PREDICCIÓN DE LOCALIZACIÓN DE PUESTOS DE TRABAJO CON REDES NEURONALES SOBRE IMÁGENES SATELITALES

Paulina Pizarro, Pontificia Universidad Católica de Chile papizarro@uc.cl
Hans Löbel, Pontificia Universidad Católica de Chile halobel@ing.puc.cl
Juan Carlos Muñoz, Pontificia Universidad Católica de Chile jcm@ing.puc.cl

RESUMEN

La localización de trabajos dentro de las ciudades es fundamental para planificar adecuadamente los sistemas de transporte. Nuevas metodologías como el *deep learning* y una mayor disponibilidad de imágenes satelitales permite desarrollar nuevas técnicas para obtención de datos de forma automatizada y menos costosa que las metodologías tradicionales actuales. En esta investigación se propone el uso de redes neuronales convolucionales sobre imágenes satelitales para predecir la localización de trabajos en la ciudad de Santiago. Extrayendo características visuales mediante *transfer learning* y utilizando información de uso de suelos se diseña una estructura de red neuronal de dos inputs. Se proponen modelos de clasificación obteniendo un nivel de exactitud de 85% para un modelo binario que predice la presencia o no presencia de trabajos sobre recortes de imágenes. Los experimentos muestran que es posible utilizar CNNs sobre imágenes satelitales para interpretar presencia de actividades laborales.

Palabras claves: machine learning, imágenes satelitales, Smart city

ABSTRACT

Job localization within cities is fundamental for proper planning for transportation systems. New technologies such as deep learning and a broader availability of satellite imagery allows the development of new techniques for more automatized and less costly way of gathering data. In this research we propose the use of convolutional neural networks over satellite images to predict job localization in the city of Santiago. We extract visual features through transfer learning and utilize land use information proposing a two-input neural network. Classification models are proposed obtaining an accuracy of 85% for a binary model that predicts the presence or absence of jobs over images. The experiments show that it is possible to use CNNs over satellite images to interpret the presence of work force and locate it to a certain level.

Keywords: machine learning, satellite image, smart city

1 INTRODUCCIÓN

Para planificar efectivamente los sistemas de transporte en las ciudades, es necesario conocer una serie de indicadores como densidad urbana, uso de suelo y accesibilidad a servicios (Weber et al., 2016). Entre estos, la localización de puestos de trabajos es fundamental para comprender los traslados que ocurren dentro de la ciudad, en especial los de hora punta que definen la capacidad del sistema que es necesario ofrecer.

En Chile, las fuentes de información de puestos laborales presentan problemas de baja precisión, poca actualización o incompletitud. Por esta razón, es relevante cuestionarse si es posible usar datos actualmente disponibles para facilitar la obtención de esta información. Nuevas metodologías de *machine learning*, como las redes neuronales profundas, permiten la extracción de patrones de grandes cantidades de datos de forma eficiente y menos costosa que otros métodos, como recolección directa mediante encuestas. Con el reciente desarrollo de estas técnicas se han logrado avances que permiten desprender indicadores de uso de suelo mediante imágenes satelitales como: localización de campos de refugiados (Quinn *et al.*, 2018), clasificación de ambientes urbanos en ciudades (Albert et al., 2017), detección de vías, vehículos y árboles (Napiorkowska et al., 2018), identificación de sectores de pobreza (Wurm et al., 2019), entre otros. En esta línea, esta investigación propone una metodología novedosa para predecir la localización de puestos de trabajo utilizando redes neuronales sobre imágenes satelitales para la ciudad de Santiago de Chile. Al no existir una metodología de este tipo propuesta anteriormente en la literatura, esta investigación pretende ser un aporte interdisciplinar haciendo uso de la ciencia de datos e inteligencia artificial en ingeniería de transporte.

2 MARCO TEÓRICO

En esta sección se profundiza en conceptos y metodologías de *deep learning* que serán de ayuda para comprender de mejor manera los experimentos realizados.

2.1 *Deep learning*

El aprendizaje profundo, más conocido por su nombre en inglés, *deep learning* se caracteriza por el uso de redes neuronales profundas para tareas de aprendizaje (Lecun, Bengio y Hinton, 2015). Específicamente, estos modelos utilizan múltiples capas de operaciones no lineales, o neuronas, que trabajan de manera secuencial para realizar la estimación requerida. Los parámetros de las capas son entrenados en conjunto y a mayor cantidad de capas se habla de una red más profunda.
Una de las redes neuronales más utilizadas es el perceptrón multi capa (MLP). En términos generales, los perceptrones multi capa se interpretan como conjuntos de neuronas y se modelan como grafos acíclicos dirigidos, donde cada nodo representa una neurona (para más detalle sobre las estructuras matemáticas fundamentales de aprendizaje profundo revisar el libro de Zhang *et al.* (2019)).

Las redes se entrenan por *batches*, que son divisiones del conjunto de datos que se entregan a la red de a uno. Cuando cada objeto del set de datos ha pasado por la red, se cumple una época. El entrenamiento se realiza utilizando optimizadores sobre funciones de pérdida. Uno de los

optimizadores más comunes es el descenso de gradiente estocástico (SGD), que funciona de la misma forma que el descenso de gradiente común, excepto que disminuye cómputos al utilizar el gradiente de un objeto dentro de cada *batch* de forma aleatoria. Un optimizador muy relevante derivado de SGD es Adam, que innova utilizando los primeros y segundos momentos del gradiente para actualizar los parámetros (para más detalle revisar el artículo de Kingma y Ba, (2015)).

2.2 *Transfer learning*

Actualmente, una de las principales barreras para entrenar efectivamente redes neuronales es la gran cantidad de datos necesaria. Aquí es donde el *transfer learning* (o transferencia de aprendizaje en español) genera un relevante aporte, al permitir el uso de redes pre entrenadas con grandes sets de datos para otros usos con conjuntos de datos diferentes y de menor tamaño. Razavian *et al.* (2014) sientan las bases de esta metodología en *deep learning*, usando redes neuronales convolucionales (CNNs) previamente entrenadas para extraer características de set de datos diferentes de manera muy efectiva. Las CNNs se utilizan para procesar imágenes por tener la capacidad de computar eficientemente conocimiento por áreas o por conjuntos vecinos de pixeles.

ResNet, la CNN propuesta por He et al. (2016) se ha convertido en una de las redes más utilizadas en *transfer learning*. La ResNet integra el uso de capas residuales de aprendizaje con referencia a las capas de input que permiten facilitar la optimización y, por lo tanto, mejorar la precisión al poder utilizar redes más profundas. He et al. (2016) entrenaron la red con el conjunto de datos ImageNet para clasificar imágenes dentro de 1.000 categorías distintas. Para *transfer learning* se utilizan los pesos previamente entrenados para este propósito, pero obviando la última capa densa de clasificación, entregando un vector representando 2.048 características. De esta manera, la interpretación de características generada por el entrenamiento con ImageNet de la ResNet puede ser utilizado para entregar como input a otras redes con diferentes propósitos y entrenarlas.

3 TRABAJOS RELACIONADOS

Entre los modelos de clasificación que utilizan imágenes de alta resolución se encuentra el de Chen *et al.* (2019) que utilizando imágenes de 0.6m de resolución identifica entre sectores construidos o caminos y carreteras. Otros modelos más complejos que clasifican entre un mayor número de clases son los de Pritt y Chern (2018) y Christie *et al.* (2018), en los cuales predicen entre 62 categorías de objetos e instalaciones sobre imágenes satelitales globales utilizando como inputs las imágenes y sus respectivos metadatos asociados, como resolución, temporalidad, zona UTM, entre otros. Las imágenes utilizadas en estos modelos no comparten necesariamente el mismo nivel de resolución, sin embargo, se pueden clasificar en la misma categoría en el rango de alta o muy alta resolución. Con un propósito similar, pero con 21 categorías, Liu *et al.* (2018) proponen el uso de una red neuronal convolucional profunda con agrupación espacial piramidal más conocida por SPP por sus siglas en inglés, *spatial pyramid pooling*, logrando un entrenamiento más rápido y facilitando el uso de imágenes de distintos tamaños.

Hadzic et al. (2020) diseñan un modelo de regresión que intenta predecir la cantidad de población en sectores de campamentos de refugiados con imágenes captadas por drones de 30 cm de resolución en Bangladesh. Utilizan una ResNet50 pre entrenada con el set de datos ImageNet

modificada para regresión obteniendo MAE (error medio absoluto) y MAPE (error porcentual medio absoluto) de 3.341 y 7,02% respectivamente, superando los resultados de metodologías sin uso de redes neuronales convolucionales.

Finalmente, una investigación que no utiliza imágenes satelitales, sino aéreas es la de Jeon et al. (2018). Generan un control de señales de tráfico adaptativo utilizando aprendizaje por refuerzo con redes neuronales convolucionales sobre secuencias de videos aéreos. Esta es la única investigación a conocimiento de los autores que utiliza imágenes aéreas con CNNs para ingeniería de transporte.

4 METODOLOGÍA

Los modelos generados en la presente investigación utilizan como inputs características visuales de imágenes y vectores de usos de suelo asociados al área correspondiente de la imagen analizada. Las redes, conformadas por una interconexión de múltiples capas son entrenadas y finalmente entregan como resultado la predicción de categoría o valor numérico estimada por la red, según si es un modelo de clasificación o regresión.

En este capítulo se describe cómo se generó el set de datos y de qué forma se configuraron las imágenes para ser procesadas por la red. Además, se detallan las estructuras de red generales para los diferentes modelos de clasificación que luego se describen en la sección de evaluación experimental.

4.1 Generación de la base de datos

Para generar el set de datos fue necesario juntar información proveniente de distintas fuentes y procesarla para llegar a un conjunto de datos consolidado y utilizable para entregar a una red neuronal.

Se utilizó como base una imagen satelital de la ciudad de Santiago tomada el 2014 con una resolución de 0,35 metros por pixel (Ministerio de Economía, 2015). Con el uso de herramientas computacionales como Python, QGIS y paquetes preprogramados como rasterio y gdal, se configuró el formato de la imagen satelital para georreferenciar en un mapa virtual.

La información de puestos de trabajo se obtuvo del catastro de empresas del Servicio de Impuestos Internos correspondiente al año 2012. El catastro contiene las empresas inscritas en la ciudad de Santiago con su dirección asociada y la cantidad de trabajadores. Se procesaron las direcciones de cada empresa para convertirlas en coordenadas geográficas mediante la api Nominatim y de esta manera fue posible asignar cada empresa a un punto en un mapa virtual con su respectiva cantidad de trabajadores. Si bien cada empresa puede tener múltiples lugares en donde sus trabajadores desempeñan funciones, en la base de datos utilizada solo se registra una ubicación, esto genera que algunos puntos muestren una cantidad muy alta de puestos laborales cuando en la realidad no es así. Con el fin de disminuir este error, al momento de unir los trabajos con las imágenes, se consideraron solo los puntos o empresas que indicaran una cantidad menor o igual a 200 trabajos. Esta decisión de diseño para definir un número límite se realizó en base a la observación de los

datos de manera general y particular. De forma general, la distribución de cantidad de trabajos de empresas muestra una diferenciación entre empresas con menos de 200 trabajos, donde la cantidad de empresas son más abundantes. De forma particular se observaron los datos identificando empresas revisando la cantidad de trabajos indicados y si coincidía con la realidad. La base de datos inicial del Servicio de Impuestos Internos contabilizaba 4.408.208 trabajadores en sus registros. Con el filtro descrito, la base de datos final utilizada considera 1.032.278 trabajos. Se utilizó el catastro de empresas del 2012 (y no 2014 como la imagen satelital) para utilizar en conjunto con la información de la Encuesta Origen Destino 2012 (EOD). Sin embargo, la georreferenciación de la EOD no presenta un nivel de precisión y al añadirla al set de datos se vieron perjudicados los resultados.

Con ambas fuentes de información georreferenciadas, se recortaron sub-imágenes de la imagen satelital inicial registrando la cantidad de trabajadores totales contemplados dentro del área y asignando ese valor a la imagen recortada. Las imágenes recortadas se guardaron en formato png y tienen un tamaño de 260x260 pixeles, que corresponden a un área cuadrada de 91x91 metros. El proceso de corte se hizo de forma que cada imagen tiene un traslape del 50% con la más cercana, hacia los lados y de arriba abajo, de esta manera se puede obtener un set de imágenes más cuantioso. Como resultado se obtiene un conjunto de aproximadamente 131.000 imágenes con su respectiva etiqueta de puestos laborales.

Es relevante aclarar que a lo largo de esta investigación se utiliza el concepto trabajos o puestos laborales, haciendo referencia a sectores sin, con o con cierta cantidad de trabajos, con la intención de simplificar la escritura. El catastro utilizado para localizar trabajos considera solo una porción de la verdadera fuerza laboral existente en Santiago y por lo tanto, al utilizar los conceptos trabajos o puestos laborales, los autores se refieren a trabajos adscritos a empresas y contabilizados por el Servicio de Impuestos Internos.

Con el fin de aportar un mayor contexto al conjunto de imágenes, se extrajo información de los usos de suelo de la base de datos de bienes y raíces del SII que contiene los metros cuadrados de 21 tipos de uso para cada manzana de la ciudad de Santiago. Las manzanas son considerablemente más grandes en área que la cubierta por las imágenes, además, en la mayoría de los casos, hay más de una manzana presente en una imagen y viceversa. Por esta razón, mediante la librería geopandas, se calculó el porcentaje del área de cada manzana que interseca con la imagen y para cada una de las 21 categorías se sumaron los metros cuadrados ponderados por dicho porcentaje.

4.2 Correcciones al conjunto de datos

En las experimentaciones iniciales, se pudo observar dificultades en predecir correctamente algunas imágenes etiquetadas con cero trabajos, que visualmente son similares a imágenes con presencia laboral. Un mayor análisis de los datos indicó que en gran parte de estos casos, se podría estar tratando con imágenes mal etiquetadas. Por esta razón, se depuró la base de datos excluyendo imágenes cuyo uso de suelo no hace sentido con la etiqueta de cantidad de trabajos. Tomando en cuenta usos con potencial laboral tales como: comercio, oficina, industria, hotel, administración pública y salud, se filtraron todas las imágenes que abarcaran al menos 50 metros cuadrados como suma de estos usos y cuya etiqueta indicara cero puestos de trabajo. Por ejemplo, una imagen etiquetada con cero trabajos, con una gran cantidad de metros cuadrados de uso de suelo asociado

a comercio ya no se incluiría en el set de datos. Con este procedimiento se disminuyó considerablemente el tamaño de los datos pasando de 97.673 a 48.247 imágenes con cero trabajos.

La reducción de la cantidad de datos puede tener efectos negativos en el entrenamiento de la red, generando un sobre entrenamiento más acelerado y por consecuencia peores resultados de predicción. Por otro lado, una base de datos con menos errores reduce el ruido y mejora el rendimiento. En este caso, el impacto de reducir el tamaño de datos fue menos significativo que el de minimizar los errores, ya que los resultados mejoraron significativamente con el uso de la base de datos depurada.

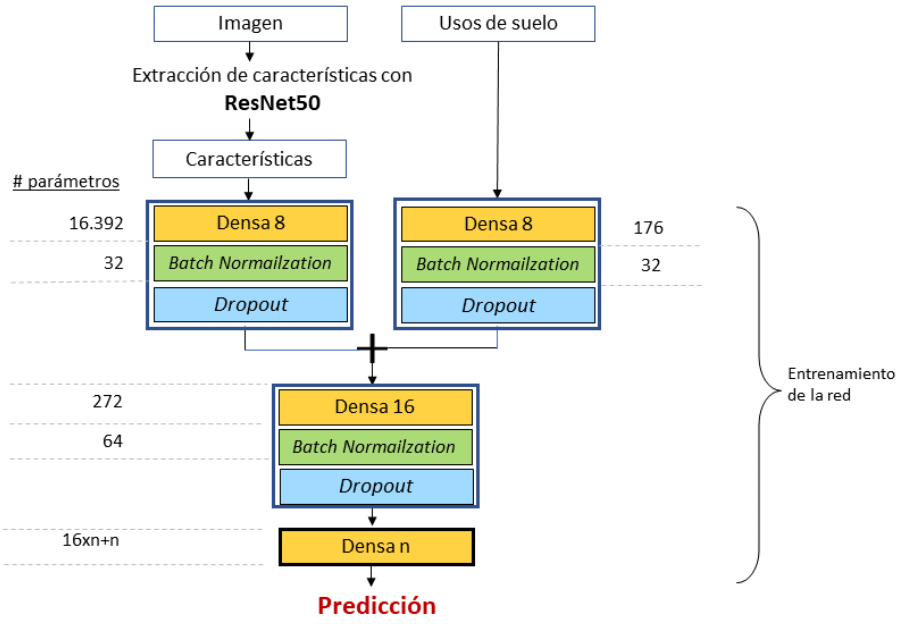
4.3 Estructura de red

En todos los modelos, previo al entrenamiento, se utiliza la ResNet50 pre entrenada sobre ImageNet para extracción de características de las imágenes. Para procesar las imágenes adecuadamente por la ResNet es necesario adaptar el set de datos para coincidir con su formato de recepción. El tamaño de imagen más grande que recibe la ResNet50 es de 224x224 pixeles, por este motivo se redujo el tamaño de las imágenes utilizando funciones de la librería TensorFlow. Esta operación genera una pérdida de resolución de 0,35 a 0,41 metros por pixel, pero también disminuye el tamaño total de datos procesados por la red. Fue preferible hacer esto a recortar las imágenes inicialmente en tamaño 224x224, ya que al hacerlo se disminuye el área total cubierta por la imagen perdiendo contexto que se considera relevante para el análisis de características. Otra adaptación necesaria que se aplicó a las imágenes para poder utilizar la ResNet fue implementar una transformación de los valores numéricos de los pixeles para que distribuyeran entre -1 y 1.

La red recibe dos inputs de forma paralela: las características visuales obtenidas de la ResNet y los vectores de uso de suelo (Figura 1). Ambos se procesan por bloques conformados por una capa densa (o *fully connected*) y *batch normalization*, esta última se usa para normalizar los valores de entrada y facilitar la regularización (Ioffe y Szegedy, 2015). Lo resultante de ambos bloques se concatena para luego seguir procesando en otro bloque de forma conjunta. A este último bloque se le añade una capa de dropout en modelos en los que se experimenta un sobreentramiento acelerado. Finalmente se añade una capa densa última que entrega la predicción.

Las arquitecturas de red utilizadas en la evaluación experimental son aquellas que lograron los mejores resultados en su respectivo modelo. Las combinaciones posibles de capas son muy numerosas, por no decir infinitas. Por lo tanto, siguiendo estructuras comunes en la literatura, considerando el tamaño de vectores iniciales y probando una gran cantidad de combinaciones se decidió la arquitectura que se muestra en la Figura 1.

Figura 1: Estructura general de red



4.4 Métricas de evaluación de modelos

Para la comparación de los modelos, se utilizaron tres métricas de evaluación muy comunes en modelos de clasificación, F1-score, *accuracy* y *balanced accuracy*. F1-score es una métrica clásica y se interpreta como el promedio ponderado de la precisión y *recall*. *Accuracy* mide la proporción de predicciones correctas con respecto al total de datos y *balanced accuracy* es una versión modificada de *accuracy* pero considerando cada categoría con igual peso. Si bien no es una métrica, también se utilizaron matrices de confusión, que son una buena herramienta para visualizar la distribución de las predicciones por cada clase, mostrando intuitivamente en qué categoría fueron correcta o equivocadamente asignadas las imágenes.

Es relevante mencionar que comúnmente en el entrenamiento de redes neuronales el conjunto de datos se divide aleatoriamente en tres subconjuntos: entrenamiento, validación y prueba (o testeo). Es importante hacer esta separación para obtener de la forma más fiable posible la real predictibilidad de los modelos. En esta investigación, el set de entrenamiento se conforma por el 80% de los datos totales y se utiliza para entrenar la red. El conjunto de validación se utiliza para evaluar durante el entrenamiento de la red y determinar cuándo detener el entrenamiento. Finalmente, el set de testeo se utiliza para la evaluación final del modelo ya que los datos pertenecientes a este conjunto no se utilizaron ni para entrenar la red ni para determinar la época de parada del entrenamiento. Tanto el set de validación como el de testeo se conforman por un 10% de los datos totales.

5 EVALUACIÓN EXPERIMENTAL

En la presente sección se detallan modelos de regresión y clasificación de dos, cuatro y diez categorías que entregaron los mejores resultados. Para todos se utilizó *transfer learning* con una ResNet50 pre entrenada en imagenet y los dos inputs previamente descritos: imágenes y vector de uso de suelos. Se utilizaron los *frameworks* TensorFlow y Keras para la codificación de las redes neuronales y computaciones matemáticas. Los modelos se entrenaron en los servidores IALAB del Departamento de Ciencias de la Computación de la Pontificia Universidad Católica y se utilizó una GPU GeForce GTX1080Ti.

5.1 **Modelo de regresión**

Se construyó un modelo de regresión para predecir el total de trabajos contenidos en cada imagen utilizando una función de pérdida de error cuadrático medio. Para medir la predictibilidad del modelo se utilizó el error medio absoluto, que compara la diferencia absoluta entre el valor a predecir de la imagen y la predicha por el modelo. Entrenando por 5 épocas se llegó a un resultado en el set de testeo se llegó a un error medio absoluto de 48. Este resultado se considera sub-óptimo para resolver el problema en cuestión. Sin embargo, tomando en cuenta que hay imágenes con miles de trabajos asociados, el tipo de dispersión de los datos hace que este problema sea altamente complejo. Probando distintas configuraciones de red, funciones de pérdida y filtros al set de datos, no fue posible superar los resultados. La dificultad del problema y el rápido sobre-entrenamiento genera que la red prediga con mayor probabilidad los valores más frecuentes en el set de datos, que son los valores más bajos.

5.2 **Clasificación binaria**

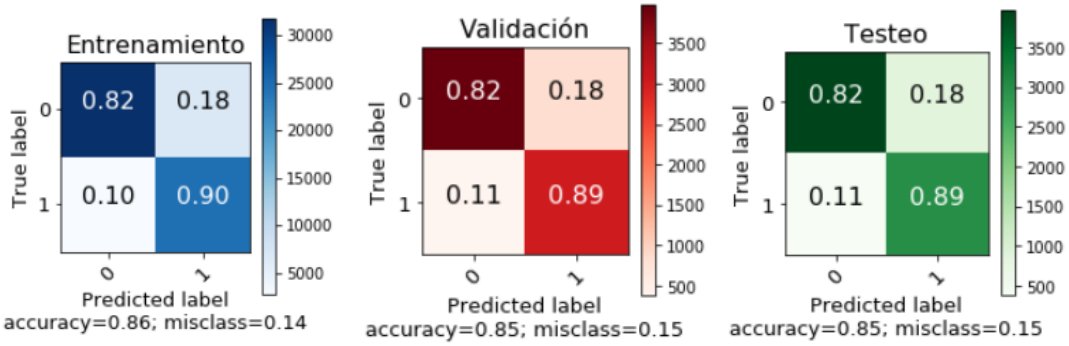
El modelo de regresión mostró dificultades para estimar números exactos de la cantidad de trabajos sobre registros que inicialmente no son altamente precisos. El modelo de dos clases invita a vislumbrar la posibilidad de discernir mediante la visualización de imágenes la presencia de trabajos. Una selección de categorías más general difumina los errores iniciales de la base de datos y, consecuentemente, genera un conjunto de datos de entrenamiento más acertado. La relativa simpleza de este modelo entrega posibilidades relevantes al momento de identificar puntos de concentración laboral. Si bien el análisis es menos complejo por imagen, la agrupación de estas genera áreas notorias de concentraciones contiguas de presencia o no presencia de actividad laboral. Las categorías se definieron como imágenes con trabajos, clase *1*, y sin trabajos, clase *0*, y se modificaron las etiquetas acordes a esto, obteniendo un set de datos relativamente parejo con 48.000 imágenes con cero trabajos y 34.000 imágenes con uno o más.

Se usa una activación ReLU en las capas densas intermedias y activación Softmax en la capa densa final. La función de pérdida implementada es *sparse categorical crossentropy* y se utiliza Adam como optimizador (estas características también son aplicables para los siguientes modelos de clasificación que se ven en esta sección). En modelos de clasificación, es común utilizar la función de pérdida *cross-entropy* (Zhang et al., 2019), y en este caso se usa la versión *sparse categorical cross-entropy* de la librería Keras, que debido a los datos utilizados en esta investigación es más conveniente y ahorra cómputos previos en comparación con la *categorical cross-entropy*. Se entrenó el modelo durante dos épocas y se obtuvo una pérdida de 0,25 y 0,32 para los sets de entrenamiento y validación respectivamente. Fue necesario entrenar por una muy acotada cantidad de épocas debido al rápido sobre entrenamiento que presentó este modelo en particular. En el set

de testeo se obtuvo un *accuracy* y *balanced accuracy* de 85%, similar el resto de los sets que se pueden observar en la Tabla 2 al final de la sección.

En la Figura 2 se muestran las matrices para cada conjunto de datos de entrenamiento, validación y testeo en colores azul, rojo y verde respectivamente. Si bien el set de entrenamiento recibe una cantidad mucho mayor de elementos, la distribución de predicción por clases es similar a los sets de validación y testeo. También, es destacable que, a pesar de que la clase *0* se encuentre sobre representada respecto a la clase *1* en los sets de datos, el modelo no aprende a sobre representarla en la predicción. El hecho de que la red falle más al asignar correctamente la clase *0*, puede indicar que un conjunto de imágenes con etiqueta *0* comparte características muy similares a imágenes con etiqueta 1.

Figura 2: Matrices de confusión modelo binario



5.3 **Clasificación de cuatro categorías**

Las categorías de este modelo se generaron intentando que la distribución de datos por clase sea lo más balanceada posible y que al mismo tiempo, las categorías resulten significativamente distintas y resulten útiles para analizar localización laboral. Las categorías son las siguientes:

- ***0***: 0 trabajos
- ***1***: Entre 1 y 10 trabajos
- ***2***: Entre 11 y 50 trabajos
- ***3***: 51 trabajos o más

Esta división de categorías con la cantidad de trabajos y la densidad que representan sobre áreas de 91x91 metros consideran focos muy distintos de agrupación laboral y por lo tanto, atracciones de viajes razonablemente distintas.

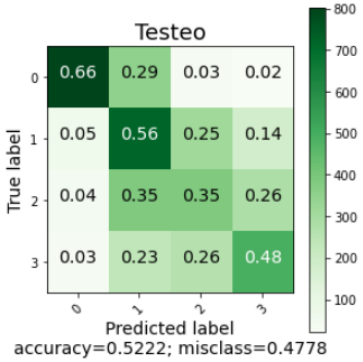
Para hacer un set de datos con clases equilibradas se usaron 12.000 imágenes con trabajos, es decir, con categoría igual a *0*, seleccionadas al azar. Como resultado, la clase *0* tiene una proporción de 25% en el conjunto de datos, la clase *1* un 28%, la clase *2* un 24% y la clase *3* un 21%.

La red utilizada es similar a la del modelo binario, con la diferencia de que en esta se añade dropout de 0,2 después de la unión de las capas paralelas. A diferencia del modelo anterior, para entrenar este modelo no se utilizó todo el conjunto de datos. Para conseguir un set de cuatro categorías balanceado se filtraron las imágenes con 0 trabajos. Con esto se disminuyó la totalidad de datos para entrenar el modelo y esto generó un sobre entrenamiento más acelerado, lo que se compensó

con el uso de dropout. Las características del entrenamiento son las mismas al modelo anterior con respecto a funciones de activación, optimizador y función de pérdida. Se entrenó la red durante siete épocas llegando a un *accuracy* y *balanced accuracy* en el set de testeo es de 52% para ambas métricas.

Al observar la matriz de confusión (Figura 3), se puede ver que en casi todas las categorías el modelo predijo con mayor proporción correctamente, exceptuando la categoría *2*, que confundió fuertemente con la categoría *1*. Se observa que las categorías *1, 2* y *3* son rara vez mal clasificadas en la categoría *0*, lo que es un posible indicador de que las imágenes con trabajos son visualmente distinguibles de las imágenes sin. Por otro lado, la clase *0* fue la clase con mayor proporción de predicciones correctas, pero con una proporción de 29% etiquetadas erróneamente en la clase *1*. Esto tiene similitudes con lo ocurrido en el modelo binario, pero con el lado positivo de que se etiquetó equivocadamente casi exclusivamente en la categoría más cercana.

Figura 3: Matriz de confusión modelo de 4 clases



5.4 **Clasificación de diez categorías**

De manera similar al modelo anterior, las categorías de este modelo se dividieron intentando tener tamaños balanceados de cada una. En la Tabla 1 se muestran las categorías con su cantidad de imágenes asociadas y su porcentaje equivalente el set de datos. Debido a la cantidad de categorías y distribución de los datos no fue posible hacer un set de datos balanceado, es por esto que en este modelo se asignaron pesos específicos durante el entrenamiento a cada clase para compensar la sobre o subrepresentación.

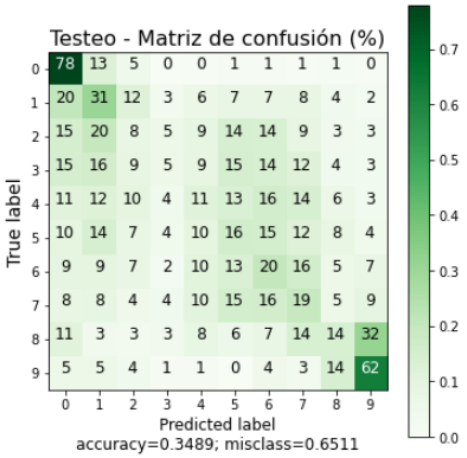
Tabla 1: Categorías modelo de 10 clases

Categoría	Rangos de trabajos	Cantidad de imágenes	Porcentaje	Peso
0	[0]	12.000	26%	0,38
1	[1, 5]	9.199	20%	0,50
2	[6, 10]	3.911	8%	1,18
3	[11, 20]	4.668	10%	0,99
4	[21, 30]	2.780	6%	1,66
5	[31, 50]	3.396	7%	1,36
6	[51, 100]	3.941	9%	1,17
7	[101, 200]	3.157	7%	1,46

8	[201, 500]	1.902	4%	2,42
9	[500, …[	1.140	2%	4,04

Se entrenó la red durante dos épocas llegando a una pérdida de 2,03 y 1,86 para el set de entrenamiento y validación respectivamente. En el conjunto de prueba se obtiene un *accuracy* y *balanced accuracy* de 35% y 27% respectivamente. A niveles generales la precisión lograda es baja, pero considerando que son diez clases posibles, el modelo es un 25% mejor que una selección completamente aleatoria. En ese sentido, es más preocupante la diferencia entre las métricas de *accuracy* y *balanced accuracy* ya que esto indica que el modelo predice diferenciadamente mejor algunas categorías sobre otras. Esto se puede confirmar al observar la matriz de confusión en la figura 4. El modelo logra predecir relativamente bien las clases más extremas, asignando correctamente el 78% de la categoría *0* y el 62% de la categoría *9*, pero erra fuertemente en las categorías intermedias.

Figura 4: Matriz de confusión modelo 10 clases



Como el modelo tiene una gran cantidad de clases, se calcularon métricas de *top k accuracy* para valores de k iguales a 2, 3, 4 y 5, con la finalidad de observar la precisión del modelo entregando mayor holgura. Los resultados de estas métricas se pueden ver en la Tabla 2. Se destaca el resultado de *top 3 accuracy* que alcanza un 60% en el set de testeo, que además en un análisis más detallado se obtuvo que el 66% de las primeras tres predicciones son clases contiguas, indicando un aprendizaje consecuente.

Tabla 2: Resultados modelos de clasificación

Ent: Entrenamiento, Val: Validación, Test: Testeo									
	Binario			**4 Clases**			**10 Clases**		
	Ent	**Val**	**Test**	**Ent**	**Val**	**Test**	**Ent**	**Val**	**Test**
F1-score	0.86	0.85	0.85	0.55	0.51	0.53	0.33	0.32	0.33
Accuracy	85.5%	84.9%	84.6%	54.7%	50.3%	52.2%	35.1%	33.8%	34.8%
Balanced accuracy	86.2%	85.5%	85.4%	54.0%	49.6%	51.6%	26.8%	26.5%	26.5%
Top k=2 accuracy	-	-	-	80.9%	78.4%	79.4%	49.4%	48.1%	49.3%

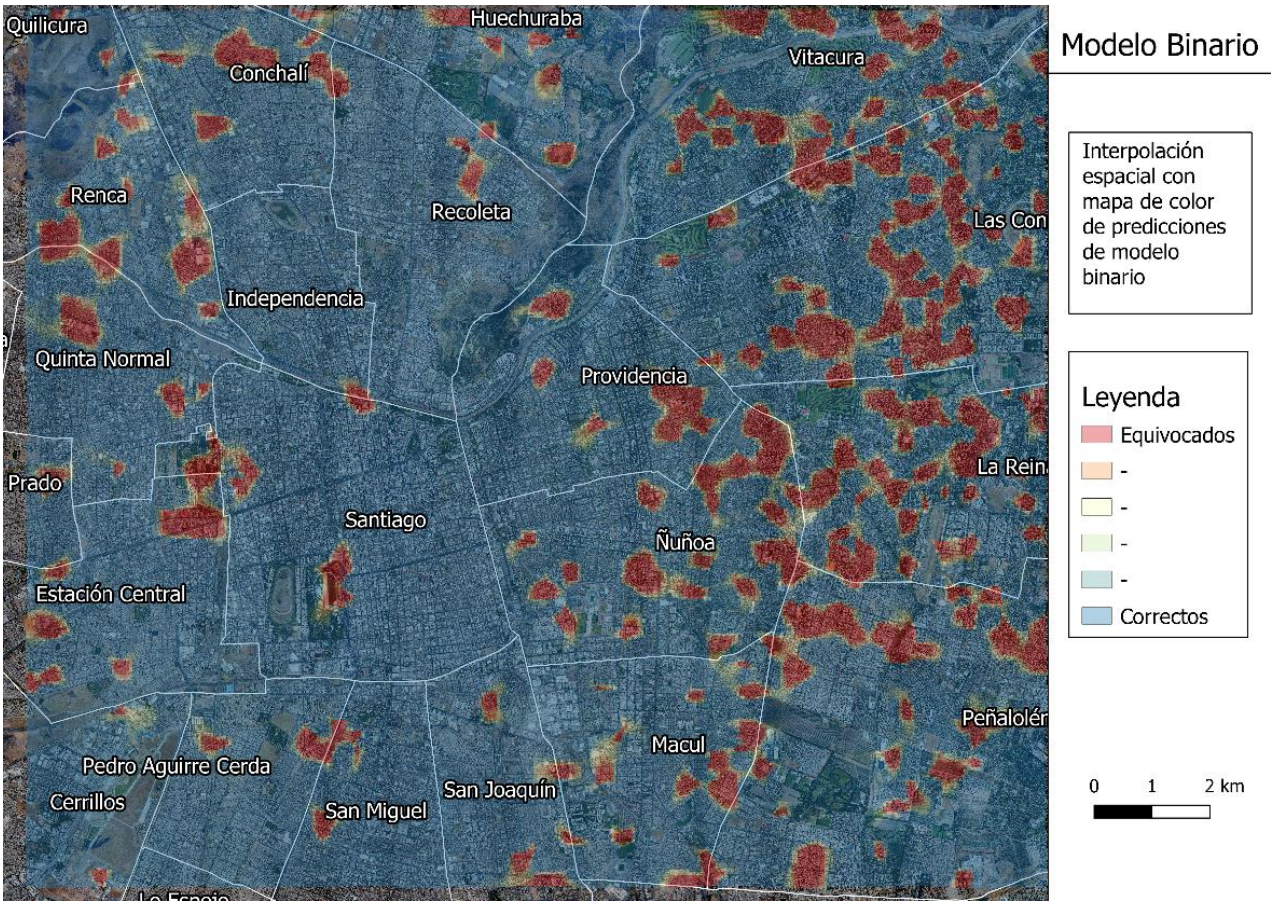
Top k=3 accuracy	-	-	-	-	-	-	59.8%	58.2%	59.5%
Top k=4 accuracy	-	-	-	-	-	-	68.8%	67.4%	68.7%
Top k=5 accuracy	-	-	-	-	-	-	76.8%	75.8%	76.6%

6 ANÁLISIS DE RESULTADOS, CONCLUSIONES Y DISCUSIÓN

6.1 Análisis de resultados

Graficando en el mapa las predicciones sobre el set de testeo del modelo binario, se pueden realizar análisis de los errores que comete la red. En la Figura 5 se muestra un mapa que interpola espacialmente los resultados de las predicciones con rojo donde el modelo predice erróneamente y en azul correctamente. Fue necesaria una interpolación, ya que el set de testeo corresponde a un 10% del total de datos y observarlos individualmente no permitía una clara interpretación. El mapa presenta claras áreas rojas donde el modelo falló para predecir correctamente, con una concentración pronunciada de áreas rojas en el sector oriente. Por la ubicación de las áreas, es posible que parte de los errores se deban el modelo no logre distinguir imágenes con presencia laboral que visualmente parecen viviendas.

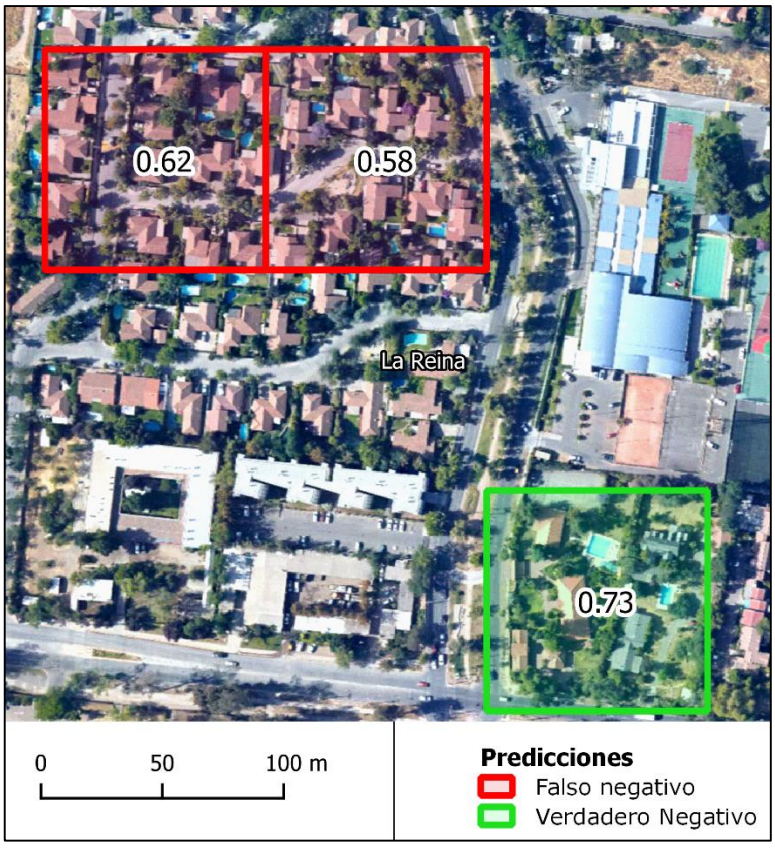
Figura 5: Mapa de errores interpolados de modelo binario



En la Figura 6 se muestra un área en la comuna de La Reina donde hay tres recuadros correspondientes a las predicciones sobre el set de testeo. Dos rojos, que representan errores de predicción donde el modelo predijo que no hay trabajos cuando si había y uno verde, donde el modelo predijo correctamente que no había trabajos. Visualmente (al ojo humano), las tres imágenes tienen apariencia habitacional, además, las tres tienen exclusivamente uso de suelo habitacional, según su vector asociado. En este caso, el error incurrido por la red se ve asociado a que los datos entregados pueden ser muy similares en apariencia otros datos con una clasificación diferente.

Si bien a simple vista, las dos imágenes predichas erróneamente tienen apariencia principalmente habitacional, hay algo de positivo en cuanto a la interpretación de la red. La capa final de la red asigna una probabilidad a cada categoría y en este caso ambas imágenes con predicción errónea fueron asignadas a la categoría 0 con aproximadamente 60% de probabilidad, mientras que la imagen predicha correctamente fue asignada con un 73% de probabilidad. Esto indica que, si bien la predicción final no fue correcta, la red no las asignó con alta seguridad y al contrario, cuando sí predijo correctamente, tuvo mayor seguridad.

Figura 6: Área de análisis con cuadros de predicción de modelo binario



6.2 **Conclusiones**

Los resultados experimentales de los modelos de clasificación muestran que la metodología implementando CNNs sobre imágenes satelitales es capaz hasta cierto punto de interpretar

características visuales e información de uso de suelo para predecir presencia de actividad laboral. El modelo binario presenta un aporte de un 35% de predictibilidad sobre un sorteo totalmente aleatorio, a su vez los modelos de cuatro y diez clases presentan un aporte de 27% y 25% respectivamente.

El análisis final más detallado de los resultados indica una de las principales dificultades que enfrenta la metodología, en poder diferenciar correctamente imágenes que visualmente tienen características muy similares, pero etiquetas diferentes. Esto se relaciona fuertemente con el desafío, que a criterio de los autores es el más importante y perjudicial para los resultados, correspondiente a la falta de acceso a una mejor calidad de datos, principalmente la base de datos de localización de puestos de laborales. Los datos iniciales son uno de los principales factores que se recomienda tener en cuenta en trabajos futuros y además se sugiere complementar con vectores de uso de suelo más detallados, que estén recabados a nivel predial, a diferencia de manzanas.

6.3 **Discusión**

Debido al contenido interdisciplinar de esta investigación se considera relevante dedicar una sección discusión con la finalidad de integrar los resultados obtenidos con la ingeniería de transporte y los aportes a la disciplina.

Los autores consideran dos puntos principales:

- En primer lugar, los resultados del modelo binario entregan una alta precisión de predicción facilitando la detección de sectores atractores de trabajos con un nivel de detalle muy alto. La metodología presenta mayores dificultades para predecir correctamente cuando hay actividades laborales en sectores habitacionales, sin embargo, aún así logra identificar una porción de estas imágenes. La interpretación de características visuales de este tipo sobre la ciudad permite ayudar a identificar potenciales puntos de atracción laboral de forma automatizada.

- En segundo lugar, este trabajo se presenta como un puntapié inicial que abre la rama de investigación de modelos de CNNs sobre imágenes satelitales para la predicción de localización de trabajos, como aporte en planificación de transporte. Se espera que investigaciones futuras logren progresivamente mejores resultados a medida que se obtengan mejores fuentes de datos y avances en metodologías de *deep learning* y CNNs.

7 REFERENCIAS

Albert, A., Kaur, J. and Gonzalez, M. C. (2017) 'Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale', *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Part F1296, pp. 1357–1366. doi: 10.1145/3097983.3098070.

Chen, J. *et al.* (2019) 'Deep Learning from Multiple Crowds: A Case Study of Humanitarian Mapping', *IEEE Transactions on Geoscience and Remote Sensing*. IEEE, 57(3), pp. 1713–1722. doi: 10.1109/TGRS.2018.2868748.

Christie, G. *et al.* (2018) 'Functional Map of the World', *Proceedings of the IEEE Computer*

Society Conference on Computer Vision and Pattern Recognition, 2, pp. 6172–6180. doi: 10.1109/CVPR.2018.00646.
Hadzic, A. *et al.* (2020) 'Estimating Displaced Populations from Overhead', pp. 1–4. Available at: http://arxiv.org/abs/2006.14547.
Ioffe, S. and Szegedy, C. (2015) 'Batch normalization: Accelerating deep network training by reducing internal covariate shift', *32nd International Conference on Machine Learning, ICML 2015*, 1, pp. 448–456.
Jeon, H., Lee, J. and Sohn, K. (2018) 'Artificial intelligence for traffic signal control based solely on video images', *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 22(5), pp. 433–445. doi: 10.1080/15472450.2017.1394192.
Kingma, D. P. and Ba, J. L. (2015) 'Adam: A method for stochastic optimization', *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15.
Lecun, Y., Bengio, Y. and Hinton, G. (2015) 'Deep learning', *Nature*, 521(7553), pp. 436–444. doi: 10.1038/nature14539.
Liu, Q. *et al.* (2018) 'Learning multiscale deep features for high-resolution satellite image scene classification', *IEEE Transactions on Geoscience and Remote Sensing*. IEEE, 56(1), pp. 117–126. doi: 10.1109/TGRS.2017.2743243.
Ministerio de Economía (2015) *Fotografía aérea del Gran Santiago, Catálogo Nacional de Información Geoespacial*. Available at: http://www.ide.cl/descargas/capas/economia/Fotografia-aerea-Gran-Santiago.rar.
Napiorkowska, M., Petit, D. and Martí, P. (2018) 'Three applications of deep learning algorithms for object detection in satellite imagery', in *International Geoscience and Remote Sensing Symposium (IGARSS)*. Institute of Electrical and Electronics Engineers Inc., pp. 4839–4842. doi: 10.1109/IGARSS.2018.8518102.
Pritt, M. and Chern, G. (2018) 'Satellite image classification with deep learning', *Proceedings - Applied Imagery Pattern Recognition Workshop*. IEEE, 2017-Octob, pp. 1–7. doi: 10.1109/AIPR.2017.8457969.
Quinn, J. A. *et al.* (2018) 'Humanitarian applications of machine learning with remote-sensing data: Review and case study in refugee settlement mapping', *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2128). doi: 10.1098/rsta.2017.0363.
Weber, R. *et al.* (2016) *A Spatial Analysis of City-Regions: Urban Form & Service Accessibility, Nordregio Working Paper*.
Wurm, M. *et al.* (2019) 'Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks', *ISPRS Journal of Photogrammetry and Remote Sensing*. doi: 10.1016/j.isprsjprs.2019.02.006.
Zhang, A. *et al.* (2019) *Dive Into Deep Learning, Unpublished Draft*. doi: 10.1016/j.jacr.2020.02.005.