



Predicción Geoespacial de Crímenes en Bogotá: Un Enfoque Basado en Machine Learning para Mejorar la Seguridad Ciudadana.

Sergio David Salazar Isairias

Universidad de los Andes
Facultad de Ingeniería, Departamento de Ingeniería Eléctrica y Electrónica
Bogotá D.C, Colombia, 2024

Predicción Geoespacial de Crímenes en Bogotá: Un Enfoque Basado en Machine Learning para Mejorar la Seguridad Ciudadana.

Sergio David Salazar Isairias

Universidad de Los Andes

Presentado en cumplimiento parcial de los requisitos para el grado de:
Ingeniero Electrónico

Asesor:

PhD. Fernando Enrique Lozano Martínez

Sustentación de proyecto de grado:
Enero, 2024

Universidad de los Andes

Facultad de Ingeniería, Departamento de Ingeniería Eléctrica y Electrónica
Bogotá D.C, Colombia, 2024

Agradecimientos

Agradezco a mi familia, especialmente a mi hermana Nicolle, por siempre creer en mí a pesar de mis errores y debilidades. También agradezco a la universidad, por brindarme espacios que no sabía que necesitaba.

Abstract

Currently, artificial intelligence models play a crucial role in solving everyday problems, ranging from summarizing paragraphs to discovering new songs or creating presentations. Modern artificial intelligence models not only address an increasing array of problems but also tackle challenges that were previously deemed insurmountable. Among these challenges are time series forecasting problems. The aim of the current project is to build an artificial intelligence model that predicts crime occurrences in the city of Bogotá, thereby contributing to the efforts to combat insecurity.

Keywords: Attention, Deep Learning, Graph, Time Serie

Tabla de Contenido

Lista de Figuras	viii
Lista de Tablas	x
1 Introducción	1
1.1 Motivación	1
1.2 Objetivos	1
1.2.1 General	1
1.2.2 Específicos	2
1.3 Alcance	2
2 Marco Teórico	4
2.1 Bogotá	4
2.2 Crime Forecasting	5
2.3 Serie de Tiempo	5
2.3.1 Medidas de Error	5
2.4 Deep Learning	6
2.4.1 Redes Neuronales	6
2.4.2 Perceptrón	7
2.5 Grafo	7
2.5.1 Red de Información	8
2.6 Graph Neural Networks	8
2.6.1 Message Passing	8
2.7 Attention	10
2.8 Graph Attention Networks	10

3 Metodología	12
3.1 Planteamiento del problema	12
3.2 Recolección de datos	12
3.2.1 Reducción de datos	13
3.3 Tratamiento de los datos	13
3.4 Visualización	13
3.5 Selección del modelo	14
3.6 Preparación del modelo	15
3.6.1 Definición de nodos	15
3.6.2 Transformación de datos	16
3.7 Modelo	17
3.7.1 Implementación	17
4 Resultados	20
4.1 Entrenamiento	20
4.2 Desempeño	21
5 Conclusiones	24
References	25
6 Appendix A Feature Engineering	27
6.1 Datos a Disposición	27
6.2 Descripción	28

List of Figures

2.1	Conjunto de las Unidades de Planificación Zonal, distribuidas en las 19 localidades urbanas de Bogotá [5].	4
2.2	Biological neurons to Artificial neurons [2]	6
2.3	Comparación en la estructura de una red con respecto a una imagen y una oración [3].	8
2.4	Visión general de cómo un único nodo agrega mensajes de su vecindario local. El modelo agrega mensajes de los vecinos del grafo local de A (es decir, B, C y D) y, a su vez, los mensajes procedentes de estos vecinos se basan en la información agregada de sus respectivos vecindarios, y así sucesivamente. Esta visualización muestra una versión de dos capas de un modelo de paso de mensajes [4].	9
2.5	Concepto de atención en la lectura de un texto.	10
2.6	Ilustración de la capa de atención y de la actualización de los embedding a través de los coeficientes de atención [18].	11
3.1	Mapa de calor de los homicidios reportados en la zona urbana de Bogotá durante el mes de julio del 2020.	14
3.2	Comparación de las divisiones de los barrios legalizados durante dos instantes de tiempo. La zona es aquella cercana a la Universidad de Los Andes. El mapa de la izquierda contiene las divisiones establecidas por el distrito para el año 2016, el mapa de la derecha contiene las divisiones a corte de agosto del 2023.	15
3.3	Comparación entre Bogotá y el grafo generado. El mapa de la izquierda presenta la división de Bogotá a través de UPZs. El mapa de la derecha contiene el grafo, donde cada punto representa una UPZ del mapa de la izquierda.	16
3.4	Proceso de transformación de la información contenida en los reportes a valores numéricos que se usan como entradas del modelo.	16
3.5	Ilustración del modelo de pronóstico de criminalidad.	17
3.6	Arquitectura del modelo de crime forecasting.	18
4.1	Evolución del MSE a través del tiempo de entrenamiento, medido en épocas.	20
4.2	Las gráficas contienen la cantidad de días entre el 1 de enero de 2016 y el 31 de diciembre de 2019 en que se realizó al menos un reporte, para cada una de las UPZ.	21
4.3	Comparación entre los datos reales y las predicciones de la distribución de los homicidios durante el mes de enero del año 2020.	22

4.4 Comparación entre los datos reales y las predicciones de la distribución de los hurtos a personas durante el mes de enero del año 2020	23
4.5 Comparación entre los datos reales y las predicciones de la distribución de los hurtos de celulares durante el mes de enero del año 2020	23

List of Tables

3.1	Estructura de la red neuronal usada para predecir la proporción de ocurrencias.	18
3.2	Estructura del modelo de predicción del crimen.	19
4.1	MSE de los modelos en los datos de entrenamiento.	21
4.2	Medidas de desempeño en los datos de validación.	21
4.3	MAPE de los modelos en los pronósticos de enero del año 2020.	22
6.1	Reporte de Datos a Disposición. Un archivo es útil si los datos que contiene abarcan un periodo de al menos 5 años o si dicho archivo, junto a otro archivo, cumplen con dicho criterio. En caso de que un archivo contenga información de un año que ya se encuentra en otro archivo, este archivo no es de utilidad.	27
6.2	Reporte de columnas que contienen los archivos a disposición. La tabla contiene el nombre de la columna, así como una descripción de la información que se encuentra en esta. Cabe mencionar que hay variaciones entre los diferentes archivos con respecto al nombre de las columnas, pero la información que almacenan es la misma. Por ejemplo, se tienen registros donde en lugar de <i>Escolaridad</i> se tiene una columna <i>Nivel Educativo</i>	28

1 Introducción

1.1 Motivación

Bogotá desempeña un papel central en el desarrollo nacional, alberga las principales instituciones gubernamentales, además de ser un gran centro empresarial, por lo que tiene alta influencia política y económica. Debido a ello, la ciudad es más atractiva que otras grandes ciudades del país, lo cual explica que Bogotá sea la ciudad más poblada de Colombia y la novena más poblada en Latinoamérica [12].

El alto nivel de ciudadanos que alberga Bogotá (aproximadamente 8 millones según cifras del DANE) enfatiza problemáticas de salud y seguridad pública. En particular, la inseguridad de los bogotanos es uno de los principales retos de la alcaldía. En 2021 según información del DANE la percepción de inseguridad alcanzo su valor más alto en los últimos años, siendo del 88% [15]. En 2022 el índice presentó mejoras, pero la percepción continúo situándose arriba del 80%, siendo ésta del 84.3% [13]. Para el presente año el DANE no ha recolectado información para estimar la percepción de inseguridad, no obstante, las estadísticas publicadas por la Secretaría de Seguridad, Convivencia y Justicia (SCJ) no son alentadoras. En la primera mitad del año (1 de enero a 30 de junio) el hurto a personas aumento en un 28.1% respecto al mismo intervalo para el año 2022 y en el mismo sentido el hurto a residencias aumento en un 31.7%. Es decir, el panorama acerca de la evolución de la inseguridad en la capital es negativa.

Ante dicho panorama, la Administración Distrital, en cabeza de la alcaldesa mayor Claudia López crearon siete comandos especializados para mitigar el multicrimen en Bogotá [14]. No obstante, dicha medida tiene una limitante. La ciudad es de tal tamaño que no hay personal suficiente para ejercer control territorial y para laborar en las unidades especializadas [14], por lo que la efectividad y velocidad con que se perciban los resultados de la propuesta dependen de la cantidad de personal capacitado. Para contrarrestar dicho escenario se deben implementar medidas no tradicionales que optimicen los recursos a disposición; en particular, la tecnología se puede usar como herramienta en pro de la lucha contra el crimen, por ejemplo, utilizar modelos para estimar la probabilidad de ocurrencia de un crimen en una zona dada de la ciudad.

1.2 Objetivos

1.2.1 General

Desarrollar un modelo de aprendizaje automático que permita examinar la distribución espacial y la evolución temporal de los crímenes ocurridos en Bogotá entre 2016 y 2022, el cual pueda ser empleado como herramienta para combatir la criminalidad.

1.2.2 Específicos

- Recopilar y analizar datos históricos de crímenes en Bogotá durante el período 2016-2022, incluyendo ubicaciones y fechas precisas, para alimentar el modelo de aprendizaje automático.
- Desarrollar algoritmos de procesamiento espacial y temporal que permitan identificar patrones y tendencias significativas en la distribución y evolución de los crímenes en diferentes áreas de la ciudad a lo largo del tiempo.
- Evaluar la precisión y eficacia del modelo de aprendizaje automático mediante pruebas y validaciones exhaustivas.

1.3 Alcance

El proyecto se centra en desarrollar un modelo de aprendizaje supervisado, y la principal limitación reside en la calidad y cantidad de los datos disponibles. Para construir el modelo, se han empleado los informes de criminalidad de la policía nacional. Aunque estos informes brindan una visión de la criminalidad en la ciudad, no necesariamente representan la fuente de información más óptima debido a posibles sesgos.

En este proyecto, los alcances se basan en suposiciones clave. Se supone que los datos recopilados sobre crímenes en Bogotá son precisos, completos y representativos de la realidad. También se asume que las características utilizadas en el análisis, como el tipo de crímenes, ubicaciones y fechas, son relevantes para identificar patrones y predecir la criminalidad. Se presupone además que los patrones criminales mantienen cierta estabilidad a lo largo del tiempo y que las relaciones geoespaciales persisten durante el período de análisis.

No obstante, el proyecto enfrenta restricciones importantes. La disponibilidad de datos sobre crímenes en Bogotá puede ser limitada en términos de calidad y cantidad, lo que podría afectar la robustez y precisión de los resultados del análisis. Los recursos computacionales, como la potencia de procesamiento y la memoria, pueden imponer limitaciones en la complejidad del modelo de Machine Learning y en la velocidad de ejecución del análisis.

Por último, el proyecto conlleva factores de riesgo importantes que deben ser tenidos en cuenta. Entre ellos se incluye la posibilidad de sesgos en los datos, como subreportes de crímenes en ciertas áreas, lo que podría influir en los resultados y llevar a conclusiones incorrectas. También existe el riesgo de que el modelo de Machine Learning no capture con precisión la complejidad de los datos criminales, lo que podría resultar en predicciones inexactas. Estas suposiciones y restricciones guiarán tanto la ejecución del proyecto como la interpretación de sus resultados. Como resultado de estas restricciones, al finalizar el proyecto se espera contar con los siguientes entregables:

- Informe de Recopilación y Preparación de Datos: Un informe detallado que documenta el proceso de recopilación, limpieza y preparación de los datos de crímenes en Bogotá, incluyendo la descripción de las fuentes de datos, métodos de limpieza y transformación aplicados.
- Análisis Exploratorio de Datos: Un conjunto de visualizaciones y resúmenes estadísticos que revelan patrones iniciales y tendencias en los datos criminales, lo que proporciona una comprensión preliminar del problema.
- Informe de Patrones y Clusters de Crímenes: Un informe que detalla los patrones y agrupaciones de crímenes identificados en diferentes áreas de Bogotá, respaldado por análisis estadísticos y visualizaciones geoespaciales.

- **Modelo de Machine Learning Entrenado:** El modelo de Machine Learning desarrollado, entrenado y optimizado para predecir patrones de criminalidad en diferentes áreas geográficas y períodos de tiempo.
- **Informe de Evaluación del Modelo:** Un informe que detalla los resultados de la evaluación del modelo, incluyendo métricas de rendimiento, precisiones y limitaciones identificadas.

2 Marco Teórico

A continuación se presentan los principales conceptos que se requieren para entender el desarrollo del presente proyecto:

2.1 Bogotá

Bogotá es la capital de Colombia. Esta ciudad se divide por medio de 20 localidades: 19 urbanas y 1 rural. A su vez, cada localidad esta conformada por unidades de planeación: UPZs para las zonas urbanas y UPRs para las zonas rurales.

Las UPZ son áreas urbanas más pequeñas que las localidades y más grandes que el barrio. La función de las UPZ es servir de unidades territoriales o sectores para planificar el desarrollo urbano en el nivel zonal. Son un instrumento de planificación para poder desarrollar una norma urbanística en el nivel de detalle que requiere Bogotá, debido a las grandes diferencias que existen entre unos sectores y otros [6].

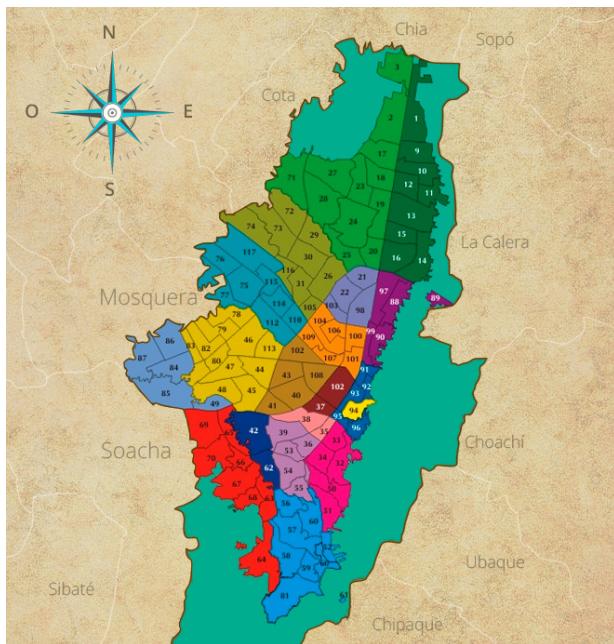


Figure 2.1: Conjunto de las Unidades de Planificación Zonal, distribuidas en las 19 localidades urbanas de Bogotá [5].

En la figura (2.1) se encuentra la distribución de las 117 unidades de planeamiento dispuestas por el plan de ordenamiento territorial.

2.2 Crime Forecasting

El Crime Forecasting es el concepto asociado a la tarea de predecir la ocurrencia de crímenes. Utilizando datos de entrada como el tipo de crimen, el momento del día, el día de la semana y la ubicación geográfica, se busca anticipar la cantidad de crímenes que ocurrirán. Inicialmente, el problema de Crime Forecasting se puede abordar como una tarea de regresión, donde se analiza la cantidad de crímenes como una variable que se puede explicar a través de un conjunto de datos, aunque esto no siempre sea cierto. La ocurrencia de crímenes es un fenómeno que depende de las dinámicas sociales, por lo que comprender el crimen equivale a comprender esas dinámicas, lo cual no es sencillo. El análisis de las dinámicas de la criminalidad requiere varios enfoques, incluyendo una perspectiva espacio-temporal y demográfica. Afortunadamente, los avances recientes en Machine Learning permiten crear modelos que incorporan estas perspectivas, lo que ha beneficiado la tarea de Crime Forecasting.

2.3 Serie de Tiempo

Una serie de tiempo \mathcal{S} es un conjunto de observaciones de una variable cuantitativa, cuyos valores se encuentran indexados en el tiempo. El elemento y_i indica el valor de la variable en el tiempo i .

$$\mathcal{S} = \{y_1, \dots, y_T\} \quad (2.1)$$

Para analizar las series de tiempo se utilizan modelos predictivos, los cuales utilizan observaciones pasadas para pronosticar los siguientes valores de la variable. Dichos modelos son de alto valor para las empresas, pues permiten tomar decisiones para garantizar el buen desempeño. Por ejemplo, en cadenas de insumos se utilizan modelos de series de tiempo para estimar la demanda y así planificar su producción.

2.3.1 Medidas de Error

Para la evaluación del desempeño predictivo se emplean diferentes medidas que cuantifican la distancia entre el pronóstico y su valor real. Tres de las medidas más utilizadas son el Promedio del Error Porcentual Absoluto (MAPE), el error cuadrático medio (MSE) y la raíz del error cuadrático medio (RMSE).

$$MAPE = \frac{1}{T} \left(\sum_{t=1}^T \frac{|y_t^R - y_t^P|}{y_t^R} \right) \quad (2.2)$$

$$MSE = \frac{1}{T} \sum_{t=1}^T (y_t^R - y_t^P)^2 \quad (2.3)$$

$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T (y_t^R - y_t^P)^2} \quad (2.4)$$

donde T es la cantidad de periodos en los que se realizó la predicción. y_t^R es el valor real en el periodo t . y_t^P es el pronóstico de la variable para el periodo t .

2.4 Deep Learning

Deep Learning es un subconjunto del Machine Learning que imita la forma en que los humanos adquieren conocimiento. Los modelos de Deep Learning se utilizan para simular el comportamiento del cerebro humano, permitiendo "aprender" o adquirir conocimiento de grandes cantidades de información [11].

Los modelos corresponden a redes neuronales que identifican características de los conjuntos de datos con mayor facilidad y en un menor tiempo que los seres humanos. Es decir, el Deep Learning destaca por su capacidad de procesar grandes cantidades de datos, esto sin sacrificar la precisión en las tareas que se desempeñe, por lo que es altamente usado en áreas como visión por computador o en procesamiento de lenguaje natural [17].

2.4.1 Redes Neuronales

Una red neuronal es un modelo constituido por un conjunto de capas, las cuales a su vez están formadas por nodos, generando así la estructura de la red. La primera capa que compone el modelo es la capa de entrada, encargada de recibir la información que se va a procesar. A continuación, pueden existir capas intermedias, conocidas como capas ocultas, las cuales se encargan de procesar e identificar características relevantes de la información. Por último, existe una capa de salida que genera una respuesta a la información inicial; esta respuesta se utiliza para evaluar el rendimiento del modelo. Por ejemplo, en un problema de clasificación de imágenes, la capa de entrada recibe la información de los píxeles de una imagen. Luego, las capas ocultas extraen las características relevantes de la misma y, finalmente, la capa de salida genera la etiqueta de la imagen.

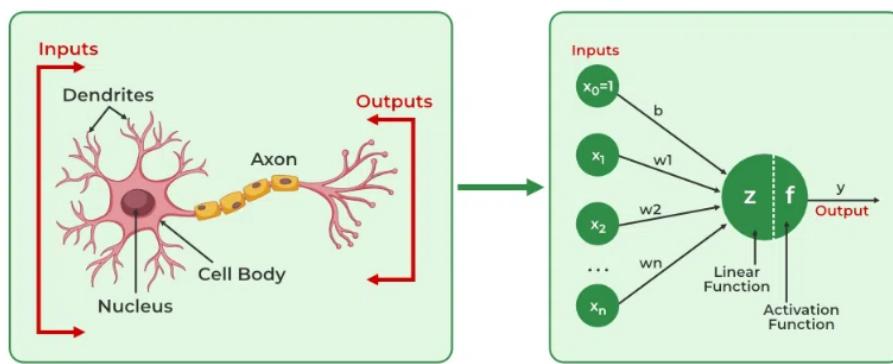


Figure 2.2: Biological neurons to Artificial neurons [2]

Capas

En una red neuronal, las neuronas se estructuran en capas, facilitando el procesamiento simultáneo de datos. Típicamente, estas redes consisten en tres tipos de capas principales. La primera, la capa de entrada, recibe los datos iniciales, con cada neurona correspondiente a una característica de entrada. El número de neuronas aquí coincide con la dimensión de los datos. Las capas ocultas, entre la entrada y la salida, transforman los datos para extraer características clave, con el número de neuronas y capas siendo hiperparámetros críticos para el rendimiento. La capa de salida produce la respuesta final, variando en configuración según la tarea, con una sola neurona para regresión o varias para clasificación.

La organización de neuronas en capas, la capacidad de procesamiento en paralelo y la adaptabilidad de las capas ocultas y de salida son aspectos fundamentales en el diseño y rendimiento de las redes neuronales.

Nodos

Los nodos hacen referencia a las neuronas del cerebro humano. Estos nodos son unidades de procesamiento que reciben información de entrada a través de enlaces de información. A cada enlace de información se le asigna un peso, denotado como w_i , que se utiliza para amplificar o atenuar el efecto de la entrada correspondiente [2]. En la figura 2.3, la entrada z es una función lineal de las entradas x_i , cuyos parámetros son los pesos w_i , es decir, $z = \sum_i w_i x_i$. Es importante mencionar que es común incluir un término constante conocido como "bias" en la forma lineal de z , es decir, $z = \sum_i w_i x_i + b$.

Función de Activación

Las funciones de activación son transformaciones no lineales que se utilizan para determinar si una neurona se activa o no. En otras palabras, una función de activación toma la forma lineal de la entrada de una neurona (z) y, en función de su valor, decide si activarla o restringir su salida a cero. Es importante destacar que las funciones de activación permiten modelar tanto comportamientos lineales como no lineales. Existen diversas funciones de activación, siendo las más comunes la función escalón, la sigmoide, ReLU y Leaky ReLU [1].

Pesos

Los pesos son los parámetros fundamentales de una red neuronal, ya que se utilizan para transformar la información a medida que atraviesa las capas ocultas. Durante la construcción del modelo, los valores de los pesos se inicializan de manera aleatoria. Posteriormente, a través de un proceso conocido como "entrenamiento", se ajustan hasta alcanzar aquellos que aseguran un óptimo rendimiento del modelo [7].

2.4.2 Perceptrón

El Perceptrón es un algoritmo de aprendizaje supervisado utilizado principalmente en tareas de clasificación. En su versión más básica, es un modelo de red neuronal que consta de una sola capa y se destaca por su capacidad para aprender patrones lineales para la identificación de clases [8]. Es importante señalar que existen modelos más complejos de Perceptrón, conocidos como Perceptrón Multicapa, que contienen más de una capa y más de una salida (clase).

2.5 Grafo

Un grafo es una pareja no ordenada con la forma $G = (\mathcal{V}, \mathcal{E})$. \mathcal{V} es un conjunto que almacena elementos denominados vértices o nodos. \mathcal{E} es un conjunto que almacena elementos denominados aristas o enlaces, que tienen la forma (v, w) con $v, w \in \mathcal{V}$. En caso de que (v, w) no represente el mismo objeto que (w, v) se dice que el grafo es dirigido.

2.5.1 Red de Información

Una red de información es un grafo dirigido $G = (\mathcal{V}, \mathcal{E})$ con funciones de enlace $\tau : \mathcal{V} \rightarrow \mathcal{A}$ y $\phi : \mathcal{E} \rightarrow \mathcal{R}$. A cada objeto $v \in \mathcal{V}$ se le asigna la clase $\tau(v) \in \mathcal{A}$. Además, a cada enlace $e \in \mathcal{E}$ se le asigna la relación $\phi(e) \in \mathcal{R}$.

Cuando $|\mathcal{A}| + |\mathcal{R}| > 2$ se dice que la red de información es heterogénea, en caso contrario es homogénea.

2.6 Graph Neural Networks

Los grafos proporcionan una estructura para analizar dinámicas complejas, como el flujo vehicular, las interacciones en una red social y las búsquedas en línea, entre otras. Aunque la estructura de red es útil para modelar sistemas complejos, su análisis puede resultar desafiante. Los grafos pueden tener vértices pertenecientes a diferentes clases, como en una red de información. Además, los nodos pueden estar altamente conectados, lo que complica aún más su análisis. Ante esta situación, en el campo del *Deep Learning* se desarrollaron los *Graph Neural Networks* (GNNs), modelos diseñados para analizar información con estructura de grafo.

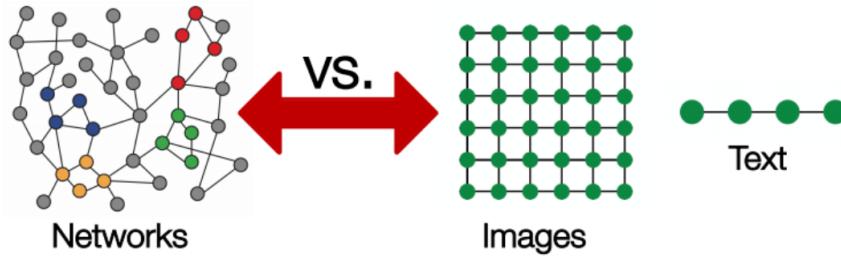


Figure 2.3: Comparación en la estructura de una red con respecto a una imagen y una oración [3].

Los GNNs se fundamentan en el concepto de *Representation Learning*. Cuando se presenta un grafo, el modelo tiene como objetivo representarlo en un espacio de baja dimensión, conocido como *embedding space*, lo cual facilita su análisis. Para describir un grafo, se utilizan dos elementos: una matriz de adyacencia que indica las conexiones entre los nodos y un vector que almacena la información de cada nodo. El embedding se utiliza para representar esa misma información en un vector de menor dimensión. Por ejemplo, en una red social, los nodos representan a los usuarios y la matriz de adyacencia identifica si dos personas son *amigas* o no. Luego, cada nodo tiene un vector que almacena información del usuario como género, edad, escolaridad, etc. Así, un GNN busca generar un *embedding* que, en el *embedding space*, agrupe a usuarios que son amigos y aún más a aquellos que comparten características comunes, como gustos musicales y pasatiempos.

2.6.1 Message Passing

Message Passing es el proceso que utilizan los GNNs para aprender las características de un grafo. La intuición de *Message Passing* es que a través de las iteraciones, el *embedding* de cada nodo capture información de las regiones más lejanas del grafo. En la iteración $k = 1$ el *embedding* contempla la información del nodo, así como aquella de sus vecinos directos. Luego, en la iteración

$k = 2$ el *embedding* también considera información de los vecinos a 2 pasos. Por lo que en k iteraciones el *embedding* contempla información de los nodos vecinos a k pasos de distancia [4].

Dado un grafo $G = (\mathcal{V}, \mathcal{E})$, el *embedding* de $v \in \mathcal{V}$ es h_v . *Message Passing* actualiza los *embedding* a través de k iteraciones. Las iteraciones se describen de la siguiente forma.

$$h_v^{(k)} = \text{UPDATE}(h_u^{(k-1)}, m_{\mathcal{N}(v)}^{(k-1)}) \quad (2.5)$$

donde *UPDATE* es una función que se utiliza para actualizar los valores del *embedding*, por ejemplo la función sigmoide σ .

$$\text{UPDATE}(h_u^{(k-1)}, m_{\mathcal{N}(uv)}^{(k-1)}) = \sigma(W_{self}h_v^{(k-1)} + W_{neigh}m_{\mathcal{N}(v)}^{(k-1)}) \quad (2.6)$$

donde W_{self} y W_{neigh} son parámetros que se deben entrenar a partir de una función de perdida. Dicha función de perdida varía según el objetivo del modelo, por ejemplo, si el GNN está acompañado de un *decoder* una función apropiada es el error de reconstrucción. Por otra parte, el elemento $m_{\mathcal{N}(v)}^{(k-1)}$ es el resultado de agregar los *embedding* de los vecinos inmediatos.

$$m_{\mathcal{N}(v)}^{(k-1)} = \text{AGGREGATION}(h_u^{(k-1)} \forall u \in \mathcal{N}(v)) \quad (2.7)$$

donde la función de *agregación* hace referencia a una función que no sea sensible al orden en que se revisan los nodos, es decir, una función conmutativa. Algunas funciones de agregación son: suma, multiplicación y promedio.

$$\text{AGGREGATION}(h_u^{(k-1)} \forall u \in \mathcal{N}(v)) = \sum_{u \in \mathcal{N}(v)} h_u^{(k-1)} \quad (2.8)$$

donde $\mathcal{N}(v)$ es el conjunto de vecinos inmediatos del nodo $v \in \mathcal{V}$.

En la figura (2.4) se encuentra un ejemplo de message Passing. En la imagen se muestra el proceso para obtener el *embedding* del nodo A. El GNN tiene 2 capas, es decir, el *embedding* contempla la información de los vecinos directos del nodo, así como los vecinos de sus vecinos. Nótese que el gráfico de cálculo de la GNN forma una estructura de árbol desplegando el vecindario alrededor del nodo objetivo.

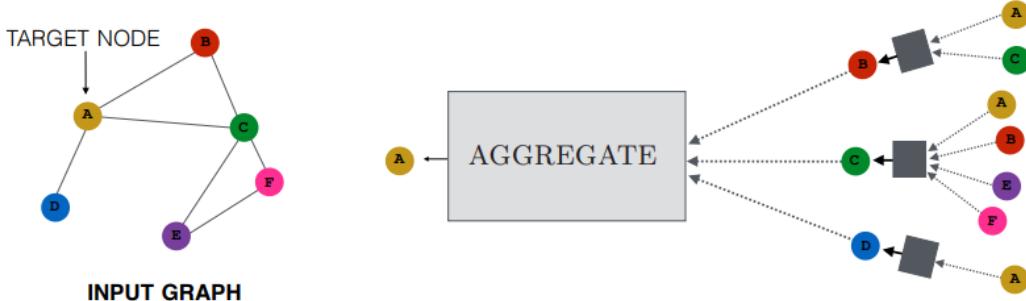


Figure 2.4: Visión general de cómo un único nodo agrega mensajes de su vecindario local. El modelo agrega mensajes de los vecinos del grafo local de A (es decir, B, C y D) y, a su vez, los mensajes procedentes de estos vecinos se basan en la información agregada de sus respectivos vecindarios, y así sucesivamente. Esta visualización muestra una versión de dos capas de un modelo de paso de mensajes [4].

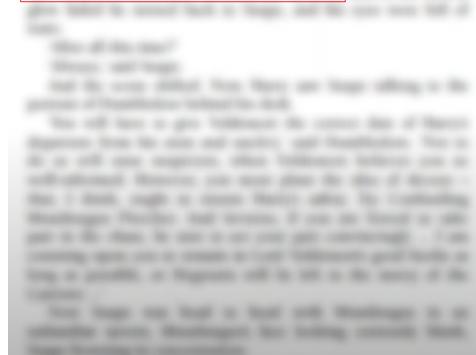
2.7 Attention

Attention es un concepto utilizado en Deep Learning, cuyo origen se encuentra en el sistema de visión humano. Los ojos del ser humano tienen un sistema de atención visual, el cual permite comprender imágenes. Cuando una persona observa una imagen, el ojo enfoca una región de esta, dicho enfoque permite obtener la información más importante. Luego, el ojo enfoca otras regiones, hasta que la información obtenida es suficiente para entender la imagen.

Attention es el concepto detrás de mecanismos que se usan para replicar el funcionamiento del ojo humano. Los mecanismos de atención han permitido mejorar el desempeño de modelos de Machine Learning, así como desarrollar modelos modernos como *GPT-3* [16].

the window. Dumbledore watched her fly away, and as her silvery glow faded he turned back to Snape, and his eyes were full of tears.
 'After all this time?'
 'Always,' said Snape.
 And the scene shifted. Now, Harry saw Snape talking to the portrait of Dumbledore behind his desk.
 'You will have to give Voldemort the correct date of Harry's departure from his aunt and uncle's,' said Dumbledore. 'Not to do so will raise suspicion, when Voldemort believes you so well-informed. However, you must plant the idea of decoys – that, I think, ought to ensure Harry's safety. Try Confounding Mundungus Fletcher. And Severus, if you are forced to take part in the chase, be sure to act your part convincingly ... I am counting upon you to remain in Lord Voldemort's good books as long as possible, or Hogwarts will be left to the mercy of the Carrows ...'
 Now Snape was head to head with Mundungus in an unfamiliar tavern, Mundungus's face looking curiously blank, Snape frowning in concentration.

the window. Dumbledore watched her fly away,



(a) Página de un libro.

(b) Enfoque del ojo humano.

Figure 2.5: Concepto de atención en la lectura de un texto.

2.8 Graph Attention Networks

Graph Attention Network (GAN) es una arquitectura de redes neuronales que utiliza mecanismos de atención para resolver problemas en que la información se estructura por medio de un grafo [18]. En los modelos GNN el aprendizaje, a grandes rasgos, es un proceso de intercambio y actualización de información. El mecanismo de message passing es útil para compartir información entre nodos vecinos, y así obtener un *embedding* que capture tanto la información local, como el contexto de esta. Ahora bien, el problema del proceso de message passing es que la información que se comparte no es necesariamente útil o hay información que es más relevante que otra. Ante esta problemática, al modelo GNN se introduce una capa de atención, la cual se encarga de generar un ranking sobre la relevancia de la información que se comparte.

En los modelos GAN la capa de atención se encarga de generar un coeficiente de atención entre dos nodos, como se observa en (2.9). El coeficiente de atención es una medida entre los *embedding* de dos nodos que están conectados.

$$e_{ij} = a(\vec{W}\vec{h}_i, \vec{W}\vec{h}_j) \quad (2.9)$$

a es una función escalar que recibe el nombre de mecanismo de atención, que en los modelos GAN es el resultado de aplicar la función LeakyReLU al producto escalar entre un vector fijo $\vec{a} \in \mathbb{R}^{2F'}$ y la composición de los *embedding* $[\vec{W}\vec{h}_i || \vec{W}\vec{h}_j]$.

$$a(\vec{W}\vec{h}_i, \vec{W}\vec{h}_j) = \text{LeakyReLU} \left(\vec{a}^T [\vec{W}\vec{h}_i || \vec{W}\vec{h}_j] \right) \quad (2.10)$$

Para comparar los coeficientes de atención entre los diferentes nodos, se realiza una normalización de estos, por medio de la función *softmax*.

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})} \quad (2.11)$$

Luego, los coeficientes de atención se utilizan para ponderar la función de *agregación* del message passing.

$$\vec{h}'_i = \sigma \left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} W \vec{h}_j \right) \quad (2.12)$$

\vec{h}'_i es el nuevo *embedding* del nodo i , σ es la función de *actualización* y W es una matriz $\mathbb{R}^{F' \times F}$ que contiene los pesos de la primera capa oculta. Cabe mencionar que F es la dimensión inicial de los *embeddings* y F' es la dimensión final de estos.

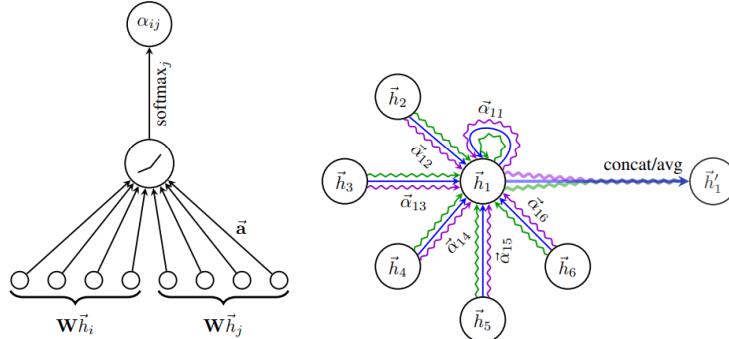


Figure 2.6: Ilustración de la capa de atención y de la actualización de los embedding a través de los coeficientes de atención [18].

En la figura (2.6) se encuentra un ejemplo del funcionamiento de los coeficientes de atención para generar la nueva representación de un nodo. En esta imagen se generaliza el uso de mecanismos de atención, ya que cada color de la flecha indica una función diferente para calcular los coeficientes.

3 Metodología

3.1 Planteamiento del problema

Para el problema de predicción de crimen se utiliza un enfoque numérico, donde se busca predecir la cantidad de veces que ocurre un crimen en un punto particular del espacio-tiempo. La definición formal es la siguiente.

Dado un escenario E con R regiones urbanas, se considera una ventana de tiempo de longitud T y C categorías de crimen. Se define el tensor de crimen $\mathbf{X} \in \mathbb{R}^{R \times T \times C}$, donde el elemento $\mathbf{X}_{r,c}^t$ representa el número de ocurrencias del crimen c en la región r en el elemento t de la ventana de tiempo. Con base en la definición anterior, el problema consiste en aprender una función de predicción que tome el tensor de crimen \mathbf{X} como entrada y prediga $\mathbf{X}^{T+1} \in \mathbb{R}^{R \times C}$.

3.2 Recolección de datos

En primer lugar, se obtuvieron datos sobre los crímenes de alto impacto en la ciudad de Bogotá por medio de una solicitud en el portal queremosdatos.co¹. La información recibida consta de las denuncias de crímenes de alto impacto de las regiones principales de Colombia, entre ellas Bogotá. En los archivos permitidos por la plataforma de queremosdatos.co se tienen a disposición los reportes de al menos el año 2022, ya que algunos delitos cuentan con información de años pasados, lo cual es positivo para el entrenamiento del modelo.

Como respuesta a la solicitud de los reportes de criminalidad se obtuvieron los informes para los siguientes delitos.

- Amenazas.
- Delitos Sexuales.
- Extorsión.
- Homicidios.
- Hurto de Automotores.
- Hurto de Motocicletas.
- Hurto de Bicicletas.
- Hurto a Comercio.
- Hurto a Residencias.
- Hurto a Personas.
- Hurto de Celulares.
- Lesiones Personales.
- Violencia Intrafamiliar.

¹Queremosdatos.co es una plataforma web que permite solicitar información a cualquier institución pública o a cualquier entidad que maneje o administre recursos públicos. Cualquier ciudadano puede enviar solicitudes de acceso a la información a través de esta página [9].

3.2.1 Reducción de datos

Uno de los propósitos del proyecto es emular la correlación entre el tiempo y el acontecimiento de un crimen. Por lo tanto, es relevante contar con reportes de diferentes días del año, así como reportes de diferentes años. Con ello en mente, del conjunto de datos se eliminaron aquellos crímenes con reportes incompletos o inferiores a un año.

Además, dada la definición del problema, la cantidad de crímenes a analizar repercute negativamente en la construcción de la función de predicción, ya que a mayor cantidad de categorías de crimen, mayor será la cantidad de parámetros que se deben entrenar. Por tanto, se omitieron algunos crímenes, de modo que la función solo considere 3 tipos de crimen.

- Homicidios
- Hurto a personas
- Hurto de celulares

3.3 Tratamiento de los datos

Los reportes de crímenes de la Policía Nacional no se pueden usar directamente para el entrenamiento del modelo, ya que estos cuentan con problemas de información faltante e inconsistencias en los reportes pertenecientes a años diferentes. Debido a ello se limpian los datos, con lo cual se tienen más de 900.000 reportes desde el año 2010 hasta el año 2022. Todos los reportes tienen la estructura descrita en la tabla (6.2).

A partir del planteamiento del problema se entiende que los componentes temporales y espaciales son los más relevantes para la construcción del modelo, por lo que información adicional a estos no se tuvo en cuenta. Así pues, el conjunto de reportes a utilizar contiene las siguientes columnas.

- | | |
|---|--|
| <ul style="list-style-type: none"> ■ Fecha ■ Barrio | <ul style="list-style-type: none"> ■ Localidad ■ Tipo de crimen. |
|---|--|

La información a disposición contiene los reportes de un periodo de 13 años. Si bien es cierto que se requieren reportes de diferentes fechas, la cantidad actual es problemática. La formulación del ejercicio establece que cada punto del espacio se describe por medio de una matriz $X^{T \times C}$. Si se analizan los 3 tipos de crimen durante una ventana de 10 años, cada región se describe por medio de una matriz con más de 10000 entradas, lo cual es computacionalmente costoso. En adición a esto, en los años recientes aconteció el confinamiento por COVID-19, lo cual provocó que los reportes de crímenes descendieran drásticamente. El comportamiento de los reportes durante 2020 y años próximos es diferente a aquél que se esperaría en caso de no haber ocurrido la pandemia por COVID-19, por lo que analizar estos años es incorrecto. Para dar solución a este problema, se utilizaron los reportes de 2016 a 2019, los demás no se utilizaron.

3.4 Visualización

Los datos obtenidos son de carácter geo espacial, por lo que es correcto realizar una visualización de estos a través de mapas de calor. Para analizar la distribución de los crímenes a lo largo de la ciudad se generaron estas ilustraciones por medio de *python*.

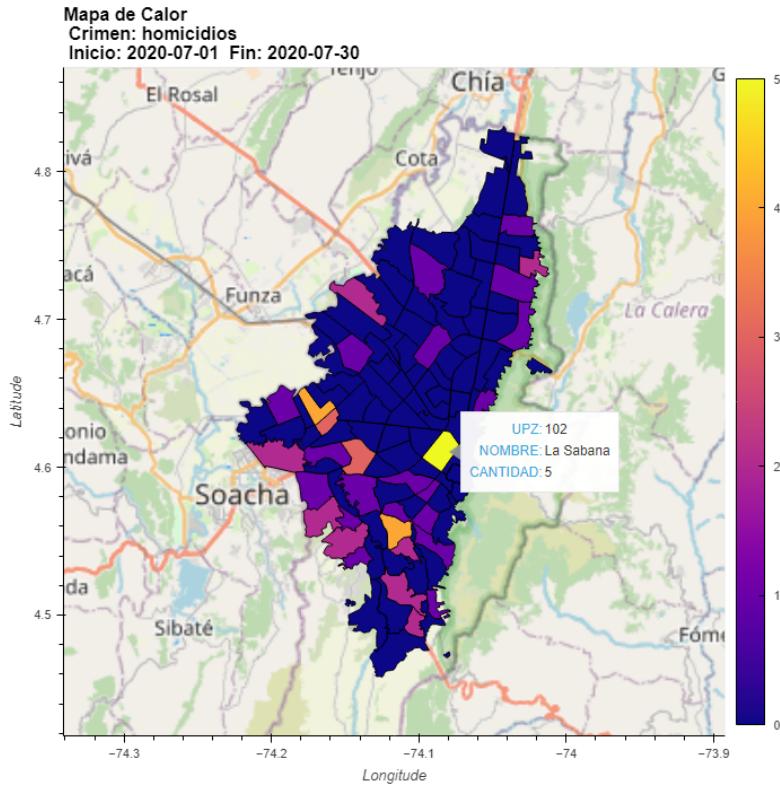


Figure 3.1: Mapa de calor de los homicidios reportados en la zona urbana de Bogotá durante el mes de julio del 2020.

En la figura 3.1 se encuentra un ejemplo de mapa de calor. La ilustración provee una división de la zona urbana de Bogotá a través de las UPZs. A través de estas se comprende la distribución de los homicidios durante el mes de julio de 2020. Por ejemplo, se evidencia que en la zona occidente hay mayor ocurrencia de crímenes, en comparación con la zona comprendida por Teusaquillo y Fontibon.

3.5 Selección del modelo

A partir de la conceptualización de Bogotá como un conjunto de puntos interconectados, se seleccionó como función de predicción a un modelo de red neuronal basada en grafos, que además utilice el concepto de atención. El buen desempeño de modelos GNN para generar *embeddings* en diferentes tareas de machine learning justifica que sea un modelo adecuado para analizar la criminalidad de la ciudad de Bogotá. Además, el uso de capas de atención permite administrar la información de los *embedding* de forma eficiente, por lo que es positivo incorporarlas al modelo de crime forecasting.

3.6 Preparación del modelo

3.6.1 Definición de nodos

Para construir el modelo es necesario establecer la cantidad de regiones del espacio a analizar. Los reportes de crímenes tienen como uno de sus elementos el nombre del barrio en el cual se registró el suceso, por lo que en una primera aproximación se planteó el modelo de predicción a través de los barrios de Bogotá. Sin embargo, se descartó esa idea por dos motivos. El primero de ellos es que las divisiones de los barrios han variado con el pasar de los años, ya sea por la ejecución de obras públicas, por legalización de barrios o por decisiones del distrito. Es decir, las divisiones de los barrios no son transversales al tiempo, por lo que utilizarlos como puntos de referencia es problemático para el modelo. Por otra parte, Bogotá cuenta con más de 1900 barrios [10], por lo que utilizarlos como elemento espacial es computacionalmente costoso, de modo que se deben buscar alternativas para considerar menos puntos del espacio.

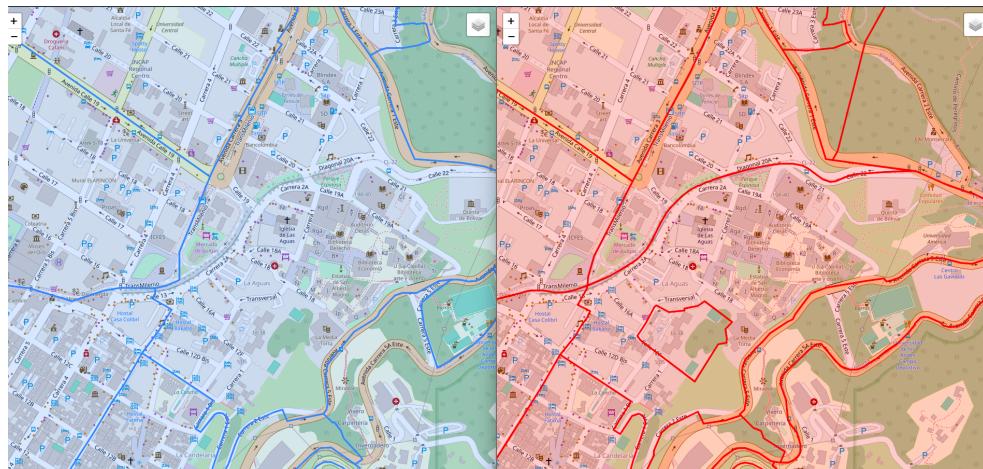


Figure 3.2: Comparación de las divisiones de los barrios legalizados durante dos instantes de tiempo. La zona es aquella cercana a la Universidad de Los Andes. El mapa de la izquierda contiene las divisiones establecidas por el distrito para el año 2016, el mapa de la derecha contiene las divisiones a corte de agosto del 2023.

En la figura (3.2) se comparan las divisiones de los barrios durante dos años diferentes. En caso de que ocurriera el mismo crimen durante los dos años concebidos, y en el mismo punto del espacio, los reportes indicarían que los crímenes ocurrieron en barrios diferentes.

Ante la problemática que se muestra en la figura se decidió agrupar barrios, y considerar dicha unión como un punto del espacio a analizar. Con esta idea la cantidad de puntos a estudiar se reduce, además se evitan problemas con relación a las divisiones establecidas por el distrito. En particular, se utilizaron las UPZs. Los barrios que componen una UPZ tienen características similares, en cuanto a población y economía, por lo que agruparlos de esta manera es consistente con el propósito del modelo.

Bogotá cuenta con 112 UPZs, las cuales definen el grafo que representa a la ciudad en el modelo de Crime Forecasting. En la figura (3.3) se presenta el grafo que se generó a partir de las UPZs. Cabe mencionar que los enlaces definen la relación *Vecino Directo*, es decir, dos nodos conectados representan dos UPZs que son vecinas.

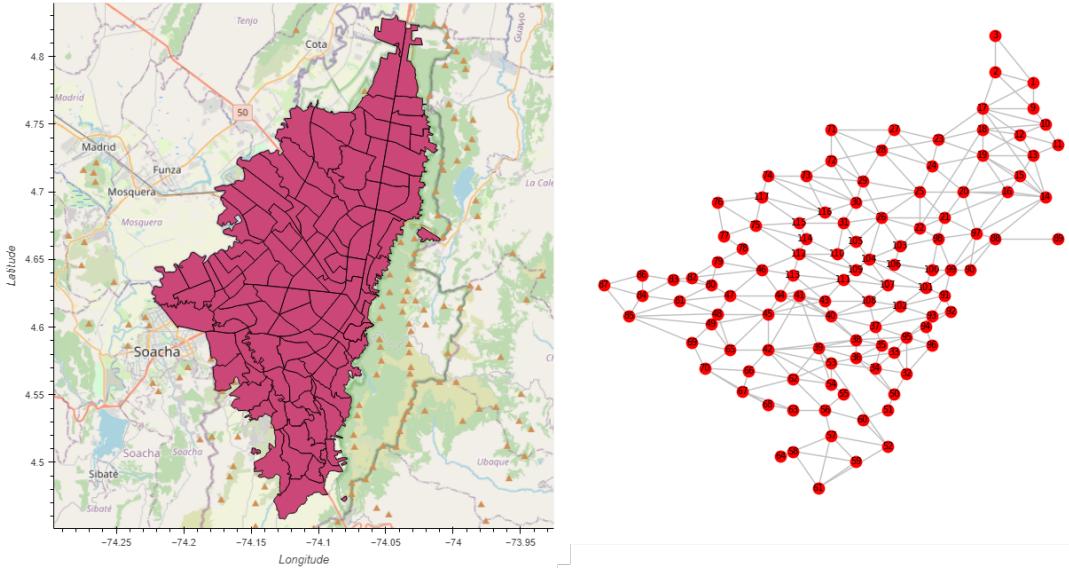


Figure 3.3: Comparación entre Bogotá y el grafo generado. El mapa de la izquierda presenta la división de Bogotá a través de UPZs. El mapa de la derecha contiene el grafo, donde cada punto representa una UPZ del mapa de la izquierda.

3.6.2 Transformación de datos

Los datos a disposición son los reportes de crímenes entre 2016 y 2019, los cuales se pueden analizar como una serie de tiempo. Es decir, cada reporte se entiende como una muestra de una variable aleatoria, que es la cantidad de ocurrencias de un crimen, en un día particular del año.

El rango de tiempo es de 4 años, por lo que cada UPZ tiene muestras en 1460 días (desde el 1 de enero de 2016 hasta el 31 de diciembre de 2019) para los 3 tipos de crimen que se analizan. Es decir, cada nodo se caracteriza por medio de una matriz de información $X \in \mathbb{R}^{4 \times 1460}$. Con dicha estructura, la columna j de X indica: la fecha en que se tomó la muestra, la cantidad de homicidios reportados, la cantidad de hurtos a personas y la cantidad de hurtos de celulares. Ahora bien, dicha información se usará como entrada del modelo, pero se deben realizar transformaciones sobre sus valores, pues el formato actual no se puede usar para entrenar un modelo de machine learning.

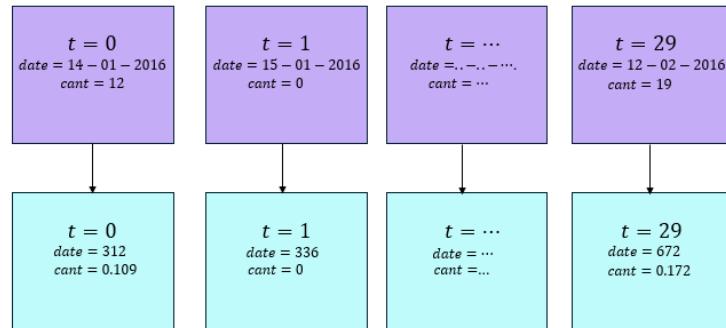


Figure 3.4: Proceso de transformación de la información contenida en los reportes a valores numéricos que se usan como entradas del modelo.

El elemento *fecha* tiene un formato que no es numérico, por lo que se debe transformar en

un valor continuo, como se observa en la figura (3.4). El mecanismo para sustituir la fecha por un valor numérico consiste en contar los días que hay entre la fecha y el primer día del año (1 de enero). Además, la cantidad de ocurrencias de cada crimen se transformó por medio de una normalización. Para cada día se contabiliza la cantidad de ocurrencias de un determinado crimen, luego, la cantidad de ocurrencias en cada UPZ se divide entre el total, de modo que cada columna de la nueva matriz de información indica la proporción de crímenes.

3.7 Modelo

El modelo utiliza la matriz de información de cada nodo para generar pronósticos de la cantidad normalizada de crímenes en un instante de tiempo futuro. Dado el funcionamiento de un modelo GNN, se utiliza message passing para generar *embeddings* de los nodos. El *embedding* inicial es una matriz de información de tamaño 4×1460 , lo cual no es eficiente, pues el comportamiento de la ocurrencia de crímenes en un instante futuro no depende de instantes considerablemente alejados en el tiempo, como lo son aquellos de más de 3 años en el pasado.

En lugar de utilizar la matriz de información en su totalidad, se utilizó una muestra más pequeña para realizar los pronósticos. En particular, se construyó el modelo de la figura (3.5).

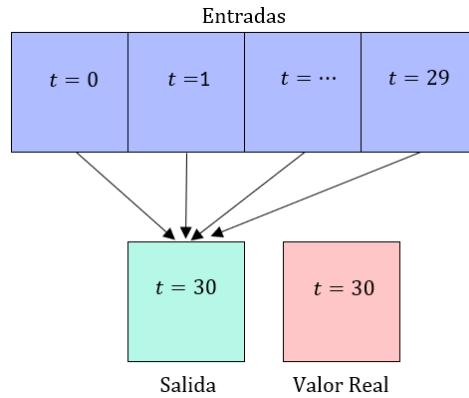


Figure 3.5: Ilustración del modelo de pronóstico de criminalidad.

Con base en modelos autorregresivos en series de tiempo, se construyó el modelo de la figura (3.5). La función de predicción utiliza la información de los reportes en los últimos 30 días para generar una predicción de los reportes en el día 31 (también normalizados).

$$MSE(\text{nodo}) = \frac{1}{T} \sum_t (y_t^i - GANN(x_{t-1}^i, \dots, x_{t-30}^i))^2 \quad (3.1)$$

$$MSE(\text{modelo}) = \frac{1}{112} \sum_{i=1}^{112} MSE(\text{nodo}) \quad (3.2)$$

La métrica de desempeño del modelo es el error cuadrático medio de todos los nodos.

3.7.1 Implementación

Finalmente, se optó por la implementación de un modelo de predicción específico para cada tipo de crimen. Para este propósito, se adoptó la arquitectura detallada en la figura (3.6). El modelo

se compone de un módulo *ENCODER*, encargado de generar el primer *embedding* para cada nodo. Posteriormente, se incorpora un módulo de atención que utiliza la función sigmoide para actualizar el *embedding*. En la fase final, se emplea un módulo de predicción (P) compuesto por una red neuronal con 2 capas ocultas. La salida de este módulo representa la proporción estimada de crímenes que ocurrirán en el siguiente intervalo de tiempo en el nodo actual.

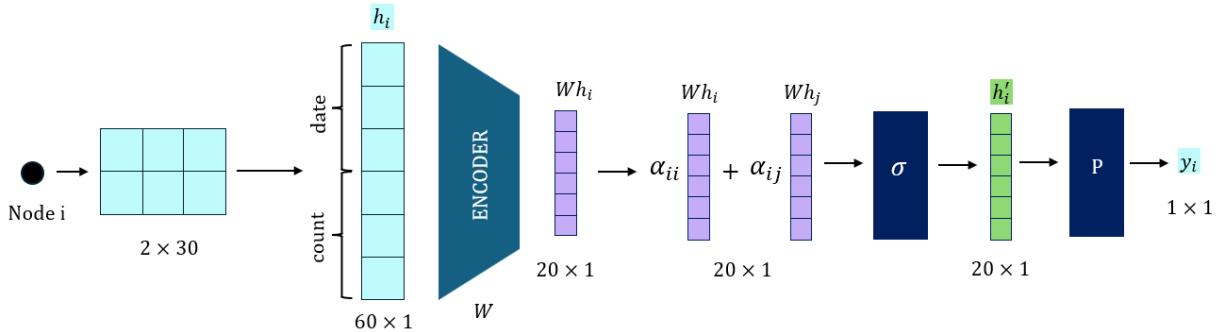


Figure 3.6: Arquitectura del modelo de crime forecasting.

Capa (Tipo)	Dimensión de salida	Número de parámetros
Densa	10	230
LeakyReLU	10	0
Densa	1	20
Número total de parámetros: 240		

Table 3.1: Estructura de la red neuronal usada para predecir la proporción de ocurrencias.

Para entrenar el modelo se utilizó el algoritmo (1). Allí se precisa que se utilizó un método de descenso para actualizar los parámetros del modelo. Por último, en la tabla (3.2) se describen los parámetros de cada módulo.

Algorithm 1 Entrenamiento del modelo

Require: Tamaño del batch (m), cantidad de epochas (E), cantidad de batchs (B)

Definir nodos

Iniciarizar parámetros aleatoriamente W , \vec{a}

for Epoch < E **do**

 loss = 0

 step = 0

while step ≤ B **do**

 Generar batch $\{x^1, \dots, x^m\}$ y $\{y^1, \dots, y^m\}$

 Estimar los valores por medio del modelo $\hat{y} \leftarrow \{f(x^i) | 1 \leq i \leq m\}$

 Calcular MSE

 loss \leftarrow loss + MSE

 step \leftarrow step + 1

end while

 loss \leftarrow loss/step

 Estimar gradiente $\hat{g} \leftarrow f(\text{loss})$

 Actualizar los parámetros

end for

Modulo	Dimensión de salida	Número de parámetros
Encoder	20	1220
Atención	20	40
P	1	240
Parámetros por nodo: 240		
Número total de parámetros: 168.000		

Table 3.2: Estructura del modelo de predicción del crimen.

4 Resultados

4.1 Entrenamiento

Para el entrenamiento del modelo se utilizó el 70% de los datos, que equivale a los reportes de 1001 días, el 30% restante se destinó a la validación. Para el algoritmo de entrenamiento se utilizó una tasa de aprendizaje $\eta = 0.1$, un tamaño de batch $m = 30$, $B = 15$ para la cantidad de batches y se consideraron 30 épocas.

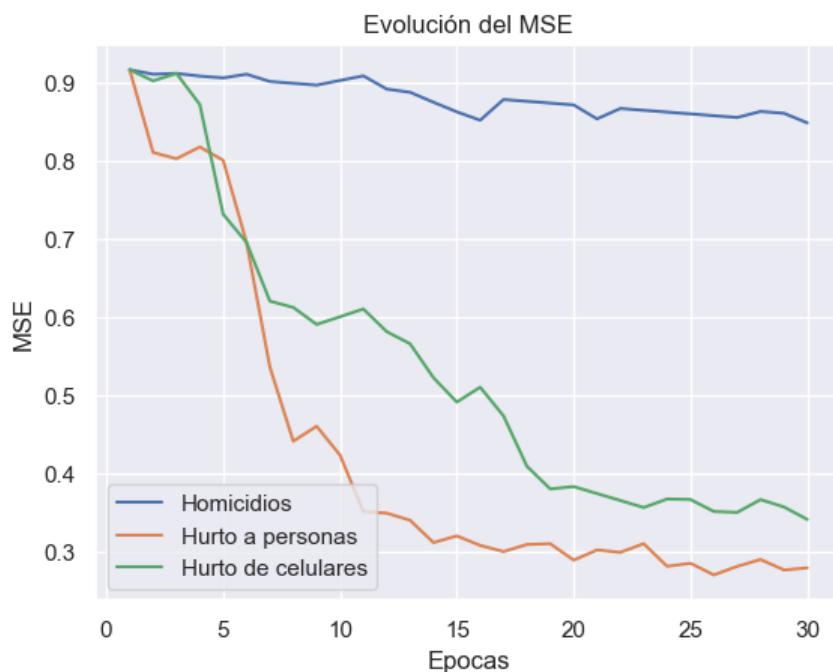


Figure 4.1: Evolución del MSE a través del tiempo de entrenamiento, medido en épocas.

En la figura (4.1) se presenta la evolución de la media cuadrática de los modelos a través de las épocas, cuyos valores en la época 30 se reportan en la tabla (4.1). En la gráfica se observa que el error disminuye a medida que avanza el tiempo de procesamiento, lo cual es consistente con el método de actualización de los parámetros (método de descenso), no obstante el desempeño no es igual para todos los modelos. En particular, el modelo de predicción de homicidios tiene el peor desempeño, pues en 30 épocas el cambio en la función de error es menor que aquel en los otros modelos.

Modelo	MSE
Homicidios	0,8478
Hurto a personas	0,279
Hurto de celulares	0,341

Table 4.1: MSE de los modelos en los datos de entrenamiento.

4.2 Desempeño

En la tabla (4.2) se presenta el desempeño de cada modelo en el conjunto de validación. Para medir el desempeño se utilizaron tres métricas: MSE, RMSE y MAPE.

Modelo	MSE	RMSE	MAPE
Homicidios	0,8613	0,9281	52,93%
Hurto a personas	0,619	0,7868	33,71%
Hurto de celulares	0,6831	0,8265	40,11%

Table 4.2: Medidas de desempeño en los datos de validación.

Los valores de las métricas de desempeño indican que el modelo de predicción de hurto a personas es aquel con mejor rendimiento. Por su parte, el modelo con peor desempeño es el modelo de predicción de homicidios.

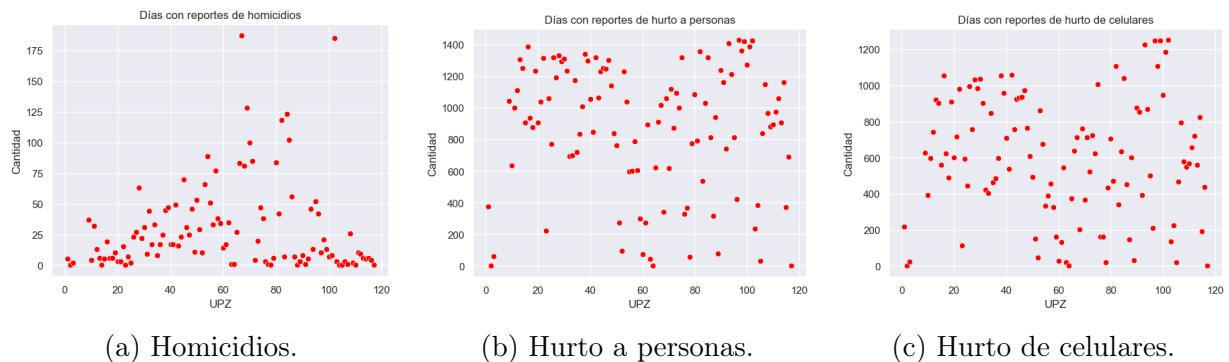


Figure 4.2: Las gráficas contienen la cantidad de días entre el 1 de enero de 2016 y el 31 de diciembre de 2019 en que se realizó al menos un reporte, para cada una de las UPZ.

Los resultados de las métricas de desempeño muestran que los modelos tienen dificultad para realizar pronósticos. El modelo de hurto a personas tiene un MAPE de 33,71% es decir, en promedio, el pronóstico de cada nodo tienen un error de 33,71%. Por ejemplo, si la proporción del crimen hurto a personas que ocurre en una UPZ es igual a 0.10, entonces, el modelo estima 0.1337 o 0.06629.

En la figura (4.2) se comparan la cantidad de días en que se reportaron crímenes en cada una de las UPZ. Se observa que la cantidad de denuncias es menor en el delito de homicidio. En la figura (4.2a) se evidencia que la cantidad de fechas en que se denuncia un homicidio, es pequeña en comparación con los demás delitos. De hecho, el valor máximo es de 190 fechas, es decir, en

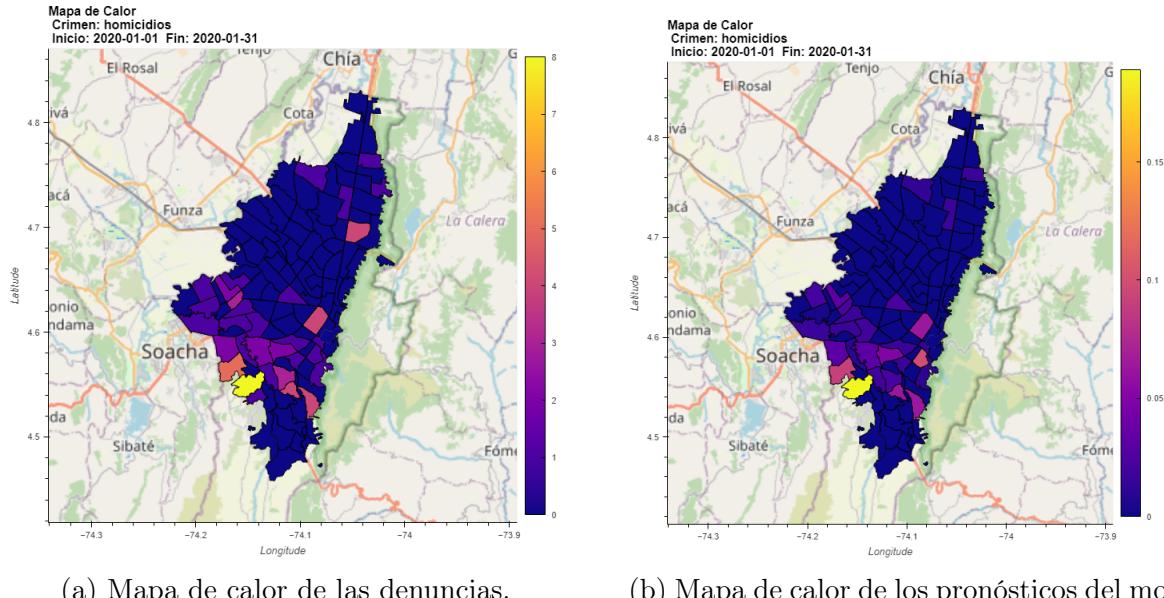
la mayor parte de las fechas (86.71%) no hay denuncias de homicidios, por lo que las matrices de información están compuestas en su mayoría por ceros. Como consecuencia, los procedimientos realizados para entrenar el modelo no obtienen un buen desempeño, pues la mayor parte de la información que ingresa es nula.

Por último, se utilizó el modelo para pronosticar las proporciones de los 3 crímenes para una ventana de tiempo que no estaba contenida en los datos. En particular, se realizaron pronósticos para el mes de enero del año 2020.

Modelo	MAPE
Homicidios	45,29%
Hurto a personas	34,68%
Hurto de celulares	63,13%

Table 4.3: MAPE de los modelos en los pronósticos de enero del año 2020.

El desempeño de los modelos se encuentra en la tabla (4.3). En las imágenes (4.3) a (4.5) se comparan los valores reales, con las predicciones de las proporciones por medio de mapas de calor. La métrica MAPE indica que los modelos no tienen un buen desempeño, no obstante, los modelos son útiles. En los mapas de calor se evidencia que los modelos emulan la distribución de los crímenes, lo cual es de utilidad para generar medidas para controlar la ocurrencia de estos.



(a) Mapa de calor de las denuncias.

(b) Mapa de calor de los pronósticos del modelo.

Figure 4.3: Comparación entre los datos reales y las predicciones de la distribución de los homicidios durante el mes de enero del año 2020.

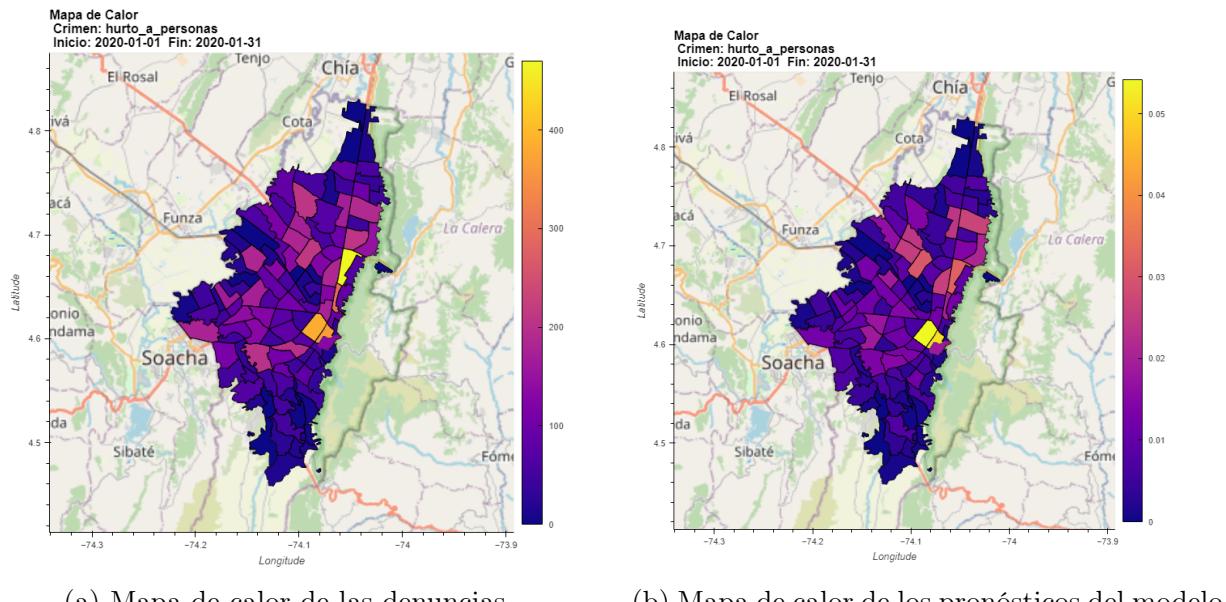


Figure 4.4: Comparación entre los datos reales y las predicciones de la distribución de los hurtos a personas durante el mes de enero del año 2020

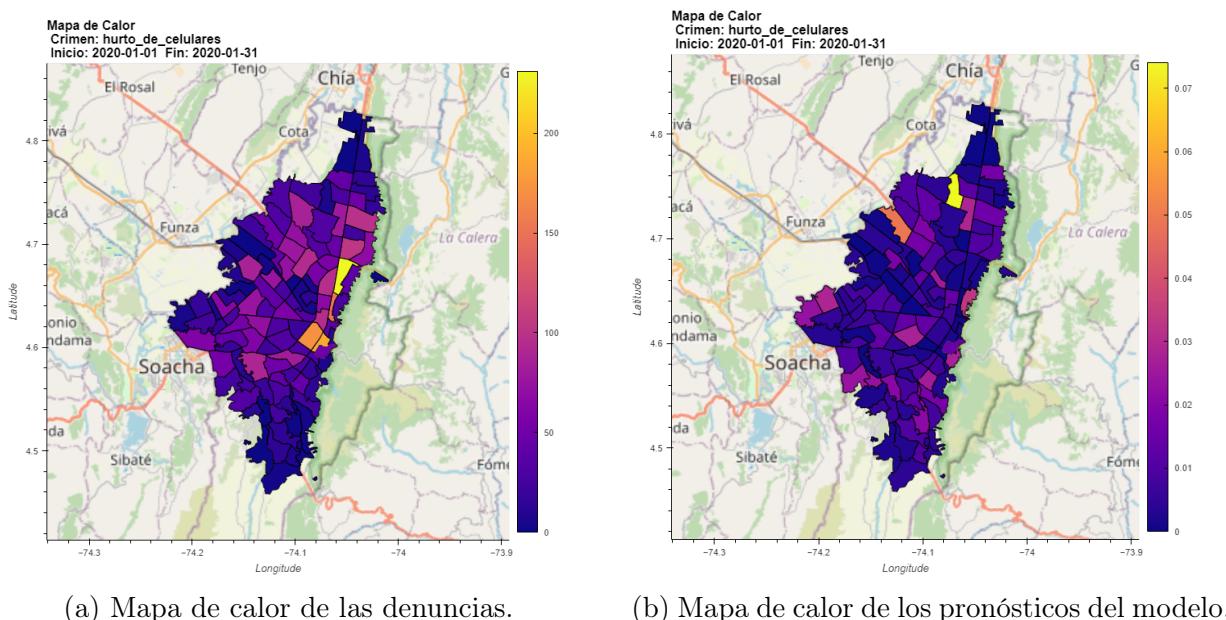


Figure 4.5: Comparación entre los datos reales y las predicciones de la distribución de los hurtos de celulares durante el mes de enero del año 2020

5 Conclusiones

Las herramientas de Deep Learning se revelan como recursos de gran valía, fundamentales para la construcción de los modelos predictivos de criminalidad. Al utilizar un modelo basado en la estructura de un grafo, se logró capturar las relaciones espaciales que subyacen a los patrones de ocurrencia de crímenes. Además, al concebir el problema de predicción como un modelo autorregresivo en series de tiempo, se exploraron las dinámicas temporales asociadas a los crímenes. Como resultado, se han creado tres funciones de predicción que pueden desempeñar un papel crucial en la lucha contra la inseguridad en la ciudad de Bogotá. Estos modelos permiten estimar la distribución de crímenes en la ciudad y, en manos de las autoridades, posibilitan una asignación más eficiente de los recursos en el ámbito de la seguridad.

Finalmente, para futuros proyectos relacionados, sería importante considerar conceptos de Deep Learning que no se utilizaron en el proyecto, como el concepto de memoria, así como el uso de otras arquitecturas de red neuronal, como redes neuronales recurrentes. Además, sería de gran valor contar con mejores datos de la criminalidad de Bogotá, como reportes más precisos en cuanto a la ubicación geográfica en que ocurrió un crimen, lo cual permitiría construir un grafo más complejo que aquel considerado en el proyecto.

References

- [1] Activation functions. [Online]. Available: <https://www.geeksforgeeks.org/activation-functions/>. Last accessed 2023-11-1.
- [2] Artificial neural networks and its applications. [Online]. Available: <https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/>. Last accessed 2023-11-1.
- [3] Graph neural network and some of gnn applications: Everything you need to know. [Online]. Available: <https://neptune.ai/blog/graph-neural-network-and-some-of-gnn-applications>. Last accessed 2023-11-10.
- [4] The graph neural network model. [Online]. Available: https://www.cs.mcgill.ca/~wlh/grl_book/files/GRL_Book-Chapter_5-GNNs.pdf.
- [5] Mapa interactivo. [Online]. Available: <https://recursos.ccb.org.co/ccb/pot/PC/files/3distribucion.html>. Last accessed 2023-11-10.
- [6] Unidades de planificaciÓn zonal / definiciÓn. texto explicativo. [Online]. Available: <https://recursos.ccb.org.co/ccb/pot/PC/files/3definicion.html>. Last accessed 2023-11-10.
- [7] Weight (artificial neural network). [Online]. Available: <https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network>. Last accessed 2023-11-1.
- [8] What is perceptron: A beginners guide for perceptron. [Online]. Available: <https://www.simplilearn.com/tutorials/deep-learning-tutorial/perceptron>. Last accessed 2023-11-10.
- [9] ¡bienvenido a queremosdatos.co! [Online]. Available: <https://www.queremosdatos.co/help/about>. Last accessed 2023-10-12.
- [10] RedacciÓn Ciencuadras. ¿quÉ tanto sabes de bogotá y tu barrio? [Online]. Available: <https://www.ciencuadras.com/blog/por-que-vivir-en/cuanto-conoces-a-bogota-y-a-tu-barrio-descubrelo>, 2023. Last accessed 2023-11-10.
- [11] IBM. What is deep learning? [Online]. Available: <https://www.ibm.com/topics/deep-learning>. Last accessed 2023-11-1.
- [12] RedacciÓn Libretilla. Las 40 ciudades mÁs pobladas de colombia. [Online]. Available: <https://libretilla.com/ciudades-mas-grandes-de-colombia-por-poblacion/>, 2021. Last accessed 2023.
- [13] Tatiana Munevar. Creció la percepción de inseguridad en colombia: estas son las ciudades donde la gente se siente mÁs amenazada. [Online]. Available: <https://acortar.link/gSzqo5>, 2023. Last accessed 2023.

- [14] Redacción Nacional. ¿cómo combatir la inseguridad en cada localidad de bogotá? [Online]. Available: <https://acortar.link/h95kwd>, 2023. Last accessed 2023.
- [15] Portafolio. Bogotá, con la percepción de inseguridad más alta en 6 años. [Online]. Available: <https://www.portafolio.co/tendencias/bogota-con-la-percepcion-de-la-inseguridad-mas-alta-en-6-anos-561756>, 2022. Last accessed 2023.
- [16] Derya Soydaner. Attention mechanism in neural networks: where it comes and where it goes. *Neural Computing and Applications*, 34(16):13371–13385, May 2022.
- [17] Matthew Urwin. 20 deep learning applications you should know. [Online]. Available: <https://builtin.com/artificial-intelligence/deep-learning-applications>. Last accessed 2023-11-1.
- [18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

6 Appendix A Feature Engineering

6.1 Datos a Disposición

Crimen	Archivo (.xlsx)	Año(s)	Utilidad
Amenazas	AMENAZAS, AMENAZAS_2	2022 & 2023	NO
Delitos Sexuales	DELITOS_SEXUALES_1, DELITOS_SEXUALES_2	2010 a 2023	SI
Extorsión	EXTORSION_1, EXTORSION_2	2010 a 2023	SI
Homicidios	HOMICIDIOS_1, HOMICIDIOS_2	2010 a 2023	SI
Hurto de Automotores	HURTO_A_AUTOMOTORES_1, HURTO_A_AUTOMOTORES_2	2010 a 2022	SI
Hurto de Motocicletas	HURTO_DE_MOTOCICLETAS_1, HURTO_DE_MOTOCICLETAS_2	2010 a 2023	SI
Hurto de Bicicletas	HURTO_DE_BICICLETAS_1, HURTO_DE_BICICLETAS_2	2010 a 2023	SI
Hurto a Comercio	HURTO_A_COMERCIO_1, HURTO_A_COMERCIO_2	2010 a 2023	SI
Hurto a Residencias	HURTO_A_RESIDENCIAS_1, HURTO_A_RESIDENCIAS_2	2010 a 2023	SI
Hurto a Personas	HURTO_A_PERSONAS_1, HURTO_A_PERSONAS_2	2010 a 2023	SI
Hurto de Celulares	HURTO_DE_CELULARES_1, HURTO_DE_CELULARES_2	2010 a 20223	SI
Lesiones Personales	LESIONES_PERSONALES_1, LESIONES_PERSONALES_2	2010 a 2023	SI
Secuestro	SECUESTRO_1, SECUESTRO_2	2010 a 2023	SI
Violencia Intrafamiliar	VIOLENCIA_INTRAFAMILIAR, VIOLENCIA_INTRAFAMILIAR_2	2022 & 2023	NO

Table 6.1: Reporte de Datos a Disposición. Un archivo es útil si los datos que contiene abarcan un periodo de al menos 5 años o si dicho archivo, junto a otro archivo, cumplen con dicho criterio. En caso de que un archivo contenga información de un año que ya se encuentra en otro archivo, este archivo no es de utilidad.

6.2 Descripción

Columna	Descripción
Fecha	Fecha de ocurrencia del crimen.
Departamento	Departamento de ocurrencia del crimen.
Municipio	Municipio de ocurrencia del crimen.
Barrio	Barrio del municipio en el cual ocurrió el crimen.
Zona	Indica si la zona en que ocurrió el crimen es urbana o rural.
Clase de Sitio	Indica el tipo de sitio en el cual ocurrió el crimen. La clase contempla espacios públicos como vías, espacios privados como casas y espacios no físicos como redes sociales.
Arma empleada	Arma empleada por el agresor/victimario.
Edad	Edad de la víctima.
Estado Civil	Estado civil de la víctima.
Genero	Genero de la víctima.
Móvil Agresor	Mecanismo de desplazamiento que utilizo el agresor.
Móvil Víctima	Mecanismo de desplazamiento que utilizo la víctima.
País de Nacimiento	País en que nació la víctima.
Ocupación	Ocupación de la víctima. Por ejemplo: estudiante o empleado particular.
Profesión	Profesión de la víctima.
Escolaridad	Mayor nivel educativo alcanzado por la víctima.
Descripción de la Conducta	Indica el artículo de la ley que explica el delito. La descripción contempla aspectos como el arma que se utilizó, así como la ubicación en que se dio el delito.

Table 6.2: Reporte de columnas que contienen los archivos a disposición. La tabla contiene el nombre de la columna, así como una descripción de la información que se encuentra en esta. Cabe mencionar que hay variaciones entre los diferentes archivos con respecto al nombre de las columnas, pero la información que almacenan es la misma. Por ejemplo, se tienen registros donde en lugar de *Escolaridad* se tiene una columna *Nivel Educativo*.