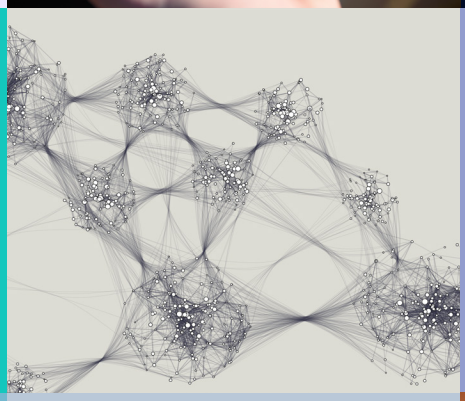
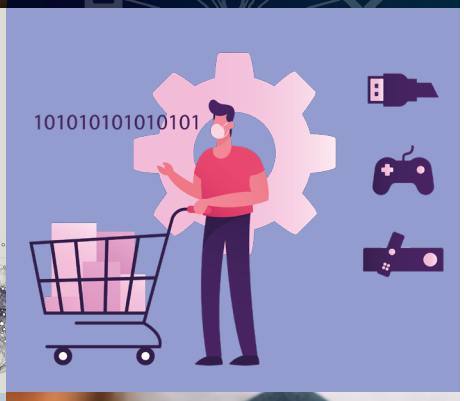
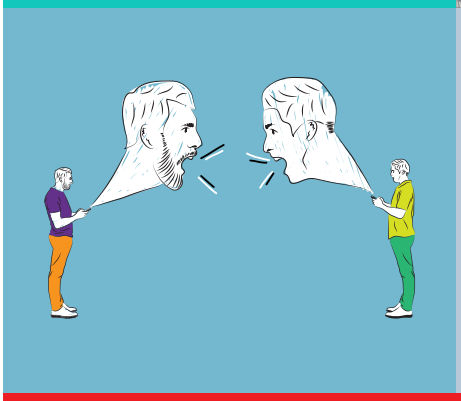
Aplicaciones de la inteligencia artificial







A través de una serie de miniartículos independientes, ilustramos cómo la inteligencia artificial y sus diferentes métodos permiten abordar problemas en una amplia y creciente diversidad de dominios. Por cuestiones de extensión, la enumeración no pretende ser exhaustiva y muchas áreas quedarán pendientes para una futura edición de la Revista.

¿Puede una máquina ver mejor que un humano?

JAVIER CARRASCO Ingeniero Civil en Computación de la Universidad de Chile y egresado del Instituto Milenio Fundamentos de los Datos.

AIDAN HOGAN Profesor Asociado del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador Asociado del Instituto Milenio Fundamentos de los Datos.

JORGE PÉREZ Profesor Asociado del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador Asociado del Instituto Milenio Fundamentos de los Datos.

La última década ha sido testigo de avances extraordinarios en el área de la inteligencia artificial, impulsados, en particular, por el concepto de redes neuronales profundas, combinado con la disponibilidad de enormes cantidades de datos para entrenar estas redes. Entre las subáreas de la computación que se han beneficiado con esta tecnología, podemos destacar, por ejemplo, la visión computacional, y la tarea específica de reconocimiento de imágenes. En esta tarea, la máquina recibe una imagen de un objeto y tiene que devolver la clase de ese objeto, diciendo, por ejemplo, que la imagen representa un perro, una flor, una taza, etc.

El conjunto de datos más usado para entrenar y evaluar métodos de reconocimiento de imágenes se llama ImageNet; contiene millones de imágenes etiquetadas según mil clases distintas. Según Russakovsky et al. [1], un experto humano puede lograr una tasa de error (top-5) de 5,1% en un subconjunto de 1.500 imágenes de ImageNet. En la misma tarea, una red neuronal profunda del estado del arte (SeNetResNet50 [2]) puede lograr una tasa de error (top-5) de 2,3%, es decir que tiene mejor rendimiento que un humano experto en esta tarea. ¿Este resultado significa que las máquinas, ahora, pueden "ver" mejor que los humanos? No necesariamente, pues es una pregunta multifacética. En esta tarea, las clases son muy finas, e incluyen ejemplos como un *cucal*, un *Sealyham terrier*, etc., que pueden ser difíciles de recordar y distinguir para un humano. También, la tarea siempre considera imágenes de calidad total. Entonces surge una duda: si las imágenes tuvieran menos calidad que las vistas en los ejemplos de entrenamiento, ¿cómo afectaría el rendimiento de las máquinas y de los humanos? ¿Los humanos necesitan más o menos información para poder clasificar una imagen correctamente en comparación con las máquinas? ¿Qué tipo de información les importa más?

Imágenes mínimas positivas

Para poder entender y comparar la dependencia que las máquinas y los humanos tienen para poder clasificar bien una imagen, definimos el concepto de una *imagen mínima positiva* [3]: dada una imagen etiquetada con su clase, y un clasificador de imágenes, la imagen mínima positiva es la versión de la imagen con la peor calidad tal que el clasificador siga dando la clase correcta. Con respecto a la calidad de la imagen, hablamos más específicamente de

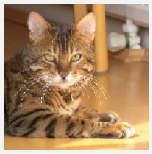
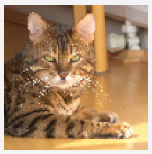
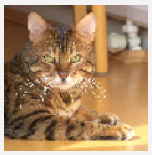
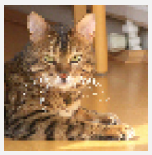
Modelo	Color	Resolución	Zona	Combinación
SqueezeNet				
GoogLeNet				
ResNet50				
SeNetResNet50				
Humano				

Figura 1. Imágenes mínimas positivas para un gato.

la información que contiene, medida usando el tamaño de la imagen comprimida (sin pérdida; usamos compresión de PNG). Se pueden considerar varias formas de reducción de imágenes; en nuestro trabajo, hemos considerado las reducciones de color, de resolución, de zona, y la combinación de las tres. La tabla de la Figura 1 ejemplifica las imágenes mínimas para una imagen de un gato, tal que el modelo (clasificador) indicado puede reconocer que la imagen es de un gato, pero con más reducción, no puede más.

Para calcular las imágenes mínimas en el caso de las máquinas, tomamos una imagen de prueba (no vista antes durante el proceso de entrenamiento), e implementamos una búsqueda sobre los parámetros de reducción, empezando con la imagen completa, y reduciendo la información hasta que se encuentre la imagen mínima. Para calcular las imágenes mínimas en el caso de las máquinas, no se puede usar la misma estrategia, pues el humano recordará la clase de la imagen completa. Así que diseñamos una interfaz que empieza con la imagen "nula" (con una reducción completa), tal que el humano pueda aumentar la información hasta que pueda reconocer el objeto de la imagen y clasificarla (si la clasificación es incorrecta, descartamos la imagen y pasamos a la próxima).

Experimentos y resultados

Para ver qué tan sensibles son los clasificadores frente a la pérdida de diferentes tipos de información, hicimos experimentos con 20 clases simplificadas de ImageNet, tomando 15 imágenes para cada clase. Tomamos cuatro modelos que usan redes neuronales profundas, que han logrado el mejor resultado sobre ImageNet en algún momento, y que han sido entrenados con las imágenes (completas) de entrenamiento de ImageNet. Los cuatro modelos, en orden de su rendimiento sobre ImageNet, son SqueezeNet, GoogLeNet, ResNet50, y SeNetResNet50. Se pueden ver ejemplos de las imágenes mínimas de cada modelo en la Figura 1 considerando varias formas de reducción.

Luego medimos la proporción de reducción para las imágenes mínimas positivas como el cociente entre el tamaño de la imagen original y la imagen mínima positiva (ambas comprimidas con PNG). Un menor cociente significa que el modelo es más robusto a la pérdida de información correspondiente. En la Figura 2, podemos ver los resultados, presentados como un diagrama de caja. Se puede ver que los humanos son mejores para clasificar imágenes con menos colores y resolución, pero que las máquinas pueden clasificar las imágenes basado en zonas más pequeñas. Estos resultados apoyan la observación de Geirhos *et al.* [4] de que la textura de la imagen es una característica importante para las redes neuronales profundas, las cuales pueden diferenciar, por ejemplo, entre el pelo de un gato y un perro. Por eso sólo necesitan una zona pequeña de una imagen, pero sufren más con una pérdida de resolución o color. Otra observación es que los modelos más robustos frente a la pérdida de información también tienen mejor rendimiento para las imágenes completas.

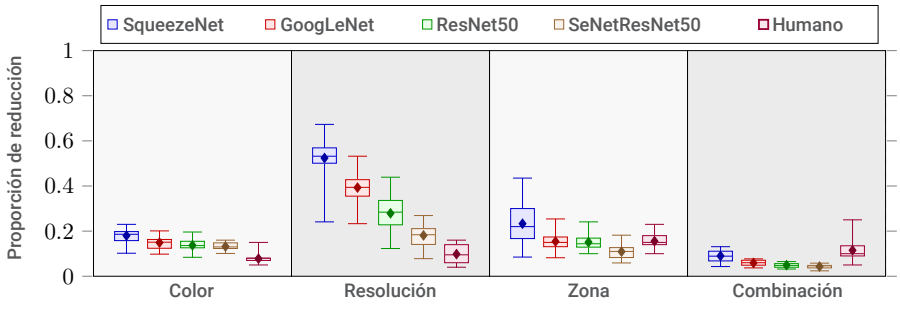


Figura 2. Proporción de reducción para las imágenes mínimas positivas.

Finalmente, hicimos un experimento usando cada clasificador para clasificar las imágenes mínimas positivas de los otros clasificadores. Se pueden encontrar los resultados completos en nuestro artículo [3]. En resumen, observamos que los humanos pueden clasificar mejor las imágenes mínimas positivas de las máquinas que al revés, logrando una precisión de 0,89-0,92 para color, 0,86-0,93 para resolución, 0,76-0,87 para zona, y 0,74-0,85 para combinación, con mejor precisión para las imágenes mínimas positivas, respectivamente, de SqueezeNet (más fáciles), GoogLeNet, ResNet50, y SeNetResNet50 (más difíciles). Al revés, clasificando las imágenes mínimas positivas de los humanos, los modelos de máquina lograron una precisión de 0,14-0,42 para color, 0,03-0,29 para resolución, 0,11-0,42 para zona, y 0,07-0,35 para combinación; los mejores modelos fueron, respectivamente, SeNetResNet50 (mayor precisión), ResNet50, GoogLeNet y SqueezeNet (menor precisión).

Conclusiones

¿Puede una máquina ver mejor que un humano? Es una pregunta cada vez más compleja, que puede ser interpretada de varias formas. En la Clasificación de Imágenes, nuestros resultados han indicado que los humanos proveen resultados más robustos frente a la pérdida de información. En la práctica, esto implica que los resultados dados por las redes neuronales profundas entrenadas y evaluadas en el contexto de conjuntos de imágenes completas pueden no aplicarse a condiciones reales, en las cuales un objeto (por ejemplo, una cara) está parcialmente oculto, o está a distancia, o iluminado parcialmente, etc.

Una pregunta que nos interesa ahora, entonces, es la siguiente: ¿se puede mejorar la robustez de los clasificadores de máquinas frente a la pérdida de información? Los modelos que usamos en este trabajo fueron entrenados sobre imágenes completas. Quizás se puedan entrenar las redes con imágenes reducidas o mínimas, para mejorar su robustez en situaciones de información parcial. ■

REFERENCIAS

[1] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, y Fei-Fei Li. 2015. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision 115, 3 (2015), 211–252.

[2] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, y Enhua Wu. 2019. Squeeze-andExcitation Networks. arXiV:1709.01507v4.

[3] Javier Carrasco, Aidan Hogan y Jorge Pérez. 2020. Laconic Image Classification: Human vs. Machine Performance. En el acta de la International Conference on Information and Knowledge Management (CIKM), Galway, Ireland, [Online], October 19–23, 2020.

[4] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, y Wieland Brendel. 2019. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. En el acta de la International Conference on Learning Representations (ICLR). OpenReview.net.

Procesamiento de Lenguaje Natural: dónde estamos y qué estamos haciendo

FELIPE BRAVO-MÁRQUEZ Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador Joven del Instituto Milenio Fundamentos de los Datos.

JOCELYN DUNSTAN Profesora Asistente de la Iniciativa de Datos e Inteligencia Artificial de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile e Investigadora del Centro de Modelamiento Matemático.

El Procesamiento de Lenguaje Natural (PLN) es una rama de la Inteligencia Artificial (IA) centrada en el diseño de métodos y algoritmos que toman como entrada o producen como salida datos en la forma de lenguaje humano [1]. Esto puede venir en forma de texto o audio, y una vez que el audio es transcrito, ambos tipos de datos tienen un análisis común.

Tal como argumentan Julia Hirschberg y Chris Manning [2], tareas actuales donde el PLN entra en nuestras vidas son la traducción automática, los sistemas de pregunta-respuesta y la minería de texto en redes sociales. Ahondemos en la primera de ellas: la Web está en su mayoría en inglés, y el poder traducir páginas en forma casi instantánea es algo extraordinario. Traducir un texto no es fácil pues no hay una biyección entre palabras en ambos lenguajes, sino que una frase puede requerir menos palabras en un idioma que en otro (pensar por ejemplo traducir del español al inglés). Pero además, la traducción de una palabra requiere información del contexto en la que aparece para saber el sentido en la que se está usando. Asimismo, puede ocurrir que la palabra no tenga sentido en sí misma sino que en conjunto con la palabra que la acompaña (piense en las *phrasal verbs* del inglés). Actualmente los traductores automáticos usados por Google o DeepL están basados en sofisticadas redes neuronales.

PLN suele confundirse con otra disciplina hermana llamada Lingüística Computacional (LC). Si bien ambas están estrechamente relacionadas, tienen un foco distinto. La LC busca responder preguntas fundamentales sobre el lenguaje mediante el uso de la computación, es decir, cómo entendemos el lenguaje, cómo producimos lenguaje o cómo aprendemos lenguaje. Mientras que en PLN el foco está en resolver

problemas específicos, tales como las transcripción automática del habla, la traducción automática, la extracción de información de documentos y el análisis de opiniones en redes sociales. Es importante señalar que en PLN, el éxito de una solución se mide en base a métricas concretas (por ejemplo: qué tan similar es la traducción automática a una hecha por un humano) independientemente de si el modelo hace uso de alguna teoría lingüística.

Comprender y producir el lenguaje computacionalmente es extremadamente complejo. La tecnología más exitosa actualmente para abordar PLN es el aprendizaje automático supervisado que consiste en una familia de algoritmos que "aprenden" a construir la respuesta del problema en cuestión en base a encontrar patrones en datos de entrenamiento etiquetados.[1] Por ejemplo, si queremos tener un modelo que nos diga si un *tweet* tiene un sentimiento positivo o negativo respecto a un producto, primero necesitamos etiquetar manualmente un conjunto de *tweets* con su sentimiento asociado. Luego debemos entrenar un algoritmo de aprendizaje sobre estos datos para poder predecir de manera automática el sentimiento asociado a *tweets* desconocidos. Como se podrán imaginar, el etiquetado de datos es una parte fundamental de la solución y puede ser un proceso muy costoso, especialmente cuando se requiere conocimiento especializado para definir la etiqueta.

Los orígenes de PLN se remontan a los años cincuenta con el famoso test de Alan Turing: una máquina será considerada inteligente cuando sea capaz de conversar con una persona sin que ésta pueda determinar si está hablando con una máquina o un ser humano. A lo largo de su historia la disciplina ha tenido tres grandes periodos: 1) el racionalismo, 2) el empirismo, y 3) el aprendizaje profundo [3] que describimos a continuación.

El racionalismo abarca desde 1950 a 1990, donde las soluciones consistían en diseñar reglas manuales para incorporar mecanismos de conocimiento y razonamiento. Un ejemplo emblemático es el agente de conversación (o *chatbot*) ELIZA desarrollado por Joseph Weizenbaum que simulaba un psicoterapeuta rogeriano. Luego, a partir de la década de los noventa, el diseño de métodos estadísticos y de aprendizaje automático construidos sobre corpus llevan a PLN hacia un enfoque empirista. Las reglas ya no se construyen sino que se "aprenden" a partir de datos etiquetados. Algunos modelos representativos de esta época son los filtros de *spam* basados en modelos lineales, las cadenas de Markov ocultas para la extracción de categorías sintácticas y los modelos probabilísticos de IBM para la traducción automática. Estos modelos se caracterizaban por ser poco profundos en su estructura de parámetros y por depender de características manualmente diseñadas para representar la entrada.[2]

A partir del año 2010, las redes neuronales artificiales, que son una familia de modelos de aprendizaje automático, comienzan a mostrar resultados muy superiores en varias tareas emblemáticas de PLN [4]. La idea de estos modelos es representar la entrada (el texto) con una jerarquía de parámetros (o capas) que permiten encontrar representaciones idóneas para la tarea en cuestión, proceso al cual se refiere como "aprendizaje profundo". Estos modelos se caracterizan por tener muchos más parámetros que los modelos anteriores (superando la barrera del millón en algunos casos) y requerir grandes volúmenes de datos para su entrenamiento. Una gracia de estos modelos es que pueden ser preentrenados con texto no etiquetado como libros, Wikipedia, texto de redes sociales y de la Web para encontrar representaciones iniciales de palabras y oraciones (a lo que conocemos como *word embeddings*), las cuales pueden ser posteriormente adaptadas para la tarea objetivo donde sí se tienen datos etiquetados (proceso conocido como *transfer learning*). Aquí destacamos modelos como Word2Vec [5], BERT [6] y GPT-3 [7].

Este tipo de modelos ha ido perfeccionándose en los últimos años, llegando a obtener resultados cada vez mejores para casi todos los problemas del área [8]. Sin embargo, este progreso no ha sido libre de controversias. El aumento exponencial en la cantidad de parámetros[3] de cada nuevo modelo respecto a su predecesor, hace que los recursos computacionales y energéticos necesarios para construirlos sólo estén al alcance de unos pocos. Además, varios estudios han mostrado que estos modelos aprenden y reproducen los sesgos y prejuicios (por ejemplo: género, religión, racial) presentes en los textos a partir de los cuales se entrenan. Sin ir más lejos, la investigadora Timmnit Gebru fue despedida de Google cuando se le negó el permiso para publicar un artículo que ponía de manifiesto estos problemas [9].

1 | En PLN se le suele llamar a estos conjuntos de datos textuales (etiquetados o no etiquetados) como "corpus".

2 | La mayor parte de algoritmos de aprendizaje operan sobre vectores numéricos, donde cada columna es una característica del objeto a modelar. En PLN esas características pueden ser las palabras de una oración, las frases u otra propiedad (por ejemplo: el número de palabras con mayúsculas, la cantidad de emojis en un *tweet*, etc.).

3 | Word2Vec [5] tiene del orden de cientos de parámetros, BERT [6] tiene 335 millones de parámetros y GPT-3 [7] tiene 175 mil millones de parámetros.

Representations for Learning and Language (ReLeLa)[4] es un grupo de investigación del Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile, donde también participan académicos y estudiantes de otros departamentos y centros. Sus miembros investigan varios temas en PLN: análisis de sentimiento y emociones en redes sociales, texto clínico, educación, textos legales, lenguas indígenas y el análisis de argumentos políticos.

Una línea de ReLeLa liderada por Jorge Pérez, ha sido el desarrollo de modelos preentrenados para el idioma español. Una contribución destacada ha sido BETO[5], la versión en español de BERT, que es ampliamente utilizado por investigadores y desarrolladores del mundo hispano.

En el ámbito del texto clínico, la creación de recursos para la extracción de información relevante requiere un trabajo fuertemente interdisciplinario. Recientemente fue presentado en el *workshop clínico de EMNLP*[6] el primer corpus clínico chileno etiquetado y resultados preliminares para el reconocimiento automático de entidades nombradas.

Finalmente, *The Word Embeddings Fairness Evaluation Framework* (WEFE)[7], es una herramienta de código abierto que permite medir y mitigar el sesgo de los modelos preentrenados señalados anteriormente. La principal característica de WEFE es estandarizar los esfuerzos existentes en un marco común para ser libremente utilizado.

A pesar de los grandes avances en los últimos años, aún estamos lejos de responder todas las interrogantes de PLN. En problemas como el diseño de *chatbots* las soluciones del estado del arte aún distan mucho de lo esperado y ni siquiera es claro cómo evaluarlas correctamente, luego para muchos otros problemas del mundo real simplemente no es posible obtener los recursos necesarios (datos etiquetados, hardware) para construir una solución adecuada. En RELELA confluyen visiones provenientes de la computación, las matemáticas, la lingüística y la salud para discutir esas interrogantes y sobre todo para mantenernos al día con los constantes avances del área. Todo esto ocurre en nuestros seminarios semanales donde escuchamos exposiciones de miembros del grupo o de algún charlista invitado. ■

REFERENCIAS

[1] Eisenstein, J. (2018). Natural language processing.

[2] Hirschberg, J., & Manning, C. D. (2015). Advances in natural language processing. Science, 349(6245), 261–266.

[3] Deng, L., & Liu, Y. (Eds.). (2018). *Deep learning in natural language processing*. Springer.

[4] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. Journal of machine learning research, 12(Aug):2493–2537.

[5] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'13).

[6] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 4171–4186.

[7] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., et al. (2020). Language models are few-shot learners. In Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 202.

[8] NLP-progress: Repository to track the progress in Natural Language Processing (NLP), including the datasets and the current state-of-the-art for the most common NLP tasks: http://nlpprogress.com/.

[9] Bender, Emily M., et al. (2021). "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜." *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*.

4 | https://relela.com/.

5 | https://github.com/dccuchile/beto.

6 | https://www.aclweb.org/anthology/2020.clinicalnlp-1.32/.

7 | https://wefe.readthedocs.io/en/latest/.

Inteligencia artificial para restauración de material arqueológico

ALEXIS MENDOZA Estudiante de pregrado de la Escuela de Ciencia de la Computación, Universidad Nacional San Agustín, Perú.
ALEXANDER APAZA Estudiante de pregrado de la Escuela de Ciencia de la Computación, Universidad Nacional San Agustín, Perú.
IVÁN SIPIRÁN Profesor Asistente del Departamento de Ciencias de la Computación, Universidad de Chile.
CRISTIÁN LÓPEZ Profesor Asistente del Departamento de Ingeniería, Universidad de Ingeniería y Tecnología, Perú.

En 2018, el museo Josefina Ramos de Cox en Lima - Perú inició un proceso de digitalización de los objetos arqueológicos que albergan en su colección. El museo administra más de siete mil piezas provenientes de diferentes culturas prehispánicas, principalmente culturas de la costa central del Perú. Para el proceso de digitalización, el museo usó un escáner 3D de escritorio que utiliza tecnología de luz estructurada. Sin embargo, el proceso de digitalización no se desarrolló de forma satisfactoria por dos razones:

1. La mayoría de los objetos eran frágiles y, al no poder sostenerse sobre la base del escáner, se tuvo que colocar bases artificiales. Estas bases artificiales fueron posteriormente removidas en las superficies 3D generadas, dejando grandes porciones de la base de los objetos sin información.

2. El escáner de luz estructurada tiene problemas para escanear superficies cuyo ángulo con respecto al haz de luz es casi perpendicular. Por lo tanto, hay bases de objetos que no fueron correctamente escaneadas por la limitación del escáner.

El problema en la digitalización trajo como consecuencia que un gran número de objetos tengan una superficie incompleta después del escaneo (ver Figura 1). Nosotros propusimos una forma de solucionar el problema de la geometría faltante desde un enfoque basado en datos y usando inteligencia artificial.

Nuestra propuesta

Nuestro método consiste de una red neuronal que recibe un objeto 3D con superficie incompleta y produce el objeto completo reparado. Nuestra premisa es que si contamos con suficientes ejemplos de objetos dañados y objetos completos, la red neuronal puede encontrar una buena correspondencia entre la geometría de la superficie incompleta y

Figura 1. Vista frontal y superior de algunos objetos escaneados. Note la falta de geometría en la base de los objetos.

la superficie de los objetos completos. Además, si seguimos un protocolo de entrenamiento adecuado, podemos esperar que la red neuronal generalice bien a diferentes geometrías faltantes.

El problema es que la colección escaneada del museo Josefina Ramos de Cox no contiene muchos ejemplos de objetos completos, como para permitir hacer un entrenamiento adecuado de una red neuronal. En este punto, hicimos una observación clave para solucionar el problema. Lo que requerimos de la red neuronal es que aprenda la estructura de objetos arqueológicos, por lo que cualquier otro conjunto de datos con estructura similar podría servir para nuestro cometido. Así, logramos recolectar un conjunto de 1458 objetos desde el 3D Pottery Benchmark [1] y las clases "Bowl" y "Jar" del dataset ShapeNet [2]. Todos estos objetos tienen estructura común a objetos arqueológicos y sirvieron para entrenar nuestra red neuronal.

Con respecto a la arquitectura de la red neuronal, típicamente el problema de "*shape completion*" se aborda desde una perspectiva de un modelo tipo *encoder-decoder*, en donde el encoder procesa la geometría de entrada y la transforma en un vector numérico. Posteriormente, el vector numérico es la entrada al decoder, que finalmente reconstruye la geometría completa [3, 4]. Sin embargo, un problema con este tipo de arquitectura es que generan una representación transformada de la geometría completa. En nuestro caso, la geometría de entrada no tiene que ser cambiada ni transformada, y más bien lo que necesitamos es generar una buena representación de la superficie que falta. Es así que nosotros presentamos una nueva arquitectura para este problema específico, en donde una primera red neuronal produce una región faltante candidata. La unión del objeto incompleto y la región candidata es posteriormente refinada con una segunda red neuronal, la cual produce el objeto completo. Ambas redes neuronales son entrenadas en conjunto y en forma *end-to-end*. Para la representación de los modelos 3D, escogimos las nubes de puntos [5]. La arquitectura puede verse en la Figura 2.

Para entrenar este modelo, usamos el conjunto de datos recolectado y

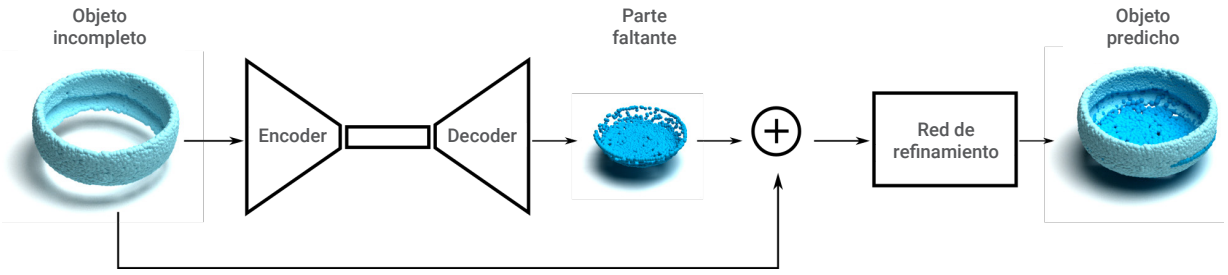


Figura 2. Arquitectura de nuestra red neuronal. El modelo consiste en un *encoder-decoder* para generar la parte faltante a partir del objeto incompleto. Ambos objetos son luego usados por la red de refinamiento para obtener el objeto reparado final.

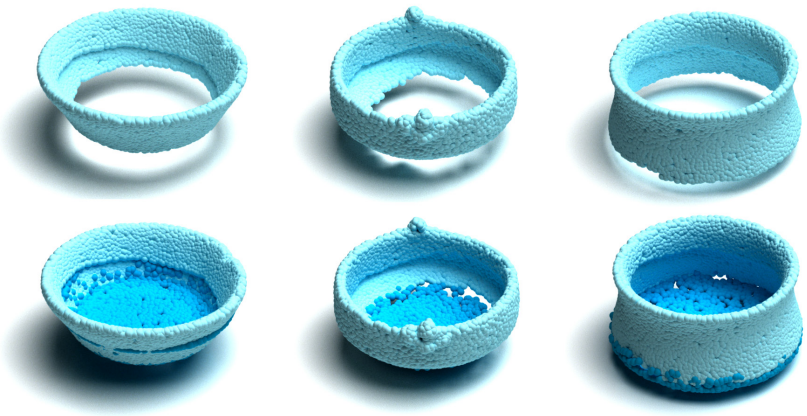


Figura 3. Ejemplos de objetos reparados con nuestra herramienta.

realizamos la generación de pares de entrenamiento (objeto incompleto, objeto completo) durante el mismo entrenamiento. Creamos un protocolo para generar pares aleatorios de objetos, aplicando un algoritmo que simula la eliminación de geometría en la base de un objeto de entrada. Este algoritmo nunca genera dos objetos iguales durante el entrenamiento, por lo que esto garantiza que la red no memorice los ejemplos de entrenamiento.

Una vez que la red fue entrenada, usamos el conjunto de objetos arqueológicos del museo como objetos de prueba. Como la red procesa nubes de puntos, implementamos un algoritmo que reconstruye la superficie de los objetos 3D. La Figura 3 muestra algunos resultados de nuestro método.

Consideraciones finales

Abordamos un problema de restauración de piezas arqueológicas desde una perspectiva de datos. Este trabajo se pudo llevar a cabo gracias a los recientes avances en análisis de formas y procesamiento geométrico a través del uso de técnicas de aprendizaje automático. Nuestros resultados muestran que las redes neuronales que procesan geometría pueden extraer información de estructura de los objetos. Esta estructura puede ser empleada para el diseño asistido por computadora, y específicamente en nuestro caso fue útil para predecir la geometría faltante de objetos con defectos de escaneo. ■

REFERENCIAS

[1] Koutsoudis A., Pavlidis G., Liami V., Tsiafakis D., Chamzas C., "*3D Pottery content-based retrieval based on pose normalisation and segmentation*". Journal of Cultural Heritage, 11(3), pp 329-338, 2010.

[2] Chang A., Funkhouser T., Guibas L., Hanrahan P., Huang Q., Li Z., Savarese S., Savva M., Song S., Su H., Xiao J., Yi L., Yu F., "*ShapeNet: An Information-Rich 3D Model Repository*". CoRR abs/1512.03012. Arxiv, 2015.

[3] Yuan W., Khot T., Held D., Mertz C., Hebert M., "*PCN: Point Completion Network*". In Proc: *International Conference on 3D Vision (3DV)*, pp. 728-737. 2018.

[4] Tchapmi L., Kosaraju V., Rezatofighi H., Reid I., Savarese S., "*TopNet: Structural Point Cloud Decoder*". *In Proc: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 383-392. 2019.

[5] Qi R., Su H., Kaichun M., Guibas L., "*PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*". *In Proc: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 77-85. 2017.

Neuroevolución: ¿Cómo evitar los datos masivos de entrenamiento?

ALEXANDRE BERGEL Profesor Asociado del Departamento de Ciencias de la Computación de la Universidad de Chile.

Contexto

Según la teoría de Darwin, el cerebro de los mamíferos es el resultado de una larga evolución. Frente a cualquier otra especie, los humanos tienen el cerebro más grande en relación a su peso. Hace decenas de milenios, nuestro cerebro no tenía la sofisticación que tiene hoy. El cerebro evolucionó, en parte, para solucionar problemas complejos como la necesidad de los humanos de comunicarse en forma eficiente. Siguiendo un proceso de evolución similar al de nuestro cerebro, la neuroevolución es una técnica de la inteligencia artificial que combina un algoritmo genético con una red neuronal. Su idea central es producir modelos que sean lo suficientemente desarrollados para solucionar un problema que no se puede expresar a través de ejemplos. Es una idea casi opuesta a la forma en que se entrena un modelo con grandes cantidades de imágenes o de texto, como se hace en el área de *deep learning*.

Ejemplo y aplicaciones

Consideren la red neuronal de la Figura 1. Esta red describe el comportamiento del operador booleano *XOR*, usando una función de activación de tipo *step*. Tiene, además, nueve parámetros, tres por cada neurona. Un algoritmo de aprendizaje, como el *backpropagation* usado en *deep learning*, tendrá que deducir estos nueve parámetros desde un conjunto de ejemplos. En este caso, tener ejemplos no representa un problema para entrenar la red, pero en otros casos, según el problema a abordar, tener ejemplos puede representar un lujo que no siempre es alcanzable.

La neuroevolución es una técnica alternativa al *backpropagation* para deducir estos nueve parámetros y consiste en la aplicación de un algoritmo genético con redes neuronales. En vez de entrenar una red usando mecanismos de aprendizaje, la neuroevolución usa un algoritmo evolutivo para buscar los parámetros que generan redes de “mejor calidad”.

Un algoritmo genético es una metáfora computacional del mecanismo de evolución natural, tal como lo describió Charles Darwin. En la naturaleza, los individuos más fuertes tienen mayores probabilidades de sobrevivir y de reproducirse. Aplicado a nuestro ejemplo de

redes naturales, un individuo es una serie de nueve números y la probabilidad de evolucionar depende de la cantidad de errores que comete la red neuronal bajo este individuo.

En cada generación, los individuos más fuertes (i.e., las redes neuronales que cometen menos errores) se combinan usando operaciones genéticas, tales como la mutación y el *cross-over*. La población inicial de individuos está compuesta por series de nueve números aleatorios, pero en cada generación se genera una red mejor, que produce menos errores que en la generación anterior.

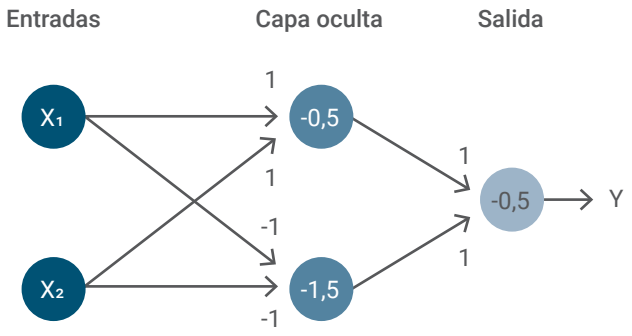


Figura 1. Red neuronal que simula al operador booleano *XOR*.

Algoritmos sofisticados de neuroevolución, como NEAT y HyperNEAT, permiten evolucionar no solamente los parámetros, sino también la topología de la red, algo que no se puede lograr con el *deep learning* clásico.

En el grupo ISCLab[1] del Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile, usamos la neuroevolución para desarrollar inteligencia artificial de videojuego, estilo Mario Bros. La neuroevolución es particularmente conveniente para producir dicho tipo de IA ya que, en comparación al *deep learning*, no requiere datos de jugadas.

Por otro lado, estamos desarrollando técnicas de visualización que permiten caracterizar el proceso de evolución. La neuroevolución, como cualquier otro algoritmo de *machine learning*, es una caja negra que entrega un resultado sin dar cuenta del camino tomado para obtener dicho resultado. Nuestras visualizaciones ayudan a entender las diferentes decisiones tomadas por el algoritmo de neuroevolución, lo que ayuda a explicar su resultado.

Beneficios

La neuroevolución no tiene las limitaciones que imponen un uso de cantidades *masivas* de datos. Un modelo basado en neuroevolución puede superar a un modelo basado en ejemplos producidos por humanos. Ejemplos prominentes de esta situación son la robótica y los videojuegos. Si un jugador virtual tuviese que aprender de los humanos cómo jugar, no lograría superarlos. Pero un algoritmo evolutivo (al que la neuroevolución pertenece) puede superar, y por mucho, a los mejores jugadores del mundo. AlphaGo y Dota2 demuestran la amplia capacidad de los algoritmos evolutivos para superar a los humanos.

El artículo "Designing neural networks through neuroevolution", publicado en 2019 en la revista Nature Machine Intelligence, describe los últimos progresos en el área de la neuroevolución. Además de presentar una retrospectiva de cómo la naturaleza y la evolución del cerebro han tenido un enorme impacto en el área de la inteligencia artificial, este artículo describe una extraordinaria forma de acercarse a una inteligencia artificial genérica. Ahora, es reconocido que la neuroevolución es competidora de las técnicas modernas usadas en aprendizaje supervisado, al que pertenecen las técnicas de aprendizaje de redes neuronales. ■

1 | https://isclab.dcc.uchile.cl/.

Inteligencia artificial en la educación

JÉRÉMY BARBAY Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile.

La tecnología siempre se ha incorporado a la docencia de manera desigual, y las técnicas de Inteligencia Artificial (IA) no son una excepción. Con el fin de contribuir a reducir dicha desigualdad, presentamos una vista superficial de: 1) algunas técnicas de inteligencia artificial, 2) algunos sistemas de manejo del aprendizaje, y 3) algunas aplicaciones de técnicas de IA a lo señalado en el punto 2. Con la finalidad de (intentar) guiar desarrollos futuros, presentamos una discusión corta sobre los desafíos presentes y futuros de las técnicas de inteligencia artificial sobre los sistemas de manejo de aprendizaje.

Historia y definiciones

Tecnologías educacionales

El campo de "tecnologías educacionales" corresponde al estudio y la práctica ética de facilitar la educación y mejorar el rendimiento creando, usando y manejando los recursos y procesos adecuados. Desde la perspectiva del uso de la tecnología en educación, tecnologías educacionales se puede entender como el uso de tecnologías existentes y emergentes para mejorar la experiencia de aprendizaje en una variedad de contextos instruccionales, como el aprendizaje formal, informal, no-formal, a demanda (*on-demand*) o *"just-in-time"* [1].

Tales tecnologías educacionales incluyen una gran variedad de dominios de desarrollo. Respecto al material, se consideran una gran cantidad de dispositivos, desde proyectores de apuntes copiados sobre láminas transparentes, computadoras personales e interconectadas, hasta tecnologías "inteligentes" como teléfonos, entornos virtuales, computación en la nube, aparatos *"wearable"* y *"location-aware"*. Respecto del software, se considera, por un lado el software dirigido a quienes aprenden, como los software de simulación y de visualización, y las interfaces de gamificación mejorando la motivación; y, por otro lado, el software dirigido a la administración del aprendizaje, con los "*Learning Management Systems*" (LMS)[1] y su integración vía "Learning Tools Interoperability" (LTI).[2]

1 | https://en.wikipedia.org/wiki/Learning_management_system.

2 | https://en.wikipedia.org/wiki/Learning_Tools_Interoperability.

La digitalización del material educativo, una tendencia que existía pero se desarrollaba relativamente lenta hasta 2019 [2], se ha acelerado con la transición súbita hacia la docencia online en el contexto de la pandemia por COVID-19. En este contexto, se han digitalizado muchos aspectos de la docencia. Por un lado, las charlas, tradicionalmente en anfiteatros y en vivo, y raramente grabadas, han sido reemplazadas en muchos casos por la difusión en tiempo real de tales charlas en video, y en otros casos por la difusión de cápsulas de videos cortas, grabadas y editadas con anticipación: en ambos casos, los alumnos pueden mirar tales videos en momentos de su elección, desde su hogar, y muchas veces las ven mientras hacen otras actividades y/o en modo acelerado. Por otro lado, las evaluaciones teóricas, tradicionalmente entregadas sobre papel, en instancias de exámenes presenciales, están siendo reemplazadas por entregas digitales, generando sospechas de copias y de usurpación de identidades en los cuerpos docentes.

En tal desarrollo de las tecnologías educacionales, era esperable ver llegar las técnicas de inteligencia artificial, las cuales intentamos definir en la siguiente sección, antes de desarrollar sus interacciones con el campo de *"Learning Management Systems"* en la sección "Aplicaciones de la IA a los LMS".

Técnicas de inteligencia artificial

Conviene primero aclarar el concepto de "inteligencia artificial". En la vida diaria, el término se aplica cuando una máquina imita las funciones "cognitivas" que los humanos asocian con mentes humanas, como por ejemplo: "percibir", "razonar", "aprender" y "resolver problemas" [4]. Una definición más formal y menos antropomórfica sería *"la capacidad de un sistema para interpretar correctamente datos externos, para aprender de dichos datos y emplear esos conocimientos para lograr tareas y metas concretas a través de la adaptación flexible"*. En ambos casos, mientras que las máquinas se vuelven más capaces, tecnologías que alguna vez se consideraban del campo de "inteligencia artificial" se reevalúan.

La expresión "inteligencia artificial" fue introducida en 1956 por John McCarthy, quien la definió como *"la ciencia e ingenio de hacer máquinas inteligentes, especialmente programas de cómputo inteligentes"*. Pero el concepto existía desde hace mucho más tiempo, lo que hace que siga evolucionando en paralelo con las tecnologías [3].

En 2021, los objetivos de "inteligencia artificial" se pueden clasificar en cuatro tipos [4]:

- *Sistemas que piensan como humanos.* Estos sistemas tratan de emular el pensamiento humano, por ejemplo, las redes neuronales artificiales. La automatización de actividades que vinculamos con procesos de pensamiento humano, actividades como la toma de decisiones, resolución de problemas y aprendizaje.

- *Sistemas que actúan como humanos.* Estos sistemas tratan de actuar como humanos, es decir, imitan el comportamiento humano, por ejemplo, la robótica. El estudio de cómo lograr que los computadores realicen tareas que, por el momento, los humanos hacen mejor.

- *Sistemas que piensan racionalmente.* Esto es, con lógica (idealmente), tratan de imitar el pensamiento racional del ser humano, por ejemplo, los sistemas expertos. El estudio de los cálculos que hacen posible percibir, razonar y actuar.

- *Sistemas que actúan racionalmente.* Tratan de emular de forma racional el comportamiento humano, por ejemplo los agentes inteligentes. Está relacionado con conductas inteligentes en artefactos.

En la siguiente sección veremos cómo las técnicas de inteligencia artificial se han relacionado y siguen relacionándose con las técnicas de educación y de aprendizaje.

Aplicaciones de la IA a los LMS

Desde muy temprano se relacionaron los temas de educación (humana) e inteligencia artificial, quizás porque en ambos casos se trata de desarrollar habilidades "inteligentes"', ya sea en humanos o en máquinas. Seymour Papert, uno de los cofundadores del Instituto de Inteligencia Artificial del MIT, en 1963 (con Marvin Minsky, considerado uno de los padres de la inteligencia artificial[3], había tenido previamente un rol mayor en la evaluación y el desarrollo de técnicas de educación, en colaboración con el psicólogo educativo Piaget.[4]

En 2021, técnicas de inteligencia artificial presentan aplicaciones en varios aspectos de la docencia. En un survey publicado en 2020, Chen et al. [5] describen varias aplicaciones de inteligencia artificial en áreas relacionadas con la educación, en particular aplicadas a los aspectos de la administración de la docencia. Tales aplicaciones permiten, entre otros, detectar ocurrencias de plagio, automatizar algunos aspectos de la evaluación de trabajos, e identificar a un alumno presente cuyo perfil sea similar al perfil de alumnos anteriores que tuvieron problemas en fases siguientes.

Por otro lado, software como Duolingo[5] usa técnicas de gamificación para mantener la motivación de sus alumnos, y técnicas de repetición espaciada [6] para programar qué ejercicio darle a un alumno en función de modelos.

En el futuro, técnicas de inteligencia artificial tendrán otras aplicaciones en educación. Investigadores como la Dra. Shaghayegh Sahebi están proponiendo diseñar, desarrollar y evaluar sistemas capaces de realizar recomendaciones personalizadas de material docente en función de varios parámetros [7].

Conclusiones

Las técnicas descritas como "inteligencia artificial" no son más que nuevas tecnologías que apuntan a acercar las capacidades de las máquinas a las capacidades de los humanos. En varias épocas se sobrepromotió lo que se podía lograr con dichas técnicas, y la época presente no es una excepción. Pero aún permiten automatizar algunas tareas humanas, y apoyar otras.

El área de la educación, y en particular el área de la educación en línea, tiene un gran potencial de mejoras vía técnicas digitales en general, y técnicas propias de "inteligencia artificial" en particular, y ha sido un poco lenta en adoptar dichas técnicas. Es esperable que con la digitalización acelerada debido a la pandemia por COVID-19, dicha transición se vea acelerada.

Como siempre con la tecnología, será importante no dejar el efecto de novedad, ni quitar el foco de problemas importantes existentes (por ejemplo, desigualdades) ignorados o amplificados por nuevas técnicas, ni de nuevos problemas creados por dichas técnicas (por ejemplo, sesgos en favor de minorías producidos por técnicas de inferencias, impacto ecológico de las digitalizaciones, etc.). ■

REFERENCIAS

[1] R. Huang, J. Spector y J. Yang (2019). Educational Technology: A Primer for the 21st Century. 10.1007/978-981-13-6643-7.

[2] J. Barbay y V. Peña-Araya (2019). El Académico Digital. En Revista Bits de Ciencia nº18.

[3] Historia de la inteligencia artificial. En *Wikipedia*. Accedido desde https://es.wikipedia.org/wiki/Historia_de_la_inteligencia_artificial, [2021-04-19 Mon].

[4] Inteligencia artificial. En *Wikipedia*. Accedido desde https://es.wikipedia.org/wiki/Inteligencia_artificial, last accessed, [2021-04-19 Mon].

[5] L. Chen, P. Chen y Z. Lin. (2020). Artificial Intelligence in Education: A Review. En IEEE Access, vol. 8, pp. 75264–75278, 10.1109/ACCESS.2020.2988510.

[6] Spaced repetition. En *Wikipedia*. Accedido desde https://en.wikipedia.org/wiki/Spaced_repetition, [2021-04-19 Mon].

[7] https://www.nsf.gov/awardsearch/showAward?AWD_ID=2047500, [2021-04-19 Mon].

3 | https://es.wikipedia.org/wiki/Marvin_Minsky.

4 | https://es.wikipedia.org/wiki/Seymour_Papert.

5 | https://www.duolingo.com/.

Aprendizaje de representaciones en grafos y su importancia en el análisis de redes

MARCELO MENDOZA Profesor Asociado del Departamento de Informática de la Universidad Técnica Federico Santa María e Investigador Asociado del Instituto Milenio Fundamentos de los Datos.

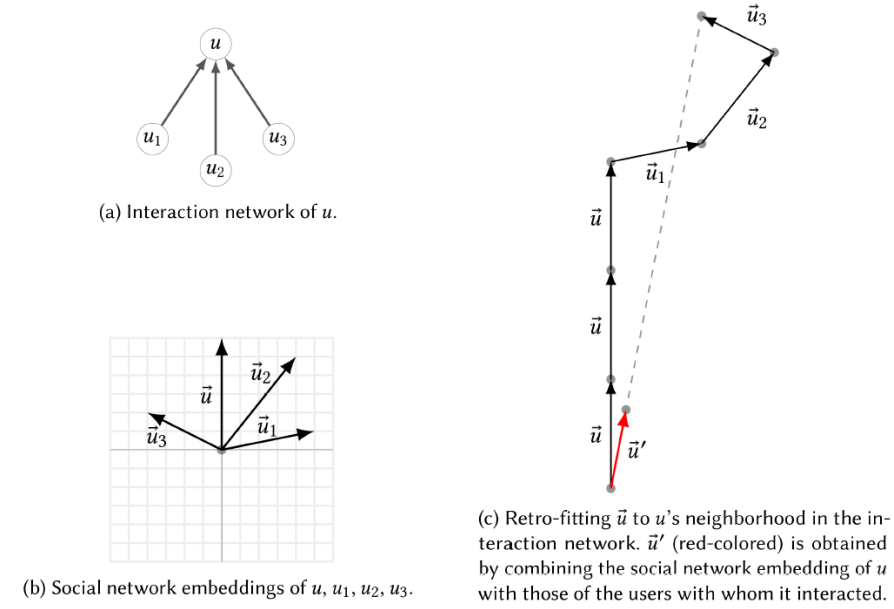
Una de las líneas de investigación en inteligencia artificial más fructíferas de la última década es el aprendizaje de representaciones. Mostraremos dos ejemplos en los cuales el aprendizaje de representaciones de nodos en grafos ha permitido abordar exitosamente tareas de análisis de redes.

Detección de *bots*

Los *bots* tienen un nefasto efecto en la diseminación de información engañosa o tendenciosa en redes sociales [1]. Su objetivo es amplificar la alcanzabilidad de campañas, transformando artificialmente mensajes en tendencias. Para ello, las cuentas que dan soporte a campañas se hacen seguir por cuentas manejadas por algoritmos. Muchas de las cuentas que siguen a personajes de alta connotación pública son *bots*, las cuales entregan soporte a sus mensajes con *likes* y *retweets*. Cuando estos mensajes muestran un inusitado nivel de reacciones, se transforman en tendencias, lo cual aumenta aún más su visibilidad. Al transformarse en tendencias, su influencia en la red crece, produciendo un fenómeno de bola de nieve.

La detección de *bots* ha sido una tarea difícil. Mientras que las primeras generaciones de *bots* eran sencillas de detectar, las nuevas generaciones de *bots*, conocidas como *social bots*, alternan periodos de propaganda y periodos de baja actividad [2]. En estos últimos, los *bots* muestran un comportamiento cercano al de un usuario promedio, con participación esporádica en la red. En periodos de campaña, la actividad de estas cuentas aumenta. El cambio en el régimen de interacciones es una pista que nosotros usamos para detectarlos.

En [3], mostramos cómo extender una representación de nodos aprendida a partir de la red de conexiones sociales en Twitter. La estrategia de aprendizaje usada se denomina ComplEx [4], la cual permite aprender *node embeddings* de la red de conexiones para predicción de *links*. Para capturar el régimen de interacciones entre cuentas, extendemos ComplEx reescalando los *node embeddings* en la dirección de los vecinos con los cuales tienen más interacciones. La Figura 1 muestra la estrategia de reescalamiento basada en interacciones, lo cual permite recalcular los *node embeddings* combinando ambas redes (social e interacción). Para aprender los *node embeddings* usamos una estrategia denominada *retrofitting* [5], que busca una

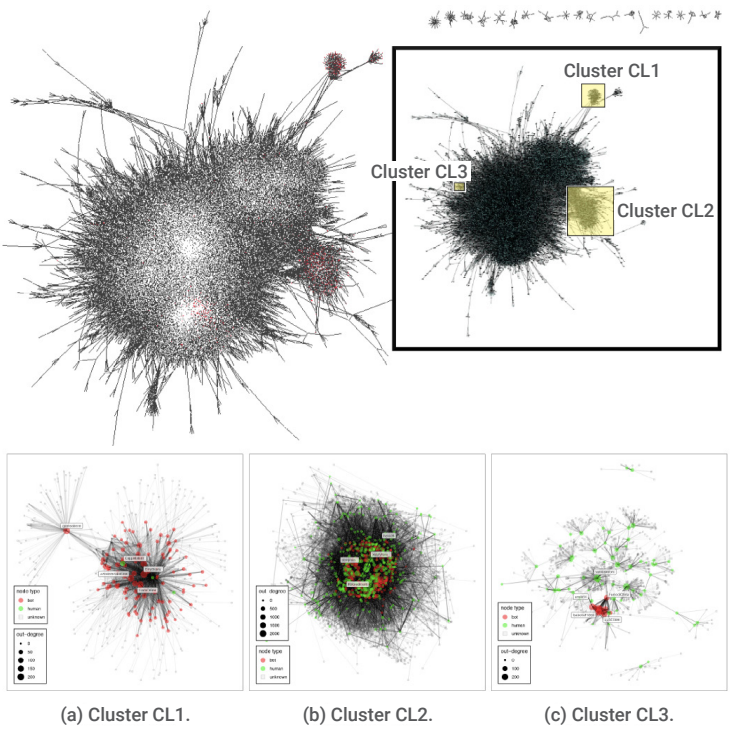


(a) Interaction network of u.

(b) Social network embeddings of u, u_1, u_2, u_3.

(c) Retro-fitting $\vec{u}$ to u's neighborhood in the interaction network. $\vec{u}'$ (red-colored) is obtained by combining the social network embedding of u with those of the users with whom it interacted.

Figura 1. Extensión de ComplEx [4] que incorpora la red de interacciones entre usuarios de Twitter.



(a) Cluster CL1. (b) Cluster CL2. (c) Cluster CL3.

Figura 2. Red de proximidad entre *node embeddings* en Twitter, que muestra tres *clusters* con presencia de *bots* (nodos rojos). Mientras que el *cluster* 1 (CL1) no logra interactuar con humanos (nodos verdes), los *clusters* 2 (CL2) y 3 (CL3) se mimetizan, promoviendo contenido propagandístico.

representación consistente entre ambas fuentes de información.

Para detectar *bots*, aplicamos un algoritmo de propagación de etiquetas en la red de proximidad de *node embeddings*. El método de propagación permite trabajar con un número reducido de nodos etiquetados como *bots*, usando una estrategia semisupervisada sobre la red. La estrategia semisupervisada permite que el método funcione sobre redes de enorme tamaño con sólo una fracción de sus nodos etiquetados por expertos (app. 1% del total de la red). Mostramos que el método de imputación de etiquetas es análogo a una estrategia de paso de mensajes en una red neuronal de grafos que aborda una tarea de clasificación de nodos [6].

Nuestro método superó al estado del arte (Botometer [7] y Holoscope [8]). Su principal habilidad está en la detección de *botnets*, lo cual le permite sacar ventaja de sus más directos competidores que abordan la tarea como clasificación de nodos. El método de propagación de etiquetas tiene la ventaja de identificar grupos de cuentas *clusterizadas* según interacciones inusuales, detectando patrones de coordinación temporal. La Figura 2 muestra una red de proximidad entre *node embeddings* y tres *clusters* con alta presencia de *bots* (nodos rojos) en Twitter. Mientras que el *cluster* 1 (CL1) es una *botnet* que no ha logrado interactuar con humanos (nodos verdes), los *clusters* 2 (CL2) y 3 (CL3) muestran una mimetización de los *bots* en las redes de humanos, con interacción cruzada entre ambos tipos de usuarios.

Predictibilidad en redes sociales *offline*

En [9], analizamos las relaciones filiales entre personas, observables a través de los vínculos de apellidos paternos-ma-

ternos. La red construida con los datos del servicio electoral y cruzada con datos del Indice de Bienestar Territorial nos permitió construir un mapa de las conexiones familiares de los habitantes de la Región Metropolitana. Usando el método de Mateos *et al.* [10], identificamos los vínculos cuyas ocurrencias superaban el valor esperado dado por una red de conexiones aleatorias. Una vez construida la red, visualizamos su estructura agrupando nodos según modularidad. Las comunidades detectadas muestran etnias y también una fuerte *clusterización* de apellidos de clase alta según índice socioeconómico (ver Figura 3, al tope).

La misma red, ahora *clusterizada* según ingreso socioeconómico (ver Figura 3, al medio), muestra dos particiones, una con una fuerte interacción entre apellidos poco frecuentes y muchos nodos articuladores (comunidad azul de los tres deciles de ingreso más alto), y una partición mucho más desarticulada, con una vinculación más débil entre apellidos y menos nodos articuladores (comunidad roja de los siete deciles más bajos de ingreso). Estudiamos la predictibilidad de esta red, donde la tarea corresponde a predecir vínculos entre familias no conectadas (*link prediction*). Para hacer esto, aplicamos una técnica de aprendizaje de representaciones de nodos basada en factorización tensorial denominada método de TuckER [11]. Probamos el desempeño de otros métodos de representación a nivel de nodos, como ComplEx [4], RESCAL [12] y RotatE [13], usados en *knowledge-base completion*. TuckER mostró mejor desempeño en *link prediction* que sus competidores, factor atribuible a su habilidad de trabajar con datos *sparse*.

Al pie de la Figura 3 mostramos los resultados de predicción de vínculos segmentados por decil de ingreso. Los deciles de mayor ingreso (d1 - d3) muestran mejor predictibilidad, la cual disminuye progresivamente para los deciles de menor ingreso (d4 – d10).

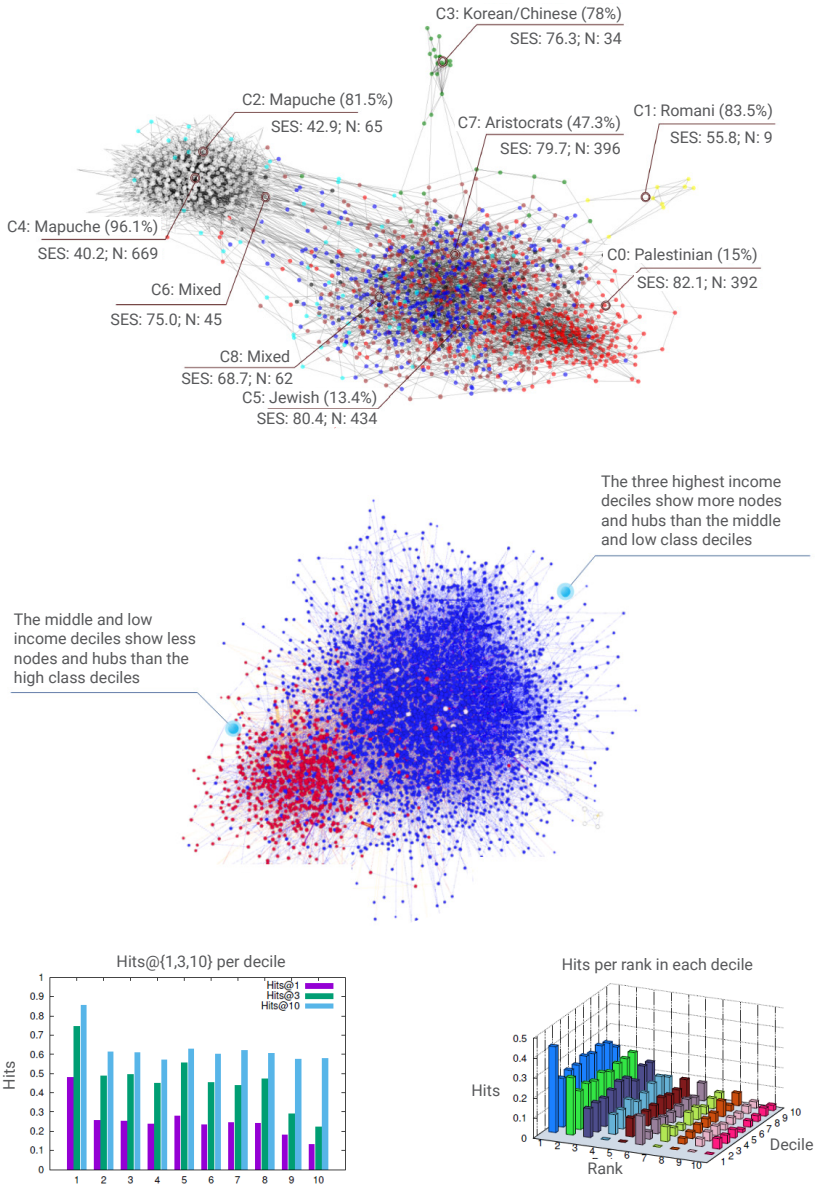


Figura 3. Redes de vínculos paternos-maternos en la Región Metropolitana (al tope), la misma red *clusterizada* según ingreso socioeconómico (al medio), y la predictibilidad de vínculos usando TuckeER [11] (al pie).

Conclusión

La inteligencia artificial a través de su área denominada aprendizaje de representaciones ofrece enormes posibilidades en tareas complejas, tanto en redes sociales en línea como en redes *offline*. Su habilidad para codificar características esenciales en distintos dominios permite generar representaciones que mejoran las posibilidades de análisis de datos. ■

REFERENCIAS

[1] Stefano Cresci: A decade of social bot detection. Commun. ACM 63(10): 72–83 (2020).

[2] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, Maurizio Tesconi: The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. WWW (Companion Volume) 2017: 963–972.

[3] Marcelo Mendoza, Maurizio Tesconi, Stefano Cresci: Bots in Social and Interaction Networks: Detection and Impact Estimation. ACM Trans. Inf. Syst. 39(1): 5:1–5:32 (2020).

[4] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, Guillaume Bouchard: Complex Embeddings for Simple Link Prediction. ICML 2016: 2071–2080.

[5] Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard H. Hovy, Noah A. Smith: Retrofitting Word Vectors to Semantic Lexicons. HLT-NAACL 2015: 1606–1615.

[6] Franco Scarselli, Sweah Liang Yong, Marco Gori, Markus Hagenbuchner, Ah Chung Tsoi, Marco Maggini: Graph Neural Networks for Ranking Web Pages. Web Intelligence 2005: 666–672.

[7] Onur Varol, Emilio Ferrara, Clayton A. Davis, Filippo Menczer, Alessandro Flammini: Online Human-Bot Interactions: Detection, Estimation, and Characterization. ICWSM 2017: 280–289.

[8] Shenghua Liu, Bryan Hooi, Christos Faloutsos: HoloScope: Topology-and-Spike Aware Fraud Detection. CIKM 2017: 1539–1548.

[9] Naim Bro, Marcelo Mendoza. Surname affinity in Santiago, Chile: A network-based approach that uncovers urban segregation. PLOS ONE, 16(1): e0244372, 2021.

[10] Pablo Mateos, Paul Longley, David O'Sullivan. Ethnicity and population structure in personal naming networks. PLOS ONE, 6(9): e22943, 2011.

[11] Ivana Balazevic, Carl Allen, Timothy M. Hospedales: TuckER: Tensor Factorization for Knowledge Graph Completion. EMNLP/IJCNLP (1) 2019: 5184–5193.

[12] Maximilian Nickel, Volker Tresp, Hans-Peter Kriegel: A Three-Way Model for Collective Learning on Multi-Relational Data. ICML 2011: 809–816

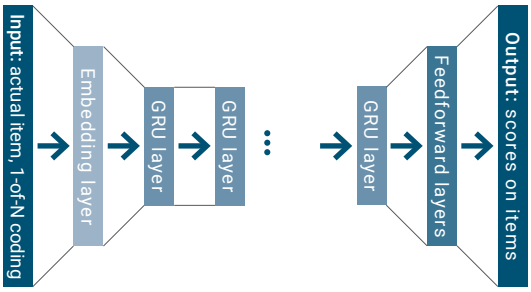
[13] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, Jian Tang: RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. ICLR 2019.

Aprendizaje profundo en sistemas de recomendación

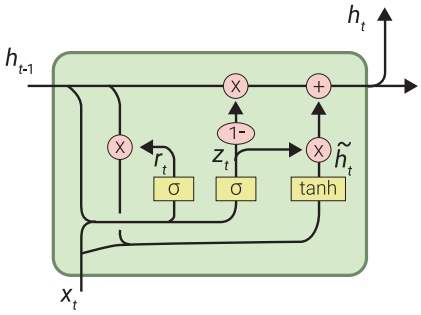
DENIS PARRA Profesor Asociado del Departamento de Ciencia de la Computación de la Pontificia Universidad Católica de Chile e Investigador Adjunto del Instituto Milenio Fundamentos de los Datos.

Corría el año 2010 y yo cursaba mi doctorado enfocado en personalización y sistemas de recomendación en la Universidad de Pittsburgh, ubicada en la ciudad homónima (Pittsburgh) al oeste del estado de Pennsylvania en Estados Unidos. Las técnicas más avanzadas de mi tema de investigación eran del área conocida como Aprendizaje Automático (en inglés, *Machine Learning*), por lo que sentía la necesidad de tomar un curso avanzado para completar mi formación. En el semestre de otoño finalmente me inscribí en el curso de Aprendizaje Automático, y gracias a un convenio académico pude cursarlo en la universidad vecina, Carnegie Mellon University. Yo estaba realmente emocionado de tomar un curso en un tema de tan creciente relevancia en unas de las mejores universidades del mundo en el área de computación. Recuerdo que vimos muchas técnicas que permitían aprender modelos a partir de datos, con especial énfasis en modelos gráficos —por ejemplo, el famoso *Latent Dirichlet Allocation* [1]— así como en métodos *kernel* como *Support Vector Machines* (SVM). Casi al final del curso, tuvimos una clase algo tímida sobre redes neuronales artificiales, un método interesante pero que poca gente usaba. Las redes neuronales artificiales datan de los años cincuenta [2], renacieron en los ochenta luego del invierno de la IA [3], para luego volver a perder tracción en los noventa. Cuál fue mi sorpresa cuando el año 2012 las redes neuronales artificiales pasaban a ser el método que todos querían usar y del cual todos hablaban. El motivo fue el sorprendente resultado del equipo SuperVision de la Universidad de Toronto[1] —Krizhevsky, Sutskever y Hinton—, que usando una red neuronal convolucional profunda (*deep convolutional neural network*) con 60 millones de parámetros y 650 mil neuronas, entrenado con dos GPUs durante una semana, ganaba el ImageNet challenge 2012 con un error top-5 del 15,3% y más de 10 puntos de mejora en relación al segundo lugar. Las redes neuronales profundas tenían algunos antecedentes importantes de buen rendimiento [4], pero el resultado del 2012 en el ImageNet challenge catapultó su popularidad. La arquitectura de red neuronal creada empezó a ser popularmente conocida como AlexNet [5], debido al nombre del primer autor, Alex Krizhevsky. A partir de ese momento, ingenieros e investigadores de diferentes áreas de la inteligencia artificial querían escribir los términos *deep learning*

1 | https://www.image-net.org/challenges/LSVRC/2012/results.html.

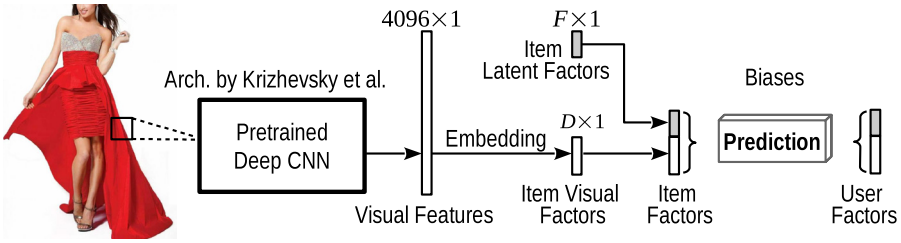


Fuente: [16].



Fuente: http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

Figura 1. Arquitectura de GRU4Rec donde cada capa GRU tiene celdas GRU como la que se observa a la derecha, que pueden recordar y olvidar, selectivamente, permitiendo el aprendizaje de secuencias.



Fuente: [17].

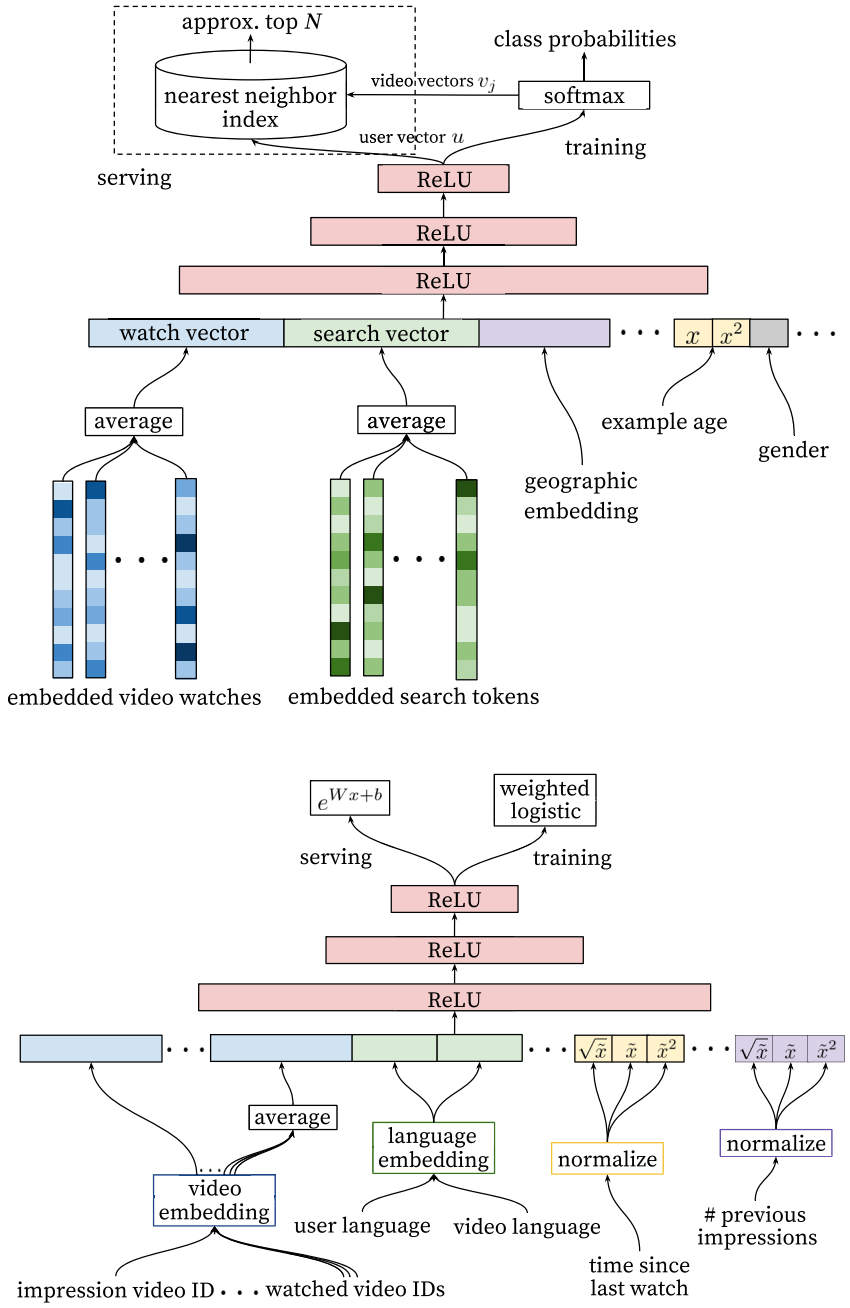
Figura 2. Diagrama de VBPR que indica cómo las características visuales obtenidas con una red neuronal convolucional profunda son incorporadas en el predictor de preferencia.

o *neural network* en el título de sus artículos, y es así cómo este método empieza a permear desde el campo de visión por computador a otras áreas como recuperación de información [6], traducción automática [7], describir imágenes con texto de forma automática [8], o incluso áreas creativas como generación visual [9] y musical [10].

A pesar del frenesí de distintas áreas por usar aprendizaje profundo, no fue hasta el 2015 que aparecen *papers* relevantes de aprendizaje profundo aplicados a Sistemas Recomendadores (de aquí en adelante, *SisRec*). Recordemos que los SisRec tienen como rol principal ayudarnos a encontrar ítems relevantes dentro de una sobreabundancia de información [11] considerando nuestras preferencias individuales. Compañías tan diversas como Amazon, Netflix, Google, Booking y Spotify basan buena parte de sus funcionalidades y modelos de negocio en sistemas recomendadores. Estos sistemas se han desarrollado por más de treinta años, pero han evolucionado especialmente rápido en la última década.

Volviendo a la aplicación de aprendizaje profundo aplicado a SisRec, es posible rescatar como antecedente previo a ImageNet el uso de *restricted Boltzman machines* [12], un tipo de red neuronal probabilística, entre los mejores métodos que compitieron en el Netflix prize [13]. Sin embargo, los primeros trabajos utilizando aprendizaje profundo ya sea a través de representaciones preentrenadas o para el modelo completo fueron los trabajos de Van den Oord *et al.* [14], un recomendador de música que utilizaba representaciones de audio aprendidas con una red neuronal profunda. Luego, se presenta en 2015 "aprendizaje profundo colaborativo para SisRec" [15], un método que combina las técnicas de filtrado colaborativo con *denoising autoencodeers*. El mismo 2015 aparece GRU4Rec [16] que modela secuencias de interacciones usando redes recurrentes con celdas GRU (ver Figura 1) para recomendar productos, y el mismo año se publica VBPR [17], método que utiliza la representación de imágenes que entrega una red convolucional preentrenada para mejorar recomendaciones visuales (ver Figura 2) realizadas por el modelo BPR [18].

Es difícil saber por qué el área de SisRec demoró tanto (alrededor de tres años) en ingresar a la ola de las redes neuronales profundas, pero es posible argumentar algunas razones en base a los pilares que posibilitaron el crecimiento del aprendizaje profundo: (a) gran cantidad de datos, (b) algoritmos de aprendizaje más eficientes, y (c) hardware especializado para el entrenamiento. En el área de sistemas de recomendación no era trivial encontrar *datasets* de gran tamaño, como el ImageNet, para entrenar modelos con tantos millones de parámetros como una red neuronal profunda. Esto se debe a que las grandes



Fuente: [26].

Figura 3. Las dos redes neuronales que formaban parte del sistema recomendador de videos, de aprendizaje profundo, del portal YouTube, activo hasta el 2019.

compañías han sido reticentes a compartir *datasets* que indiquen preferencias de usuarios por productos, ya sea por temas de competencia como para evitar violaciones de privacidad [19]. En los últimos años la disponibilidad de grandes *datasets* para entrenar modelos de recomendación ha mejorado mucho, con *datasets* como el de Spotify[2], Goodreads[3] o la versión 25M del tradicional movielens dataset[4]. En cuanto a algoritmos, si bien es posible adaptar métodos existentes de clasificación de imágenes o ranking de documentos para tareas de recomendación, el hecho de tener que incorporar el modelo de usuario en el método complejiza un poco su modelamiento e implementación. No es lo mismo usar un modelo de ranking de imágenes dada una imagen de entrada, que un modelo de ranking de imágenes personalizado, que considere tanto el historial de consumo de un usuario [17, 20, 21] así como el contexto de dicho consumo —día de la semana, hora, haciendo qué actividad, etc. [22]. En relación a hardware, no es un secreto que son grandes compañías como NVidia, Google, Amazon, o Facebook quienes disponen de los mejores recursos de hardware para entrenar modelos que crecen sin cesar en cantidad de parámetros: como muestra, el reciente modelo de lenguaje GPT-3 tiene 175 mil millones de parámetros [23], comparado con los 60 millones de parámetros de la AlexNet. Esto dificulta la investigación que provenga exclusivamente desde la academia, donde los incentivos permiten investigar temas diferentes a los que empujan la investigación en la industria. A pesar de estas dificultades, una propiedad interesante de estos modelos es la posibilidad de hacer *transfer learning* [24], es decir, entrenarlos inicialmente para una tarea y luego actualizar todos o parte de sus pesos para otro *dataset* o para otras tareas. Esto permite que el costo mayor de entrenamiento lo lleven a cabo grandes compañías, fundaciones y universidades,

2 | https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge.
3 | https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/home.
4 | https://grouplens.org/datasets/movielens/25m/.

y luego otros usuarios con menores recursos de hardware tienen sólo que adaptar (*finetuning)* los pesos para la nueva tarea o *dataset* que se aborda.

A partir del año 2016 el aprendizaje profundo aterriza con fuerza en la conferencia internacional ACM de sistemas recomendadores, donde se publica "*Ask the GRU*" [25], un recomendador con aprendizaje multitarea de artículos científicos que usa una red recurrente con celdas del tipo Gated Recurrent Unit. Además de este *paper*, autores de Google [26] presentan la nueva versión del sistema recomendador de videos de YouTube, basado en dos redes neuronales profundas (ver Figura 3), una red que selecciona cientos de candidatos a partir de millones de opciones, y una segunda red que ordena los videos candidatos previamente filtrados. La nueva arquitectura del portal YouTube [27] tiene algunos aspectos interesantes, por ejemplo que considera los likes de los usuarios para generar el perfil del usuario para recomendar, cosa que no hacía el recomendador anterior [26].

Luego de estas publicaciones, es común encontrar SisRec implementados con métodos de aprendizaje profundo en temas como recomendación de música, películas, libros, pareja sentimental, ropa de temporada, entre muchos otros. Los sistemas han evolucionado en los últimos años de la arquitecturas como Transformer [28], integrados con otras técnicas como aprendizaje reforzado profundo [29], así como explotando avances en áreas como NLP [30] o modelos generativos [31].

Discusión y conclusión

El aprendizaje profundo ha impactado positivamente el área de SisRec, tanto como a otras áreas de aplicación de la inteligencia artificial. Hay, sin embargo, dos aspectos importantes a mencionar que generan inquietud en el área: cuánto es el progreso real que ha traído el aprendizaje profundo, y cómo estos modelos afectan el avance en temas de temas de equidad, explicabilidad y transparencia.[5]

¿Cuánto se ha progresado? El artículo de [32] pone en entredicho el impacto del aprendizaje profundo en los SisRec, mostrando que cuando métodos tradicionales de factorización matricial que se conocen por más de una década son entrenados adecuadamente, tienen tanto o mejor rendimiento que métodos de aprendizaje profundo. Si bien este *paper* es relevante por mostrar una crisis de reproducibilidad en SisRec y que no siempre el aprendizaje profundo puede mejorar el rendimiento los métodos ya conocidos, hay un aspecto relevante a considerar. La investigación de Dacrema sólo considera tuplas usuario-ítem como entrada, pero no considera información adicional como imágenes, video, metadata, contexto, etc. Justamente es con esta gran cantidad y diversidad de datos donde es esperable el rendimiento mejorado de técnicas de aprendizaje profundo, por lo cual se recomienda revisar con cautela los resultados de este análisis, y ponerlo en perspectiva sólo para el filtrado colaborativo tradicional.

FAccT. Considerar los desafíos que se plantean en la inteligencia artificial en relación a equidad (*fairness*), explicabilidad (*accountability*) y transparencia (*transparency*) es un gran desafío para los modelos de aprendizaje profundo en SisRec [33]. Considere el caso en que usa GPT-3, un modelo de 175 mil millones de parámetros, para recomendar un documento y el usuario solicita una explicación sobre dicha sugerencia ¿cómo explicaría dicha recomendación intentando ser transparente? Los métodos de explicabilidad para inteligencia artificial están en activa investigación en estos días [34] y si deseamos que los sistemas de recomendación permeen áreas críticas de toma de decisiones como medicina, finanzas o seguridad, se debe avanzar en esta área. En relación a asegurar que estos sistemas no están sesgados existe una inquietud similar: cómo hacer que provean recomendaciones justas a diferentes grupo de usuarios finales, por ejemplo de un sistema de recomendación de empleo, así como a creadores de contenido: que un portal de libros recomiende con la misma probabilidad tanto a escritores hombres como mujeres o de otros grupos LGBTQ.

Conclusión. El aprendizaje profundo tomó algunos años en permear el área de sistemas de recomendación en comparación con otras áreas de inteligencia artificial, pero se instaló con fuerza a partir de 2016 gracias a su gran capacidad para encontrar representaciones de usuarios y datos para posteriormente ser usadas en tareas de filtrado de información. Con el avance de modelos de visión por computador, modelos de lenguaje, arquitecturas como atención y más recientemente modelos de redes neuronales para grafos, el impacto de las redes neuronales profundas en SisRec no ha dejado de crecer. La integración de estas técnicas con otras como aprendizaje reforzado para SisRec y el crecimiento en los últimos años de los sistemas de recomendación conversacionales [35] le siguen dando fuerza a esta área de investigación. Los desafíos en términos de mostrar los avances reales en rendimiento de estas técnicas [32] así como su adaptación para lidiar con necesidades de equidad, transparencia, explicabilidad [33], nos harán ver sin duda mucha más investigación en este tema en los años venideros. ■

5 | FAccT 2018. ACM Conference on Fairness, Accountability, and Transparency https://facctconference.org/.

REFERENCIAS

[1] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, *3*, 993-1022.

[2] Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.

[3] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ. San Diego La Jolla Inst. for Cognitive Science.

[4] Ciresan, D. C., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2011). Flexible, high performance convolutional neural networks for image classification. In *Twenty-second international joint conference on artificial intelligence*.

[5] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*, 1097-1105.

[6] Severyn, A., & Moschitti, A. (2015). Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 373-382).

[7] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[8] Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156-3164).

[9] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks. arXiv preprint arXiv:1406.2661.

[10] Roberts, A., Engel, J., Raffel, C., Hawthorne, C., & Eck, D. (2018). A hierarchical latent vector model for learning long-term structure in music. In *International Conference on Machine Learning* (pp. 4364-4373). PMLR.

[11] McNee, S. M., Kapoor, N., & Konstan, J. A. (2006). Don't look stupid: avoiding pitfalls when recommending research papers. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work* (pp. 171-180). ACM.

[12] Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). Restricted Boltzmann machines for collaborative filtering. In Proceedings of the 24th international conference on Machine learning (pp. 791-798).

[13] Bennett, J., & Lanning, S. (2007, August). The Netflix Prize. In Proceedings of KDD cup and workshop (Vol. 2007, p. 35).

[14] Van Den Oord, A., Dieleman, S., & Schrauwen, B. (2013). Deep content-based music recommendation. In Neural Information Processing Systems Conference (NIPS 2013) (Vol. 26). Neural Information Processing Systems Foundation (NIPS).

[15] Wang, H., Wang, N., & Yeung, D. Y. (2015). Collaborative deep learning for recommender systems. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1235-1244).

[16] Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939.

[17] He, R., & McAuley, J. (2016). VBPR: visual bayesian personalized ranking from implicit feedback. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 30, No. 1).

[18] Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2012). BPR: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618.

[19] Narayanan, A., & Shmatikov, V. (2006). How to break anonymity of the Netflix Prize dataset. arXiv preprint cs/0610105.

[20] Chen, J., Zhang, H., He, X., Nie, L., Liu, W., & Chua, T. S. (2017). Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval (pp. 335-344).

[21] Messina, P., Domínguez, V., Parra, D., Trattner, C., & Soto, A. (2019). Content-based artwork recommendation: integrating painting metadata with neural and manually-engineered visual features. User Modeling and User-Adapted Interaction, 29(2), 251-290.

[22] Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In Recommender systems handbook (pp. 217-253). Springer, Boston, MA.

[23] Brown, T. B., Mann, B., Ryder, N., et al. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165

[24] Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. IEEE Transactions on knowledge and data engineering, 22(10), 1345-1359.

[25] Bansal, T., Belanger, D., & McCallum, A. (2016). Ask the gru: Multi-task learning for deep text recommendations. In proceedings of the 10th ACM Conference on Recommender Systems (pp. 107-114).

[26] Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (pp. 191-198). ACM.

[27] Zhao, Z., Hong, L., Wei, L. et al. (2019). Recommending what video to watch next: a multitask ranking system. In Proceedings of the 13th ACM Conference on Recommender Systems (pp. 43-51).

[28] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

[29] Zheng, G., Zhang, F., Zheng, Z., Xiang, Y., Yuan, N. J., Xie, X., & Li, Z. (2018, April). DRN: A deep reinforcement learning framework for news recommendation. In Proceedings of the 2018 World Wide Web Conference (pp. 167-176).

[30] Penha, G., & Hauff, C. (2020). What does BERT know about books, movies and music? Probing BERT for Conversational Recommendation. In Fourteenth ACM Conference on Recommender Systems (pp. 388-397).

[31] Kang, W. C., Fang, C., Wang, Z., & McAuley, J. (2017). Visually-aware fashion recommendation and design with generative image models. In 2017 IEEE International Conference on Data Mining (ICDM) (pp. 207-216). IEEE.

[32] Dacrema, M. F., Cremonesi, P., & Jannach, D. (2019). Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems* (pp. 101-109).

[33] Ekstrand, M. D., & Sharma, A. (2017). FATREC Workshop on Responsible Recommendation. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 382-383).

[34] Gunning, D. (2017). Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA), nd Web, 2(2).

[35] Christakopoulou, K., Radlinski, F., & Hofmann, K. (2016). Towards conversational recommender systems. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 815-824).

Detección de discurso de odio

AYMÉ ARANGO Estudiante de Doctorado del Departamento de Ciencias de la Computación de la Universidad de Chile

Las redes sociales se han convertido en un medio importante de interacción entre usuarios de todo el mundo. El contenido compartido puede ser de gran utilidad, como fuente de información inmediata que permite el análisis de eventos, estudio de fenómenos, la difusión de arte, ciencia, entre otras. Junto con esta información, también se encuentran manifestaciones de ciertos fenómenos comunicacionales como noticias falsas y discurso de odio que pueden producir efectos colaterales dañinos.

A pesar de que hay cierta discrepancia en cómo definir el término "discurso de odio", una de las definiciones más usadas es: expresiones derogatorias a individuos o grupos atendiendo a cierta característica como color de la piel, origen étnico, género, orientación sexual, entre otros.[1] La propagación de este tipo de contenido en los medios digitales tiene como efectos la molestia e intimidación de los usuarios. En casos extremos puede trascender el ámbito virtual y llegar a ocasionar daños físicos en individuos. Estudios recientes han encontrado vínculos entre el odio en las redes y los crímenes de odio [1]. Desde diversas disciplinas se trabaja para entender y tratar de identificar a tiempo este fenómeno.

Revisar el contenido publicado consiste en una ardua tarea para los proveedores de redes sociales. Debido al gran flujo de datos a analizar en un red social, y a su variedad, se requieren técnicas automatizadas para detectar este tipo de contenido y tomar medidas necesarias a tiempo. Dada la complejidad de la tarea, esto no ha podido lograrse satisfactoriamente hasta el momento.

Desde el punto de vista de la ciencia de datos, la detección de discurso de odio puede ser planteada como un problema de clasificación en el cual la entrada es un mensaje (*tweet*, comentario, fotografía, etc.) y la salida es la clasificación de éste como contenido odioso o no. Sin embargo, algunos investigadores consideran categorías más específicas y construyen modelos capaces de predecir el tipo específico de odio que está siendo expresado, como sexismo, racismo, xenofobia, entre otros.

Técnicas de inteligencia artificial se han venido utilizando para intentar resolver este problema. Específicamente, los modelos de aprendizaje automático han sido ampliamente utilizados como herramientas en la detección de discurso de odio [2, 3], incluyendo, en los últimos años, modelos basados en arquitecturas de redes neuronales [4]. Para que tales modelos "aprendan" a diferenciar el contenido "odioso" del contenido "normal", se necesitan datos previamente etiquetados. Idealmente, estos datos deberían contener ejemplos representativos de los diferentes tipos de expresiones de odio existentes. Obtener este tipo de datos etiquetados es costoso y debido a la información sensible que manejan y a políticas de cada

1 | https://www.encyclopedia.com/international/encyclopedias-almanacs-transcripts-and-maps/hate-speech.

plataforma, muy pocos conjuntos de datos son públicos y la mayoría son pequeños.[2] Adicionalmente, algunos de los conjuntos de datos publicados han sido reportados como sesgados [5], lo que reduce las posibilidades de utilizar datos de calidad, y como consecuencia, de construir buenos detectores de discurso de odio.

Como parte de mi tesis doctoral, junto con los profesores Bárbara Poblete y Jorge Pérez, estamos investigando técnicas para la construcción de modelos que sean generalizables a diferentes idiomas. Tal y como sucede en otras tareas relacionadas con el Procesamiento del Lenguaje Natural, la mayoría de los modelos desarrollados hasta el momento han sido principalmente explotados para resolver el problema en el idioma inglés. Como consecuencia, la gran parte de los recursos construidos son de utilidad solamente para este idioma, mientras la tarea avanza más lentamente para el resto. Analizando dos de los mejores modelos reportados en la literatura de idioma Inglés [6], encontramos que los resultados mostrados estaban sobreestimados debido a problemas experimentales, y uso de datos sesgados. Además, estos modelos presentan una pobre generalización a datos en el mismo idioma inglés y a datos en español.

Siendo el odio en medios digitales un fenómeno del cual hay evidencia a lo largo de todo el mundo, se requieren soluciones efectivas en los distintos idiomas para afrontar el problema. La idea de nuestro enfoque es aprovechar los recursos existentes (mayormente en inglés) y construir modelos generalizables a diferentes idiomas, ahorrando así el esfuerzo necesario en la creación de nuevos recursos para cada idioma separadamente. Para que los modelos de aprendizaje automático sean capaces de transferir conocimiento de un idioma a otro, se requieren representaciones de los datos a través de un conjunto de características que puedan ser comunes para diferentes idiomas. Ejemplo de esto pueden ser representaciones vectoriales multilingües o información que no esté directamente relacionada con un idioma específico. Particularmente, nuestro equipo de investigación ha trabajado en encontrar dichas características que sean comunes al odio en diferentes idiomas que nos permitan construir modelos generalizables. Bajo nuestro foco de atención, se encuentran aquellas representaciones que puedan ser extraídas del contexto del mensaje, del autor del mensaje (meta-información) y que por su naturaleza no estén atadas a un único idioma [7]. Además, estamos interesados en construir representaciones específicas para el lenguaje de odio, siendo este un fenómeno con características especiales donde ciertas palabras o expresiones pueden tomar connotaciones de odio, en dependencia del contexto. Dichas expresiones no son únicas y pueden depender no sólo del idioma, sino del contexto cultural en el que se exprese. Nos interesaría resaltar estas diferencias culturales en aras de construir modelos que generalicen mejor.

Este tipo de generalización presenta aún varios retos debido a las diferentes características de los idiomas y a la complejidad que puede tener la tarea, siendo el odio un fenómeno no sólo lingüístico, sino social y cultural. Definitivamente, todavía hay mucho que investigar en esta área. Los resultados aún no son concluyentes respecto a qué modelo o representación de datos resulta mejor para esta tarea y aunque se han logrado algunos avances, la tarea aún está por resolverse. ■

REFERENCIAS

[1] Williams ML, Burnap P, Javed A, Liu H, Ozalp S. Hate in the machine: anti-black and anti-Muslim social media posts as predictors of offline racially and religiously aggravated crime. Br J Criminol (2020), 60(1), pp. 93–117.

[2] Anzovino, M., Fersini, E., and Rosso, P. Automatic Identification and Classification of Misogynistic Language on Twitter. In International Conference on Applications of Natural Language to Information Systems (2018), Springer, pp. 57–64.

[3] Papegnies, E., Labatut, V., Dufour, R., and Linares, G. Graph-based Features for Automatic Online Abuse Detection. In International Conference on Statistical Language and Speech Processing (2017), Springer, pp. 70–81.

[4] Gambäck, B., and Sikdar, U. K. Using Convolutional Neural Networks to Classify Hate-Speech. In Proceedings of the First Workshop on Abusive Language Online (2017), Association for Computational Linguistics, pp. 85–90.

[5] Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. The Risk of Racial Bias in Hate Speech Detection. In Proceedings of the Association for Computational Linguistics (2019), pp. 1668–1678.

[6] Arango, A., Pérez, J., Poblete, B.: Hate Speech Detection is Not as Easy as You May Think: A Closer Look at Model Validation. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (2019), ACM, pp. 45–54.

[7] Arango, A., Pérez, J., & Poblete, B. Hate Speech Detection is Not as Easy as You May Think: A Closer Look at Model Validation (extended version). Information Systems , 101584 (2020).

2 | https://github.com/aymeam/Datasets-for-Hate-Speech-Detection.

Conectando la visión y el lenguaje

JESÚS PÉREZ-MARTÍN Estudiante de Doctorado del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador del Instituto Milenio Fundamentos de los Datos.

BENJAMÍN BUSTOS Profesor Titular del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador Asociado del Instituto Milenio Fundamentos de los Datos.

JORGE PÉREZ Profesor Asociado del Departamento de Ciencias de la Computación de la Universidad de Chile e Investigador Asociado del Instituto Milenio Fundamentos de los Datos.

En este minuto más de 500 horas de video se están publicando en YouTube.[1] Además, el último *Digital Global Overview Report* estima que diariamente se visualizan mil millones de horas de video en la misma plataforma. Con los videos ganando tanta popularidad, YouTube Creator Academy[2] recomienda que las descripciones transmitan información valiosa para ayudar a los espectadores a encontrar videos en los resultados de búsquedas y comprender lo que mirarán.[3] En este sentido detalla: *"Las descripciones bien redactadas con las palabras clave correctas pueden ayudar a mejorar las visualizaciones y el tiempo de reproducción, ya que ayudan a que el video tenga una mayor visibilidad en los resultados de la búsqueda"*.

La forma de comunicación que más usamos los humanos es el lenguaje natural. Es entonces esencial que sistemas interactivos de Inteligencia Artificial (IA) y robots auxiliares sean capaces de generar texto automáticamente a partir de datos no lingüísticos. Reiter y Dale [1] caracterizan *Natural Language Generation* (NLG) como la producción de textos comprensibles a partir de una representación no lingüística subyacente de la información. Esta definición de NLG generalmente se asocia con la de *data-to-text generation*, asumiendo que la entrada exacta puede variar sustancialmente.

Hoy en día, la generación de texto a partir de una entrada perceptiva no estructurada —como una imagen sin

1 | Estadísticas de YouTube 2021 [infografía] - 10 datos fascinantes de YouTube: https://cl.oberlo.com/blog/estadisticas-youtube.

2 | Academia de creadores de YouTube, educación y cursos: https://creatoracademy.youtube.com.

3 | Consejos de YouTube para crear descripciones inteligentes: https://creatoracademy.youtube.com/page/lesson/descriptions?hl=es-419#strategies-zippy-link-1.

procesar o un video— se ha convertido en un desafío importante en el campo de investigación reciente que combina Visión y Lenguaje (V+L). Específicamente, obtener texto a partir de un video (*video-to-text*) puede efectuarse, principalmente, recuperando las descripciones más significativas de un corpus o generando una nueva descripción dado el video de contexto. Estas dos formas representan tareas esenciales para las comunidades de procesamiento de lenguaje natural y visión computacional, y son ampliamente conocidas como *video-to-text retrieval* y *video captioning/description*, respectivamente. Ambas tareas son sustancialmente más complejas que generar o recuperar una oración desde una única imagen. La información espacio-temporal presente en los videos introduce diversidad y complejidad respecto al contenido visual y a la estructura de las descripciones de lenguaje asociadas.

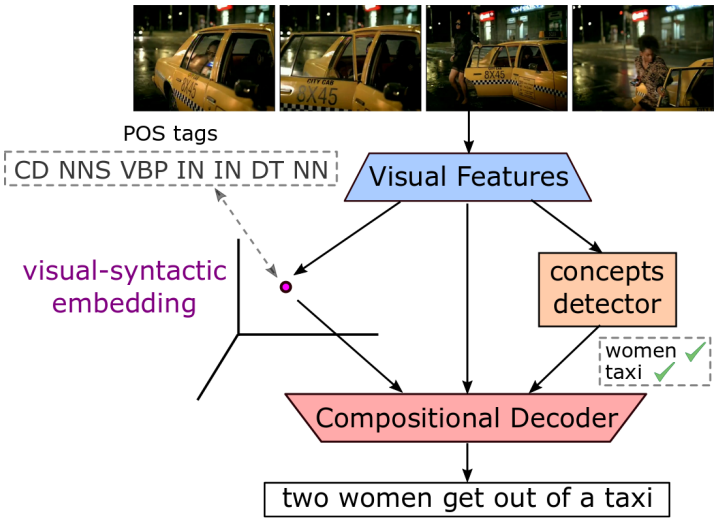


Figura 1. *Video captioning* usando un *embedding visual-sintáctico*. El método obtiene representaciones semánticas y sintácticas de alto nivel a partir de la representación visual del video. A continuación, el decodificador genera una oración a partir de ellos.

Con gran atención de ambas comunidades, V+L incluye otras tareas desafiantes que conectan o combinan las modalidades de la visión y el lenguaje, como *visual question-answering* (responder preguntas basadas en texto sobre imágenes), *caption-based image/video retrieval* (dados un texto y un grupo de imágenes, debemos recuperar la imagen que mejor se describe con el texto), *video generation from text* (generar un video plausible y diverso a partir de un texto de entrada) y *multimodal verification* (dada una o más imágenes y un texto, debemos predecir alguna relación semántica).

Sintaxis y semántica de un video

Es impresionante el progreso que los investigadores han logrado en conjuntos de datos específicos, pero a pesar de este progreso, la conversión de video a texto sigue siendo un problema abierto. Las técnicas del estado del arte aún están lejos de lograr un desempeño similar al humano. No obstante, las técnicas basadas en *deep learning* han logrado resultados prometedores, tanto para la generación de descripciones como para los métodos basados en la recuperación.

Como una tarea de generación de texto, el proceso de describir videos requiere predecir una secuencia de palabras semántica y sintácticamente correcta dado el contexto presente en el video. Los primeros trabajos en esta área siguieron la estrategia de, primero, detectar sujeto, verbo y objeto, formando un *triplete SVO*; y luego, generar una oración usando un conjunto reducido de plantillas que aseguran la correctitud gramatical. Este enfoque requiere que los modelos reconozcan a los sujetos y objetos que participan en la acción que debemos describir, logrando sus mejores resultados en videos cortos de entornos específicos, como deporte o cocina. En este tipo de videos, la cantidad de objetos y acciones que se debe detectar es limitada.

A partir de esta idea, podemos notar que para los modelos de *video captioning* dos aspectos esenciales son la identificación de contenidos visuales de forma explícita y la intención de producir oraciones correctas. Desarrollar técnicas que aborden alguno de estos aspectos ha guiado la investigación en los últimos años. Por un lado tenemos métodos que intentan conectar las palabras generadas a regiones específicas dentro del video (*visual grounding*) [2] y modelar las relaciones entre ellas [3, 4]. Mientras que por el otro tenemos métodos que consideran el aprendizaje de una representación sintáctica como un componente esencial de los enfoques de *video captioning* [5, 6, 7].

En el Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile nos encontramos desarrollando métodos de *video captioning* que extraen información valiosa sobre las posibles descripciones a partir de dimensiones implícitas en la información visual. Nuestros resultados recientes muestran que los videos contienen, además de la apariencia y el movimiento, información semántica y sintáctica que podemos extraer directamente de la información visual para guiar el proceso de generación de

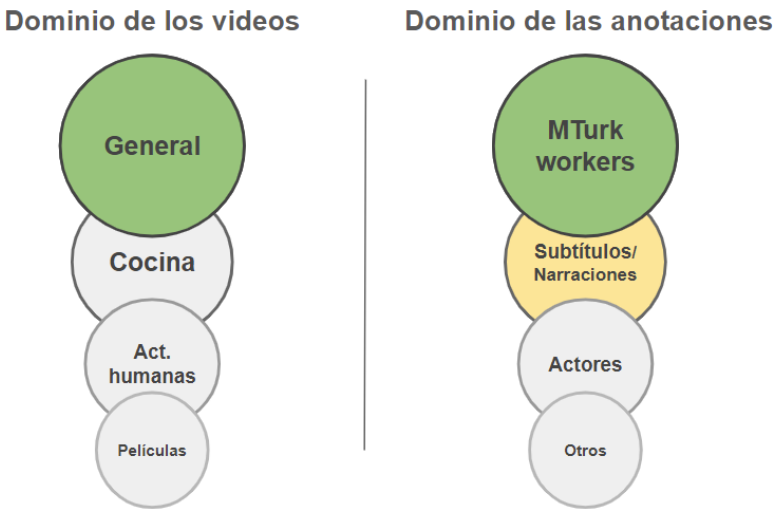


Figura 2. Para entrenar estos métodos, existen más de veinticinco conjuntos de datos anotados que podemos agrupar según el dominio de video y de diferentes formas se obtienen las descripciones.

texto. Sin embargo, tener una fuerte dependencia de sólo una de ellas puede perjudicar el rendimiento de los modelos, produciendo brechas semánticas u oraciones sintácticamente incorrectas. Por eso, para nosotros es fundamental determinar cómo fusionar estos canales de información de forma adaptativa. En dos artículos que presentamos recientemente en las conferencias internacionales ICPR 2020 [8] y WACV 2021 [7], proponemos estrategias efectivas que combinan técnicas de recuperación y generación para evitar estas brechas y aprender representaciones de forma multimodal.

Específicamente, en nuestro trabajo propusimos un modelo llamado *Visual-Semantic-Syntactic Aligned Network* (SemSynAN) [7]. Este modelo basado en el esquema *encoder-decoder* es capaz de generar oraciones con semántica y sintaxis más precisas. Una de las innovaciones más importante fue proponer una técnica de recuperación de secuencias de etiquetado gramatical (POS por sus siglas en inglés)[4] provenientes de las descripciones de video, para generar representaciones sintáctica de alto nivel directamente desde la información visual (ver Figura 1). Con este trabajo mostramos que prestar atención especial a la sintaxis puede mejorar sustancialmente la calidad de las descripciones. Además, nuestro método garantiza la relación contextual entre las palabras de la oración, controlando el significado semántico y la estructura sintáctica de las descripciones generadas [7].

Conjuntos de datos de entrenamiento

V+L es un área de investigación recientemente planteada. Aunque ha recibido mucha atención en los últimos años, todavía se necesitan más datos para entrenar y evaluar nuevos modelos. Para distinguir con precisión entre diferentes clases de información visual, los modelos deben entrenarse a escala, con descripciones diversas y de alta calidad que contengan una amplia variedad de videos.

La creación de conjuntos de datos a gran escala requiere un esfuerzo humano significativo y costoso para su anotación, ya que recopilar una gran cantidad de referencias puede llevar mucho tiempo y ser difícil para los idiomas menos comunes. Debido a esto —y a pesar de que la mayor cantidad de *datasets* ha sido creada a partir de videos de dominio general anotados por humanos (ver Figura 2)—, el *dataset* más grande a la fecha ha sido creado a partir de la generación automática de subtítulos y narraciones (*dataset HowTo100M* [9]).

Con trabajos recientes como CLIP [10], el campo se ha movido a nuevas arquitecturas y modelos (*transformers* [11], *pre-training* y *fine-tuning* ahora se han convertido en el enfoque dominante). Básicamente, estos estudios han mostrado los beneficios de preentrenar los modelos para tareas de V+L y luego ajustar el modelo para tareas específicas.

Por ejemplo, podemos aprender previamente representaciones genéricas a partir de tareas de V+L, como *visual question-answering* o *cross-modal retrieval* (recuperación a través de diferentes modalidades, como imagen-texto, video-texto y audio-texto), y luego ajustar su codificación visual en la tarea de *video captioning*. Esta técnica requiere un gran volumen de datos para aprender dicha representación en un espacio común entre la información visual y textual. Por ejemplo, para entrenar CLIP se usaron 400 millones de pares (imagen, texto) obtenidos de Internet.

Los modelos de *video captioning* basados en esta estrategia, como COOT [12],

4 | Categorizar y etiquetar palabras de acuerdo a categorías léxicas: https://www.nltk.org/book/ch05.html.

generalmente son preentrenados sobre datos obtenidos de forma automática de los subtítulos y narraciones (ver Figura 2) que brindan las plataformas de video *online*. Sin embargo, un gran inconveniente de este tipo de corpus es la gran cantidad de *tokens* desconocidos (términos que no se pueden asociar a una palabra del vocabulario) que se producen. Por ejemplo, en *HowTo100M* [9] sólo el 36,64% de las palabras del vocabulario (217.361 de las 593.238 palabras únicas) aparecen en el vocabulario ampliamente utilizado *GloVe-6B*[5] [13], que tiene 400.000 *tokens*. Este alto nivel de "ruido" en los subtítulos es un aspecto interesante del proceso de entrenamiento que debemos aprender a aprovechar.

Conclusiones

Hace diez años pocos hubieran imaginado que sistemas de V+L serían capaces de generar descripciones textuales plausibles como las que se logran hoy. Los investigadores han logrado modelos que extraen, hasta cierto sentido, información espacio-temporal compleja presente en los videos. No obstante, una característica de la que carecen los sistemas actuales es la capacidad de representar el *sentido común*, por lo que aún queda mucho para comprender y representar la diversidad en cuanto a contenido visual de los videos y la estructura de sus descripciones textuales.

Es muy probable que en el futuro la cantidad de videos que los buscadores deberán procesar sea mayor que en la actualidad. Siempre ha sido así y al día de hoy, que la pandemia nos incita a ser más digitales, no hay ningún indicador que señale que esta dinámica cambiará. Al contrario, esta tendencia aumentará la necesidad de transformar la información visual en descripciones textuales que la resuman, verbalicen y simplifiquen de forma precisa. ▪

REFERENCIAS

[1] Reiter, E. & Dale, R. Building natural language generation systems. (Cambridge University Press, 2000).

[2] Pan, B. et al. Spatio-Temporal Graph for Video Captioning with Knowledge Distillation. In Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 10870–10879 (2020).

[3] Zhou, L., Kalantidis, Y., Chen, X., Corso, J. J. & Rohrbach, M. Grounded Video Description. In Proc. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 6571–6580 (IEEE, 2019).

[4] Zhang, Z. et al. Object Relational Graph with Teacher-Recommended Learning for Video Captioning. In Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 13278–13288 (2020).

[5] Hou, J., Wu, X., Zhao, W., Luo, J. & Jia, Y. Joint Syntax Representation Learning and Visual Cue Translation for Video Captioning. In Proc. IEEE International Conference on Computer Vision (ICCV) (2019).

[6] Wang, B. et al. Controllable Video Captioning with POS Sequence Guidance Based on Gated Fusion Network. In Proc. IEEE International Conference on Computer Vision (ICCV) (2019).

[7] Pérez-Martín, J., Bustos, B. & Pérez, J. Improving Video Captioning with Temporal Composition of a Visual-Syntactic Embedding. In Proc. IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) (2021).

[8] Pérez-Martín, J., Bustos, B. & Pérez, J. Attentive Visual Semantic Specialized Network for Video Captioning. In Proc. 25th International Conference on Pattern Recognition (2020).

[9] Miech, A. et al. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In Proc. IEEE/CVF International Conference on Computer Vision (ICCV) 2630–2640 (IEEE, 2019).

[10] Radford, A. et al. Learning Transferable Visual Models From Natural Language Supervision. (2021).

[11] Vaswani, A. et al. Attention is all you need. In Proc. 31st International Conference on Neural Information Processing Systems 6000–6010 (Curran Associates Inc., 2017).

[12] Ging, S., Zolfaghari, M., Pirsiavash, H. & Brox, T. COOT: Cooperative Hierarchical Transformer for Video-Text Representation Learning. In Proc. Conference on Neural Information Processing Systems (2020).

[13] Pennington, J., Socher, R. & Manning, C. D. Glove: Global vectors for word representation. IN EMNLP (2014).

5 | Proyecto Stanford GloVe (vectores globales) que usa aprendizaje no supervisado para obtener vectores representativos para un gran conjunto de palabras: https://nlp.stanford.edu/projects/glove/.

¿Cómo la inteligencia artificial puede ayudar al e-commerce?

EQUIPO IMPRESEE

CAMILA ÁLVAREZ Chief Technology Officer (CTO)
JUAN MANUEL BARRIOS Chief Executive Officer (CEO)
MAURICIO PALMA LIZANA Chief Financial Officer (CFO)
JOSÉ M. SAAVEDRA Chief Research Officer (CRO)

El *e-commerce* es un mercado mundial que se ha vuelto indispensable en el último tiempo. Basa su éxito en la satisfacción de los usuarios que necesitan comprar y en el consecuente incremento de las ventas en las tiendas. Es un contexto en el que modelos de Inteligencia Artificial (IA) y Ciencia de Datos se vuelven cada vez más relevantes tanto para atraer visitantes, mostrar productos relevantes, diseñar campañas de marketing, etc.

Impresee es una empresa SaaS que ofrece servicios de alta tecnología para el *e-commerce*. Tenemos clientes en diversas partes del mundo como Estados Unidos, Canadá, Alemania, China y Sudamérica, entre otras. Fundamos Impresee con el deseo de desarrollar servicios que combinen áreas de inteligencia artificial, visión por computadora, procesamiento del lenguaje natural y ciencia de datos para lograr soluciones innovadoras que mejoren el *e-commerce*.

La investigación científica la hacemos en Impresee eCommerce Labs[1], donde trabajamos en conjunto con retailers y colaboradores académicos para hacer investigación aplicada para el *e-commerce* y crear tecnología novedosa usando datos reales de ambientes reales. Nos enorgullece haber sido reconocidos por la comunidad científica en el año 2015 con el Premio a Mejor Demo basada en Visión por Computadora en la IEEE International Conference on Computer Vision (ICCV).

IA en la industria del *e-commerce*

En un principio nos enfocamos principalmente en mejorar la experiencia de los consumidores a través de un motor de búsqueda moderno, eficiente y efectivo. Potenciamos la tradicional búsqueda por texto con modelos basados en visión por computadora para permitir la búsqueda de productos por medio de fotos. Además desarrollamos una novedosa modalidad de consulta: la búsqueda basada en dibujos (*sketch-based image retrieval*), que tiene sus raíces en la tesis de doctorado

1 | https://impresee.com/ecommerce-labs/.

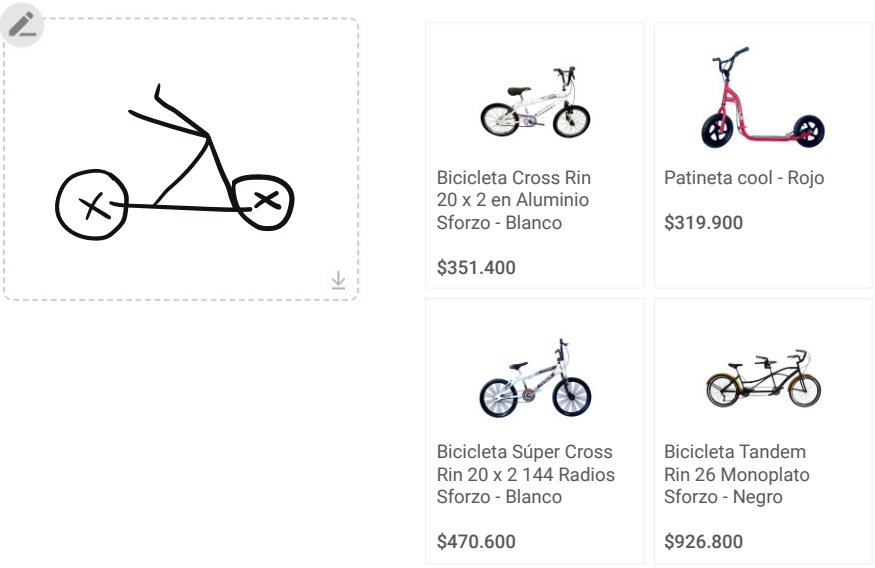


Figura 1. Resultado de búsqueda a través de dibujos.

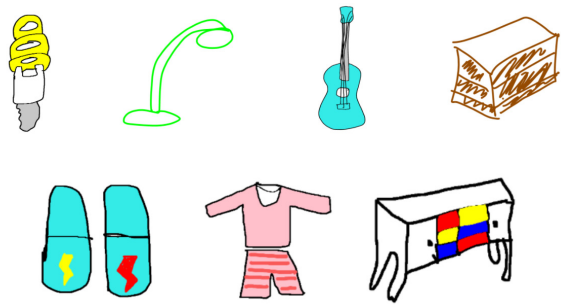


Figura 2. Ejemplo de consultas tipo sketch con color.

de José M. Saavedra. Por ejemplo, la Figura 1 muestra el resultado de búsqueda de un dibujo.

Luego observamos que la gran cantidad de datos que capturamos de una tienda (tráfico, visitantes, ventas) y los datos que generamos desde las búsquedas (consultas, fotos, dibujos, clicks) se complementan para formar un conjunto valioso para distintas áreas de la tienda. Trabajamos en crear métodos para analizar datos y generar información útil para la tienda, como el comportamiento de los visitantes y su apreciación de los productos para apoyar las áreas de marketing y ventas.

Nos dimos cuenta que los *dashboards* no son suficientes para generar valor, sino que debemos ir más allá, apoyando las conclusiones y automatizando las acciones posteriores. Por ejemplo, mediante análisis de datos es posible localizar productos con un buen potencial de ventas y que tienen baja visibilidad. Luego con *machine learning* es posible generar modelos para identificar las mejores acciones de marketing a realizar en una tienda para aumentar sus ventas.

Los sistemas de recomendación son otra arista que estamos trabajando. En esta línea investigamos modelos para integrar recomendadores y buscadores. En el buscador el visitante escribe lo que desea comprar y, además, en nuestro caso, puede subir una foto o dibujarlo. Según nuestros análisis, es tres veces más probable que un usuario que usa el buscador compre un producto comparado con uno que solo navega por el sitio. Por tanto, analizando la gran cantidad de imágenes de un catálogo (fotos de *influencers*, catálogos de temporada, etc.) junto con las imágenes de búsqueda, es posible entrenar modelos basados en redes convolucionales que permitan recomendar de forma automática prendas de vestir, dada una prenda de consulta. En términos técnicos, se trata de modelar un espacio de características donde las prendas complementarias se acercan entre sí.

Trabajos de investigación recientes

Trabajar en investigación en casos reales nos permite detectar problemas anticipadamente y desarrollar soluciones que tienen alto impacto. Así, en los siguiente párrafos describiremos tres trabajos aceptados para presentación oral en workshops de la International Conference on Computer Vision and Pattern Recognition (CVPR) 2021.

Color-Sketch-based Image Retrieval

Luego de lanzar el buscador basado en dibujos, observamos que en contextos como Fashion & Apparel y Home-Decor los usuarios debieran poder agregar información a la consulta como color y texturas. Así comenzamos investigar sobre cómo modelar dibujos incluyendo color y texturas y cómo compararlos con las imágenes de productos. La Figura 2 muestra algunas consultas. El resultado se plasmó en el

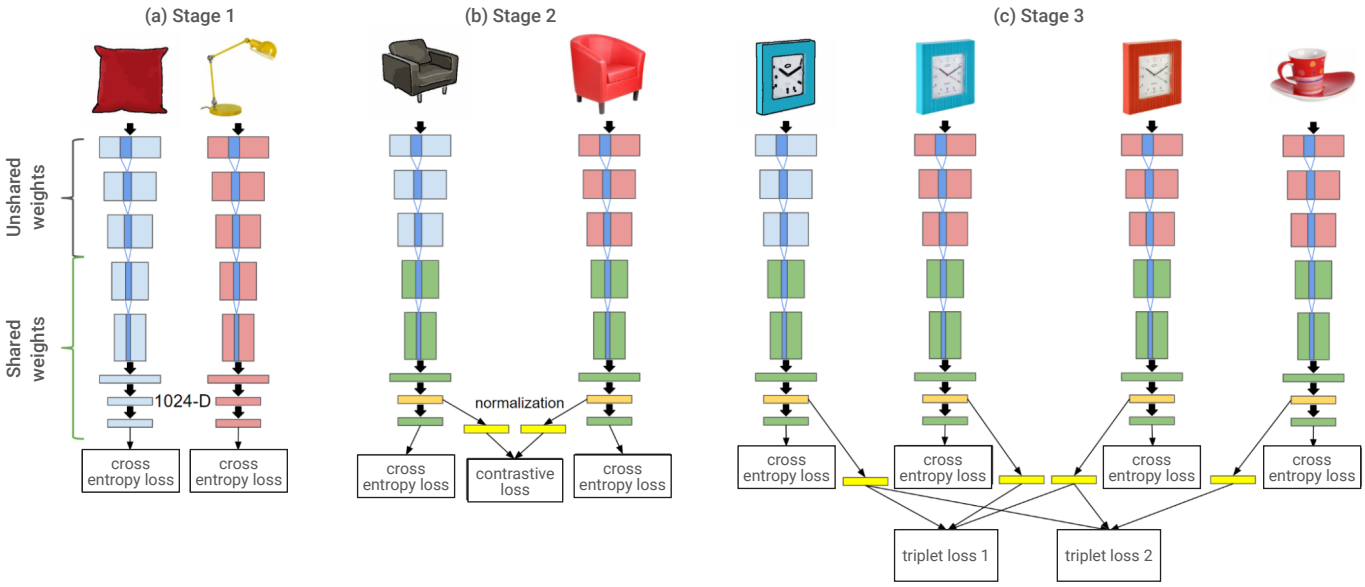


Figura 3. Arquitectura Sketch-QNet.

Figura 4. Ejemplo de resultado en un espacio de características de 8 dimensiones.

trabajo titulado "Sketch-QNet: A Quadruplet ConvNet for Color Sketch-based Image Retrieval"[2], que fue aceptado recientemente en el 1st Workshop on Sketch-Oriented Deep Learning (SketchDL) de CVPR 2021.

En ese trabajo proponemos una nueva arquitectura de red neuronal convolucional a la que llamamos Sketch-QNet para resolver el problema de *color-sketch based image retrieval*. La Figura 3, muestra la arquitectura propuesta que es entrenada por medio de cuadrupletas (cuatro pares de entrada). Con esto, extendemos la búsqueda de imágenes basada en dibujos a consultas que incluyan información de color. El objetivo es generar un espacio de características que pueda contener sketches con color y fotografías al mismo tiempo. El entrenamiento se realiza de modo que una consulta en forma de *sketch* con color quede muy cerca, en el espacio inducido, de fotos que expresen la misma información semántica de la consulta. Fotos que comparten solamente el concepto pero difieren en color deben quedar un poco más lejos. Finalmente, fotos con una semántica diferente a la consulta deben estar mucho más lejos de ella.

Representaciones compactas para Sketch-based Image Retrieval

La eficiencia de los espacios de características juegan un rol muy importante en sistemas reales. Comúnmente los vectores característicos para la recuperación de imágenes son de alta dimensión, variando entre 256 a 4096 dimensiones. Esto resulta impráctico para soportar catálogos con millones de imágenes, impactando negativamente el tiempo de búsqueda y la memoria requerida. Decidimos investigar modelos que nos permitan crear espacios reducidos (por ejemplo, menos de 10 dimensiones) sin perder efectividad. En esta línea desarrollamos el trabajo titulado "Compact and Effective Representations for Sketch-based Image Retrieval"[3], recientemente aceptado en el 1st Workshop on Sketch-Oriented Deep Learning (SketchDL) de CVPR 2021.

2 | https://impresee.com/sketch-qnet/.

3 | https://impresee.com/sketch-based-image-retrieval/.

En este trabajo, observamos que los espacios de características actuales forman una topología local que puede ser aprovechada por métodos de reducción de dimensión que preserven la localidad. Nuestros experimentos muestran que el uso de UMAP como método de reducción permite obtener espacios de baja dimensión (por ejemplo, 4 u 8) incrementando, además, la efectividad del método original. Este incremento en la efectividad se debe a que al preservar la localidad se extraen características relevantes a la vecindad de cada punto, descartando características ruidosas. Así, objetos que comparten una semántica similar tienden a ser atraídos entre sí. La Figura 4 muestra algunos resultados de recuperación de imágenes usando *sketches*, en un espacio reducido a 8 dimensiones. Estos resultados representan un nuevo estado del arte en este contexto.

Extracción de atributos visuales

Los atributos visuales juegan un rol muy importante en la búsqueda de productos. La manera tradicional de extraer estos atributos es entrenando una red CNN que se ajusta a un conjunto determinado de clases. Esta aproximación no escala a problemas donde los atributos de interés pueden cambiar con frecuencia. En nuestro trabajo titulado "Scalable Visual Attribute Extraction through Hidden Layers of a Residual ConvNet"[4] proponemos un método para extraer atributos visuales de imágenes, particularmente como las que podemos encontrar en un *e-commerce*, aprovechando la capacidad que tienen las capas ocultas de una red convolucional para aprender características visuales (ver Figura 5).

(a) Color (b) Textura (b) Color y Textura

Figura 5. Agrupación no supervisada de imágenes por atributos visuales.

Proyectos en curso

Además, mantenemos diversos trabajos de investigación activos con participación de estudiantes de pre y postgrado, y colaboradores académicos nacionales e internacionales. Aquí algunos de estos trabajos.

Unsupervised Learning for Sketch-Based Image Retrieval

Muchos de los modelos exitosos de visión por computador se basan en tener una gran cantidad de datos etiquetados. Sin embargo, en ambientes reales no es práctico etiquetar tal cantidad de datos. Así, con Javier Morales, memorista del Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile, y Nils Murrugarra, investigador de Snap, estamos trabajando en métodos autosupervisados para el aprendizaje de representaciones visuales (*embeddings*) en el contexto de recuperación de imágenes. Además, apuntamos a crear modelos híbridos que aprendan a partir de datos etiquetados en forma supervisada y que al mismo tiempo se alimenten de datos no etiquetados para mejorar la generalización.

ColoSketch2Photo

Convertir una expresión abstracta como lo es un dibujo a un objeto fotorrealista es de gran importancia en el *e-commerce*, especialmente en los rubros de personalización de productos. Los usuarios podrían dibujar lo que necesitan y obtener una representación real de esa abstracción. Junto a Diego Donoso, estudiante de magíster del DCC, estamos trabajando en diseñar modelos que permitan explotar la diversidad de dibujos que representan la semántica de una consulta y producir imágenes fotorealistas guiados por atributos adicionales como colores y texturas.

Invitación a colaborar

En Impresee eCommerce Labs buscamos producir conocimiento que permita mejorar el *e-commerce* tanto para los vendedores como para los mismos usuarios. Nos gusta colaborar con investigadores y formar equipos. Te invitamos a formar parte de estos y otros proyectos que ¡siempre tendrán un alcance nada menos que global! ▪

4 | https://impresee.com/scalable-visual-attribute-extraction/.