza la Optimización de Política Proximal (PPO).

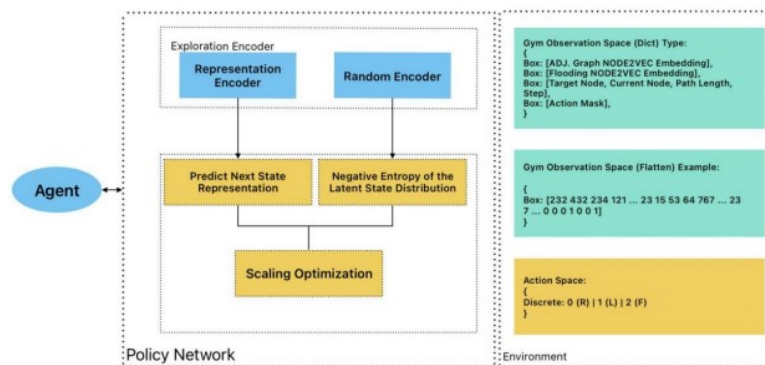
Para hacer frente a los entornos complejos y dinámicos, se requiere de una red neuronal que memorice los nodos y elija una acción válida a través de Q-learning, este algoritmo evita no seleccionar rutas inválidas con la red de políticas.

### 2.1.7.3. Resultados obtenidos

Se comparó con el método de Dijkstra y el algoritmo clásico de aprendizaje por refuerzo mediante puntuación por métricas de seguridad y recompensa de cada ciudad, como también su



**Figura 2.15:** Puntuaciones de evaluación



**Figura 2.16:** Puntuaciones de evaluación

tiempo. Los resultados indican que el mejor algoritmo es el modelo propuesto, ya que tienen un mayor puntaje de seguridad, como también un mayor indicador de recompensa. Sin embargo, tiene una mayor duración al ejecutar cada enrutamiento o detectar rutas seguras. También se destaca el hecho de que el modelo propuesto no atraviesa los puntos inundados, superando en términos de seguridad a los algoritmos tradicionales.

Method	Environment	Metric.	Score
Dijkstra	New York City (Large)	Safety	1450.23
ReinforceRouting	New York City (Large)	Safety	1663.01
Dijkstra	Houston (Large)	Safety	1798.06
ReinforceRouting	Houston (Large)	Safety	1963.90
Vanilla-RL	Houston (Small)	Reward	-0.54
RE-RL	Houston (Small)	Reward	0.71
Vanilla-RL	Houston (Small)	Episode Length	20.5
RE-RL	Houston (Small)	Episode Length	87.7
RE-RL	Houston (Large)	Reward	0.12
ReinforceRouting	Houston (Large)	Reward	0.62
Vanilla-RL	Houston (Large)	Episode Length	20.5
ReinforceRouting	Houston (Large)	Episode Length	87.7

**Figura 2.17:** Puntuaciones de evaluación



### 2.1.8. An Enhanced Motif Graph Clustering-Based Deep Learning Approach for Traffic Forecasting (pr'zhang)

**pr'zhang** realizaron un artículo de investigación el cual fue publicado en el « GLOBE-COM 2020 » en el año 2020. Este fue titulado **pr'zhang** la cual traducida al español significa «Un enfoque de aprendizaje profundo basado en agrupamiento de gráficos de motivos mejorado para la previsión del tráfico».

#### 2.1.8.1. Planteamiento del Problema y objetivo

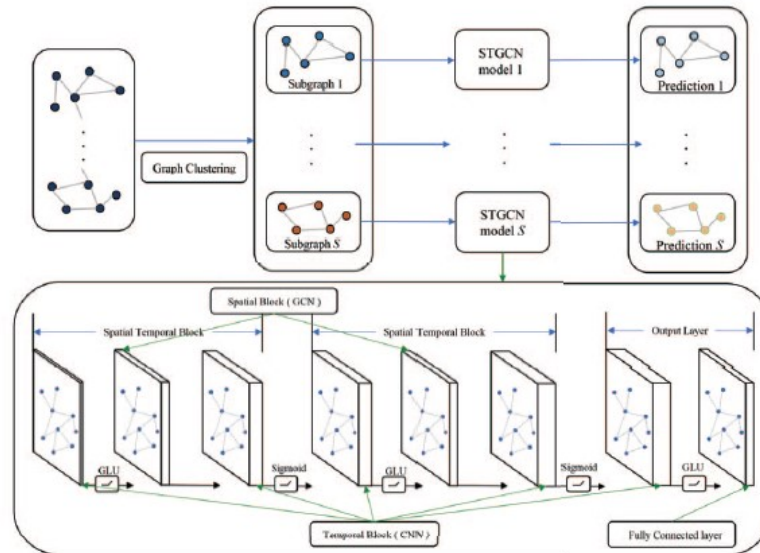
Por la gran complejidad que existe para el logro de una predicción precisa, como es el gran tamaño y condiciones de redes de tráfico, los métodos de Aprendizaje Profundo basados en grafos ha sido una alternativa de solución en tendencia. Además, mientras más complejo sea las redes de tráfico, el rendimiento de los modelos de Graph Convolution Network (GCN) son cada vez más bajos. Para afrontar estos problemas, los autores del presente artículo proponen un enfoque basado en agrupamiento de grafos para aplicar un modelo de Spatio-Temporal Graph Convolutional Networks (STGCN) para la predicción de la velocidad de tráfico en lugares urbanos a gran escala.

#### 2.1.8.2. Metodología empleada por los autores

La metodología propuesta empieza utilizando dos conjuntos de datos, el primer conjunto de datos pertenece a Nav-BJ y contiene datos de 1159 estaciones de sensores desplegadas en las diferentes carreteras de la ciudad de Beijing del mes de marzo de 2019, mientras que el segundo conjunto de datos pertenece a Caltrans Performace Measurement System (PeMS) que contiene 228 estaciones de sensores en el distrito 7 de California de los meses de mayo y junio de 2012. Luego se realiza un preprocesamiento de los datos, se realiza una normalización Z-Score y se aplica una interpolación lineal para la recuperación de datos faltantes.

Se utilizan otros tres enfoques del modelo propuesto EMGC-STGCN (agrupación de grafos), el primero es el modelo STGCN sin agrupación de grafos, el segundo es Random-STGCN, el cual utiliza el método de agrupación de grafos aleatorios, mientras que el último enfoque es Metis-STGCN que utiliza el método ingénuo de Metis. Para el entrenamiento del modelo propuesto, se necesita como parámetro las matrices de adyacencia de los dos conjuntos de datos y que representa el 60 % de estas; además, se utiliza el optimizador Adam durante 50 épocas. Luego se realiza el proceso de prueba del 20 % del conjunto de datos y por último el proceso de validación que representa el 20 %.

Se compara el modelo propuesto de los 3 enfoques distintos con los métodos de Promedio Histórico (HA) y ARIMA, así como el modelo GRU. Para esto, se hace uso de las métricas de Error Absoluto Medio (MAE), Error Porcentual Absoluto Medio (MAPE) y Raíz del Error Cuadrático Medio (RMSE).



**Figura 2.18:** Metodología EMGC-STGCN propuesta por los autores

### 2.1.8.3. Resultados obtenidos

Mediante la comparación de los distintos modelos y métodos, se evidencia que el modelo propuesto es mejor que sus distintos enfoques, como también los métodos HA, ARIMA y el modelo GRU.

Model	15 min			30 min			45 min		
	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)
HA	7.54	4.19	12.28	7.54	4.19	12.28	7.54	4.19	12.28
ARIMA	8.03	4.39	10.88	12.21	5.91	12.66	19.07	8.85	18.56
GRU	7.34	4.02	13.56	7.61	4.23	14.32	7.56	4.32	14.67
STGCN	4.53	3.10	11.13	4.84	3.28	11.84	5.02	3.40	12.30
Random-STGCN	4.23	2.89	9.90	4.48	3.05	10.60	4.68	3.19	11.36
Metis-STGCN	4.20	2.84	9.72	4.41	3.02	10.38	4.56	3.14	10.97
EMGC-STGCN	<b>4.11</b>	<b>2.81</b>	<b>9.52</b>	<b>4.33</b>	<b>2.99</b>	<b>10.26</b>	<b>4.46</b>	<b>3.08</b>	<b>10.68</b>

**Figura 2.19:** Resultados de modelos propuestos con el conjunto de datos Nav-BJ

Model	15 min			30 min			45 min		
	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)
HA	7.20	4.01	10.61	7.20	4.01	10.61	7.20	4.01	10.61
ARIMA	9.00	5.55	12.92	9.13	5.86	13.94	9.38	6.27	15.20
GRU	4.15	2.35	7.25	5.36	3.04	9.12	6.19	3.52	10.14
STGCN	3.55	2.02	4.82	4.91	2.85	7.10	5.45	3.14	7.67
Random-STGCN	3.52	2.06	4.74	4.73	2.67	6.51	5.37	3.04	7.52
Metis-STGCN	3.49	1.99	4.67	4.59	2.58	6.19	5.34	3.03	7.45
EMGC-STGCN	3.47	1.98	4.56	4.62	2.58	6.20	5.40	3.03	7.42

**Figura 2.20:** Resultados de modelos propuestos con el conjunto de datos PeMS

## 2.2. Bases Teóricas

### 2.2.1. Machine Learning

Es un subcampo de ejecutar dificultosos procesos aprendiendo de datos, en lugar de seguir reglas preprogramadas ([Royal Society Working Group, 2017](#)).

es importante mencionar que existen también cinco tipos de problemas de aprendizaje que se pueden enfrentar: regresión, clasificación, simulación, optimización y clusterización ([Gollapudi, 2016](#)). Por otro lado, el aprendizaje automático también posee una división por subcampos que se puede observar en la Figura 14.

### 2.2.2. Natural Language Processing (NLP)

Naturalmano ([Goyal et al., 2018](#)). Otra definición para este término implica que es un campo especializado de la informática que es

De acuerdo con [2018](#), e

## 2.3. Marco Conceptual

**Inteligencia Artificial:** Según Lasse Rouhiainen, la Inteligencia artificial es un conjunto de algoritmos que tienen la capacidad de tomar decisiones cómo lo haría un ser humano, aplicable a la mayoría de situaciones, como la detección y clasificación de un objeto, aumentar el desempeño de algunos procesos o la detección de riesgos (Rouhiainen, 2018).

**Aprendizaje Automático:** Es una rama importante de la Inteligencia Artificial que es aplicada a cualquier disciplina, utilizando conocimientos de informática, estadísticas e ingeniería para la predicción o evaluación de los objetos después de haber sido entrenado con datos históricos. (Tao, Q. e.at., 2021).

**Aprendizaje Profundo:** Subconjunto del Aprendizaje automático, con la capacidad de reconocer patrones con un gran volumen de datos, estructurados y no estructurados, entrenados por redes neuronales artificiales que intentan emular el cerebro humano y que, a comparación del aprendizaje automático, no requiere de la intervención humana para el modelo (IBM, s.f.).

**Aprendizaje por Refuerzo Profundo (DRL):** Técnica que combina el aprendizaje profundo con aprendizaje por refuerzo. Se utiliza una red neuronal de aprendizaje profundo que aprende por cada acción en tiempo real se presente, un ejemplo de esto es en un juego de Ajedrez (gamco, s.f).

**Graph Neural Networks:** Técnica que utiliza la potencia de predicción del aprendizaje profundo para predecir objetos(nodos) que están relacionados (bordes). Tiene la capacidad de predecir a nivel de nodo, edges o gráficos (Merritt, 2022).

**Red convolucional-LSTM basada en grafos (GCLSTM):** Es una red neuronal que integra LSTM para la predicción dinámica de enlaces de red. **Regresión LOESS:** Técnica no paramétrica para eliminar los efectos de ruido de los datos para suavizar una curva.

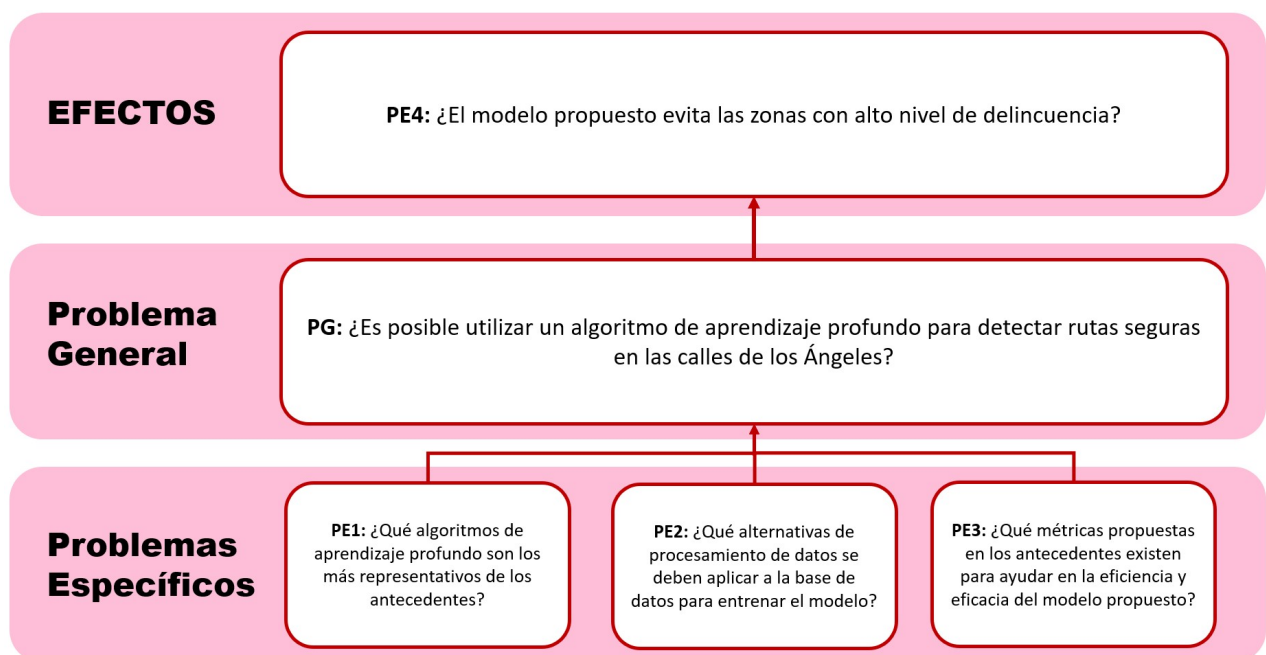
**Optimización de Política Proximal (PPO):** Es un método que impide al modelo realizar modificaciones adicionales que causen inestabilidad, además de ser escalable y emplea un enfoque basado en regiones de confianza y eficiencia del muestreo (Greyrat, 2022).

## **Anexos**

# Anexos A

## Anexo I: Matriz de Consistencia

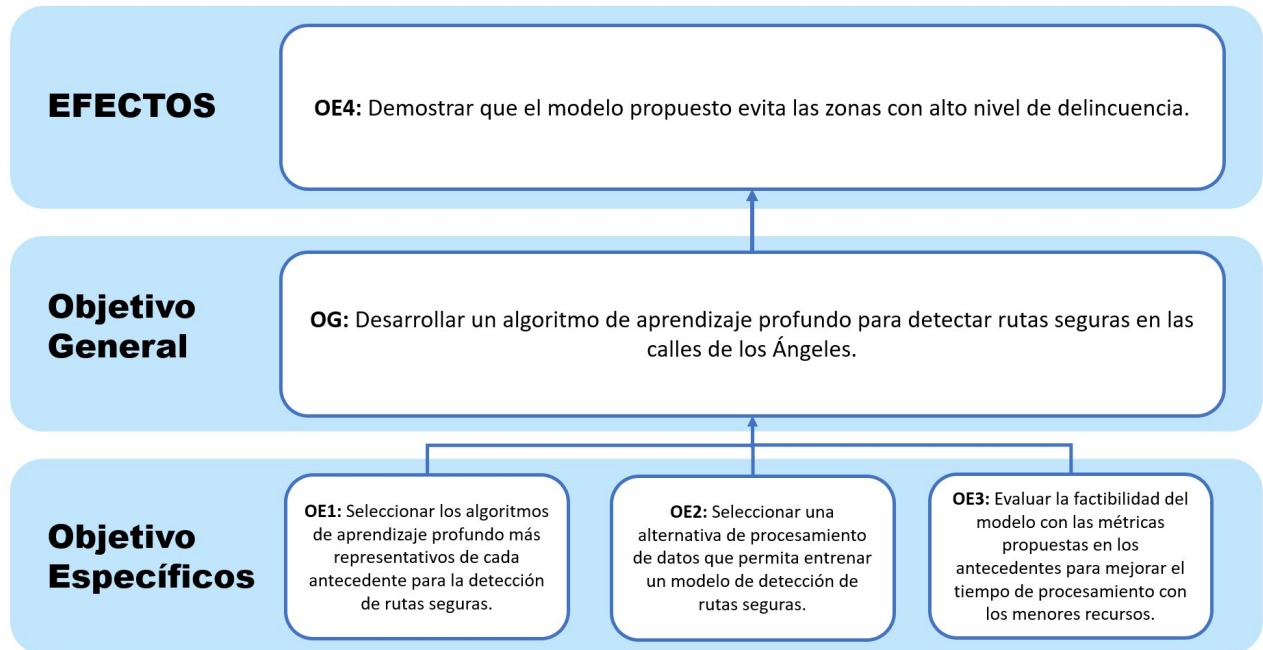
### A.1. Árbol de Problemas



PROBLEMAS	OBJETIVOS	HIPÓTESIS
Problema General	Objetivo General	Hipótesis General
¿Es posible utilizar un algoritmo de aprendizaje profundo para detectar rutas seguras en las calles de los Ángeles?	Desarrollar un algoritmo de aprendizaje profundo para detectar rutas seguras en las calles de los Ángeles.	Mediante el desarrollo de un algoritmo de aprendizaje profundo se logra detectar rutas seguras en las calles de los Ángeles.
Problemas Específicos	Objetivos Específicos	Hipótesis Específicas
¿Qué algoritmo de aprendizaje profundo es el más representativos de los antecedentes?	Determinar qué algoritmo de aprendizaje profundo es el más representativo de cada antecedente para la detección de rutas seguras.	La selección del algoritmo de aprendizaje profundo más representativos de cada antecedente mejora significativamente la precisión y eficiencia al detectar rutas seguras.
¿Qué alternativas de procesamiento de datos se deben aplicar a la base de datos para entrenar el modelo?	Seleccionar una alternativa de procesamiento de datos que permita entrenar un modelo de detección de rutas seguras.	La identificación de una alternativa de procesamiento de datos influye positivamente para poder entrenar un modelo de detección de rutas seguras.
¿Qué métricas propuestas en los antecedentes existen para ayudar en la eficiencia y eficacia del modelo propuesto?	Evaluar la factibilidad del modelo con las métricas propuestas en los antecedentes para mejorar el tiempo de procesamiento con los menores recursos.	La utilización de métricas propuestas en los antecedentes es factible para optimizar el tiempo de procesamiento del modelo.
¿El modelo propuesto evita las zonas con alto nivel de delincuencia?	Demostrar que el modelo propuesto evita las zonas con alto nivel de delincuencia.	Se demuestra que el modelo propuesto evita las zonas con alto nivel de delincuencia.

**Tabla A.1:** Matriz de consistencia. Fuente: Elaboración propia

## A.2. Árbol de Objetivos





## **Anexos B**

### **Anexo II: Resumen de Papers investigados**

Tipo	N°	Título	Autor	Año	País	Fuente
Problema	1	Predicting motor vehicle theft in Santiago de Chile using graph-convolutional LSTM.	Esquivel, N., Nicolis, O., & Márquez, B. P	2020	Chile	39th International Conference of the Chilean Computer Science Society (SCCC)
	2	SafeRoute: Learning to navigate streets safely in an urban environment	Levy, S., Xiong, W., Belding, E., & Wang, W. Y.	2020	USA	ACM Transactions on Intelligent Systems and Technology (TIST)
Propuesta	3	Route-the safe: A robust model for safest route prediction using crime and accidental data.	Soni, S., Shankar, V. G., & Chaurasia, S.	2019	India	Int. J. Adv. Sci. Technol
	4	Real-time deep reinforcement learning based vehicle navigation	Koh, S., Zhou, B., et al.	2020	Reino Unido	Applied Soft Computing
Técnica	5	Algoritmo para calcular la ruta más segura y óptima	Cano, S., & Tabares, S. A. A.	2022	Colombia	OSF Repository
	6	A reinforcement learning-based routing algorithm for large street networks. International Journal of Geographical Information Science	Li, D., Zhang, Z., Alizadeh, B., et al.	2024	USA	International Journal of Geographical Information Science
	7	Real-time safest route identification: Examining the trade-off between safest and fastest routes	Ghoul, T., Sayed, T., & Fu, C.	2023	Canada	Analytic methods in accident research
	8	Análisis y desarrollo de modelos de predicción de links en redes mediante técnicas de Deep Learning.	Muñoz, A.	2022	España	Archivo Digital UPM

**Tabla B.1:** Cuadro Resumen de Papers investigados. Fuente: Elaboración propia

# BIBLIOGRAFÍA

- Akita, R., Yoshihara, A., Matsubara, T., & Uehara, K. Deep learning for stock prediction using numerical and textual information. En: *En 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*. IEEE. 2016, 1-6.
- Cortez, C. T., Saydam, S., Coulton, J., & Sammut, C. (2018). Alternative techniques for forecasting mineral commodity prices. *International Journal of Mining Science and Technology*, 28(2), 309-322.
- Dehghani, H., & Bogdanovic, D. (2018). Copper price estimation using bat algorithm. *Resources Policy*, 55, 55-61.
- Gartner. (2019). Gartner IT Glossary. <https://www.gartner.com/it-glossary/>
- Gollapudi, S. (2016). *Practical machine learning*. Packt Publishing Ltd.
- Goyal, P., Pandey, S., & Jain, K. (2018). Deep learning for natural language processing. *Deep Learning for Natural Language Processing: Creating Neural Networks with Python [Berkeley, CA]: Apress*, 138-143.
- IBM. (2019). IBM AI glossary. <https://www.ibm.com/cloud/garage/architectures/cognitiveArchitecture/glossary>
- Jingrui, Z., Qinglin, W., Yu, L., & Yuan, L. A method of optimizing LDA result purity based on semantic similarity. En: *En 2017 32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE. 2017, 361-365.
- Kulkarni, A., & Shivananda, A. (2019). Exploring and Processing Text Data. En *Natural Language Processing Recipes* (pp. 37-65). Springer.
- Lagos, G. (2017). ¿Cómo lo han hecho los especialistas?: aciertos y desaciertos al proyectar el precio del cobre. <https://gyn.claseejecutiva.uc.cl/como-lo-han-hecho-los-especialistas-aciertos-y-desaciertos-al-proyectar-el-precio-del-cobre/#>
- Martínez, R., & Cohen, E. (2018). Manual formulación, evaluación y monitoreo de proyectos sociales. <https://dds.cepal.org/redesoc/publicacion?id=242>
- Ministerio de Energía y Minas (ESTAMIN). (2019). *Perú: país líder de los metales del futuro* (Boletín Estadístico Minero).

- Rao, D., Deng, F., Jiang, Z., & Zhao, G. Qualitative Stock Market Predicting with Common Knowledge Based Nature Language Processing: A Unified View and Procedure. En: En *2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics*. 2. IEEE. 2015, 381-384.
- Real Academia Española. (2014). Diccionario de la lengua española (Twenty-third). <https://dle.rae.es/?w=diccionario>
- Royal Society Working Group. (2017). *Machine learning: the power and promise of computers that learn by example* (inf. téc.). Technical report.
- Study Group International Copper. (2018). The World Copper Factbook 2018. [www.icsg.org](http://www.icsg.org)
- TensorFlow. (2019). Vector Representations of Words. <https://www.tensorflow.org/tutorials/representation/word2vec>
- Tipirisetty, A. (2018). *Stock Price Prediction using Deep Learning* [Tesis de maestría, San José State University] [Master's Projects]. <https://doi.org/https://doi.org/10.31979/etd.bzmm-36m7>
- Trastour, S., Genin, M., & Morlot, A. (2016). Prediction of the crude oil price thanks to natural language processing applied to newspapers. <http://cs229.stanford.edu/proj2016/report/>
- U.S. Geological Survey. (2019). Mineral commodity summaries 2019: U.S. Geological Survey. <https://doi.org/https://doi.org/10.3133/70202434>
- Yun, H., Sim, G., & Seok, J. Stock Prices Prediction using the Title of Newspaper Articles with Korean Natural Language Processing. En: En *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*. IEEE. 2019, 019-021.
- Zhao, R., & Mao, K. (2017). Fuzzy bag-of-words model for document representation. *IEEE Transactions on Fuzzy Systems*, 26(2), 794-804.