



Tarea 2: reconocimiento visual urbano explicable

Introducción

En esta tarea tendrá la oportunidad de experimentar con el uso de redes neuronales convolucionales para el procesamiento de datos visuales y de evaluar técnicas de explicabilidad para estas. En particular, deberá entrenar CNNs para hacer regresión o clasificación de la percepción visual de imágenes urbanas, para luego interpretar, a través de técnicas de explicabilidad, los elementos visuales que más influyen en los resultados. Para el desarrollo se utilizará el framework Pytorch sobre la plataforma [Colab](#) de Google. Dado que existe abundante material disponible en línea para colaborar con el desarrollo de la tarea, se espera que todo recurso externo utilizado, sea este código, librerías o papers, esté debidamente indicado.

Set de datos

La fuentes primaria de datos será un conjunto de más de 100.000 imágenes de la ciudad de Santiago, extraídas de la plataforma Google Street View. Para cada una de estas imágenes, se cuenta con su ubicación espacial (latitud, longitud) y un vector de valores (puntajes/utilidad) para indicadores visuales asociados a estas escenas. Específicamente, estos datos capturan variables continuas asociadas a la percepción visual de barrios urbanos, especificada a partir de 6 atributos distintos: bello, deprimente, aburrido, seguro, adinerado, animado. Todas las imágenes cuentan con la misma resolución y con los 6 atributos. Más detalles sobre la metodología utilizada para construir esta base de dato pueden leerse [acá](#).

Para crear conjuntos de entrenamiento, validación y test independientes, utilice las funcionalidades de `scikit-learn`, como han sido indicadas en los ejemplos de código.

Modelos

Para esta tarea, debe utilizar modelos de redes convolucionales profundas como los descritos en el capítulo 3 del curso, sin limitante en relación al tipo de capas que se puede utilizar (convolucional, densa, *dropout*, *batchnorm*, etc). Se recomienda revisar bibliografía relacionada para esto. Considere además preprocesar las entradas de acuerdo a lo utilizado en la literatura (por ej., restar la media, normalizar en el rango $[0,1]$, etc.). No hay problema en basarse completamente en modelos propuestos previamente en la literatura o en tutoriales. En cualquier caso, debe justificar sus elecciones. Tampoco existe problema en utilizar modelos preentrenados, para luego ser adecuados mediante finetuning o transferencia directa.

Para los aspectos de interpretación de los elementos visuales más influyentes, puede utilizar cualquier método propuesto en la literatura. Una lista y revisión bastante completa de los métodos existentes puede ser encontrada [aquí](#). En particular, los más utilizados son LIME, CAM, RAM, Grad-CAM.

Tareas a realizar

Para la tarea se espera se espera que realice al menos las siguientes actividades:

- Entrenamiento: entrene o transfiera un modelo de CNN en los datos indicados anteriormente, de modo que sea capaz de predecir simultáneamente los 6 puntajes asociados a cada imagen. Muestre el rendimiento y su evolución en los conjuntos de entrenamiento, validación y test, y comente sobre el sobreentrenamiento observado.
- Entrenamiento 2: entrene o transfiera un nuevo modelo, pero esta vez para realizar clasificación sobre los atributos perceptuales, en vez de regresión. Para esto, cuantice cada puntaje/utilidad en 5 niveles distintos (muy bajo, bajo, medio, alto, muy alto) de acuerdo a algún criterio definido por ud. Muestre el rendimiento y su evolución en los conjuntos de entrenamiento, validación y test, y comente sobre el sobreentrenamiento observado, además de las diferencias con el caso de regresión.
- Interpretabilidad y visualización: utilice alguna de las técnicas de interpretabilidad mencionadas anteriormente para analizar cuáles son los elementos visuales que más influyen para cada atributo perceptual, y presente visualmente sus resultados. Para esto es libre de utilizar cualquiera de los modelos de los ítems anteriores (no hay que hacerlo con todos, basta uno).

Desarrollo y entrega

La tarea puede desarrollarse de manera individual o en parejas, utilizando el framework Pytorch para Python. Se recomienda utilizar la plataforma Google Colab con el fin de facilitar la instalación de librerías. Esta plataforma permite utilizar gratuitamente una GPU para el entrenamiento por intervalos de 12 horas continuos. En el *notebook* desarrollado debe ir tanto el código como un informe (preferiblemente intercalados), donde se expliquen los pasos realizados, se analicen los resultados y se planteen conclusiones. La entrega de la tarea tiene como fecha límite el viernes 24 de junio a las 23:59, a través del buzón que se habilitará en el sitio del curso. Para fines de corrección, se revisará la última versión entregada.

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.