



fcfm

Ciencias de la
Computación
FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

REVISTA DEL DEPARTAMENTO DE CIENCIAS DE LA
COMPUTACIÓN DE LA UNIVERSIDAD DE CHILE

Bite

DE CIENCIA

EDICIÓN N°25 / SEGUNDO SEMESTRE AÑO 2023



Command Prompt :

INTELIGENCIA ARTIFICIAL GENERATIVA

**Un recorrido por los
modelos de lenguaje:**
Desde Shannon a GPT-4
/ Felipe Bravo Márquez

**Inteligencia artificial,
ciudadanía digital y
pedagogía**
/ Martín Cáceres

**Niñas Pro descifró el
código:** La programación
no tiene género
/ Niñas Pro

Contenidos

1 Editorial
/ Federico Olmedo

Investigación destacada

2 Mejorando la adopción de la privacidad diferencial en lenguajes de programación
/ Damián Arquez y Matías Toro

Premio Turing

10 Robert Metcalfe, coinventor de Ethernet, el estándar para redes de área local
/ José Urzúa

Inteligencia artificial generativa

16 Un recorrido por los modelos de lenguaje: Desde Shannon a GPT-4
/ Felipe Bravo Márquez

28 Inteligencia artificial, ciudadanía digital y pedagogía
/ Martín Cáceres

35 ¿Cómo navegar el camino hacia la ética en IA?
/ Claudia López Moncada, Gabriela Arriagada Bruneau y Alexandra Davidoff

44 GPT y agentes: Conversando con tus apps
/ José Manuel Peña

Computación y sociedad

50 Niñas Pro descifró el código: La programación no tiene género
/ Niñas Pro

Doctorado

56 El Doctorado en Computación de la Universidad de Chile
/ Gonzalo Navarro

Estudiantes DCC

59 Juglar Díaz / Maximiliano Canché / Pablo Estefó / Eduardo Sánchez Villegas / Gabriel Iturra Bocaz / Daniela Campos / Fabiola Rivera Salgado



COMITÉ EDITORIAL

Andrés Abeliuk
María Cecilia Bastarrica
Eduardo Graells-Garrido
Claudio Gutiérrez
Alejandro Hevia
Ana Gabriela Martínez
Jocelyn Simmonds
Iván Sipirán

EDITOR GENERAL

Federico Olmedo

EDITORA PERIODÍSTICA

Ana Gabriela Martínez

PERIODISTA

Karin Riquelme

DISEÑO

Paulette Filla

FOTOGRAFÍAS E IMÁGENES

Comunicaciones DCC

Revista BITS de Ciencia del Departamento de Ciencias de la Computación de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile se encuentra bajo Licencia Creative Commons Atribución-NoComercial-Compartir-Igual 3.0 Chile. Basada en una obra en www.dcc.uchile.cl



Revista Bits de Ciencia N°25

ISSN 0718-8005 (versión impresa)


www.dcc.uchile.cl/revista

ISSN 0717-8013 (versión en línea)

Departamento de Ciencias de la Computación

Avda. Beauchef 851, 3° piso,
edificio norte. Santiago, Chile.
837-0459 Santiago

 www.dcc.uchile.cl

 56 22 9780652

 bitsdeciencia@dcc.uchile.cl

 / [dccuchile](https://www.dcc.uchile.cl)

El contenido de los artículos publicados en esta Revista, son de exclusiva responsabilidad de sus autores y no reflejan necesariamente el pensamiento del Departamento de Ciencias de la Computación de la Universidad de Chile.



Editorial

FEDERICO OLMEDO

Editor General

Revista Bits de Ciencia



La reciente irrupción de la inteligencia artificial generativa no ha pasado desapercibida en absoluto. De hecho, a ChatGPT le tomó sólo 5 días alcanzar 1 millón de usuarios. Este fenómeno se debe, en gran medida, al inminente impacto de dicha tecnología: su capacidad para generar contenido, como texto o imágenes, ha abierto un abanico de oportunidades, prometiéndonos un aumento exponen-

cial de la productividad y obligándonos a repensar la manera en la que llevamos haciendo una variedad de tareas desde hace décadas.

No obstante, sacar el máximo provecho de esta tecnología haciendo, a la vez, un uso responsable de la misma, requiere abordar múltiples interrogantes. Por ejemplo, ¿los actuales LLMs (*Large Language Models*) como ChatGPT o Google Bard son el resultado de un mero golpe de suerte, o el fruto de múltiples desarrollos sucesivos?, ¿qué principios son relevantes para abordar los problemas éticos asociados a esta tecnología?, ¿cómo puede integrarse con otras herramientas para resolver problemas de manera más efectiva?, y en el contexto particular de la educación, ¿la presencia de esta tecnología realmente significa un deterioro en los procesos de aprendizajes, como muchos vaticinan?

Para abordar estas interrogantes, hemos invitado a especialistas de diferentes ámbitos, incluyendo la academia, la industria y el sector público, quienes comparten sus opiniones y perspectivas en el Tema Central de la Revista.

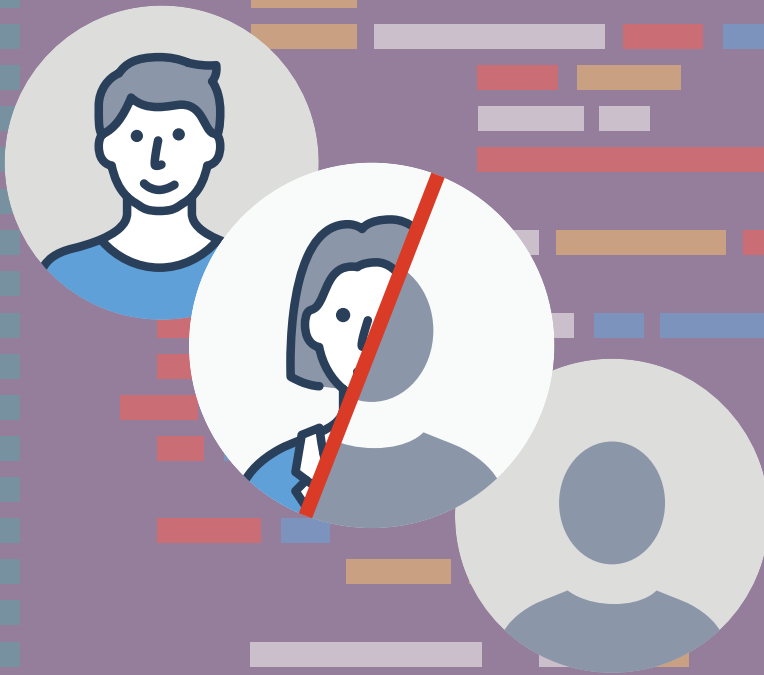
Por su parte, en la sección “Investigación destacada” Damián Arquez y Matías Toro nos cuentan cómo los lenguajes de programación pueden contribuir a la adopción de la privacidad diferencial. En la sección “Premios Turing” José Urzúa presenta el galardón otorgado a Robert Metcalfe por su contribución al desarrollo de Ethernet. Además, en la sección “Computación y sociedad” presentamos a Niñas Pro, corporación que lleva varios años trabajando para disminuir la brecha de género en las carreras STEM a través de diversas actividades.

Por último, Gonzalo Navarro hace un recorrido por el programa de Doctorado en Computación del DCC de la Universidad de Chile, y varios egresados y egresadas recientes del Departamento comparten sus trabajos finales en la sección “Estudiantes DCC”.

Esperemos que disfruten de la Revista. Como siempre, cualquier consulta o sugerencia, pueden escribirnos a bitsdeciencia@dcc.uchile.cl. ■



Mejorando la adopción de la privacidad diferencial en lenguajes de programación



DAMIÁN ARQUEZ

Estudiante de Doctorado en Computación de la Universidad de Chile. Magíster en Ciencias mención Computación por la Universidad de Chile. Líneas de investigación: lenguajes de programación, privacidad de datos.

✉ darquez@dcc.uchile.cl



MATÍAS TORO

Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile. Doctor en Ciencias mención Computación por la Universidad de Chile. Líneas de investigación: lenguajes de programación, privacidad de datos y desarrollo de software.

✉ mtoro@dcc.uchile.cl



RESUMEN. Frente a la creciente cantidad de datos personales que se almacenan en bases de datos, la privacidad de los individuos se ha vuelto un tema de gran importancia. Una técnica que ha ganado mucha atención en los últimos años es la *privacidad diferencial*, basada en una definición formal de privacidad que garantiza que un atacante no pueda distinguir si un individuo participó o no en un conjunto de datos. A pesar de haber un interés en integrar privacidad diferencial en distintas áreas de la computación y estadística, no siempre es fácil hacerlo y mucho menos probar que un programa es, en efecto, diferencialmente privado. De hecho, existen muchos casos de publicaciones que, incluso siendo revisadas por expertos, fallan en implementar privacidad diferencial correctamente. Frente a esto, durante la última década, han existido esfuerzos por integrar privacidad diferencial en lenguajes de programación, con el fin de proveer garantías de privacidad de manera automática. En este artículo, presentamos Jazz, un lenguaje de programación que permite razonar de manera estática sobre privacidad diferencial. Además, abordamos el problema de la adopción de sistemas de tipos estáticos, proponiendo un sistema de tipos gradual que permita integrar privacidad diferencial de manera incremental en programas existentes. Nuestro primer paso en esta dirección es GSoul, un lenguaje de programación que permite razonar de manera gradual sobre sensibilidad de datos.

Esta investigación fue desarrollada por los autores del presente artículo, en colaboración con David Darais, Chike Abuah, Joseph P. Near, Federico Olmedo y Éric Tanter.

Existe una inherente tensión entre privacidad y precisión: no se puede lograr, simultáneamente, perfecta privacidad y precisión.

Los datos

La información almacenada en bases de datos crece cada día, especialmente debido a áreas emergentes como big data o machine learning. Todos estamos generando nuevos datos todo el tiempo: cuando salimos de compras, cuando escuchamos música o vemos películas, cuando vamos al médico, cuando usamos nuestra cuenta bancaria, etc. Estos datos proveen un sinfín de oportunidades: la personalización de servicios en Amazon o Netflix; construir modelos computacionales para automatizar tareas, como clasificadores de correos, o traductores de texto; apoyar el desarrollo científico; guiar la toma de decisiones públicas y mejorar su transparencia, etc. Los datos nunca habían tenido un rol tan importantes como ahora.

Privacidad

El uso de datos personales tiene también sus riesgos, ya que éstos podrían tener información sensible o confidencial de las personas que puede ser filtrada, por ejemplo, información médica, financiera, amorosa, de preferencias artísticas, etc. En otras palabras, datos privados que nos exponen y vulneran como personas. Esto puede sonar alejado de la realidad, pero a la fecha ya se han acumulado una considerable cantidad de casos emblemáticos en que la privacidad de individuos es violada, muchas veces sin querer o saber.

El primero es el caso de Netflix, que realizó en el 2007 una competencia abierta para mejorar su sistema de recomendación. El premio era un millón de dólares, y los participantes tenían acceso a

un conjunto de datos de entrenamiento *anonimizado*, es decir, atributos identificadores como el nombre fueron eliminados. El problema se presentó luego de que al poco tiempo se publicó el conjunto de datos de Netflix de-anonimizado, al cruzar los datos con datos públicos de preferencias de películas de IMDB, se logró asociar nombres concretos con sus preferencias de Netflix.

Otro caso famoso fue en 1997, cuando William Weld, gobernador del estado de Massachusetts, aprobó la liberación de los registros médicos de funcionarios públicos, utilizando anonimización. Dos días después, Latanya Sweeney, en ese entonces una estudiante de doctorado del MIT, le envió un correo con todos sus registros médicos. ¿Cómo lo logró? Cruzó la información de registros médicos con el padrón electoral, usando el código postal, año de nacimiento y género como se ve en la Figura 1.

Casos como los anteriores, ha generado un gran interés en la comunidad científica respecto a técnicas que nos ayuden a preservar (y garantizar formalmente) la privacidad de los datos.

Técnicas de privacidad

Para prevenir violaciones a la privacidad existen dos grandes modelos. El primero y más conocido, es el basado en un conjunto predefinido de ataques. Decimos que la publicación de un conjunto de datos preserva privacidad si es inmune a ciertos ataques. La técnica que se usa para prevenir estos ataques es llamada anonimización pero, como vimos anteriormente, su efectividad está sujeta a la información auxiliar que pueda tener un atacante.



Figura 1. Latanya Sweeney cruzó información anonimizada de registros médicos con el padrón electoral utilizando el código postal, año de nacimiento y género, logrando asociar datos de medicamentos, diagnósticos y procedimientos médicos con el gobernador.

El segundo modelo está basado en el principio de la desinformación. Decimos que un conjunto de datos preserva la privacidad si al observarlo, un atacante gana muy poca información adicional respecto a la información que ya tenía. Aquí la técnica que se usa es llamada *privacidad diferencial* (DP, por sus siglas en inglés) [1], y es actualmente el enfoque más efectivo para lograr privacidad ya que es inmune a la información auxiliar que podría tener un atacante.

Privacidad diferencial

Privacidad diferencial es una técnica que garantiza privacidad a través de una definición formal basada en estadística. Se dice que una computación aleatoria es ϵ -DP, si para dos bases de datos que difieren en un individuo, los resultados de aplicar la computación aleatoria a ambas bases son "indistinguibles". Formalmente, un mecanismo \mathcal{M} es ϵ -DP si para cualquier par de bases de datos D

y D' que difieran en un individuo, y para cualquier subconjunto S de los posibles resultados de \mathcal{M} , se cumple:

$$\frac{\Pr[\mathcal{M}(D) \in S]}{\Pr[\mathcal{M}(D') \in S]} \leq e^\epsilon$$

La técnica más común para implementar privacidad diferencial es agregar ruido aleatorio estadístico al resultado de una computación. El parámetro ϵ , a menudo llamado presupuesto de privacidad, controla el equilibrio entre privacidad y precisión: a menor ϵ , hay más ruido y por ende el cómputo es menos preciso pero más privado; análogamente, a mayor ϵ hay menos ruido y por ende el cómputo es más preciso pero menos privado. Existe una inherente tensión entre privacidad y precisión: no se puede lograr, simultáneamente, perfecta privacidad y precisión.

Sin embargo, no cualquier cantidad de ruido es adecuada: mucho ruido puede destruir completamente la utilidad de los datos. En el artículo seminal so-

bre DP [1], Cynthia Dwork probó que el ruido adecuado debe ser proporcional a la *sensibilidad* de la computación, que intuitivamente corresponde a cuánto puede la función magnificar la distancia entre dos entradas. Por ejemplo, una función que recibe y produce números reales es s -sensible si, para cualquier par de entradas x e y , se cumple:

$$|f(x) - f(y)| \leq s \cdot |x - y|$$

Si consideramos dos funciones $f(x) = x + 1$ y $g(x) = x + x$, utilizando la definición anterior, podemos deducir que f es 1-sensible y g es 2-sensible. Sabiendo la sensibilidad de una función, podemos crear una versión diferencialmente privada de esta agregando ruido aleatorio proporcional a su sensibilidad. Por ejemplo, $F(x) = f(x) + \text{noise}(1)$ y $G(x) = g(x) + \text{noise}(2)$ son versiones diferencialmente privadas de f y g , respectivamente. La forma más básica de agregar ruido es utilizando la distribución de Laplace, específicamente fijando $\text{noise}(s) = \text{laplace}(s/\epsilon)$.

La privacidad diferencial ha ganado mucha atención en organizaciones como Google [2], Apple [3], y el censo de Estados Unidos [4, 5]. Sin embargo, aún hay muchas preguntas y desafíos. Por ejemplo, ¿cuál es un valor de ϵ adecuado?, o ¿cómo podemos verificar que un cómputo más complejo realmente satisface ϵ -DP?

¿Cómo razonar acerca de DP?

Varios sistemas que realizan análisis de privacidad diferencial han demostrado ser efectivos [6–10], y en el campo del aprendizaje automático se han diseñado varios algoritmos utilizando privacidad diferencial [11–18].

Actualmente, existen dos principales enfoques basados en lenguajes de programación de propósito general para



razonar acerca de privacidad diferencial: aquellos basados en *sistemas de tipos* [19], y otros basados en *lógica de programas* [20, 21]. Aunque estos últimos son más expresivos, en general son más difíciles de automatizar y mucho más complejos que los sistemas de tipos. Nuestro enfoque, en cambio, busca proveer herramientas automáticas y ligeras (*lightweight* en inglés) de verificación de privacidad diferencial, por lo que nos orientamos hacia sistemas de tipos.

Los sistemas de tipos ayudan a los programadores a verificar automáticamente un programa *antes de su ejecución*, asegurando que ciertas clases de errores nunca ocurran en tiempo de ejecución. Por ejemplo, un verificador de tipos básico prevendría que el programa `1 + True` llegase a ejecutarse, ya que no tiene sentido sumar un número con un Booleano. Durante la compilación de un programa, un verificador de tipos intenta clasificar las expresiones del programa en términos del tipo de valores que producen, y si no puede hacerlo, informa sobre un error de tipos estático. Uno de los principales beneficios de utilizar esta técnica es que los tipos son esenciales en el diseño de sistemas modulares. Aplicado a privacidad diferencial, nos permite razonar de manera composicional sobre esta disciplina.

FUZZ

En 2010, se publica el primer lenguaje diseñado precisamente para razonar sobre sensibilidad y privacidad diferencial: Fuzz [19]. La idea principal se funda en que el desarrollo de programas se lleve a cabo desde un principio en torno a la noción de privacidad y, por lo tanto, cualquier programa escrito en Fuzz es, por construcción, diferencial-

Nuestro enfoque [...] busca proveer herramientas automáticas y ligeras [...] de verificación de privacidad diferencial, por lo que nos orientamos hacia sistemas de tipos.



mente privado. Fuzz logra proveer esta garantía a través de un sistema de tipos que permite razonar sobre la sensibilidad de los programas y expresiones con ruido. Si un programa es válido en Fuzz, entonces no sólo sabremos los tipos de datos involucrados, sino que también podemos saber la sensibilidad de cada expresión.

Los autores de Fuzz diseñan el lenguaje con una idea clave: la sensibilidad es crucial para la privacidad diferencial. Por lo mismo, el sistema de tipos de

Fuzz se inspira en una técnica llamada *tipos lineales* [22] para modelar un seguimiento de la sensibilidad de un programa. Dado el pequeño núcleo de funcionalidades de Fuzz, el sistema de tipos aproxima la sensibilidad al número de veces que una variable es usada en un programa.¹ Por ejemplo, en Fuzz existen dos formas de darle un tipo a una función: $\mathbb{R} \rightarrow \mathbb{R}$ describe una función que recibe un número real como argumento, retorna un número real, y utiliza sin restricciones su argumento, i.e. una función ∞ -sensible en su

¹ Esto no siempre es posible pues, por ejemplo, en la expresión $x * x$, x se utiliza 2 veces pero la sensibilidad es infinita. Existen otros mecanismos para razonar sobre la sensibilidad de tales operaciones, pero escapa del alcance de Fuzz.



La idea principal se funda en que el desarrollo de programas se lleve a cabo desde un principio en torno a la noción de privacidad y, por lo tanto, cualquier programa [con el tipo adecuado, corroborado por el verificador de tipos del lenguaje, resulte], por construcción, diferencialmente privado.

argumento; en contraste, $\mathbb{R} \rightarrow \mathbb{R}$ denota una función de reales a reales, pero que utiliza su argumento en un factor de 1, i.e. una función 1-sensible en su argumento. Por ende, la función $f(x) = x + 1$ puede no sólo tener tipo $\mathbb{R} \rightarrow \mathbb{R}$ sino que podemos ser más precisos y asignarle el tipo $\mathbb{R} \rightarrow \mathbb{R}$, estableciendo además su sensibilidad. Por otro lado, si quisiéramos asignarle el tipo $\mathbb{R} \rightarrow \mathbb{R}$ a la función $g(x) = x + x$, el verificador de tipos de FUZZ nos daría un error de tipos, ya que g es de hecho 2-sensible en su argumento.

Sumado al mecanismo para razonar sobre la sensibilidad de los programas, FUZZ también provee un tipo para computaciones ruidosas: $\circ\mathbb{R}$ representa una distribución de probabilidad sobre reales. Por ejemplo, el tipo de una función `add_noise`, 1-sensitiva, y que recibe un real y produce un real añadiendo ruido según la distribución de Laplace con escala $1/\epsilon$, sería el siguiente:

```
add_noise:  $\mathbb{R} \rightarrow \circ\mathbb{R}$ 
```

Supongamos que tenemos la siguiente función, que recibe una base de datos y computa cuántas personas tienen más de 40 años:

```
over_40:  $db \rightarrow \mathbb{R}$ 
```

Podemos usarla para computar una versión ruidosa de la siguiente manera, en un lenguaje a la Python:

```
over_40_DP:  $db \rightarrow \circ\mathbb{R}$ 
def over_40_DP(d):
    add_noise(over_40(d))
```

De esta manera habremos creado una computación que *por construcción* es diferencialmente privada. De hecho, en FUZZ podemos demostrar matemáticamente que cualquier función aleatoria que tenga el tipo $db \rightarrow \circ\mathbb{R}$ es ϵ -DP.

Limitaciones

La creación de FUZZ desencadenó una serie de investigaciones en torno a la privacidad diferencial y los sistemas de tipos [23]. Sin embargo, FUZZ tiene algunas limitaciones que lo hacen poco práctico para su uso en el mundo real. En primer lugar, FUZZ sólo soporta la variante más básica de privacidad diferencial, es decir, ϵ -DP. Otras variantes más avanzadas como (ϵ, δ) -DP, Renyi-DP, o *zero concentrated DP* no son soportadas. Además, FUZZ, y otros lenguajes derivados, en muchas ocasiones son por defecto demasiado conservadores sobre la sensibilidad de programas. Si bien esta limitación se puede mitigar a través de otros mecanismos del lenguaje, requiere que el programador tenga un conocimiento no trivial sobre la sensibilidad de su programa.

JAZZ

Con el fin de hacernos cargo de estas limitaciones, y a su vez mejorar la adopción de privacidad diferencial en los lenguajes de programación, diseñamos el lenguaje JAZZ [24]. JAZZ es un lenguaje que sigue el espíritu de FUZZ, pero con un análisis de sensibilidad más preciso, sin sacrificar simplicidad. Además,

JAZZ está diseñado para soportar variantes avanzadas de DP en conjunto con programación de alto orden, donde funciones pueden tomar otras funciones como argumento.

JAZZ está compuesto de dos sublenguajes mutuamente definidos: uno para razonar acerca de sensibilidad (SAX), y otro acerca de privacidad (λ_j).

El sublenguaje de sensibilidad. SAX permite expresar explícitamente la sensibilidad de cada variable de un programa directamente en los tipos. Por ejemplo, en SAX, la función $g(x) = x + x$ tiene tipo $(x : \mathbb{R}) \xrightarrow{2x} \mathbb{R}$, que describe a una función de reales a reales donde su *sensibilidad latente* es $2x$. Esto significa que si variamos el argumento (x) en 1, amplificaremos el resultado del cálculo en 2. La novedad radica en el término “latente”, que implica que esta sensibilidad sólo afecta al resultado final del programa cuando la expresión se aplica. Por ejemplo, considere la siguiente función:

```
def f(y):
    def g(x):
        x + y
        y + 1
```

En SAX, la sensibilidad de f es 1 respecto a y , dado que el cálculo final de f no depende de g . Otra característica innovadora de SAX es que permite razonar sobre varias variables simultáneamente. Por ejemplo, la función $h(x,y) = x + x + y$ tiene sensibilidad latente $2x + y$, notación para decir que es 2-sensible en x y 1-sensible en y .

Este concepto de sensibilidad latente también se aplica a los pares. Un tipo par tiene asociado una sensibilidad latente específica para cada componente, y, en consecuencia, al acceder a un componente es que se contribuye a la sensibilidad final. Por ejemplo, el par $(2*x, x)$ tiene sensibilidades latentes $2x$ y $1x$ en el primer y segundo componente respectivamente, y por

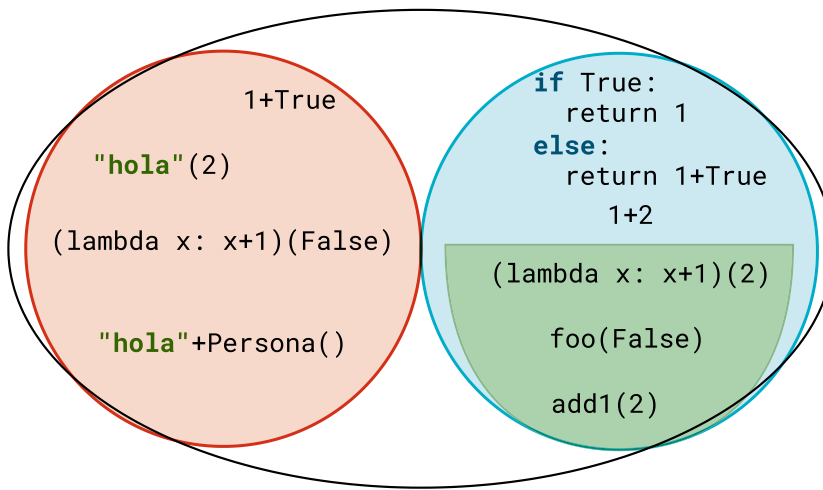


Figura 2. Los programas que funcionan correctamente se encuentran en azul, mientras que los que funcionan incorrectamente están en rojo. En verde se encuentran los programas aceptados por un sistema de tipos correcto, que aproxima de manera *conservadora* los programas que funcionan bien. La consecuencia de esto es que algunos programas que efectivamente funcionan bien, serán rechazados por el sistema de tipos (pero nunca aceptará un programa que definitivamente funcione mal).

consecuencia tiene tipo $\mathbb{R}^{2x} \times \mathbb{R}^{1x}$. Esto conduce a un análisis inherentemente más preciso, como se puede observar en el siguiente ejemplo:

```
def f(x):
    p = (2*x, x)
    if(b):
        3*p[0]
    else:
        2*(p[0] + p[1])
```

Primero, podemos notar que el cuerpo de la función es equivalente a escribir $6 \cdot x$, es decir, la función es 6-sensible en x . En lenguajes como FUZZ, el par p se clasifica como 2-sensible en x (que es el máximo de las sensibilidades de sus componentes). Luego, la expresión `if` toma, de manera pesimista, el valor máximo de las sensibilidades de ambas ramas. En este caso, la rama `else`

es clasificada como 8-sensible en x ($8 = 2 \cdot (2+2)$).² En contraste, en JAZZ, gracias a los efectos latentes, la sensibilidad de la primera rama es $6x$ (de escalar por 3 la sensibilidad latente del primer componente $2x$). De manera similar, la sensibilidad de la rama `else` es calculada como $6x$ también, al multiplicar por 2 la suma de las sensibilidades latentes $2x$ y $1x$. Como resultado, JAZZ informa de manera precisa una sensibilidad de $6x$.

El sublenguaje de privacidad. Similar a SAX, λ_j permite razonar acerca de la privacidad de cada variable. Para ello, podemos utilizar *funciones privadas* como $(x : \mathbb{R} \cdot d) \xrightarrow{\text{ex}} \mathbb{R}$. Esta función representa una función de reales a reales, cuyo nivel de privacidad es ϵ -DP respecto a su argumento x , y la *distancia relacional* de dicho argumento puede ser como máxi-

mo d . La distancia relacional cuantifica el grado de variación que el argumento puede experimentar en dos ejecuciones diferentes, con un valor comúnmente definido como $d = 1$. Este valor viene de la definición de DP, donde se compara la ejecución del mecanismo aleatorio con dos bases de datos que difieren en 1 individuo. Para ejemplificar, podemos definir la función `add_noise` que introduce ruido aleatorio basado en la distribución de Laplace de la siguiente manera:

```
add_noise: (x : ℝ · 1)  $\xrightarrow{\text{ex}}$  ℝ
def add_noise (x):
    x + Laplace(1/ε)
```

Posteriormente, el sistema de tipos garantiza que en situaciones como `add_noise(y+z)`, la distancia relacional máxima de cada variable no exceda el valor de 1.

Finalmente, establecimos la metateoría de JAZZ, demostrando que programas bien tipados no fallan en ejecución, y que además satisfacen la formulación formal de privacidad diferencial, propiedades conocidas respectivamente como *type safety* y *type soundness*.

Hacia gradual DP

Aunque ya hemos visto las ventajas que nos pueden brindar los sistemas de tipos, como los de FUZZ o JAZZ, estos por definición son una aproximación conservadora del comportamiento de programas, como se puede ver en la Figura 2. En general, es imposible diseñar un sistema de tipos que pueda distinguir con total certeza si un programa violará propiedades de sus tipos (esto es equivalente al *halting problem*). Esto significa que pueden rechazar programas que podrían funcionar correctamente, como por ejemplo:

² En muchas ocasiones sí se puede lograr un análisis preciso en Fuzz, pero hay que utilizar mecanismos complejos como el escalamiento de métricas.



```
...
if True:
    return 1
else:
    return 1+True
```

En programas más complejos, esto puede ser frustrante para desarrolladores, ya que pueden llevarse la impresión de que el sistema de tipos interfiere con la velocidad y experiencia de desarrollo, llevándolos a utilizar, en cambio, algún lenguaje *dinámicamente tipado* como Python, con menos garantías estáticas, pero más rápido para prototipado. Adicionalmente, un programador que quiera incorporar un sistema de tipos se podría enfrentar a una ardua transición, usualmente forzados a refactorizar código que ya funciona sólo para satisfacer al sistema de tipos.

El *tipado gradual* busca solucionar este tipo de problemas, soportando una transición fluida entre chequeo de tipos estático y dinámico. Consecuentemente, esta técnica permite a programadores dejar ciertas partes del código sin información

de tipos o incluso con información de tipos parcial. Luego, el sistema de tipos hace lo mejor que puede (de manera optimista) durante la fase de compilación, y retrasa lo que no puede verificar inmediatamente en tiempo de ejecución. Durante la ejecución, si alguna garantía estática se viola, el programa falla con un error.

En vías de mejorar la adopción de conceptos de privacidad para programadores, nuestro principal objetivo a futuro consiste en el desarrollo de un lenguaje gradual que permita razonar sobre sensibilidad y privacidad. Por ejemplo, el siguiente programa no es aceptado por un sistema de tipos estático y sí debería serlo por un sistema de tipos gradual:

```
x = ...
b = ...
def foo(y):
    if b:
        return y
    else:
        return y*y
add_noise (foo(x))
```

Recordemos que `add_noise` sólo funciona si el argumento es a lo más 1-sensible. Un sistema de tipos estático, de manera conservadora, rechazaría este programa porque el cuerpo del condicional es ∞ -sensible en `x` (se toma el máximo de ambas ramas). Con un sistema de tipos gradual, podemos declarar que el tipo de retorno de `foo` es desconocido. En este caso, si `b=True` entonces el programa debería correr con éxito. En cambio si `b=False` debería fallar con un error en ejecución. Ya hemos dado los primeros pasos al diseñar GSoul [25], el primer lenguaje gradual para razonar acerca de la sensibilidad de computaciones. Cómo extender GSoul para incluir privacidad diferencial es en estos momentos un tema activo de investigación y nuestro foco más inmediato. ■

REFERENCIAS

- [1] Cynthia Dwork. Differential privacy. pages 1–12.
- [2] Úlfar Erlingsson, Vasily Pihur, and Aleksandra Korolova. RAPPOR: randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014*, pages 1054–1067.
- [3] [n. d.]. Apple previews ios 10, the biggest ios release ever. <https://www.apple.com/newsroom/2016/06/apple-previews-ios-10-biggest-io>.
- [4] Samuel Haney, Ashwin Machanavajjhala, John M. Abowd, Matthew Graham, Mark Kutzbach, and Lars Vilhuber. Utility cost of formal privacy for releasing national employer-employee statistics. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, pages 1339–1354.
- [5] Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *Proceedings of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Cancún, Mexico*, pages 277–286.
- [6] Noah M. Johnson, Joseph P. Near, and Dawn Song. Towards practical differential privacy for SQL queries. *PVLDB*, 11(5):526–539, 2018.
- [7] Prashanth Mohan, Abhradeep Thakurta, Elaine Shi, Dawn Song, and David E. Culler. GUPT: privacy preserving data analysis made easy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012*, pages 349–360.
- [8] Arjun Narayan and Andreas Haeberlen. Djoin: Differentially private join queries over distributed databases. In *10th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2012, Hollywood, CA, USA, October 8-10, 2012*, pages 149–162.



- [9] Davide Proserpio, Sharon Goldberg, and Frank McSherry. Calibrating data to sensitivity in private data analysis. *PVLDB*, 7(8):637–648, 2014.
- [10] Indrajit Roy, Srinath T. V. Setty, Ann Kilzer, Vitaly Shmatikov, and Emmett Witchel. Airavat: Security and privacy for mapreduce. In *Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2010, April 28-30, 2010, San Jose, CA, USA*, pages 297–312.
- [11] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 308–318.
- [12] Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 464–473.
- [13] Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. Differentially private empirical risk minimization. *J. Mach. Learn. Res.*, 12:1069–1109, 2011.
- [14] Arik Friedman, Shlomo Berkovsky, and Mohamed Ali Kâafar. A differential privacy framework for matrix factorization recommender systems. *User Model. User-Adapt. Interact.*, 26(5):425–458, 2016.
- [15] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian J. Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.
- [16] Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. Stochastic gradient descent with differentially private updates. In *IEEE Global Conference on Signal and Information Processing, GlobalSIP 2013, Austin, TX, USA, December 3-5, 2013*, pages 245–248.
- [17] Kunal Talwar, Abhradeep Thakurta, and Li Zhang. Nearly optimal private LASSO. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3025–3033.
- [18] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey F. Naughton. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, pages 1307–1322.
- [19] Jason Reed and Benjamin C. Pierce. Distance makes the types grow stronger: a calculus for differential privacy. In *Proceeding of the 15th ACM SIGPLAN international conference on Functional programming, ICFP 2010, Baltimore, Maryland, USA, September 27-29, 2010*, pages 157–168, Baltimore, Maryland, USA, September 2010. ACM Press.
- [20] Gilles Barthe, George Danezis, Benjamin Grégoire, César Kunz, and Santiago Zanella Béguelin. Verified computational differential privacy with applications to smart metering. In *2013 IEEE 26th Computer Security Foundations Symposium, New Orleans, LA, USA, June 26-28, 2013*, pages 287–301.
- [21] Gilles Barthe, Thomas Espitau, Benjamin Grégoire, Justin Hsu, Léo Stefanescu, and Pierre-Yves Strub. Relational reasoning via probabilistic coupling. In *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015, Suva, Fiji, November 24-28, 2015, Proceedings*, pages 387–401.
- [22] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–102, 1987.
- [23] Benjamin C. Pierce. *Types and programming languages*. MIT Press, Cambridge, MA, USA, 2002.
- [24] Matías Toro, David Darais, Chike Abuah, Joe Near, Damián Árquez, Federico Olmedo, and Éric Tanter. Contextual linear types for differential privacy. *ACM Transactions on Programming Languages and Systems*, 2023. To appear. <https://arxiv.org/abs/2010.11342>.
- [25] Damián Árquez, Matías Toro, and Éric Tanter. Gradual sensitivity typing. *CoRR*, abs/2308.02018, 2023.



Premio Turing 2022:

Robert Metcalfe, coinventor de Ethernet, el estándar para redes de área local



JOSÉ URZÚA

Magíster en Ciencias mención Computación por la Universidad de Chile e ingeniero civil en computación por la Universidad de Chile. Profesor Experto Externo del Departamento de Ciencias de la Computación de la Universidad de Chile y Académico Jornada Parcial de la Facultad de Economía de la Universidad de Chile. Jefe del Área de Desarrollo de Sistemas de NIC Chile. Sus intereses incluyen aplicaciones web y tecnologías relacionadas, desarrollo de aplicaciones en red, Sistema de Nombres de Dominio (DNS) y el desarrollo de Internet en general. Lideró el Grupo de Trabajo Técnico de LACTLD (Latin American and Caribbean Top Level Domains) y actualmente lidera el desarrollo de proyectos colaborativos de la región.

 <https://jose.cl>

 jose@urzua.cl

 @jourzua



RESUMEN. Hace 24 años, mientras era alumno de la carrera de Ingeniería en computación, cursé un ramo que me presentó la programación de software de sistemas y, en particular, la programación de aplicaciones que funcionan en red. Creo que fue uno de los ramos que confirmó mi interés en la computación y en desarrollar aplicaciones que usan una red de datos para intercambiar información o ejecutar tareas de forma remota o lo que se nos pudiese ocurrir. En estas dos décadas y media creo que hemos tenido tantos avances, que casi de forma natural hoy asumimos que la conexión a una red de datos es un servicio omnipresente, disponible en cualquier dispositivo y que nos permite estar conectados, generando y recibiendo datos sin interrupción, incluso mientras dormimos. ¿Quiénes crearon lo necesario para permitirnos usar las redes tal como las usamos hoy? Entre estos creadores está Robert “Bob” Metcalfe. Su nombre lo leí por primera vez cuando tuve mis inicios en la docencia en cursos de Redes de Computadores, su trabajo era descrito en las fuentes bibliográficas y se reconocía como el coinventor de Ethernet, la tecnología desarrollada hace 50 años que sigue siendo el estándar de red más popular del mundo, el cual permite conectar computadores personales y dispositivos en una red local y con Internet global.

Ethernet

Cuando usted está conectado a la red de su lugar de estudio, trabajo o vivienda, lo más probable es que esté haciendo uso de Ethernet, el estándar para redes de área local. La historia de Ethernet, comienza a principios de 1970 en Hawái con el investigador Norman

Abramson y sus colegas de la Universidad de Hawái, quienes intentaban conectar a usuarios de islas remotas al computador principal de Honolulu con su sistema ALOHANET. Considerando la poca factibilidad para conectar cables bajo el océano Pacífico, buscaron una solución diferente usando terminales de radio de dos frecuencias: un canal ascendente en dirección al computador central y un canal descendente desde el computador central. Cuando un usuario deseaba comunicarse con el computador central, usaba el canal ascendente para enviar un paquete de datos, si nadie más estaba usando el canal, probablemente el paquete saldría, sería recibido y en el canal descendente recibiría su confirmación. Si había congestión en el canal ascendente, el terminal del usuario detectaba la falta de confirmación y realizaba otro intento de envío. Considerando que el canal de confirmaciones (canal descendente) sólo era usado por el computador central, no existían colisiones en él. Este sistema funcionaba muy bien en condiciones de poco tráfico, pero si el tráfico ascendente aumentaba, el sistema dejaba de funcionar.

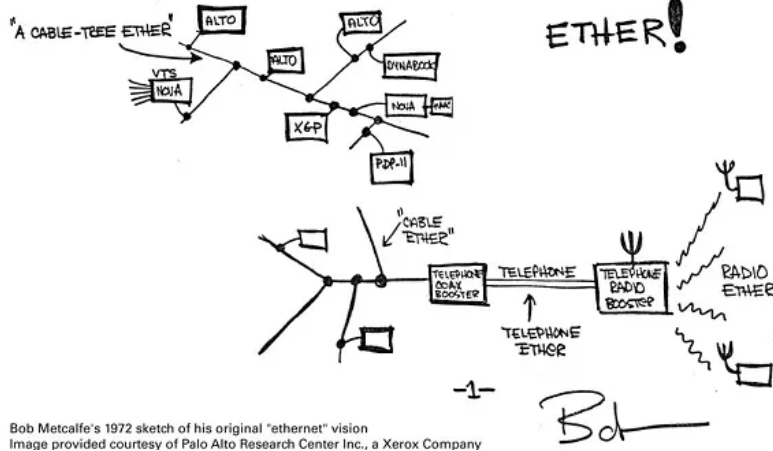
En esos años, Robert Metcalfe estaba realizando sus estudios de doctorado en la Universidad de Harvard trabajando para ARPANET (Advanced Research Projects Agency Network) y en una visita a la casa de Steve Crocker (inventor de los RFC de Internet) leyó en una revista científica sobre la red ALOHA y el trabajo de Norm Abramson. En ese momento, el problema principal al que se enfrentaban era cómo compartir el acceso a la red entre muchos usuarios. En el caso de las redes telefónicas, el problema se abordó de la forma más simple: una conexión entre dos partes bloqueaba el canal de comunicación durante la duración de la llamada; esto implicaba que ese canal no pudiese ser usado por otros usuarios, incluso si no se estaba usando en su máxima capacidad. Para las conversaciones telefónicas puede no ser un gran problema porque no se quedan en silencio por mucho

Ganador del Premio Turing 2022



tiempo, pero los computadores se comunican en ráfagas cortas que pueden estar separadas por largos periodos de tiempo sin comunicación.

Después de graduarse en Harvard, Metcalfe viajó a Hawái para conocer y trabajar un verano con Abramson. Después de esta experiencia, comenzó a trabajar en el Centro de Investigación de Palo Alto de Xerox (PARC). En este lugar, conoció el desarrollo de los primeros computadores personales que trabajaban aislados, por lo que aplicó el conocimiento del trabajo de Abramson y junto a su colega David Boggs, diseñó e implementó la primera red de área local. A este trabajo lo bautizaron con el nombre Ethernet por la teoría física (hoy abandonada) según las cuales las ondas electromagnéticas viajaban por un fluido llamado éter que se suponía llenaba el espacio. En este caso, el éter era el cable coaxial de más de 2.5 kilómetros de largo con repetidoras cada 500 metros. El sistema podía conectar hasta 256 máquinas por medio de transceptores (dispositivos que pueden transmitir y recibir comunicaciones) acoplados al cable (ver Figura 1). A diferencia de ALOHANET, en Ethernet los computadores escuchaban el cable antes de transmitir para determinar si alguien más estaba transmitiendo. Si alguien estaba transmitiendo, el computador esperaba a que terminara dicha transmisión. De esta forma, se aumentaba la eficiencia de la red porque se evitaba interferir con las



Bob Metcalfe's 1972 sketch of his original "ethernet" vision
Image provided courtesy of Palo Alto Research Center Inc., a Xerox Company

Actualmente, Ethernet es el medio principal de las comunicaciones de red por cable en todo el mundo.

los transmisores y se espera un tiempo aleatorio antes de reintentar transmitir. En caso de colisión en la transmisión, se duplica el tiempo de espera. De esta forma se lograba separar las transmisiones que están compitiendo por el canal y se da la oportunidad de lograr transmitir exitosamente. En la Figura 2 se puede ver la arquitectura original de Ethernet.

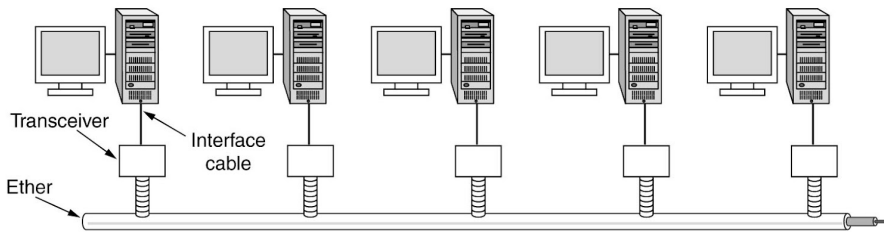


Figura 2. Arquitectura original de red Ethernet.

Estandarización

Después de dejar Xerox en 1979, Metcalfe siguió guiando el desarrollo de Ethernet y trabajando para garantizar la adopción de estándar abierto en la industria. Lideró un esfuerzo de Digital Equipment Corporation (DEC), Intel y Xerox para desarrollar una especificación de Ethernet de 10 Mbps y fue llamado estándar DIX. Se formó el comité IEEE 802 con el objetivo de establecer un estándar de red de área local (LAN) y una ligera variante de DIX se convirtió en el primer estándar IEEE 802.3 que sigue vigente en la actualidad.

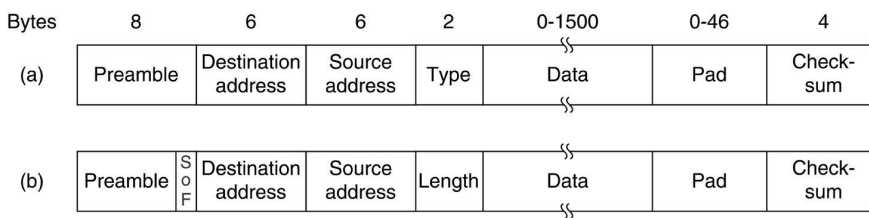


Figura 3. Formato de la trama Ethernet. (a) Estándar DIX (b) Estándar IEEE 802.3.

transmisiones existentes. En ALOHA-NET era imposible detectar la transmisión de otra terminal en una isla distante; este problema lo resolvía Ethernet al usar un cable único de conexión.

Otra mejora considerada en Ethernet, es para el caso en que dos o más

computadores están esperando para transmitir y al mismo tiempo determinan que pueden hacerlo. Para estos casos, consideraron que cada computador escuche el cable durante su propia transmisión con el fin de detectar interferencias. En caso de detectarla, se envía una señal para alertar a todos

Desde el año 2006 existe “Ethernet Alliance”, un consorcio global de proveedores de sistemas y componentes, expertos de la industria y profesionales universitarios y gubernamentales comprometidos con la expansión de la tecnología Ethernet.

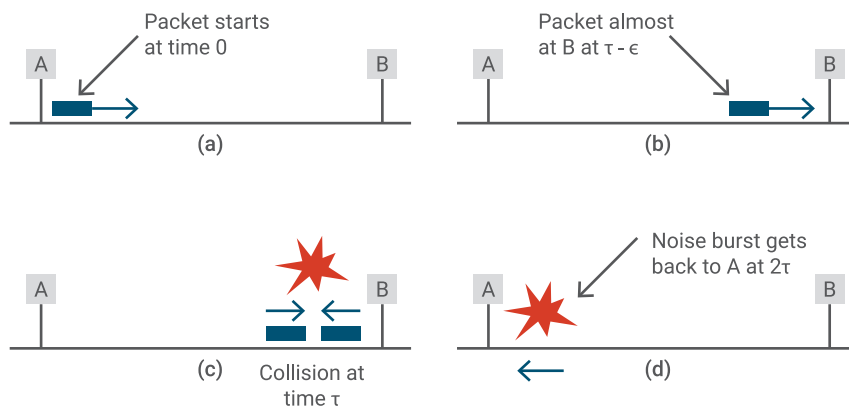


Figura 4. Detección de una colisión.

envía a una dirección de grupo, todas las estaciones del grupo la reciben. Este tipo de envío se llama multidifusión (*multicast*). También se consideran envíos para difusión (*broadcast*), en estos casos la dirección de destino consiste sólo en bits con valor 1, de esta forma, la trama es aceptada en todos los computadores de la red.

En el direccionamiento de una trama Ethernet es interesante la diferenciación de direcciones locales y globales. Las direcciones locales son asignadas por cada administrador de red y no son conocidas fuera de la red. Por otro lado, las direcciones globales son asignadas por IEEE y deben asegurar que no existan dos computadores con la misma dirección global en el mundo. Considerando que se usan 6 bytes para la dirección de destino, esto equivale a 48 bits (desde el bit 0 al bit 47), al usar el bit de mayor orden (bit 47) para diferenciar entre direcciones ordinarias o de grupo y al usar el bit 46 para diferenciar entre

direcciones locales y globales, quedan 46 bits (del 0 al 45) disponibles para direcciones globales (de orden de 7×10^{13} direcciones). De esta forma, se puede indicar la dirección exclusiva de cualquier otra estación usando el número correcto de 48 bits.

La trama Ethernet (ver Figura 3 (a)) también considera un campo que indica el tipo, el cual informa al receptor qué es lo que debe hacer. Esto permite usar múltiples protocolos y el receptor identifica el proceso al cual debe entregar la trama, de acuerdo al tipo informado. A continuación, vienen los datos con un espacio disponible de hasta 1500 bytes. Este límite se determinó al definir el estándar DIX, considerando que un transceptor necesita suficiente memoria RAM para mantener toda una trama y la RAM era muy costosa en 1978. Usar un límite mayor habría significado disponer de transceptores más costosos. También se consideró un tamaño mínimo para una trama Ethernet, de esta forma se

puede diferenciar tramas válidas de basura, por lo que se revisa que la longitud mínima sea 64 bytes desde la dirección de destino a la suma de verificación, incluyendo ambas. En caso que la parte de datos de una trama sea menor que 46 bytes se usa el campo *relleno* para llegar al tamaño mínimo.

El tamaño mínimo de una trama también tiene importancia para evitar que una estación conectada a la red termine la transmisión de una trama corta antes que el primer bit llegue al extremo más lejano en el cable, donde podría ocurrir una colisión con otra trama.

En el momento 0, la estación A ubicada en un extremo de la red envía una trama (ver Figura 4 (a)). Consideremos el tiempo de demora en llegar al otro extremo con el nombre τ . Supongamos que la estación más distante B comienza a transmitir en el momento $\tau - \epsilon$ (ver Figura 4 (b) y (c)), justo antes que llegue la trama. Cuando B detecta que está recibiendo más potencia que lo que está enviando, sabe que ha ocurrido una colisión y aborta su transmisión y genera una ráfaga de ruido de 48 bits para avisar a las demás estaciones. De esta forma llena el cable para asegurarse que el emisor tome conocimiento de la colisión. Luego, aproximadamente en el momento 2τ el emisor nota la ráfaga de ruido (Figura 4 (d)), aborta también su transmisión y espera un tiempo aleatorio para volver a transmitir.

Si una estación intenta transmitir una trama muy corta, es probable que ocurra una colisión, pero puede que la transmisión se complete antes de que la ráfaga de ruido llegue de regreso en el momento 2τ . En este caso, el emisor erróneamente supondrá que el envío de la trama fue exitoso. Para evitar esto, todas las tramas deberán tardar más de 2τ para enviarse, así la transmisión aún está en curso cuando la ráfaga de ruido llega al emisor. En una LAN de 10 Mbps con una longitud máxima de 2500 metros y

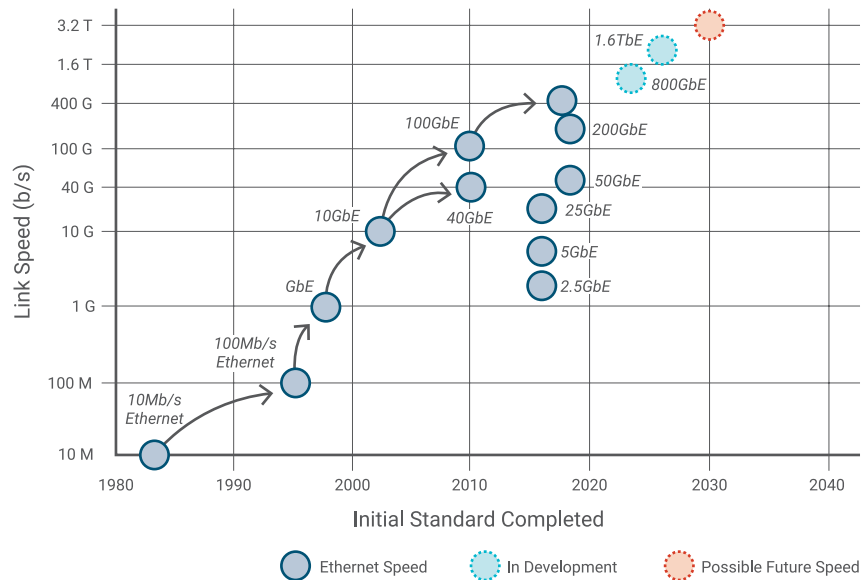
cuatro repetidores, el tiempo de ida y vuelta se ha determinado aproximadamente en $50\mu\text{seg}$ (microsegundos, millo-nésima parte de un segundo) en el peor caso, incluyendo el tiempo que pasa por los repetidores. La trama de tamaño mínimo debe tomar por lo menos ese tiempo en transmitir. A 10 Mbps un bit tarda 100 nseg, por lo que 500 bits es la trama más pequeña que se garantiza que funcionará. Para tener un margen de seguridad, este número se redondeó a 512 bits o 64 bytes. De esta forma, las tramas con tamaño menor a 64 bytes se rellenan con 64 bytes en el campo *relleno*.

En la medida que aumenta la velocidad de la red, también debe aumentar la longitud mínima de la trama o debe disminuir la longitud máxima del cable, de forma proporcional. Si tenemos una LAN operando a 1Gbps con 2500 metros, el tamaño mínimo de la trama tendría que ser de 6400 bytes. Pero, si la distancia máxima entre dos estaciones es de 250 metros, el tamaño mínimo de la trama podría ser de 640 bytes. En redes con velocidades de decenas de Gbps estas restricciones se van haciendo más complejas.

Al final de una trama Ethernet está el campo de la *suma de verificación*. Es un código de 32 bits calculado a partir de los datos. Si alguno de los bits de datos se recibe erróneamente por ruido en el cable, con una alta probabilidad la suma de verificación estará mal y se detectará el error.

Cuando se estandarizó Ethernet por parte de IEEE (ver Figura 3 (b)), el comité realizó dos cambios al formato DIX. Redujo el preámbulo a 7 bytes y usó el último byte para delimitar el inicio de la trama por compatibilidad con otros estándares. También cambió el campo tipo por un campo longitud. Con este último cambio, el receptor no sabía qué hacer con la trama que estaba recibiendo, pero esto se resolvía agregando un pequeño encabezado a la porción de datos para informar lo que se debía hacer. Cuando se publicó este estándar, se estaba usando abundante hardware y software para

ETHERNET SPEEDS



Fuente: <https://ethernetalliance.org/technology/ethernet-roadmap/>

Figura 5. Evolución de las velocidades de enlaces Ethernet y el año en que se completó (o se espera completar) el estándar.

Ethernet DIX que pocos usuarios y fabricantes tenían ganas de convertir el campo tipo en uno de longitud. Finalmente, en el año 1997 IEEE no insistió y dijo que las dos formas se ajustaban bien. En esto hubo suerte, pues todos los campos tipo que se usaban antes de 1997 tenían valores sobre 1500, por lo que cualquier valor menor que 1500 se interpretaba como longitud y cualquier valor mayor que 1500 se interpretaba como tipo.

Comercialización y actualidad

En el año 1979 Metcalfe fundó 3Com Corporation, su propia startup de Internet en Silicon Valley. Con esto, reforzó la comercialización de Ethernet vendiendo software de red, transceptores Ethernet y tarjetas Ethernet para mini computadores y estaciones de trabajo. Aprovechando la presentación del computador personal (PC) de IBM, 3Com

presentó una de las primeras interfaces Ethernet para PC y para los clones que se estaban masificando.

En 1987 3Com se fusionó con Bridge Communications, comenzando su expansión más allá de computadores personales, abarcando ahora equipos de red basados en procesadores Motorola 68000. En 1990 la junta directiva de 3COM nombró a Éric Benhamou como director ejecutivo en lugar de Metcalfe, quien luego dejó la compañía.

Actualmente, Ethernet es el medio principal de las comunicaciones de red por cable en todo el mundo y maneja velocidades de datos de 10 Mbps a 400 Gbps, con tecnologías en desarrollo de 800 Gbps y 1.6 Tbps. Ethernet también genera un gran mercado, con ingresos sólo de computadores que superan los \$30 mil millones de dólares en el año 2021 según International Data Corporation. El presidente de ACM, Yannis Loannidis, explica que "las ideas de diseño originales de Metcalfe han permitido que el



ancho de banda de Ethernet crezca de forma exponencial. Es raro ver que una tecnología escale desde sus orígenes hasta la capacidad actual de varios gigabits por segundo”, complementa que “es especialmente apropiado reconocer un invento tan impactante en su aniversario número 50”.

Desde el año 2006 existe “Ethernet Alliance”, un consorcio global de proveedores de sistemas y componentes, expertos de la industria y profesionales universitarios y gubernamentales comprometidos con la expansión de la tecnología Ethernet. Este consorcio tiene como misión apoyar las actividades desde la incubación de tecnología Ethernet hasta demostraciones de interoperabilidad y educación. Entre otras actividades, desde el año 2015 realizan publicaciones de nuevas versiones de la hoja de ruta de las actualizaciones y avances de Ethernet. Para este año, anuncian avances en automatización industrial y edificación, pronostican 700 millones de puertos Ethernet en automóviles para enlaces, mejores experiencias multimedia, sistemas autónomos de asistencia al conductor, entre otros. A nivel de proveedores de servicio, en

La demanda mundial por consumo de videos continúa ejerciendo presión para aumentar las velocidades de Ethernet, llegando actualmente a 1.6Tbps.

particular con la implementación móvil de redes 5G continúa impulsando el uso de Ethernet a mayores distancias y velocidades. La demanda mundial por consumo de videos continúa ejerciendo presión para aumentar las velocidades de Ethernet, llegando actualmente a 1.6Tbps (ver Figura 5). También se menciona SyncE (Ethernet síncrono) como una tecnología crítica usada por compañías de telecomunicaciones para el despliegue de redes y servicios 5G, con esto, se espera una adopción masiva de esta tecnología en los próximos años.

Epílogo

Robert Metcalfe contribuyó de una forma muy relevante con una solución innovadora al desarrollo de las redes de computadores de área local, proponiendo e implementando una forma de con-

trolar el uso del canal de comunicaciones y de detectar las colisiones. No sólo propuso e implementó, también se hizo cargo de la estandarización y comercialización de su invento. Ethernet se ha ido adaptando a lo largo de los años y sigue siendo la base de las redes de datos. El éxito de su trabajo lo podemos disfrutar hoy, 50 años después, al hacer uso de nuestros dispositivos conectados a una red de datos.

A los 76 años, Metcalfe inició un nuevo rumbo en su carrera y ahora es afiliado de investigación en ingeniería computacional en el Laboratorio de Ciencias de la Computación e Inteligencia Artificial del MIT, estudiando la aplicación de supercomputadoras a problemas complejos en energía, en particular en el modelado de pozos geotérmicos. “Todavía estoy en la primera parte de la curva de aprendizaje, no sé mucho, pero estoy trabajando para arreglar eso”, dijo. ■

REFERENCIAS

- [1] “ACM Turing Award Honors, Bob Metcalfe for Ethernet”, <https://amturing.acm.org/>.
- [2] “Computer Networks”, Andrew S. Tanenbaum.
- [3] “Computer History Museum”, <https://www.computerhistory.org/tdih/May/22/>.
- [4] “Bob Metcalfe, Ethernet Pioneer, Wins Turing Award”, <https://www.quantamagazine.org/bob-metcalfe-ethernet-pioneer-wins-turing-award-20230322/>.
- [5] “2023 Ethernet Roadmap”, <https://ethernetalliance.org/technology/ethernet-roadmap/>.
- [6] “Ethernet at 50: Bob Metcalfe pulls down the Turing Award”, <https://www.networkworld.com/article/3691019/ethernet-at-50-bob-metcalfe-pulls-down-the-turing-award.html>.



Un recorrido por los modelos de lenguaje:

Desde Shannon a GPT-4



FELIPE BRAVO MÁRQUEZ

Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile, Investigador Asociado del Centro Nacional de Inteligencia Artificial (CENIA) e Investigador Joven del Instituto Milenio Fundamento de los Datos (IMFD). Realizó su doctorado en el grupo *machine learning* de la Universidad de Waikato, Nueva Zelanda. Sus intereses de investigación y experiencia se centran en la adquisición de conocimientos e información a partir del lenguaje natural, abarcando las áreas del procesamiento del lenguaje natural (NLP), el aprendizaje automático (ML), la inteligencia artificial (AI) y la recuperación de información (IR). En su investigación, ha desarrollado varios métodos de NLP y ML para el análisis de opiniones y emociones en medios de comunicación social, así como otras aplicaciones centradas en la equidad, la salud y la educación, entre otras.

✉ fbravo@dcc.uchile.cl

✂ [@felipebravom](https://twitter.com/felipebravom)



RESUMEN. Los modernos modelos de lenguaje, representados por asistentes virtuales y chatbots como ChatGPT y Google Bard, han transformado la manera en la que nos relacionamos con las máquinas, permitiéndonos interactuar con ellas de la misma forma con la que interactuamos con nuestros pares humanos, usando el lenguaje.

Estas impresionantes capacidades no son el resultado de un mero golpe de suerte, sino el fruto de un progresivo desarrollo basado en descubrimientos científicos e innovaciones tecnológicas en el campo del aprendizaje automático y el procesamiento del lenguaje natural. En este artículo, trazamos el recorrido desde sus inicios, desde los primeros modelos de lenguaje estudiados por Shannon en la década de 1950, pasando por los primeros modelos de lenguaje neuronales propuestos por Bengio y otros, hasta llegar a los actuales grandes modelos de lenguaje.

Introducción

La llegada de ChatGPT a finales del año 2022, en conjunto con la proliferación de modelos de lenguaje y asistentes de conversación durante este mismo año, ha impactado profundamente en nuestro quehacer diario. Hemos sido testigos de cómo las máquinas han adquirido la capacidad de utilizar algo tan propio a nosotros los humanos: el lenguaje. Este salto tecnológico ha generado debates profundos sobre sus implicaciones en diversos ámbitos, tales como la propiedad intelectual, la diseminación de información falsa, el impacto en la salud mental y la educación, y, en última instancia, su potencial para reemplazar el trabajo humano. En este artículo, no nos adentraremos en dichos debates, sino que nos concen-

Se espera que un modelo de lenguaje atribuya probabilidades a las oraciones en función de su coherencia, tanto desde un punto de vista semántico como sintáctico.

traremos en trazar una línea histórica de los descubrimientos y avances tecnológicos que han permitido a las máquinas alcanzar la habilidad de escribir de manera similar a nosotros.

La capacidad de dominar el lenguaje humano ha estado en el imaginario de la computación desde sus inicios. En 1950, Alan Turing propuso el famoso “Test de Turing”, que planteaba la pregunta de si era posible crear una máquina capaz de mantener una conversación con otra persona sin que ésta pueda distinguir si está conversando con un humano o una máquina. En 1964, Joseph Weizenbaum creó Eliza, uno de los primeros agentes de conversación desarrollado en el MIT. Eliza era un programa que simulaba a un psicoterapeuta y utilizaba reglas predefinidas para responder a las conversaciones del usuario. En paralelo, en la década de 1950, Claude Shannon realizó los primeros estudios sobre cómo modelar el lenguaje escrito de manera estadística y predictiva [1]. Usando técnicas de teoría de la información, calculó la dificultad de predecir palabras en base a las anteriores a partir de un “corpus” de texto. Sin embargo, en 1957, Noam Chomsky, un lingüista y científico cognitivo, cuestionó la capacidad de los modelos estadísticos para capturar la gramática del lenguaje humano [2]. Para ilustrar esto, presentó dos oraciones ficticias:

1. Colorless green ideas sleep furiously.
2. Furiously sleep ideas green colorless.

Aunque ambas oraciones carecen de significado, Chomsky argumentó que sólo la primera se considera gramaticalmente correcta. Sin embargo, un mode-

lo de lenguaje como los estudiados por Shannon consideraría ambas oraciones igualmente poco probables. Las ideas de Chomsky detuvieron el progreso en el modelado estadístico de lenguaje por varios años.

Primeros modelos de lenguaje

Un modelo de lenguaje es una herramienta que asigna una probabilidad a cada posible oración que se puede formar a partir de un conjunto finito de palabras. Tomemos, por ejemplo, las siguientes oraciones: 1) “El perro ladra” y 2) “Baila tuerca alto”. Se espera que un modelo de lenguaje atribuya probabilidades a las oraciones en función de su coherencia, tanto desde un punto de vista semántico como sintáctico. Por consiguiente, debería asignar una probabilidad mayor a la primera oración en comparación con la segunda.

La motivación original de estos modelos se origina en el problema de reconocimiento del habla o de transcripción automática. Si consideramos las siguientes frases en inglés: “recognize speech” (reconocer el habla) y “wreck a nice beach” (arruinar una playa bonita), notamos que ambas suenan prácticamente idénticas (o, técnicamente hablando, producen la misma señal acústica). Un modelo de lenguaje construido a partir de un corpus de texto debería asignar una probabilidad mayor a la primera frase, dado que es más común y, por lo tanto, se encuentra con mayor frecuencia dentro del corpus. Esto, a su vez, habilitaría al sistema de transcripción para generar el texto correcto. De



hecho, los modelos de lenguaje resultan sumamente útiles en cualquier tarea de procesamiento de lenguaje natural (o NLP por sus siglas en inglés) que involucre la generación de texto (como traducción automática, la generación de resúmenes y los chatbots), ya que ayudan a los sistemas a discernir entre diversas posibles salidas mediante las probabilidades otorgadas por el modelo de lenguaje.

Matemáticamente, el modelo de lenguaje modela una oración como una secuencia de palabras, donde cada palabra es tratada como una variable aleatoria discreta que proviene de un conjunto finito de palabras llamado vocabulario. El modelo tiene la capacidad de asignar probabilidades a cualquier posible oración:

$$p(s) = p(w_1, w_2, \dots, w_n)$$

Técnicamente hablando, esta función es una función de masa de probabilidad multivariada. La regla de la cadena de probabilidades permite descomponer esta probabilidad en un producto de probabilidades condicionales, siguiendo la secuencia de la oración de izquierda a derecha y condicionando la probabilidad de cada palabra en relación a todas las anteriores, a las que llamamos "contexto":

$$p(w_1, w_2, \dots, w_n) = p(w_1) \times p(w_2|w_1) \times p(w_3|w_1, w_2) \times \dots \times p(w_n|w_1, \dots, w_{n-1})$$

Sin embargo, estimar esta función de probabilidad a partir de un corpus de entrenamiento (una colección de texto) se vuelve problemático cuando se condiciona una palabra por una secuencia muy larga de palabras anteriores. Esto se debe a que la cantidad de combinaciones posibles de palabras crece

exponencialmente con la longitud de la secuencia, lo que dificulta tener suficientes instancias en el corpus para estimar las probabilidades con precisión.

Para abordar esta dificultad, se recurre frecuentemente a un enfoque "Markoviano", que implica restringir la memoria de palabras anteriores (o del contexto) dando lugar a los modelos de lenguaje de n -gramas¹. Por ejemplo, en un modelo de lenguaje de bigramas, se asume que la probabilidad de una palabra depende únicamente de su predecesora:

$$p(w_3|w_1, w_2) = p(w_3|w_2)$$

Luego, la probabilidad de una oración se simplifica de la siguiente manera:

$$p(w_1, w_2, \dots, w_n) = p(w_1) \times p(w_2|w_1) \times p(w_3|w_2) \times \dots \times p(w_n|w_{n-1})$$

Entonces, el proceso de entrenamiento de un modelo de lenguaje de bigramas se reduce a estimar las probabilidades condicionales de una palabra dada otra palabra, lo que requiere calcular las frecuencias de palabras individuales (unigramas) y de secuencias de dos palabras (bigramas):

$$p(w_i|w_{i-1}) = \frac{\text{count}(w_{i-1}, w_i)}{\text{count}(w_{i-1})}$$

Ejemplo:

$$p(\text{york}|\text{nueva}) = \frac{\text{count}(\text{nueva}, \text{york})}{\text{count}(\text{nueva})}$$

No obstante, los modelos de lenguaje de bigramas o trigramas (que consideran dos palabras anteriores como contexto) tienen limitaciones en contextos largos y no pueden aprovechar contextos similares. Por ejemplo, consideremos los contextos:

c1. Después de comer cereales

c2. Luego de desayunar avena

Aunque esperaríamos que las distribuciones de probabilidad $p(w|c1)$ y $p(w|c2)$ fueran similares, dado que $c1$ y $c2$ casi no comparten palabras, los modelos de n -gramas que se limitan a contar frecuencia de palabras no pueden capturar estas similitudes entre contextos.

Una característica importante de los modelos de lenguaje es su capacidad generativa. Pueden crear oraciones nuevas mediante un proceso de muestreo secuencial basado en probabilidades condicionales estimadas. Cada palabra seleccionada se convierte en contexto para elegir la siguiente palabra, replicando la idea de extraer bolas de una urna donde los tamaños de las bolas representan las frecuencias relativas determinadas por el modelo. También es posible partir con un texto inicial a completar y seleccionar consecutivamente la palabra más probable en cada paso, lo que equivale a predecir la siguiente palabra. Esta mirada predictiva cobra mayor relevancia en los modelos de lenguaje neuronales, que serán discutidos a continuación.

Modelos de lenguaje neuronales

Las redes neuronales [3, 4] son una familia muy popular de modelos de aprendizaje automático, compuestos de unidades de cómputo llamadas neuronas. Cada neurona recibe entradas y salidas escalares, en donde a cada entrada se le asigna un peso escalar denotado como "w". El proceso que sigue una neurona implica la multiplicación de cada entrada por su peso correspondiente, seguida por la suma de estos productos.

1 Un n -grama es una secuencia contigua de n palabras, cuando $n=2$ tenemos un bigrama, $n=3$ un trigramo y así sucesivamente.



Lo que resulta particularmente fascinante, es que las palabras con significado relacionado (por ejemplo, sinónimos), tienden a adquirir vectores o embeddings cercanos en el espacio vectorial.

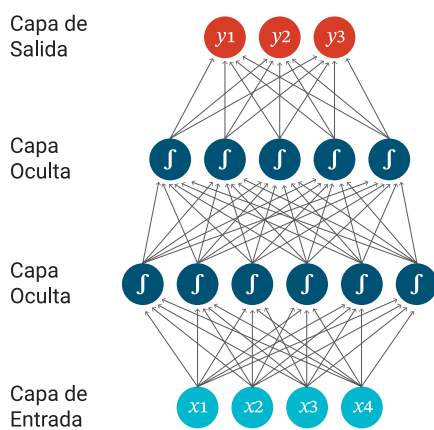


Figura 1. Ilustración de red neuronal de 4 capas: una entrada, dos capas intermedias u “ocultas” y una capa de salida.

Posteriormente, se aplica una función de activación, generalmente de naturaleza no lineal, al resultado obtenido, el cual se transmite como salida de la neurona. Típicamente, estas neuronas se organizan en capas, las cuales pueden ser apiladas para formar una red neuronal de múltiples capas conocida como “feed-forward”. Las salidas de las capas inferiores se convierten en las entradas de las capas superiores, generando así una progresión de la información a través de la red. La primera capa de la red representa la entrada del modelo, como por ejemplo una imagen o un texto, mientras que la última capa corresponde a la salida deseada, que podría ser una categoría o incluso otro texto. Las capas intermedias suelen llamarse como “capas

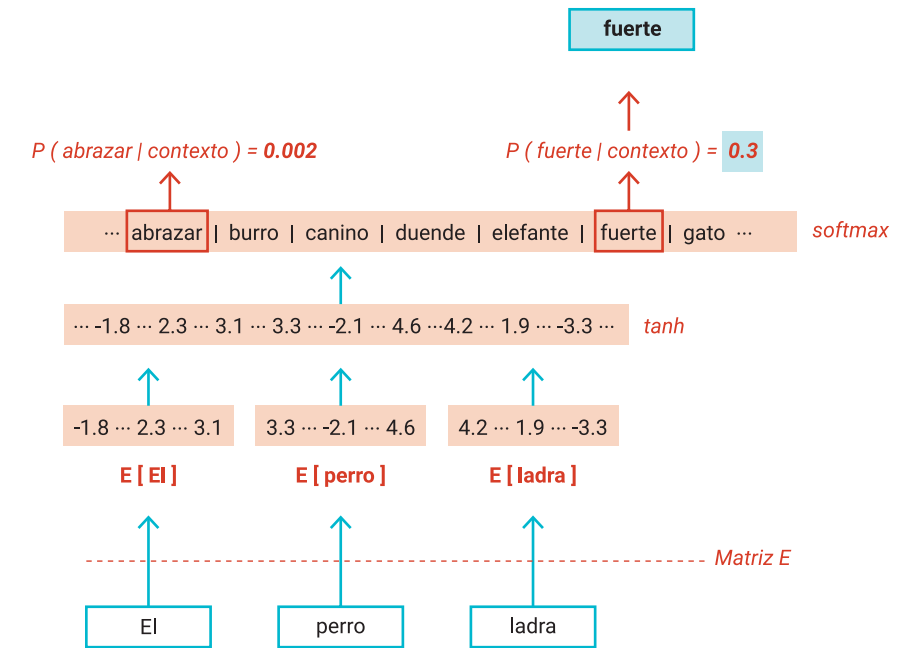


Figura 2. Modelo de lenguaje neuronal [5]. Las palabras en el contexto se transforman en vectores de dimensionalidad igual a la del vocabulario. Cada palabra se codifica utilizando la técnica *one-hot*, asignando un valor de 1 a la posición correspondiente a la palabra en el vocabulario y 0 a todas las demás posiciones. Estos vectores luego son proyectados en una capa de *embeddings*, donde se convierten en vectores densos. Estos vectores resultantes se combinan en una capa intermedia y posteriormente se proyectan hacia la salida, cuya dimensionalidad coincide con la del vocabulario. Finalmente, se aplica la función *softmax* para generar una distribución de probabilidad que abarca todas las palabras posibles en la salida.

ocultas” y el número de capas determina la profundidad de la red (ver Figura 1).

Los parámetros de una red neuronal (los pesos asociados a todas las neuronas de todas las capas) son inicializados con valores aleatorios, los cuales se ajustan y aprenden a partir de conjuntos de datos etiquetados. Estos consisten en ejemplos compuestos por pares de entradas y salidas deseadas. Este proceso de aprendizaje se lleva a cabo mediante algoritmos de optimización, siendo el algoritmo de Backpropagation el enfoque comúnmente utilizado. Una vez que una red ha sido entrenada de manera adecuada, es capaz de realizar predicciones precisas de las salidas con una alta probabilidad de acierto.

Las redes neuronales han experimentado un vaivén de entusiasmo y escepticismo a lo largo de su historia. Indudablemente, en la última década, han entrado en una fase de popularidad sin precedentes debido a su capacidad para generar “representaciones” de los datos en sus capas intermedias, lo que ha dado lugar a la creación de una disciplina conocida como aprendizaje profundo o *deep learning*. Los hitos significativos en el desarrollo de esta tecnología incluyen el surgimiento de arquitecturas especializadas de redes neuronales, como las redes convolucionales (CNN), las redes recurrentes (RNN) y la arquitectura Transformer. Además, ha sido de suma importancia el empleo de hardware especializado, como las GPUs



(graphic processing units) y TPUs (tensor processing units), para llevar a cabo un entrenamiento eficiente de estas redes, así como el acceso a grandes volúmenes de datos procedentes de la Web y plataformas de Crowdsourcing. Estas últimas permiten llevar a cabo el etiquetado masivo de datos, los cuales se utilizan en el proceso de entrenamiento de las redes neuronales.

En el año 2000, Bengio y colaboradores propusieron utilizar redes neuronales *feed-forward* para construir modelos de lenguaje [5]. Utilizando la regla de la cadena y restringiendo el contexto a un tamaño fijo de palabras (por ejemplo, 5), se puede modelar la probabilidad $p(w|c)$ mediante una red neuronal. En este enfoque, las palabras anteriores que conforman el contexto se convierten en las entradas de la red neuronal, y la palabra siguiente es la salida deseada. Luego se aplica una función denominada *softmax*, que transforma las salidas de la red (que es un vector de puntajes para cada palabra del vocabulario) en una distribución de probabilidad. De esta forma es posible tomar un corpus de texto y recorrerlo, extrayendo ventanas de palabras junto con sus palabras siguientes para entrenar la red neuronal y desarrollar un modelo de lenguaje (ver Figura 2).

Sin duda, la propiedad más interesante de los modelos de lenguaje neuronales es la noción de *word embedding* o representación vectorial de palabras. Esto implica proyectar las palabras discretas presentes en el contexto hacia vectores densos de cientos de dimensiones (ver Figura 3). Lo que resulta particularmente fascinante, es que las palabras con significado relacionado (por ejemplo sinónimos), tienden a adquirir vectores o *embeddings* cercanos en el espacio vectorial. Este fenómeno encuentra su raíz en una teoría lingüística denominada hipótesis distribucional [6, 7], que sostiene que las palabras presentes en contextos parecidos suelen compartir significados. Como resultado, los modelos de lenguaje neuronales se distinguen de sus contrapartes

Matriz de embeddings

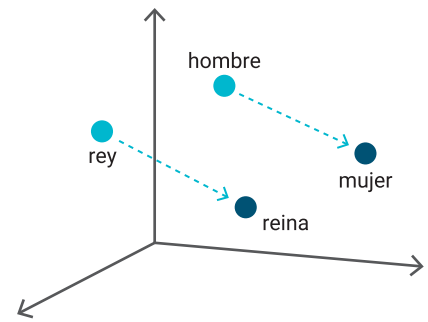
$ V \times d$				
-6.6	4.3	...	-2.4	← abrazar
3.6	-1.9	...	4.1	← canino
-1.8	2.3	...	3.1	← el
2.5	3.4	...	-2.7	← fuerte
5.0	-3.4	...	1.4	← grueso
4.2	1.9	...	4.2	← ladra
3.3	-2.1	...	4.6	← perro
-2.6	1.5	...	1.6	← zancudo

Figura 3. Ilustración de la matriz de *embeddings* aprendida por el modelo de lenguaje neuronal. La matriz consta de filas equivalentes al número de palabras en el vocabulario, con cada fila representada por un vector de dimensión d (usualmente entre 100 y 500). Cada componente en estos vectores es un valor escalar.

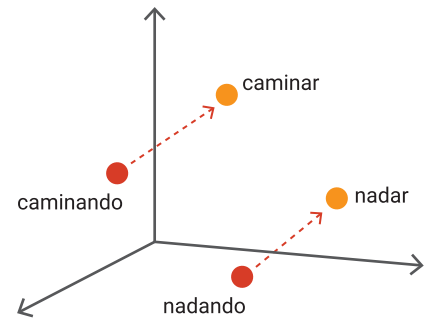
basadas en n -gramas al ser capaces de aprovechar contextos similares aunque estos difieran en las palabras utilizadas.

Tuvieron que pasar 13 años para que los modelos de lenguaje neuronales fueran adoptados de forma masiva. Esto fue posible gracias al lanzamiento del software de código abierto llamado Word2Vec [8], el cual permitía entrenar vectores de palabras empleando modelos similares a los propuestos por [5], pero con una eficiencia superior. Este software permitió a miles de usuarios entrenar sus propios vectores de palabra utilizando sus colecciones de documentos particulares, lo que les permitió explorar las características intrínsecas de sus palabras. Una propiedad descubierta fue la capacidad de realizar analogías semánticas a través de operaciones aritméticas en el espacio vectorial de las palabras tales como “hombre es a mujer como rey es a reina”, relaciones verbales como “nadar es a nadando como caminar es a caminando”,

Masculino / Femenino



Forma verbal



País / Capital

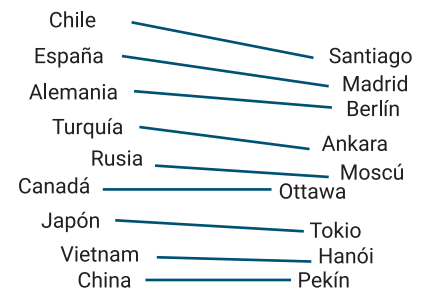


Figura 4. Visualización de relaciones capturadas por los vectores de palabras entrenados con Word2Vec. Para lograr una representación gráfica, los vectores de alta dimensionalidad se suelen reducir a 2 o 3 dimensiones, permitiendo así su visualización en un sistema de coordenadas.

además de establecer relaciones entre países y sus capitales, como “Santiago es a Chile como Madrid es a España”, entre otras (ver Figura 4).



Aunque los modelos de lenguaje neuronales originales brindan la posibilidad de aprovechar contextos similares a través de vectores de palabras, aún presentan limitaciones en su capacidad para capturar contextos largos. Consideremos el siguiente contexto como ejemplo: “Felipe nació en Chile, donde cursó sus estudios de ingeniería, y posteriormente continuó con su educación de postgrado en Nueva Zelanda. A pesar de que Felipe utiliza el idioma inglés con fluidez, su lengua materna sigue siendo el _____”.

Para lograr predecir con precisión la palabra siguiente, ya sea “español” o “castellano,” un modelo de lenguaje debería tener la capacidad de procesar un contexto lo suficientemente largo para considerar el hecho que Felipe nació en Chile, lo cual es caro computacionalmente para el modelo de lenguaje neuronal original.

Con el objetivo de potenciar a los modelos de lenguaje neuronales con la habilidad de procesar contextos extensos, se han empleado redes neuronales recurrentes [9]. Estas redes, a diferencia de las tradicionales redes *feed-forward*, tienen la capacidad de procesar secuencias de largo variable. En términos generales, cuentan con un vector que representa su estado, el cual se va actualizando a medida que se avanza en la lectura de nuevas palabras reteniendo el historial completo de la secuencia, lo que posibilita la modelación de secuencias de mayor longitud. Finalmente, este estado es utilizado para predecir la palabra siguiente con mayor precisión que el modelo neuronal original (ver Figura 5).

Los modelos de lenguaje son multitarea

Una propiedad muy poderosa de los modelos de lenguaje neuronales es que no necesitan texto etiquetado para ser

[GPT-3] tuvo un impacto revolucionario en el ámbito del aprendizaje automático y NLP, ya que introdujo la noción de contar con un modelo único para abordar múltiples tareas, evitando el costoso proceso de entrenar modelos individuales.

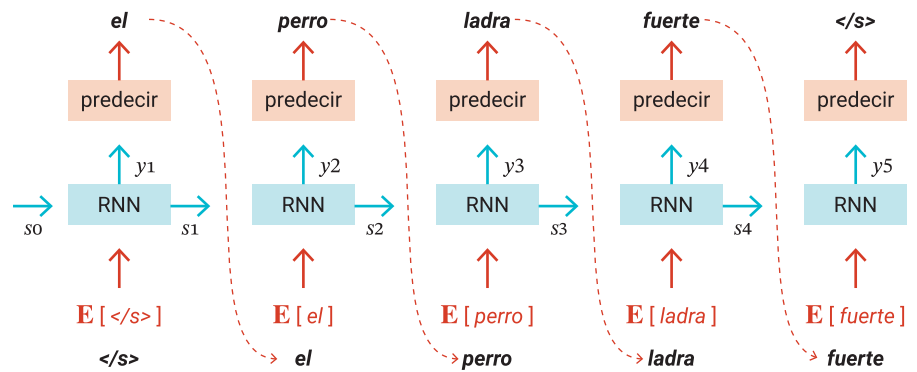


Figura 5. Modelo de lenguaje usando una red neuronal recurrente. Cada palabra en la secuencia se convierte en un vector que se procesa de manera secuencial para actualizar el vector de estado s . Se generan predicciones en cada uno de estos estados.

entrenados. En NLP, cada tipo de tarea específica, como la traducción automática, la clasificación de documentos según sentimiento, la generación de resúmenes, o la generación de respuestas a una pregunta, se resuelve entrenando redes neuronales sobre texto manualmente etiquetado con las respuestas esperadas. Etiquetar estos datos es un proceso sumamente costoso, pues requiere intervención de etiquetadores humanos. Por lo general, los datasets de entrenamiento para estas tareas no suelen tener más de miles de ejemplos, lo cual es una limitante para tener modelos que realmente sirvan (o generalicen) para datos distintos a los usados en el entrenamiento.

Si pensamos en la tarea de predecir palabras que ataca el modelo de lenguaje, podemos apreciar cómo este desafío fuerza a la red neuronal a aprender información de índole sintáctica, semántica y de conocimiento general.

Consideremos los siguientes tres contextos:

1. Abróchate el cordón de tus ...
2. Este regalo es para mi ...
3. Cristóbal Colón descubrió ...

Para predecir con alta confianza las palabras “zapatos” o “zapatillas” en el primer contexto, el modelo de lenguaje debe reconocer que estos objetos llevan cordones. Luego, en el segundo contexto, para predecir términos como “novio”, “amigo” o “mamá”, la red neuronal debe entender que un sustantivo (o una frase nominal) es la forma correcta de completar oración, lo que implica un conocimiento gramatical. Finalmente, en el tercer caso, para predecir “América”, el modelo debe poseer la información de que Cristóbal Colón fue el descubridor de América, lo que implica tener conocimiento general.

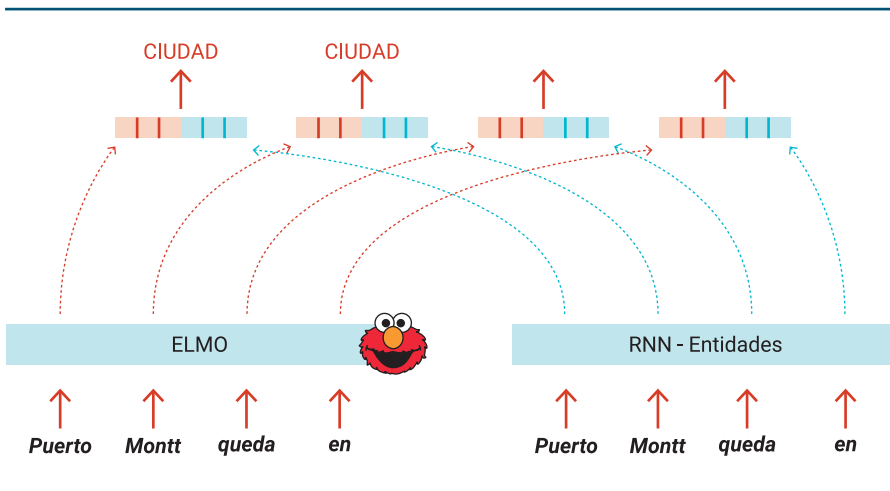


Figura 6. Ejemplo de cómo usar las representaciones de ELMO para resolver la tarea de detectar entidades (ej: ciudades) en una oración. En este ejemplo se tiene una red recurrente para resolver el problema y los estados de ésta son concatenados con los vectores obtenidos con ELMO.

Viendo el modelo de lenguaje de esta forma, hace sentido entrenar un modelo de lenguaje sobre un corpus grande (Wikipedia, Libros, la Web) y aprovechar todo el conocimiento adquirido para resolver tareas más específicas (traducir, clasificar) donde los datos etiquetados son de mucho menor tamaño que el corpus usado para entrenar el modelo de lenguaje.

Esta fue la idea que exploraron Peters *et al.* en 2018 para desarrollar ELMO [10]. ELMO es un modelo de lenguaje compuesto por aproximadamente 100 millones de parámetros y entrenado mediante una red neuronal recurrente, utilizando un corpus de texto grande. La propuesta central de ELMO consiste en aprovechar las representaciones proporcionadas por el modelo de lenguaje previamente entrenado, con el fin de enriquecer redes neuronales más específicas diseñadas para abordar tareas como la traducción o la clasificación, utilizando conjuntos de datos de entrenamiento relativamente pequeños. El proceso implica la pasada inicial de la entrada por el modelo de lenguaje preentrenado, empleando luego los estados de la red recurrente como vecto-

res de palabras contextualizados. Estos vectores son posteriormente suministrados a la red que aborda la tarea específica, lo que potencia la precisión en la resolución de dicha tarea (ver Figura 6).

Para comprender esta idea con mayor claridad, consideremos las siguientes dos oraciones:

1. Me senté en el banco a esperarte.
2. La fila del banco está muy larga.

Supongamos que nuestra tarea consiste en traducir estas oraciones del español al inglés. Una red neuronal común emplearía inicialmente una capa de *embedding*, donde cada palabra se asignaría a un vector. Sin embargo, surge un problema aquí: tanto la palabra “banco” en la primera oración como en la segunda recibirían el mismo vector, a pesar de que sabemos que poseen significados distintos debido a la situación de polisemia. No obstante, al introducir primero las oraciones en un modelo de lenguaje preentrenado, los estados de la red recurrente contextualizan los vectores, generando así vectores distintos para las dos apariciones de “banco”.

Esto permite una traducción más precisa al inglés (“bench” para la primera oración y “bank” para la segunda). De hecho, ELMO mostró mejoras significativas en varias tareas de NLP respecto al estado del arte de su momento, al aprovechar el conocimiento adquirido por el modelo de lenguaje.

ELMO estableció un precedente en la noción de que un modelo de lenguaje no sólo se encarga de asignar probabilidades a las oraciones, sino que también nos permite adquirir representaciones a partir de grandes colecciones de texto. Estas representaciones posteriormente pueden ser empleadas para resolver tareas específicas de NLP (a las cuales en inglés se les denomina *down-stream tasks*). Sin embargo, los modelos basados en redes neuronales recurrentes (RNN) como ELMO presentan una limitación crucial cuando se trata de expandir su capacidad a través de una mayor cantidad de parámetros. Las RNN requieren procesar la entrada de manera secuencial, lo que restringe la posibilidad de paralelizar sus cálculos. Sin paralelización efectiva no es factible escalar los modelos a los miles de millones de parámetros que tienen los modelos del estado del arte para entrenarlos en un tiempo razonable.

Para enfrentar estos desafíos, en 2017, Vaswani *et al.* introdujeron el Transformer, un tipo de red neuronal basada en mecanismos de (auto)atención [11]. De una manera simplificada, el Transformer recibe una oración representada como una secuencia de vectores, para luego mediante ponderaciones y multiplicaciones de matrices, ir contextualizando estos vectores (o sea, que estos cambien en función de sus vecinos), un procedimiento que puede ser repetido en varias ocasiones para producir una secuencia de vectores altamente enriquecidos y contextualizados (ver Figura 7). A diferencia de las redes neuronales recurrentes (RNN), los Transformers no están limitados por la necesidad de procesar las

entradas de forma secuencial. Esto los hace particularmente adecuados para la ejecución en hardware especializado como GPUs y TPUs, lo que, a su vez, los convierte en altamente paralelizables.

En 2019, surgió BERT (Bidirectional Encoder Representations from Transformers), que replicó la idea de ELMO pero utilizando un Transformer en lugar de una RNN [12]. BERT tenía 335 millones de parámetros y utilizaba la autoatención del Transformer para reemplazar la recursión de las RNN. Esto permitió obtener *embeddings* contextualizados de manera más escalable en términos de paralelismo y cantidad de parámetros y datos. A diferencia de ELMO, donde estos vectores contextualizados se introducían directamente en la red neuronal de la tarea específica, BERT introdujo la noción de "ajuste fino" o *fine-tuning*. Aquí, el concepto consiste en aprovechar los pesos aprendidos por BERT a partir del corpus de texto grande y focalizarse únicamente en adaptar la última capa de la red para que sea compatible con la tarea en cuestión (por ejemplo, mientras que para el modelo de lenguaje las salidas posibles son todo el vocabulario, en un problema de clasificación por sentimientos sólo se tienen categorías positivo, negativo y neutral). A continuación, la red completa se entrena con los datos etiquetados de la tarea objetivo, con la particularidad de que los pesos, en lugar de ser inicializados de manera aleatoria, parten con la configuración aprendida previamente por BERT (ver Figura 8). Este enfoque obtuvo resultados aún superiores a ELMO.

De manera paralela a BERT, la compañía OpenAI presentó una serie de modelos de lenguaje también basados en la arquitectura Transformer, denominados Generative Pretrained Transformer (GPT). Esta serie incluye GPT-1, GPT-2 y GPT-3, cada uno con una dimensión de parámetros mayor que su predecesor. No obstante, lo que más destaca en estos modelos son las propiedades

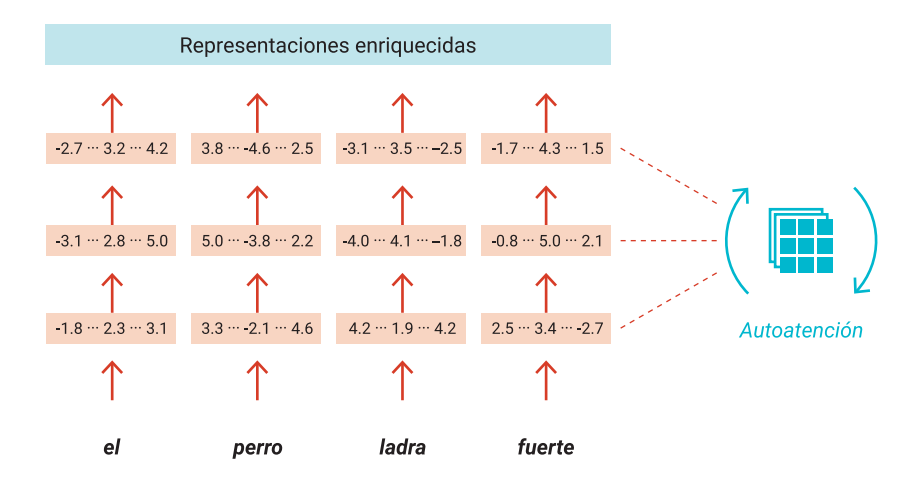


Figura 7. Diagrama de la arquitectura del Transformer. Se aplican varias capas de atención multicabeza donde los vectores de la entrada se multiplican por distintas matrices y se ponderan entre sí hasta obtener representaciones enriquecidas.

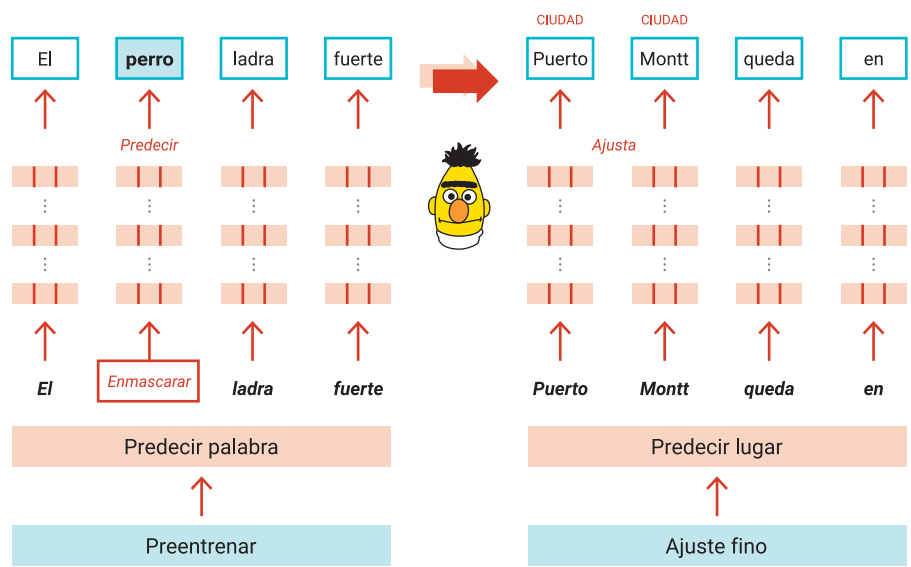
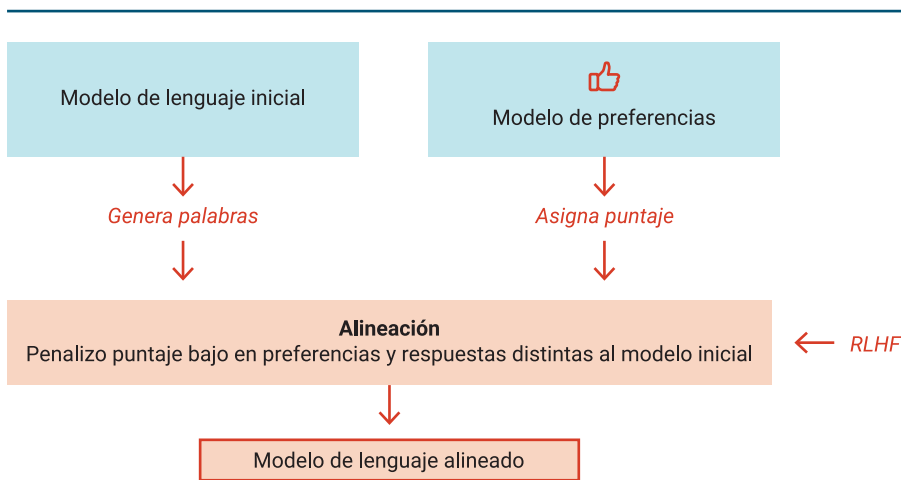


Figura 8. Ilustración del proceso de *fine-tuning* con BERT. El modelo original se entrena con la tarea de modelo de lenguaje enmascarado que consiste en predecir palabras faltantes de una oración. Luego el proceso de *fine-tuning* consiste en adaptar la capa de salida según la tarea objetivo y continuar ajustando los pesos de la red con los datos de la tarea objetivo.

“emergentes” que se manifestaron al aumentar el tamaño de los parámetros. El modelo GPT-2 [13] introdujo la noción de que un modelo de lenguaje grande puede ser visto como un modelo multitarea no supervisado, es decir, capaz

de abordar diversas tareas sin requerir el proceso de *fine-tuning* con datos etiquetados como se hacía con BERT.

En términos generales, se puede argumentar que prácticamente cualquier



Instrucción:

"Traduce esta palabra al inglés:
manzana →"

Ejemplos:

- zanahoria → carrot
- limón → lemon
- lechuga → lettuce
- pera → pear"

Este descubrimiento tuvo un impacto revolucionario en el ámbito del aprendizaje automático y NLP, ya que introdujo la noción de contar con un modelo único para abordar múltiples tareas, evitando así el costoso proceso de entrenar modelos individuales y etiquetar grandes volúmenes de datos. Es importante considerar que las habilidades emergentes sólo se manifiestan cuando escalamos los modelos a los miles de millones de parámetros.

Además, suscitó un profundo debate en la comunidad académica acerca de si estas propiedades emergentes realmente representan un aprendizaje en base al contexto o si simplemente se deben a que el extenso corpus de entrenamiento ya contenía datos relevantes para las tareas existentes. Este cuestionamiento sigue siendo ampliamente discutido [15].

El diseño apropiado de *prompts* se ha transformado en un aspecto de vital importancia para modelos tipo GPT-3, ya que la calidad de las respuestas puede variar sustancialmente según el *prompt* empleado. En consecuencia, ha surgido una nueva disciplina conocida como "ingeniería de *prompts*", cuyo enfoque radica en descubrir métodos para redactar *prompts* de manera efectiva según la tarea que se quiere resolver.

Por otro lado, GPT-3 transformó la forma en que ingenieros e investigadores acceden a estos modelos. En la época de BERT, todos los modelos eran liberados públicamente (los pesos de la red neuronal) en plataformas abiertas como

Figura 9. Ilustración del Proceso RLHF para alinear las respuestas del modelo de lenguaje. Se tiene un modelo de lenguaje inicial y un modelo de preferencias entrenado con etiquetadores humanos capaz de asignarle un puntaje a las salidas del modelo de lenguaje. El modelo resultante del ajuste fino se aprende optimizando una función que penaliza respuestas de puntaje bajo según el modelo de preferencias como también las que se alejan demasiado del modelo original.

tarea de NLP puede ser "codificada" como una instrucción dentro del contexto del modelo de lenguaje, para que luego la propiedad generativa del modelo de lenguaje le permita resolver la tarea en cuestión. Por ejemplo, sería factible usar como contexto la instrucción "traduce el siguiente texto de inglés a español: *I like dogs*", y confiar en que el modelo de lenguaje sea capaz de realizar la traducción de manera precisa al generar palabras a partir de la instrucción.

GPT-3 [14] llevó esta noción aún más allá, al contar con aproximadamente 200 mil millones de parámetros y haber sido entrenado en un corpus gigantesco que supera los 500 mil millones de palabras. Este entrenamiento, tuvo un costo energético evaluado en torno a 4.6 millones de dólares. En esta investigación, se formaliza la propiedad emergente previamente mencionada con el término "aprendizaje en contexto" (*in-context learning*), y se proponen tres enfoques para utilizar esta propiedad.

1. Aprendizaje sin ejemplos (*zero-shot learning*): se le proporciona únicamente una instrucción al modelo de lenguaje en el contexto, al cual de ahora en adelante llamaremos *prompt*.

Instrucción:

"Traduce esta palabra al inglés: manzana →"

2. Aprendizaje con un solo ejemplo (*one-shot learning*): se le proporciona un solo ejemplo junto con su respuesta correcta, además de la instrucción en el *prompt*.

Instrucción:

"Traduce esta palabra al inglés:
manzana →
Ejemplo: zanahoria → carrot"

3. Aprendizaje con pocos ejemplos (*few-shot learning*): se le proporcionan unos pocos ejemplos etiquetados en el contexto de la instrucción en el *prompt*.



HuggingFace², lo que permitía su uso y adaptación por cualquier persona interesada. En contraste, GPT-3 adoptó un enfoque diferente al ser un modelo cerrado, y el acceso a sus capacidades se ofrece mediante una API de pago, lo que generó un cambio significativo en cómo se interactúa con estos modelos avanzados.

Modelos de lenguaje como asistentes

Estos grandes modelos de lenguaje, denominados LLMs por sus siglas en inglés (Large Language Models) al ser entrenados meramente para predecir palabras (propiedad conocida como autoregresiva) no son suficientes para crear asistentes de usuario o chatbots que sean capaces de realmente asistir a los usuarios por medio del *prompt*. Pueden producir respuestas vagas, repetitivas o poco relevantes, incluso políticamente incorrectas, incluyendo discursos

de odio y prejuicios raciales o de género, para las consultas de los usuarios.

Una solución a este problema radica en llevar a cabo el ajuste fino de los modelos de lenguaje, de manera que se alineen con las intenciones del usuario. El proceso de ajuste fino implica la contratación de evaluadores humanos (también conocidos como *crowdworkers*) para interactuar con el modelo de lenguaje original y etiquetar las respuestas en función de su nivel de informatividad, seguridad, veracidad y otros criterios relevantes. Dado que esto se realiza a gran escala, posibilita la reorientación del modelo de lenguaje con el propósito de transformarlo en un asistente verdaderamente útil.

Un ejemplo de esta técnica es ChatGPT de la empresa OpenAI, que emplea un enfoque denominado aprendizaje por refuerzo a partir de retroalimentación humana (Reinforcement Learning from Human Preferences RLHF) en su proceso de ajuste fino (ver Figura 9). En este proceso, el modelo se adapta mediante una

función de preferencia que otorga una puntuación a las respuestas generadas (esta se aprende en base a etiquetadores humanos), optimizando dicha función mediante algoritmos de optimización por refuerzo (Reinforcement Learning)³. Sin embargo, es fundamental tener en cuenta que esta estrategia incrementa los costos asociados a la construcción del modelo, debido a la necesidad de contratar a un gran número de personas para que evalúen las respuestas generadas por el modelo original.

Junto a ChatGPT, diversas empresas han incursionado en una competencia por lanzar al mercado sus propios asistentes, tales como Google Bard, YouChat, entre otros. Recientemente, la empresa OpenAI presentó su producto GPT-4, cuya característica principal es su capacidad para integrar imágenes a los *prompts*. Esto le permite al modelo responder a preguntas que contienen elementos visuales, como en las pruebas estandarizadas de admisión universitaria y certificaciones en diversos campos, como

² <https://huggingface.co/>.

³ El aprendizaje por refuerzo es una forma de aprendizaje automático en el cual un agente o programa aprende una política de acciones para maximizar una recompensa. Se usa para automatizar jugadores en juegos como el ajedrez y el Go, así como también en la robótica.



El diseño apropiado de prompts se ha transformado en un aspecto de vital importancia [...], ya que la calidad de las respuestas puede variar sustancialmente según el prompt empleado.

medicina y derecho, entre otros. No obstante, se dispone de escasa información detallada sobre los aspectos técnicos de GPT-4, ya que OpenAI únicamente proporcionó un informe técnico con limitada información sobre su construcción [16].

Sin embargo, la comunidad académica y de código abierto no ha permanecido rezagada en esta carrera. Apenas una semana después del lanzamiento de LLaMA por parte de META, sus pesos fueron filtrados al público a través de BitTorrent. Esto abrió la posibilidad para que iniciativas académicas con recursos más limitados en comparación con las grandes empresas pudieran ajustar estos modelos por su cuenta a partir de los pesos de LLaMA. Han surgido tecnologías que permiten realizar este ajuste sin requerir enormes capacidades de cómputo, mediante técnicas como Lora y QLora [17] que reducen la cantidad de bits de los pesos de la red, además de reducir la cantidad de parámetros necesarios para hacer el ajuste fino. Además, en lugar de aplicar el ajuste fino directamente sobre texto libre, se realiza sobre ejemplos de pares (instrucción, salida) que abordan diversas tareas, adaptando así el modelo en función de estas muestras.

Estos ejemplos de entrenamiento incluso se generan automáticamente a través de modelos cerrados como ChatGPT,

gracias a iniciativas como ShareGPT⁴, donde las personas ponen a disposición sus conversaciones con dicho modelo de forma gratuita. En esta misma línea, se han liberado modelos como Alpaca⁵ y Vicuna⁶. En julio de 2023, META lanzó su segunda generación de LLMs abiertos bajo el nombre de LLaMA 2, con distintas versiones de 7, 13 y 70 mil millones de parámetros respectivamente [18].

Además, dado la enorme cantidad de LLMs que salen todas las semanas, la comunidad académica ha ideado métodos ingeniosos para evaluarlos, como la “Chatbot Arena”⁷, una plataforma donde diversos modelos de lenguaje compiten para resolver tareas y reciben puntuaciones de manera análoga al sistema ELO utilizado en el ajedrez.

Conclusiones

Acabamos de recorrer la historia de los modelos de lenguaje, desde los simples modelos de n -gramas hasta los complejos LLMs basados en Transformers capaces de resumir, traducir, generar programas e inventar historias.

Tanto la comunidad científica como la sociedad civil han señalado diversos riesgos vinculados a esta tecnología [19]. Uno de estos riesgos notables es

la alucinación, que se refiere a la capacidad de estos modelos para generar información falsa. Además, emerge con fuerza la preocupación por la equidad, dado que estos modelos pueden agravar los sesgos presentes en los datos de entrenamiento, lo que lleva a la propagación de prejuicios relacionados con género, raza, orientación sexual y otras minorías.

La violación de derechos de autor emerge como un riesgo legal importante, ya que los LLMs pueden vulnerar las leyes de propiedad intelectual al reproducir contenido sin la debida autorización. Además, los altos costos asociados con el entrenamiento de los LLMs plantean inquietudes acerca de la posibilidad de una concentración monopólica. La transparencia también se ve comprometida, dado que la complejidad intrínseca de estos modelos dificulta la explicación de sus respuestas. Por último, el proceso de entrenamiento de estos modelos conlleva un alto impacto ambiental con una contribución significativa a las emisiones de carbono.

Dado lo acelerado que ha sido el progreso de los modelos de lenguaje en el último tiempo, es muy difícil predecir qué estarán haciendo en el futuro. Sí podemos decir con certeza que cada vez abundarán más modelos generativos de texto y otros tipos de datos (imágenes, video, código), como también programas (o agentes)⁸ que hacen uso automatizado de estos modelos para tomar acciones como comprar pasajes, invertir, agendar consultas médicas, entre otras posibilidades que aún no nos imaginamos. ■

4 <https://sharegpt.com/>.

5 <https://crfm.stanford.edu/2023/03/13/alpaca.html>.


6 <https://lmsys.org/blog/2023-03-30-vicuna/>.

7 <https://lmsys.org/blog/2023-05-03-arena/>.

8 En esta línea destacamos Langchain, un biblioteca de programación para generar aplicaciones basadas en LLMs: <https://python.langchain.com>.



REFERENCIAS

- [3] Shannon, C. E. (1951). Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64.
- [4] Chomsky, N. (2009). Syntactic structures. In *Syntactic Structures*. De Gruyter Mouton.
- [5] Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.
- [6] Ivakhnenko, A., & Lapa, V. G. (1965). Cybernetic Predicting Devices. CCM Information Corporation. *First working Deep Learners with many layers, learning internal representations*.
- [7] Bengio, Y., Ducharme, R., and Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems*, 13.
- [8] Harris, Z. (1954). Distributional structure. *Word*, 10(23): 146–162.
- [9] Firth, J.R. (1957). A synopsis of linguistic theory 1930-1955. In *Studies in Linguistic Analysis*, pp. 1-32. Oxford: Philological Society. Reprinted in F.R. Palmer (ed.), *Selected Papers of J.R. Firth 1952-1959*, London: Longman (1968).
- [10] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.
- [11] Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In *Interspeech*, Volumen 2, No. 3, pages 1045–1048.
- [12] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- [13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gómez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- [14] Kenton, J. D. M.-W. C. and Toutanova, L. K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.
- [15] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), 9.
- [16] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- [17] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., ... & Fedus, W. (2022). Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.
- [18] OpenAI (2023). Gpt-4 technical report.
- [19] Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized LLMs. *arXiv preprint arXiv:2305.14314*.
- [20] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., ... & Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- [21] Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021, March). On the dangers of stochastic parrots: Can language models be too big? . In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency* (pp. 610–623).



Inteligencia artificial, ciudadanía digital y pedagogía



MARTÍN CÁCERES

Licenciado en Física y Doctor en Ciencias de la Ingeniería de la Pontificia Universidad Católica de Chile. Ha trabajado como profesor de física, electrónica y robótica, y en el sistema universitario ha dictado cursos de humanidades digitales y metodologías innovadoras de enseñanza y aprendizaje. Además, ha trabajado en asesorías a escuelas y universidades y en política pública relacionada con innovación pedagógica en Chile y en el extranjero. Actualmente es director del Centro de Innovación - Enlaces, del Ministerio de Educación.

✕ @marcaceresm



RESUMEN. En este artículo, propongo algunos elementos para abordar el debate sobre cómo enfrentar la irrupción de la inteligencia artificial desde el aprendizaje que como humanidad hemos tenido en momentos previos, ya que como se verá, la situación actual no es completamente inédita. Posteriormente, explico la perspectiva que hemos tomado desde el Mine-duc, centrada en el fortalecimiento de la Ciudadanía Digital y el uso de las tecnologías para promover el aprendizaje activo.

En los últimos meses han salido a la luz pública varias tecnologías basadas en LLMs (Large Language Models), como ChatGPT de OpenAI o Bard de Google, en un vertiginoso avance de la capacidades de las tecnologías que utilizan la inteligencia artificial para trabajar con lenguaje natural.

Los LLMs son entrenados con grandes bases de datos para “aprender” a identificar patrones y estructuras del lenguaje natural. A este tipo de modelos se les llama inteligencia artificial generativa, porque son capaces de generar contenido nuevo y coherente, como texto, imágenes o audio, a partir de lo que han aprendido durante el proceso de entrenamiento.

A la fecha, estos modelos son capaces de igualar o sobrepasar a los seres humanos en innumerables tareas que hasta hace poco se consideraban exclusivamente humanas, como pruebas estandarizadas de selección múltiple e ítemes abiertos en múltiples áreas del conocimiento, interpretación de imágenes, diseño de imágenes o audios a partir de texto, resolución de problemas complejos y escritura de todo tipo de textos.

Estos avances han generado un gran impacto mediático por sus posibles consecuencias en diferentes ámbitos

Las alarmas grandilocuentes sobre el impacto de las tecnologías en las formas de enseñar y aprender están documentadas desde hace siglos.

como el mundo del trabajo, donde se proyecta la obsolescencia de muchas profesiones y la aparición de nuevas ocupaciones, las implicancias éticas de sus orígenes y sus usos, los riesgos relacionados con la ciberseguridad, la protección de los datos personales y el devenir de las democracias, por nombrar algunos temas en que se podría profundizar largamente.

En el ámbito educacional, se ha planteado que estas tecnologías reemplazarían el rol de los docentes, que ya no sería posible evaluar productos generados por estudiantes al no poder certificar si el trabajo es de ellos o generado por estos sistemas, y que los riesgos relacionados con la ética y la seguridad son insoslayables.

Así, en el debate público en general y también en el educacional, se ha llamado a detener el avance y adopción de estas tecnologías, al proyectarse que los riesgos son mucho mayores a sus beneficios.

Las alarmas grandilocuentes sobre el impacto de las tecnologías en las formas de enseñar y aprender están documentadas desde hace siglos. Alrededor del año 370 AC, Platón ilustraba los peligros del avance de una disruptiva tecnología, que, según cuenta Sócrates a Fedro, el dios egipcio Theuth entregaba al rey Thamus. Thamus, preocupado por los impactos que tendría este invento en los humanos, le responde:

“Este invento dará origen en las almas de quienes lo aprendan al olvido, por descuido del cultivo de la memoria, ya que los hombres, por culpa de su confianza en la escritura, serán traídos al recuerdo desde fuera, por unos caracteres ajenos a

ellos, no desde dentro, por su propio esfuerzo (...)

Apariencia de sabiduría y no sabiduría verdadera procuras a tus discípulos. Pues habiendo oído hablar de muchas cosas sin instrucción, darán la impresión de conocer muchas cosas, a pesar de ser en su mayoría unos perfectos ignorantes; y serán fastidiosos de tratar, al haberse convertido, en vez de sabios, en hombres con la presunción de serlo”.

De la misma forma, cada avance disruptivo en las tecnologías de la información ha generado intensos debates sobre sus consecuencias en la sociedad, en las personas y en la educación.

Con la invención de la imprenta, de la fotografía, de la radio, de la televisión, de las computadoras y más recientemente del Internet, se ha planteado que la forma de enseñar y aprender cambiaría para siempre, que los docentes serían reemplazados y que la debacle social era inminente. Cada uno de estos inventos revolucionaría los sistemas educativos para siempre.

En un ensayo de 1956, Gabriela Mistral toma posición en el debate sobre la incorporación de la imagen en las salas de clase (ver Figura 1). Comparto algunos pasajes notables del ensayo:

“Al hogar de la Palabra, que llamamos Escuela o Colegio, ha llegado un competidor formidable: la imagen.

(...) Gran oposición tuvo el indiscutible en sus comienzos: los profesores le daban un ceño hostil, porque pensaban en que aquello llegaría a suprimirlos, cosa que no ocurrió ni ocurrirá nunca. Lo que nació fue la



Figura 1. Imagen generada por Dall-E, con el *prompt* “Gabriela mistral teaching with multiple images that have animals and landscapes in a 1950 rural classroom daist collage”.

alianza de la Palabra con la Imagen y tal fusión benefició a ojos vistas el gran asunto de la alfabetización.

La batalla de convencer ha sido larga y se puede decir que aún se lucha por ella en varios cantos del mundo, pero tarde o temprano, y gracias al auge que ha obtenido el cine, los profesores comprenderán que el huésped cuya presencia les pareció un peligro, es realmente el mayor y el mejor de sus aliados.

Hubo un desdén muy grande de los profesores, primarios o secundarios, respecto del valor decisivo de la ima-

gen en la enseñanza, de su utilidad y de su magia, sobre todo de las sugerencias que ella regala.

(...) La vida de los escolares suele correr en la monotonía sin apelativos de una sala de clase en la cual resuena la voz de diez o más profesores ilustres a veces y hasta amados por sus discípulos; pero ¿existe alguien que pueda gozar de una descripción larga y sin que su alegría de aprender se relaje y su pensamiento se escape huyendo al tedio?”

Como puede verse, Gabriela Mistral da respuesta a varias de las inquietudes

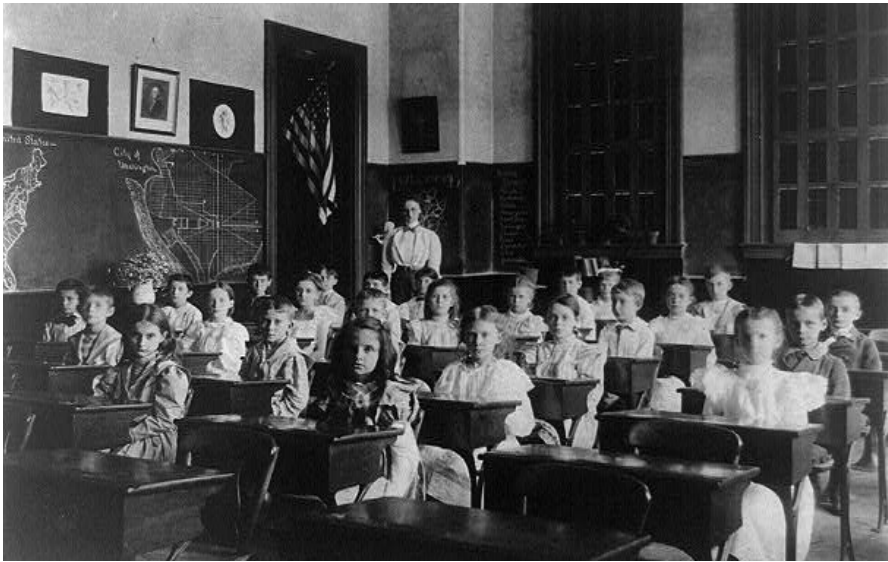
que hoy mismo se plantean con la inteligencia artificial. Tranquiliza a los docentes indicando que no serán reemplazados jamás, y muestra el gran potencial pedagógico que tiene la imagen por su contenido semiótico y el cine por su capacidad de mostrar imágenes vivas y llegar a multitudes.

Por otro lado, la poeta nos muestra lo resilientes que suelen ser los sistemas educativos a los cambios. A pesar de las grandes transformaciones socio-técnicas experimentadas en nuestras sociedades en los últimos 120 años — que efectivamente han transformado el mundo del trabajo, las formas de relacionarnos, de entretenernos, de acceder y crear conocimientos, entre muchos otros aspectos— los sistemas educativos, y sobre todo la experiencia educativa que en ellos se involucra a docentes y estudiantes, no ha cambiado demasiado (ver Figuras 2 y 3).

¿Será distinto esta vez?

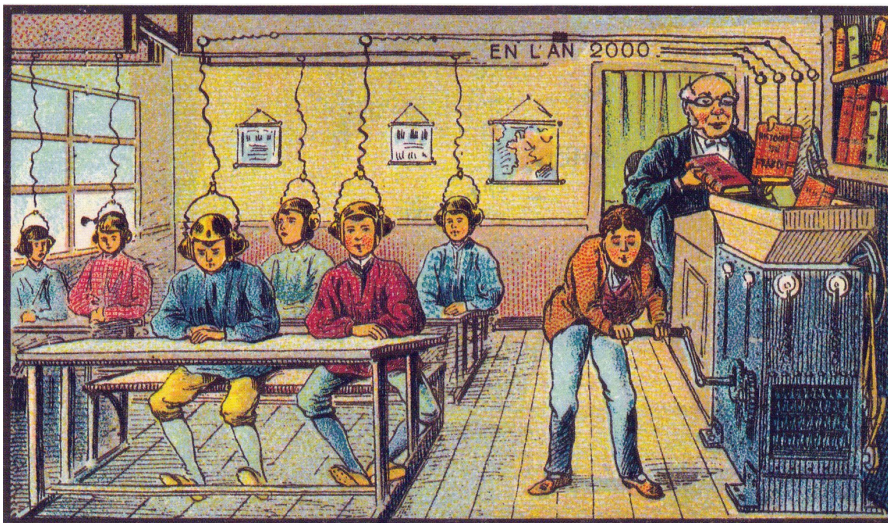
Hay razones para pensar que este cambio sí será profundo, por lo menos en el ámbito social. Hoy en día, hay más teléfonos celulares que personas en Chile. Herramientas como ChatGPT están en el bolsillo de docentes, estudiantes y todo aquel que tenga un teléfono inteligente con conexión a Internet: potencialmente toda la ciudadanía. La actual ubicuidad de los dispositivos y la conectividad nos pone en un escenario diferente a cuando irrumpieron los grandes inventos mediales anteriores: la escritura, la imprenta, la televisión, incluso las computadoras e Internet la década pasada, eran propiedad de un grupo reducido de la población.

Las características de la inteligencia artificial generativa, además, tienen implicancias profundas que llaman a cuestionamientos que atañen al corazón de la pedagogía: ¿cuáles son los conocimientos, habilidades y actitudes



Fuente: <https://www.loc.gov/item/96525653/>.

Figura 2. Aula del año 1900.



Fuente: <https://slate.com/human-interest/2015/11/en-lan-2000-is-a-series-of-visionary-illustrations-of-the-year-2000-from-paris-world-exhibition-in-1900.html>.

Figura 3. Aula del año 2000, imaginada en 1904.

que se requieren en un mundo donde es indistinguible si un texto o imagen es producido por una persona o una máquina?, ¿cómo evaluar el trabajo de los estudiantes, si tienen al alcance de la mano herramientas con las que pueden resolver cualquier proble-

ma?, ¿cuál es el rol del docente —y de la escuela— si las máquinas pueden responder —más o menos bien— cualquier pregunta que tengamos?

Estas preguntas, si bien se agudizan al ser la inteligencia artificial generativa

¿Cuáles son los conocimientos, habilidades y actitudes que se requieren en un mundo donde es indistinguible si un texto o imagen es producido por una persona o una máquina?

una herramienta mucho más poderosa que sus antecesoras, no son nuevas. Por cierto, las respuestas tampoco lo son. Volvamos a la historia.

Desde comienzos del siglo XX, el movimiento de la Nueva Escuela planteaba que, frente a los desafíos de las nacientes sociedades democráticas, era imperativo reformar las pedagogías estandarizadas y despersonalizadas, basadas en el control y en la repetición. Para el fortalecimiento de las democracias, la escuela debía ser un lugar donde se construyan vínculos entre personas y se cultiven reflexiones profundas y continuas en base a los conocimientos disciplinares y las problemáticas de las comunidades y sus entornos, poniendo en práctica los valores y principios democráticos, para, desde ahí, fortalecer las democracias.

Cien años después, vale la pena volver a esta misma reflexión (ver Figura 4). La inteligencia artificial generativa pone en crisis las pedagogías estandarizadas, ya que en la carrera de la desconfianza difícilmente ganará la capacidad de controlar. En cambio, podemos considerar las herramientas de la inteligencia artificial como potenciadores de la reflexión, el vínculo y la creatividad, lo que distintos autores llaman aprendizaje activo, profundo o auténtico. Parafraseando las palabras de Walter Benjamin en su célebre ensayo “La obra de arte en la era de su reproductibilidad técnica”, podemos decir que frente a la



Figura 4. Imagen generada por Dall-E, con el prompt “Walter Benjamin fractally reproducing himself vaporwave”.

tecnologización de la humanidad por parte del neoliberalismo, nuestra tarea es humanizar la educación.

¿Cómo concretar esta mirada en términos de política educativa y práctica pedagógica?

Desde el Mineduc proponemos abordar los desafíos de la irrupción de la inteligencia artificial a través del fortalecimiento de la Ciudadanía Digital. Desde el año pasado, estamos proponiendo un nuevo marco, que la entiende como el desarrollo de conocimientos, habilidades y actitudes fundamentales para que las personas puedan ejercer sus derechos digitales y fortalecer la convivencia democrática, mediante el uso seguro, responsable, participativo, creativo, crítico y reflexivo de las tecnologías

digitales, comprendiendo la influencia de estas en su vida personal y pública a nivel local y global.

A partir de esta definición, hemos definido cuatro dimensiones para actuar:

Alfabetización digital crítica y reflexiva

La alfabetización digital crítica y reflexiva se entiende como la adquisición de conocimientos y el desarrollo de habilidades y actitudes para usar, comprender y evaluar autónomamente las tecnologías digitales. Se trata de una dimensión base, pues, por un lado, las y los ciudadanos requieren herramientas para usar efectiva y exitosamente dichas tecnologías y, por otro, deben ser conscientes de las fuentes, brechas, sesgos y consecuencias de las tecnologías digitales, reflexionando desde diversas perspectivas.

Cuidado y responsabilidades digitales

El cuidado y las responsabilidades digitales se refieren a la construcción de ambientes digitales seguros y la participación en ellos, promoviendo el bienestar individual y colectivo. Lo anterior implica una reflexión y práctica ética respecto del uso de las tecnologías digitales y el comportamiento en redes sociales, tanto como el conocimiento de los derechos digitales, la privacidad, la protección de datos y conciencia de la huella digital. Esta dimensión tiene como horizonte la convivencia democrática y pluralista en ambientes digitales, resguardando la inclusión de la diversidad, la perspectiva de género, la interculturalidad y los derechos humanos.

Participación ciudadana digital

La participación ciudadana digital corresponde al involucramiento de las y los ciudadanos en los asuntos públicos y el cuidado de la democracia mediante el uso de las tecnologías digitales como plataformas de comunicación, difusión, debate y organización. Esta dimensión es fundamental en tanto los espacios virtuales, dotados de agencia política y social, se han convertido en una extensión del espacio público con capacidad de incidir en la democracia y su institucionalidad, ya sea para su profundización o debilitamiento.

Creatividad digital e innovación

La creatividad e innovación desde la ciudadanía digital se refiere a los procesos de generación de contenidos, expresión de ideas y planteamiento de propuestas para contribuir a la solución de problemas territoriales y globales, haciendo uso de herramientas y tecnologías digitales. El desarrollo de esta dimensión está orientado por principios relacionados con la sustentabilidad, el buen vivir, la interculturalidad, la justicia social y los



Lo que proponemos desde Mineduc es un camino diferente. Es posible utilizar la inteligencia artificial generativa como una herramienta para fomentar la creatividad, la reflexión, la colaboración y el pensamiento crítico.



<https://www.youtube.com/watch?v=JMLsHI8aV0g>.

Figura 5. Estudiantes aprendiendo con un cintillo que monitorea la atención.

derechos humanos. Implica el desarrollo de habilidades tales como el trabajo colaborativo, el pensamiento crítico y la resolución de problemas.

Proponemos abordar desde esta mirada los distintos desafíos que enfrentamos con la irrupción de las nuevas tecnologías. Por un lado, vemos necesario educar acerca de ellas. Es decir, comprender cómo funcionan, cuáles son los fundamentos y desde ahí adoptar posturas críticas sobre su uso. En el caso de la inteligencia artificial generativa, o los LLMs, necesitamos comprender, por ejemplo, cómo se entrenan, quién los entrena y cuáles son las reglas que se fijan para su entrenamiento. Esta comprensión nos permite reflexionar acerca de sus limitaciones, y desde ahí poder decidir para qué usarlo y para qué no.

Teniendo esta comprensión crítica, nuestra invitación es que aproveche-

mos su potencial para fortalecer nuestro trabajo en el proceso de enseñanza y aprendizaje, ya que puede ser de gran utilidad tanto para docentes como para estudiantes. Por un lado, podemos pensarlo como una herramienta de eficiencia. Existen múltiples formas de utilizar estas tecnologías para ahorrar tiempo, como obtener ideas para planificaciones de clases, evaluar de forma personalizada, o generar instrumentos de evaluación como pautas o rúbricas.

Sin dudas, es interesante todo lo que se puede hacer en términos de eficiencia, pero vale la pena preguntarnos si la usaremos para hacer lo mismo que hacemos actualmente, o si nos permite incorporar prácticas que antes era imposible realizar. Esta pregunta es clave ante la adopción de cualquier tecnología.

Un ejemplo para ilustrar esta definición es el uso de cintillos que leen la activi-

dad cerebral de estudiantes para determinar si están atentos o no en clase, y así generar reportes automáticos a docentes y apoderados (ver Figura 5).

¿Qué idea de pedagogía fomentamos con este tipo de aplicaciones?

Lo que proponemos desde Mineduc es un camino diferente. Es posible utilizar la inteligencia artificial generativa como una herramienta para fomentar la creatividad, la reflexión, la colaboración y el pensamiento crítico, fortaleciendo el protagonismo de los y las estudiantes y vinculando el aprendizaje con sus intereses. Podemos utilizar ChatGPT, por ejemplo, para obtener ideas de proyectos interdisciplinarios, adaptar y conectar ideas de diferentes ámbitos y diferentes contextos, generar ideas divergentes para debates en clases, darle diferentes roles para que se conviertan en tutores personalizados para reflexionar (pueden encontrar más ideas en la “Guía para docentes: cómo utilizar ChatGPT para potenciar el aprendizaje activo”¹).

Una pregunta recurrente al conversar con docentes sobre la inteligencia artificial es cómo hacerlo con la evaluación de los trabajos de estudiantes en tareas que ahora pueden hacer las máquinas. Si sólo nos centramos en los productos, por supuesto que se hace difícil, ya que existen mecanismos para evadir los detectores de plagio. Lo que proponemos, entonces, es preguntarnos cómo y para qué evaluamos.

Por ejemplo, si como docentes evaluamos una unidad mediante un ensayo escrito que no trabajamos en clase, y en el tiempo lectivo nos dedicamos sólo a “pasar la materia”, es altamente probable que el ensayo sea escrito por

¹ <https://ciudadaniadigital.mineduc.cl/wp-content/uploads/2023/05/Guía-para-Docentes-Como-usar-ChatGPT-Mineduc.pdf>.



Figura 6. Imagen generada por Dall-E, con el prompt “the creation of adam complete scene but interacting with artificial intelligence as Michaelangelo would paint it renaissance fresco”.

ChatGPT o una herramienta similar. Frente a este escenario, podemos cambiar el enfoque y desde la primera clase trabajar con ChatGPT como asistente para escribir el ensayo: obtener ideas, contraargumentos, debatir en base a sus respuestas, y utilizar todos estos diálogos para,

en clase, avanzar en la escritura. De esta forma el producto no es lo más importante, sino el proceso cognitivo en que nos involucramos (ver Figura 6).

Sin duda, cambiar las prácticas pedagógicas es difícil y nos queda un largo

camino, ya que se requiere generar condiciones para promover la innovación educativa, pero estamos convencidos de que avanzar en este camino es urgente y se debe comenzar hoy. Nos alegra ver que ya existen docentes en todo el país interesados en promover estos cambios, y desde el Mineduc estamos comprometidos en su concreción, a través de una serie de medidas como la actualización del currículo nacional, la mejora de las condiciones de trabajo de docentes y las oportunidades de desarrollo profesional, el reconocimiento y promoción de la innovación educativa y el fortalecimiento del acceso equitativo a las tecnologías.

Es importante destacar que existen ámbitos que van más allá de lo educativo en los que debemos trabajar coordinadamente desde la institucionalidad del estado, como la regulación del desarrollo de la inteligencia artificial, de forma que las compañías trabajen de forma transparente y ética, en la elaboración de sistemas de gobernanza que permitan tomar decisiones en cuanto a su desarrollo y velar por que la transformación digital no genere nuevas exclusiones. Por eso, desde el Ministerio de Ciencia, Tecnología, Conocimiento e Innovación se está trabajando en la actualización de la Política Nacional de Inteligencia Artificial para articular este marco de acción. ■

REFERENCIAS

- [1] Benjamin, W. (2008). La obra de arte en la época de su reproductibilidad técnica. *Obras Completas, Libro II, 2*.
- [2] Dewey, J. (1995). *Democracia y educación: una introducción a la filosofía de la educación*. Ediciones Morata.
- [3] Mistral, G. (1956). Imagen y palabra en educación. En: *Magisterio y niño*. Ed. de Roque Esteban Scarpa. Santiago de Chile: Editorial Andrés Bello, 1979. <http://www.gabrielamistral.uchile.cl/prosa/imgypaleneduc.html>.
- [4] Mineduc (2023). Guía para docentes: Cómo usar ChatGPT para potenciar el aprendizaje activo: <https://ciudadaniadigital.mineduc.cl/wp-content/uploads/2023/05/Guia-para-Docentes-Como-usar-ChatGPT-Mineduc.pdf>.
- [5] Mineduc (2023 B). Ciudadanía Digital para los desafíos de las comunidades educativas: <https://ciudadaniadigital.mineduc.cl/wp-content/uploads/2023/05/Ciudadania-Digital-Mineduc-2023.pdf>.
- [6] MinCiencia (2019). Política Nacional de Inteligencia Artificial: https://www.minciencia.gob.cl/uploads/filer_public/bc/38/bc389daf-4514-4306-867c-760ae7686e2c/documento_politica_ia_digital_.pdf.
- [7] Platón. FEDRO. Platón, Obras completas, edición de Patricio de Azcárate, tomo 2, Madrid 1871.



¿Cómo navegar el camino hacia la ética en IA?





CLAUDIA LÓPEZ MONCADA

Doctora en Ciencias de la Información y Tecnología por la Universidad de Pittsburgh. Profesora Asistente del Departamento de Informática de la Universidad Técnica Federico Santa María e investigadora en el Centro Nacional de Inteligencia Artificial (CENIA) y en el Núcleo Milenio Futures of Artificial Intelligence Research (FAIR). Líneas de investigación: computación centrada en personas.

✉ claudia.lopez@usm.cl



GABRIELA ARRIAGADA BRUNEAU

Magister en Filosofía por la Universidad de Edinburgh y candidata a doctora por la Universidad de Leeds. Profesora Asistente en el Instituto de Éticas Aplicadas y el Instituto de Ingeniería Matemática y Computacional, Pontificia Universidad Católica de Chile. Líneas de investigación: ética de la IA y datos, filosofía de la tecnología.

✉ gcarriagada@uc.cl



ALEXANDRA DAVIDOFF

Socióloga por la Pontificia Universidad Católica de Chile. Asistente de investigación en el Núcleo Milenio Futures of Artificial Intelligence (FAIR). Líneas de investigación: ética en IA e infancia.

✉ eadavidoff@uc.cl

RESUMEN. En este artículo relatamos cómo el Centro Nacional de Inteligencia Artificial (CENIA) busca abordar las consideraciones éticas en sus proyectos de investigación de IA. Hemos conformado un Grupo de Trabajo en Ética (GTE) con representantes de cada línea de investigación del Centro y, en conjunto, hemos explorado diferentes desafíos que aquí resumimos.

Primero, discutimos si el enfoque ético en IA requiere un análisis adicional al de los comités de ética de investigación. Luego, revisamos los principios que se han propuesto para resolver los problemas éticos que emergen alrededor de la IA. Finalmente, presentamos la iniciativa de CENIA para priorizar principios éticos propios, que involucra el desarrollo de un enfoque metodológico para evaluarlos. Como conclusión, destacamos la importancia de la ética como una herramienta para mejorar el desarrollo de la IA y subrayamos la necesidad de enfoques éticos específicos para la región latinoamericana.

Lo que inició con preocupaciones por privacidad de datos personales, imparcialidad/justicia (*fairness*) de los resultados y opacidad de los modelos de in-

teligencia artificial (IA) [1, 2, 3], hoy se ha desarrollado hacia múltiples aristas éticas, que van de la mano del despliegue masivo de la IA en nuestra vida co-

tidiana, como su impacto en el medio ambiente, su relación con los derechos humanos, la falta de mecanismos de rendición de cuentas, entre otras.

En el Centro Nacional de Inteligencia Artificial (CENIA)¹, que busca que la IA esté al servicio de las personas, esto ha inspirado una pregunta central en nuestro trabajo: ¿cómo abordamos las diversas consideraciones éticas que están involucradas en un proyecto de investigación en IA?

Con este propósito, hemos reunido a un Grupo de Trabajo en Ética en CENIA, con el objetivo de enriquecer el diálogo sobre la integración de la ética en la IA en Chile. Este artículo presenta un resumen de algunas de las preguntas e incipientes respuestas que hemos logrado desarrollar hasta el momento.

1 <https://www.cenia.cl/>.

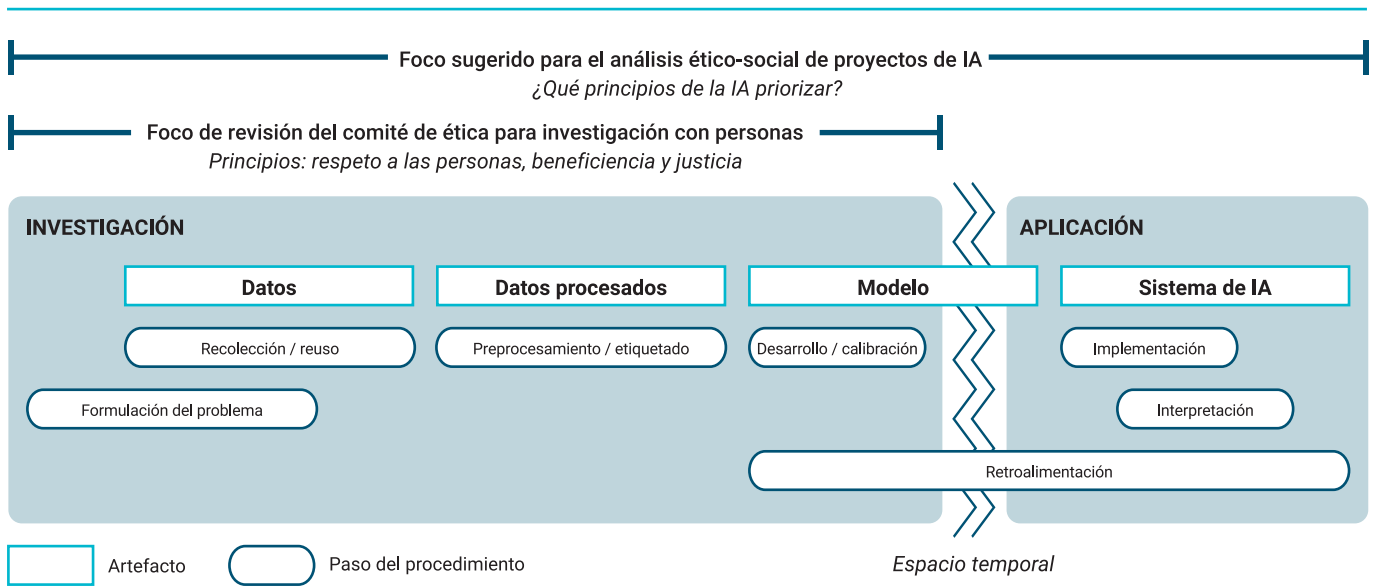


Figura 1. Ilustración del foco de análisis de los comité de ética versus análisis para proyectos de IA.

¿La ética en IA requiere un análisis diferente al de un comité de ética de investigación?

Siguiendo la literatura, hemos llegado a la conclusión que, efectivamente, existen diferencias fundamentales entre las funciones desempeñadas por estos dos enfoques. Si bien son mutuamente complementarios, los comités de ética institucionales están centrados en salvaguardar los derechos de los participantes en la investigación, poniendo un énfasis en la mitigación de los riesgos inherentes durante la ejecución de los estudios (tales como la potencial pérdida de privacidad de datos personales y los posibles efectos secundarios adversos para la salud o la integridad de los sujetos involucrados). No obstante,

la evidencia respalda la noción de que los riesgos asociados a la IA a menudo emergen en etapas posteriores a la conclusión de un proyecto de investigación, cuando la tecnología se encuentra en pleno funcionamiento [4].

En la Figura 1, ilustramos el alcance del análisis de riesgos de ambos enfoques, utilizando como ejemplo un flujo estándar en un proyecto de aprendizaje automático basado en datos. El enfoque convencional del análisis adoptado por los comités de ética tiende a centrarse en la fase inicial del proceso. De esta manera, el análisis ético vinculado a la implementación de la inteligencia artificial (IA) experimenta una ruptura temporal, considerándose como una etapa posterior o futura en la formulación de un proyecto.

Diversos casos que exponen los riesgos inherentes a la IA han sido documentados, destacándose la ocurrencia de

perjuicios hacia personas en su etapa de aplicación. Estos perjuicios abarcan desde la asignación incorrecta de categorías que afectan el acceso a beneficios gubernamentales, hasta la lesión o exclusión de individuos en procesos de contratación debido a la discriminación fundada en características como género, raza u otros atributos protegidos. A esto se agrega la ausencia de mecanismos que permitan a los usuarios solicitar correcciones o presentar apelaciones en situaciones donde un sistema basado en IA opere de manera errónea, ocasionando daños a nivel individual y grupal en la sociedad.²

En el presente contexto, académicos como Bernstein et al. [5] han propuesto poner la atención hacia los riesgos que afectan a la sociedad en fases subsiguientes a la investigación, tales como la implementación o comercialización de la inteligencia artificial, así como su

² Un reconocido caso ilustrativo es el sistema COMPAS (Correctional Offender Management Profiling for Alternative Sanctions), utilizado en los tribunales de los Estados Unidos para evaluar el riesgo de reincidencia de un individuo. Investigaciones concernientes a esta tecnología revelaron que su índice de falsos positivos era notablemente superior en el caso de personas afroamericanas en comparación con las de origen caucásico. Esta disparidad reproducía las desigualdades raciales presentes en el sistema de justicia de dicho país [6, 7].



aplicación en la formulación de políticas públicas. En esencia, abogan por asumir la responsabilidad de considerar las posibles implicancias de los resultados obtenidos en nuestras investigaciones y buscar formas de atenuar los riesgos a través de la toma de decisiones, proporcionando recomendaciones para aplicaciones futuras. Con este propósito, los autores sugieren abordar tres interrogantes fundamentales en cada proyecto: (1) ¿cuáles son los riesgos involucrados?, (2) ¿qué principios deben seguirse para mitigar estos riesgos?, y (3) ¿cómo se concretan estos principios en el diseño de la investigación?

En la construcción de estas interrogantes, Bernstein et al. [5] siguen una línea de razonamiento similar a la de los comités de ética, al conectar la mitigación de riesgos con principios que tienen como propósito resolver los dilemas éticos que surgen en el transcurso de una investigación, tal como fue propuesto en el Informe Belmont [8]. Dicho informe establece tres principios fundamentales para la investigación con personas: (1) respeto a las personas (autonomía y protección de vulnerabilidad), (2) beneficencia (bienestar y prevención de daños), y (3) justicia (trato justo y no discriminatorio).

El desafío en este momento, por tanto, es reflexionar sobre la ética en IA y poder identificar qué principios nos ayudarán a resolver los problemas establecidos y emergentes, más allá de cuestionamientos metodológicos sobre la formulación del proyecto o las responsabilidades investigativas.

¿Qué principios de la IA son relevantes según el panorama internacional?

En el contexto del rápido avance de la IA, el diálogo a nivel global enfatiza la importancia de orientar su desarrollo en base a principios que favorezcan el bien-

Hemos llegado a la conclusión de que [...] existen diferencias fundamentales entre las funciones desempeñadas por [...] los comités de ética de investigación [y] el análisis para proyectos de IA.

estar de la sociedad en su conjunto. Esta conversación abarca tanto a actores del ámbito público como del privado, con y sin fines de lucro, los cuales han articulado sus propios principios en consonancia con las prioridades específicas de sus esferas de influencia.

Hace tres años el Centro Berkman Klein de Harvard, publicó una taxonomía de los principios de la IA [9] fundamentada en un análisis de 36 documentos que enuncian principios relacionados con la IA, a nivel nacional, multilateral y organizacional. Esta taxonomía identifica ocho ejes temáticos para categorizar los principios de la IA: privacidad, rendición de cuentas, seguridad y protección, transparencia y explicabilidad, imparcialidad/justicia (*fairness*) y no discriminación, control humano de la tecnología, responsabilidad profesional y promoción de valores humanos. Asimismo, otros organismos como la UNESCO [10] y la ONU [11] también han propuesto recomendaciones, añadiendo aspectos como la alfabetización, sustentabilidad, inclusión (enfaticando género), y proporcionalidad del uso de la IA para alcanzar un objetivo legítimo.

Por otra parte, grandes empresas como Google [12], IBM [13] y Microsoft [14] han formulado sus propios protocolos y principios. Aunque en términos generales están en línea con los mencionados previamente, también ponen énfasis en la aplicación y los posibles usos de sus productos. Además, destacan la relevancia del profesionalismo y la excelencia científica en el proceso de desarrollo tecnológico. Sin embargo, es esencial notar que estos lineamientos han sido ampliamente criticados dada su limitada influencia en la práctica. De hecho, la literatura alerta sobre el uso de los linea-

mientos éticos como un mecanismo de lavado de imagen (*ethics washing*, en inglés). Más específicamente, las críticas enfatizan cómo las tendencias de operacionalización de los principios éticos sin una reflexión profunda y prudente, ha terminado por subordinar a la ética como una herramienta para los intereses económicos de las grandes corporaciones tecnológicas [15].

Esto nos deja con un escenario de amplios intereses, motivaciones, y principios, haciendo dificultosa la identificación de directrices éticas a seguir.

¿Qué principios de la IA son relevantes desde el punto de vista local?

Por esas dificultades es importante enfocarse en el punto de vista local. En 2021, el Ministerio de Ciencia, Tecnología, Conocimiento e Innovación de Chile, desarrolló una Política Nacional de IA [16], que presenta cuatro principios transversales: (1) IA con centro en el bienestar de las personas, respeto a los derechos humanos y la seguridad; (2) IA para el desarrollo sostenible, (3) IA inclusiva e (4) IA globalizada y en evolución. Además, propone tres ejes estructurales: (1) factores habilitantes, (2) desarrollo y adopción, y un último eje (3) ética, aspectos normativos, e impactos socioeconómicos. Hasta la fecha, siguen trabajando en profundizar este último eje de manera participativa en conjunto con la UNESCO, con el objetivo de presentar una actualización de la Política en octubre de 2023. Además, desde la normatividad legislativa, existe una propuesta de ley para “regular los sistemas



- 1 Preocupaciones centrales relacionadas a la presencia de sesgos, los riesgos de violación de privacidad, y la necesidad de democratizar la IA (relacionado a la representatividad y accesibilidad) tanto para usuario/as e investigadore/as.
- 2 Los principios que son frecuentemente identificados como relevantes son los de imparcialidad/justicia (*fairness*) y transparencia. También se releva la importancia de la privacidad y el resguardo del impacto general de la implementación de IA.
- 3 El género surge como temática transversal, debido a la baja participación de mujeres en IA y STEM, lo que combinado con desigualdades estructurales puede traducirse en la reproducción de sesgos de género y falta de oportunidades para las mujeres.
- 4 Hay consenso respecto a la relevancia de la ética. Pero preocupación por la carencia de lineamientos y mecanismos que permitan llevar la ética a la práctica, considerando trabas institucionales, como también individuales en la labor de cada estamento.
- 5 Un aspecto fundamental es la percepción de la ética desde la negatividad o la prohibición, una imposición restrictiva a la investigación o innovación en desarrollo. En este sentido hay una percepción de lo ético como un obstáculo, más que una oportunidad de mejora.

Figura 2. Tendencias centrales para la indagación exploratoria.

de inteligencia artificial, la robótica y las tecnologías conexas, en sus distintos ámbitos de aplicación” [17], la cual está siendo discutida en la Comisión de Desafíos del Futuro, Ciencia, Tecnología e Innovación de la Cámara de Diputadas y Diputados, y la Mesa “Legislado sobre IA” convocada por la Comisión de Desafíos del Futuro, Ciencia, Tecnología e Innovación del Senado de Chile.

Estos pasos a nivel nacional son sólo una de las realidades regionales. Vale la pena destacar las considerables diferencias en las percepciones públicas sobre los avances de IA en la región latinoamericana, según la caracterización hecha por el reciente Índice Latinoamericano de IA [18]. En el índice se enfatiza que es importante tener en cuenta las diferencias culturales y socioeconómicas al analizar la percepción de la IA en cada país, así como considerar la falta de perspectiva crítica asociada al discurso público sobre su desarrollo.

Teniendo en consideración las tendencias regionales, así como la transversal falta de entrenamiento formal en ética en IA en la formación profesional de áreas STEM [19, 20], a fines del 2022 decidimos realizar una investigación exploratoria sobre el conocimiento y las percepciones sobre ética de IA de los distintos integrantes de CENIA. A partir de esta indagación, se corroboró la escasez de conocimiento formal en el campo, aunque también se pudo identificar una gran heterogeneidad en términos de conocimiento informal, determinada por el propio interés o la relevancia del tema de acuerdo con las demandas del rol laboral de cada participante.

Para esta investigación exploratoria, se utilizó una metodología cualitativa, que constó con un total de siete entrevistas semiestructuradas individuales y una entrevista grupal, así como tres *focus groups*, a integrantes de distintos cargos como investigadores, cargos adminis-

Es importante tener en cuenta las diferencias culturales y socioeconómicas al analizar la percepción de la IA en cada país.

trativos, desarrolladores y estudiantes. A partir de una metodología de análisis de discurso por medio de rejilla, las respuestas de los participantes se codificaron de acuerdo a los objetivos de la entrevista en lo que refiere a conocimiento, valoraciones, percepciones y prácticas de los participantes. En la Figura 2 presentamos cinco aspectos esenciales que surgieron de estas entrevistas.

En relación particularmente a la percepción de la ética desde la negatividad, es que nos parece necesario reenforzar la percepción de la ética, no como una limitante, sino que como una oportunidad metodológica para mejorar el desarrollo de la IA. Para esto, nos propusimos formular una guía de trabajo para el análisis ético-social. En esta guía, el propósito es robustecer proyectos de investigación identificando nuevas o mejores preguntas base para la formulación de problemas, y el análisis de riesgo.

Ante esta serie de preocupaciones y principios, ¿por dónde empezar?

Como una forma de hacer frente a la multiplicidad de jerarquías posibles asociadas al uso de los principios y preocupaciones alrededor de la IA, el Grupo de Trabajo de Ética (GTE) de CENIA ha buscado sintetizar principios transversales que representen los intereses de nuestro centro. El GTE, constituido por solicitud del Comité Científico de CENIA, está conformado por un representante de cada



Documento	Principios
Fjeld, J., Achten, N., Hilligoss, H., Nagy, A., & Sri Kumar, M. (2020). Principled AI: A Map of Ethical and Rights-Based Approaches to Principles for AI.	Privacidad, rendición de cuentas, imparcialidad/justicia (<i>fairness</i>), seguridad, responsabilidad profesional, promoción de valores humanos, transparencia, control humano de la tecnología.
Khan, A. A., Badshah, S., Liang, P., Khan, B., Waseem, M., Niazi, M., & Akbar, M. A. (2021). Ethics of AI: A Systematic Literature Review of Principles and Challenges.	Transparencia, privacidad, responsabilidad, equidad, autonomía, explicabilidad, imparcialidad/justicia (<i>fairness</i>), no maleficencia, dignidad humana, beneficencia, responsabilidad, seguridad, seguridad de datos, sostenibilidad, libertad, solidaridad, prosperidad, efectividad, precisión, previsibilidad, interpretabilidad.
Independent High-Level Expert Group on Artificial Intelligence (European Commission). (2019). Ethics Guidelines for Trustworthy AI.	Respeto por la autonomía humana, prevención del daño, equidad y explicabilidad. Agencia humana y supervisión, robustez técnica y seguridad, privacidad y gobernanza de datos, transparencia, diversidad y no discriminación, equidad, bienestar social y ambiental, rendición de cuentas.
Jobin, A., Ienca, M., & Vayena, E. (2019). The Global Landscape of AI Ethics Guidelines.	Transparencia, justicia y equidad, no maleficencia, responsabilidad, privacidad, beneficencia, libertad y autonomía, confianza, sustentabilidad, dignidad y solidaridad.
Zeng, Y., Lu, E., & Huangfu, C. (2018). Linking Artificial Intelligence Principles.	Humanidad, colaboración, compartir (equidad), justicia, transparencia, privacidad, seguridad, protección, rendición de cuentas, AGI/ASI.
Floridi, L., Cows, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., Luetge, C., Madelin, R., Pagallo, U., Rossi, F., Schafer, B., Valcke, P., & Vayena, E. (2018). AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations.	Beneficencia, no maleficencia, autonomía, justicia, explicabilidad.
Smit, K., Zoet, M., & Van Merten, J. (2018). A Review of AI Principles in Practice.	Mejora (<i>augmentation</i>) humana, beneficencia, confiable, centrado en el ser humano. Autonomía, igualdad (diseño y ejecución), trazabilidad, dignidad humana, derechos humanos, transparencia, democratización, privacidad, seguridad, seguridad (diseño y ejecución), colaboración, responsabilidad, comprensibilidad, uso responsable de los datos, precisión y educación y promoción.

Figura 3. Revisión de literatura sobre principios de la IA.

una de las 5 líneas de investigación del centro: (RL1) Aprendizaje profundo para visión y lenguaje, (RL2) IA neuro-simbólica, (RL3) IA inspirada en el cerebro, (RL4) Aprendizaje automático basado en la física, y (RL5) IA centrada en las personas. Además, el GTE ha contado con la asesoría de una socióloga (coautora de este artículo) y dos abogadas.

La síntesis comenzó con una revisión de siete documentos recopilatorios y meta-revisiones de principios de la IA al 2022 (ver Figura 3). Una propuesta inicial de principios para CENIA, fue socializada y discutida en un workshop interno que congregó a la mayor parte de quienes integran el Centro. Con esta retroalimentación, el GTE propuso una

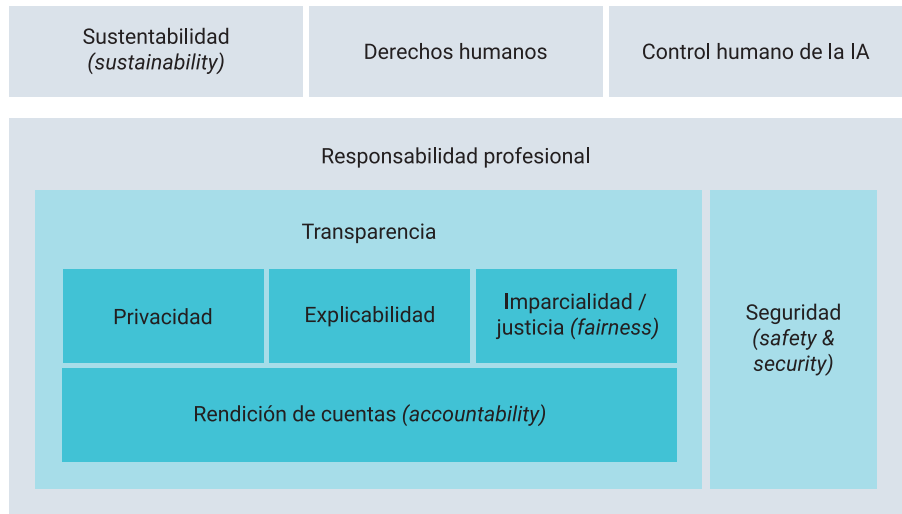


Figura 4. Síntesis de principios de la guía de trabajo para el análisis ético-social de CENIA.



Nos parece necesario reenfocar la percepción de la ética, no como una limitante, sino que como una oportunidad metodológica para mejorar el desarrollo de la IA.

síntesis de principios para CENIA, la cual ilustramos en la Figura 4, y explicamos a continuación.

En esta síntesis hay tres principios generales: sostenibilidad, derechos humanos y control humano de la IA. Estos principios generales intersectan los demás principios, a los cuales englobamos dentro del principio de responsabilidad profesional. Esta decisión busca significar que entendemos que la investigación y el desarrollo de la IA siempre ocurre en el marco de decisiones humanas que deben incorporar estándares técnicos y éticos para garantizar la concreción de los demás principios.

Bajo responsabilidad profesional, situamos a la transparencia como otro principio que ayuda a articular otros principios de la IA. La transparencia incorpora la idea de la explicabilidad, es decir, entender las decisiones puntuales del sistema. Además, es un requisito para garantizar la privacidad de las personas, donde ellas tienen control sobre cómo son usados sus datos, así como para imparcialidad/justicia (*fairness*), que requiere que las personas puedan comprender el funcionamiento e impacto de la IA, y cómo esta puede implicar sesgos y perjuicios. La transparencia, además, habilita la rendición de cuentas (*accountability*), que nos lleva de vuelta a las decisiones humanas involucradas en el proceso de la IA y cómo se deben distribuir las responsabilidades cuando se produce daño debido a la IA, y a la necesidad de desarrollar mecanismos de prevención y reparación asociados a tales daños.

Derechos humanos	Access Now [21] enfatiza la universalidad y el poder vinculante que estos tienen: “En los casos donde no existe legislación nacional, la legitimidad moral de los derechos humanos conlleva un importante poder normativo ” (p.17).
Control humano de IA	En las directrices para una IA confiable de la Unión Europea, se establece que “la supervisión humana ayuda a garantizar que un sistema de IA no socave la autonomía humana o cause otros efectos adversos” [22, p. 16].
Sustentabilidad	Para Floridi et al. [23] se trata de garantizar las condiciones básicas para la vida como una “ prosperidad continuada de la humanidad ” (p. 697) preservando el medio ambiente para generaciones futuras.
Responsabilidad profesional	Según Fjeld et al. [9] esto pasa en parte por “asegurarse de que los involucrados y afectados sean consultados y que los efectos a largo plazo sean parte de la planificación ” (p.5).
Seguridad	Los estándares de seguridad, según Smith et al. [24] son transversales: “La IA debe diseñarse y ejecutarse para proporcionar la máxima seguridad contra amenazas internas y externas , maliciosas o accidentales” (p.8).

Figura 5. Descriptores de principios en la literatura I.

Transparencia	Definida por Leslie [25], hace referencia a la habilidad de “saber cómo y por qué un modelo funciona de determinada manera , en un contexto específico” (p.35), haciendo referencia a visibilizar la lógica detrás de su comportamiento.
Explicabilidad	Este principio apunta a tener criterios para diseñar una IA “de modo que los humanos sean capaces de entender (lenguaje, presentación) la forma de trabajar de la IA” [24, p. 8].
Privacidad	Privacidad implica respetar la intimidad de las personas y sus datos, pero también su “ capacidad de decisión sobre sus datos y las decisiones que se tomen con ellos” [9, p. 4].
Imparcialidad	Se busca prevenir la discriminación de individuos y grupos, priorizando el “ evitar sesgos injustos en los sistemas de IA , que pueden propiciar injusticia social o privar a las personas de su autonomía” [26, p. 11-12].
Rendición de cuentas	Se enfatiza en la importancia de que existan mecanismos disponibles para garantizar que “ la responsabilidad por las repercusiones de los sistemas de IA se distribuya adecuadamente y que se ofrezcan soluciones adecuadas” [9, p. 4].

Figura 6. Descriptores de principios en la literatura II.



Pregunta general: ¿Se han establecido mecanismos de documentación para las decisiones de las distintas partes del proceso?

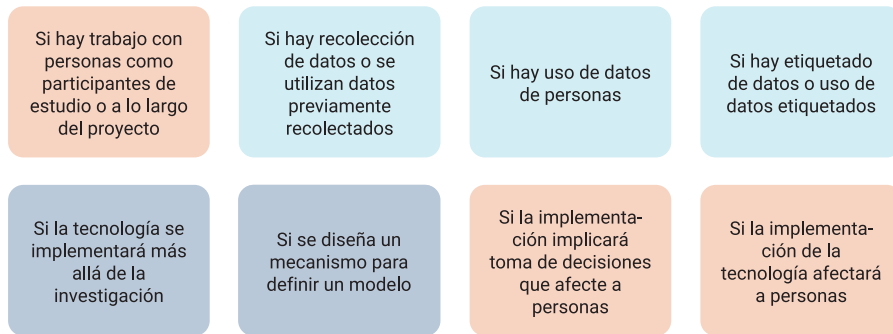


Figura 7. Categorización de preguntas de la guía de trabajo para el análisis ético-social de CENIA.

Por último, la seguridad, si bien no es necesariamente habilitada por la transparencia, sí es parte de la responsabilidad profesional que debe atravesar el diseño de la IA, al sostener los aspectos técnicos y las defensas ante posibles ataques que atenten contra los sistemas de IA o sus principios. En las Figuras 5 y 6 se incluyen descripciones de estos principios según la revisión de literatura que realizamos.

No cabe duda que ante esta lista de principios cabe preguntarse si todo proyecto de IA debe hacerse cargo de absolutamente todos esos principios. La respuesta simple es no, y justamen-

te el desarrollo de nuestros siguientes pasos busca poder generar un instancia metodológica que fomente la priorización e identificación de principios a las contextualidades propias de cada proyecto investigativo. La metodología propuesta consta de una serie de preguntas sobre principios, riesgos y estrategias de mitigación organizados en categorías (ver Figura 7). Por ejemplo, hay preguntas que se deben contestar si hay trabajo con personas, si hay uso de datos, o si existe etiquetado de datos, así como otras preguntas relacionadas al impacto esperado en caso de implementación de la IA en contextos operacionales.

En el segundo semestre del 2023, estamos implementando un piloto con proyectos de CENIA para ir refinando principios y las preguntas que motivan la reflexión ético-social, así como herramientas que permitan articular estrategias de mitigación de los riesgos identificados. Esperamos poder hacer públicos esos resultados a la comunidad de IA y computación en el futuro cercano.

Es clave resaltar que aún quedan muchas preguntas sin abordar y a las cuales deberemos poner atención desde nuestras narrativas latinoamericanas. Por ejemplo, ¿cómo hacemos esto desde América Latina donde tenemos mucha menos evidencia de los impactos de la IA? Esperamos que nuestro trabajo fomente la integración de la ética no como un accesorio o un requisito vacío, sino como un aliado en la innovación responsable de la IA centrada en las personas, apoyando los esfuerzos legislativos y normativos que están surgiendo en nuestra región. ■

Agradecimientos: Queremos agradecer a las y los investigadores que integran el Grupo de Trabajo de Ética de CENIA: Margarita Castro, Cristóbal Moenne, Cristóbal Rojas y Federico Fuentes, así como a las abogadas que nos asesoran voluntariamente, Catherine Muñoz y Evelyn López.



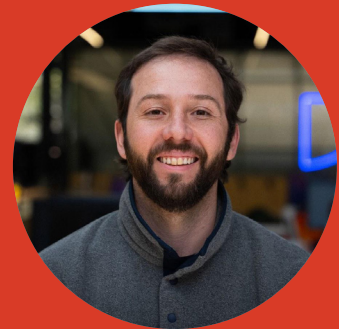
BIBLIOGRAFÍA

- [1] O'Neil, C. (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown Publishing Group.
- [2] Noble, S. U. (2018). *Algorithms of oppression: How search engines reinforce racism*. New York University Press.
- [3] Eubanks, V. (2018). *Automating inequality: How high-tech tools profile, police, and punish the poor*. St Martin's Press.
- [4] Arnold, T., & Scheutz, M. (2018). The "big red button" is too late: an alternative model for the ethical evaluation of AI systems. *Ethics and Information Technology*, 20, 59-69.
- [5] Bernstein, M. S., Levi, M., Magnus, D., Rajala, B., Satz, D., & Waeiss, C. (2021). ESR: Ethics and Society Review of Artificial Intelligence Research (Version 2). arXiv.
- [6] Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). *Machine Bias*. There is software that is used across the county to predict future criminals. And it is biased against blacks.
- [7] Larson, J., Mattu, S., Kirchner, L., & Angwin, J. (2016). How we analyzed the COMPAS recidivism algorithm.
- [8] National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research. (1979). *The Belmont report: Ethical principles and guidelines for the protection of human subjects of research*. U.S. Department of Health and Human Services.
- [9] Fjeld, J., Achten, N., Hilligoss, H., Nagy, A., & Srikumar, M. (2020). *Principled Artificial Intelligence: Mapping Consensus in Ethical and Rights-Based Approaches to Principles for AI*. Berkman Klein Center Research Publication, 1.
- [10] United Nations Educational Scientific and Cultural Organization. (2022). *Recommendation on the Ethics of Artificial Intelligence*. UNESCO. https://unesdoc.unesco.org/ark:/48223/pf0000381137_spa.
- [11] Hogenhout, L. (2021). *A Framework for Ethical AI at the United Nations (Version 1)*. arXiv.
- [12] Google AI. (2023). *Our Principles*. Google. <https://ai.google/principles/>.
- [13] World Economic Forum. (2021). *Responsible Use of Technology: The IBM Case Study (White Paper)*. <https://www.weforum.org/whitepapers/responsible-use-of-technology-the-ibm-case-study>.
- [14] Microsoft. (2022). *Microsoft Responsible AI Standard, General Requirements for External Release*. <https://aka.ms/ResponsibleAI-Questions>.
- [15] Steinhoff, J. (2023). *AI ethics as subordinated innovation network*. In *AI & SOCIETY*. Springer Science and Business Media LLC.
- [16] Ministerio de Ciencia, Tecnología, Conocimiento e Innovación. (2021). *Política Nacional de Inteligencia Artificial*. <https://minciencia.gob.cl/areas/inteligencia-artificial/politica-nacional-de-inteligencia-artificial/>.
- [17] Cámara de Diputadas y Diputados de Chile (2023) *Ley 15869-19: Regula los sistemas de inteligencia artificial, la robótica y las tecnologías conexas, en sus distintos ámbitos de aplicación*. <https://www.camara.cl/legislacion/ProyectosDeLey/tramitacion.aspx?prmID=16416&prmBOLETIN=15869-19>.
- [18] Centro Nacional de Inteligencia Artificial (2023) *Índice Latinoamericano de Inteligencia Artificial*. [/https://indicelatam.cl/wp-content/uploads/2023/08/ILIA-2023.pdf](https://indicelatam.cl/wp-content/uploads/2023/08/ILIA-2023.pdf).
- [19] Gómez Mont, C., Del Pozo, C. M., Martínez Pinto, C., & Martín del Campo Alcocer, A. V. (2020). *La inteligencia artificial al servicio del bien social en América Latina y el Caribe: Panorámica regional e instantáneas de doce países*. <https://doi.org/10.18235/0002393>.
- [20] Salas-Pilco, S. Z., & Yang, Y. (2022). *Artificial intelligence applications in Latin American higher education: a systematic review*. In *International Journal of Educational Technology in Higher Education* (Vol. 19, Issue 1). Springer Science and Business Media LLC.
- [21] AccessNow (2018) *Human Rights in the Age of Artificial Intelligence*. <https://www.accessnow.org/wp-content/uploads/2018/11/AI-and-Human-Rights.pdf>.
- [22] Independent High Level Expert Group on Artificial Intelligence. (2019). *Ethic Guidelines for Trustworthy AI*. <https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>.
- [23] Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., Luetge, C., Madelin, R., Pagallo, U., Rossi, F., Schafer, B., Valcke, P., & Vayena, E. (2018). *AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations*. In *Minds and Machines* (Vol. 28, Issue 4, pp. 689–707). Springer Science and Business Media LLC.
- [24] Smit, K., Zoet, M., & Meerten, J.V. (2020). *A Review of AI Principles in Practice*. Pacific Asia Conference on Information Systems.
- [25] Leslie, D. (2019). *Understanding artificial intelligence ethics and safety*. arXiv.
- [26] Khan, A. A., Badshah, S., Liang, P., Waseem, M., Khan, B., Ahmad, A., Fahmideh, M., Niazi, M., & Akbar, M. A. (2022). *Ethics of AI: A Systematic Literature Review of Principles and Challenges*. *ACM International Conference Proceeding Series*, 383–392.



GPT y agentes:


Conversando con tus apps



JOSÉ MANUEL PEÑA

Ingeniero industrial por la Universidad de Chile y Magíster en Applied Analytics por la Universidad de Columbia, Nueva York. Actualmente, es el manager del equipo de inversiones en Fintual, una fintech chilena con más de 700 millones de dólares en activos bajo gestión en Chile y México, donde también apoya al desarrollo de iniciativas de IA. Adicionalmente, es miembro del Comité de Inteligencia Artificial y Big Data del Instituto de Ingenieros de Chile.

✉ manu@fintual.com

 www.linkedin.com/in/jose-manuel-pena/



RESUMEN. Durante el último año, los modelos de lenguaje de gran tamaño como GPT han revolucionado la interacción humano-máquina, presentando capacidades avanzadas de procesamiento y generación de texto. Aunque estos modelos son poderosos, también tienen limitaciones como la propensión a “alucinaciones”, la incapacidad de integración nativa a sistemas tradicionales y la carencia de manejar contenido propietario.

Ante estas limitaciones, nuevas soluciones basadas en el concepto de “agentes” intentan superarlas al integrar modelos de lenguaje con otros sistemas. Estos sistemas basados en agentes utilizan herramientas externas, como bases de datos o APIs, para enriquecer y verificar las respuestas del modelo.

Un ejemplo aplicado es Fintual Copiloto, un servicio de asesoría de inversiones que utiliza estos avances para interactuar con los usuarios y proporcionar información financiera.

GPTs, una revolución incompleta

A lo largo de este año, los colosos del procesamiento del lenguaje, comúnmente conocidos como LLMs (Large Language Models) y más específicamente GPTs (Generative Pretrained Transformers) han irrumpido dramáticamente en la sociedad. Su aparición ha desencadenado una auténtica “explosión cámbrica” de herramientas y aplicaciones, todas aprovechando su notable habilidad para procesar, comprender y generar texto lógico y realista.

Si bien algunos podrían argumentar que estos modelos de última generación no son estructuralmente muy distintos de

El uso de menús y botones para navegar en servicios digitales pronto podría convertirse en cosa del pasado.

los primeros modelos basados en mecanismos de atención, surgidos entre 2016 y 2017, su incremento en potencia los ha catapultado a un nuevo nivel.

Aunado a esto, su elegante y simple interfaz —que se basa en mapear texto a texto en una modalidad similar a una conversación— ha sido un factor determinante. Estos dos elementos combinados han hecho que estos modelos entren al epicentro del debate global, tomando el podio como los emblemas de la inteligencia artificial contemporánea, por sobre el reconocimiento de imágenes y otras técnicas de *machine learning*.

Este enorme avance ha dado lugar a una nueva forma de interacción humano-máquina principalmente conversacional. El uso de menús y botones para navegar en servicios digitales pronto podría convertirse en cosa del pasado.

Sin embargo, a pesar de este creciente optimismo, los primeros intentos de implementar soluciones de preguntas y respuestas que dependen exclusivamente de los GPTs como una especie de “oráculo”, han demostrado ser totalmente insuficientes para cualquier cosa más allá de la novedad inicial.

El dilema de las alucinaciones

La primera gran limitante del uso indiscriminado de GPTs en espacios productivos es el conocido problema de las “alucinaciones”. En simples palabras, estos modelos pueden generar información que no está basada en datos reales. Esto ocurre porque estos modelos tienen como objetivo la imitación de texto realista replicando patrones de lenguaje, dejando la veracidad misma del men-

saje en un segundo plano, generando respuestas plausibles sin una fuente de verdad objetiva.

Este problema hace que no podamos (y probablemente nunca) estimar a priori si la información vertida por un LLM es factualmente correcta, haciendo que todo dato generado internamente por un LLM sea considerado sospechoso o potencialmente falso. Esta limitación es un talón de Aquiles sobre todo en los casos de uso donde la veracidad de la información es crítica (prácticamente en todos los casos, excepto para usos más creativos o artísticos).

Si bien existe la promesa de poder “dirigir” efectivamente las respuestas de un GPT a un contexto particular mediante el uso inteligente de *prompts* (el texto que se le entrega al modelo), este mecanismo sigue siendo estocástico por naturaleza y es difícil prever todos los posibles caminos que el modelo puede seguir: aunque la respuesta correcta puede existir dentro del corpus de datos sobre el cual el modelo se entrenó, no hay garantías de que el *prompt* entregado devuelva como resultado un texto basado en ese específico contenido. En otras palabras, estamos entregados a un universo de azar.

¿Y si lo entrenamos más? La verdad es que la naturaleza misma del problema hace ineficiente pensar que sólo con más entrenamiento (general o *fine-tuning*) garantice un funcionamiento 100% correcto. Más aún, el estimar cuánto entrenamiento extra (ya sea de todo el modelo o *fine-tuning*) necesitas para eliminar una “alucinación” es extremadamente difícil. Por lo que sólo nos queda tener una necesaria desconfianza de toda la información factual vertida por un GPT, siempre.



Un cerebro sin manos

En segundo lugar, parte del milagro de esta generación de modelos es también una gran limitante (a primera vista). Los GPTs, al enfocarse en mapear texto no estructurado a texto no estructurado, hacen que su integración en sistemas tradicionales sea problemático.

En la actualidad en el mundo digital, desde tu refrigerador hasta tus redes sociales, tú puedes interactuar fácilmente mediante algún tipo de lenguaje estructurado (APIs, interfaces, bases de datos, etc.). Esta limitación inicial hace que los GPTs, sin una mayor integración, sólo puedan crear conversaciones carentes de acciones, lo cual posiciona a estos modelos en el asiento del copiloto o de un consultor, incapaz de gatillar acciones útiles y sólo siendo capaz de sugerir y responder.

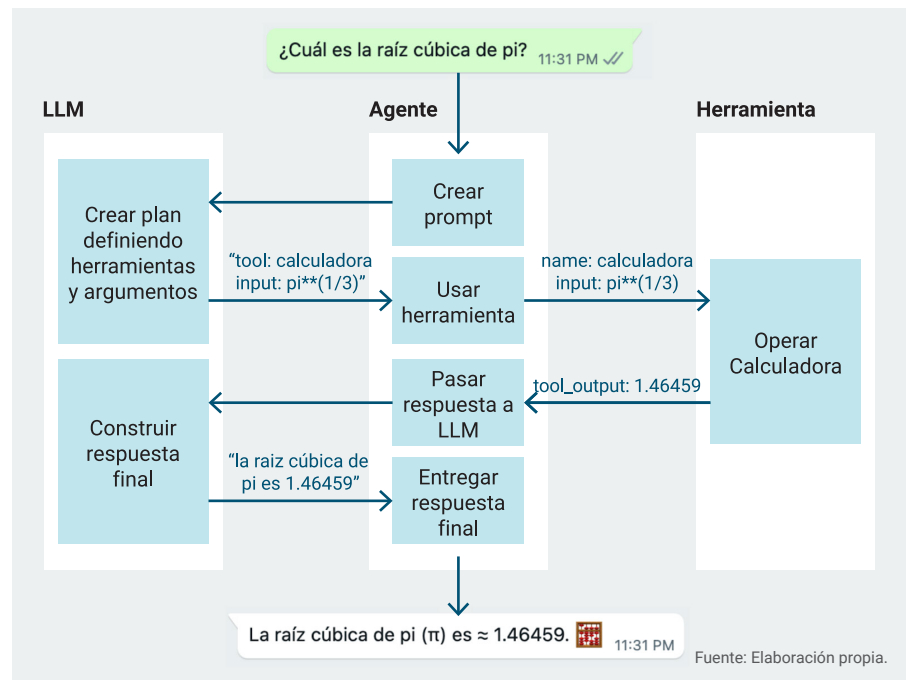


Figura 1. Diagrama estilizado de operación de un agente.

El contenido como diferenciador

Adicionalmente, desde un punto de vista productivo y comercial, las soluciones y aplicaciones basadas exclusivamente en GPTs de manera aislada no pueden sino ofrecer experiencias genéricas, carentes de detalle y expertise propias del contexto.

Alguien podría argumentar que la generación actual de LLMs ha sido entrenada con una cantidad tan vasta de información, que estos modelos pueden comportarse como "expertos" en casi cualquier área (desde pasar la prueba de admisión de abogacía en Estados Unidos, hasta completar con nota sobresaliente difíciles pruebas de conocimiento en medicina). Pero este conocimiento no abarca el universo de información privada que diferencia a una empresa: su *know-how*, datos propietarios, perfil de sus clientes/productos, etc.

En conclusión, para desarrollar sistemas basados en GPTs o LLMs que entreguen información veraz, puedan

proveer acciones más allá de simple texto, y puedan operar sobre contextos específicos de manera exitosa y robusta, debemos ser capaces de unir estos modelos con múltiples sistemas, tanto de obtención de información y datos, como de ejecución.

Agentes y herramientas: separando el conocimiento de la razón

Ante este desafío, han aparecido múltiples proyectos, como Auto-GPT o LangChain entre otros, que buscan potenciar las capacidades de GPTs al integrarlas a múltiples sistemas, convirtiéndolas en ecosistemas capaces de planear y gatillar el uso de herramientas para cumplir los requerimientos de sus usuarios.

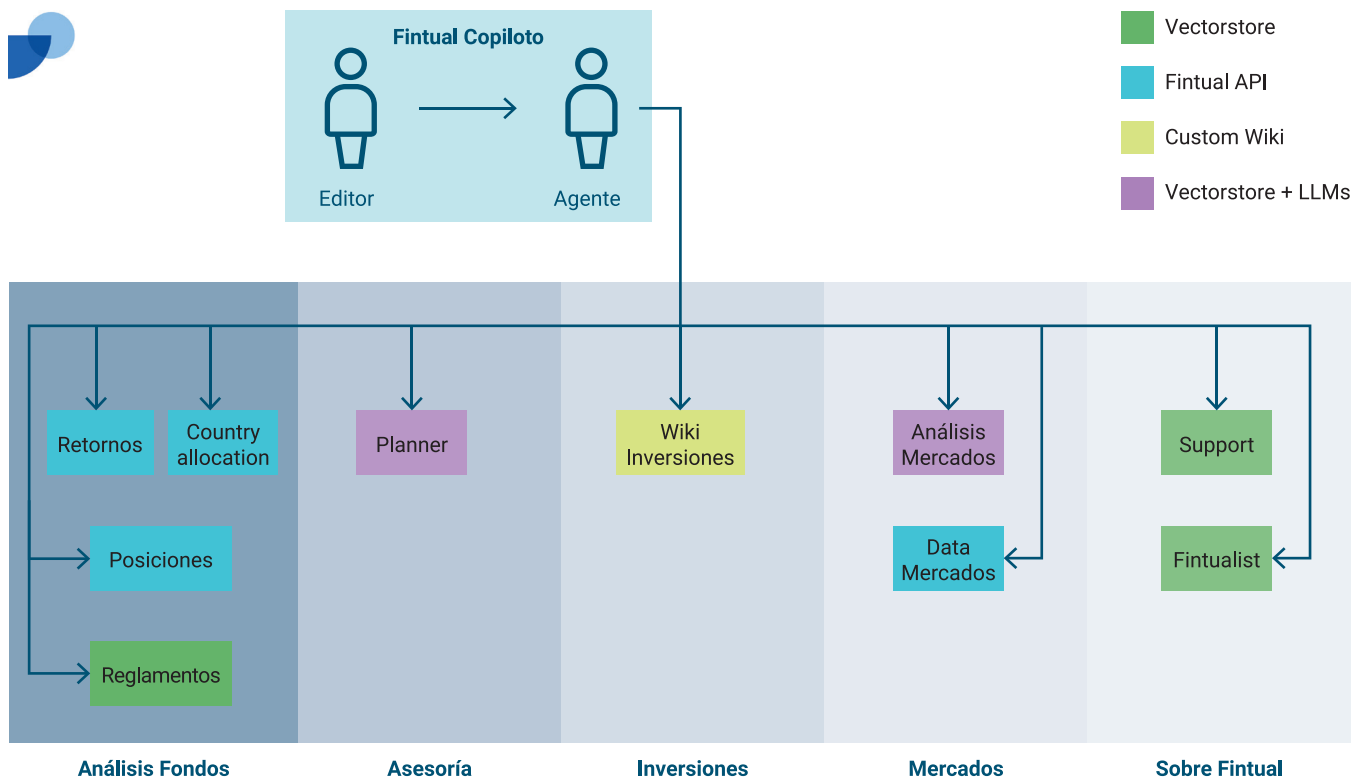
Estas librerías no reemplazan los LLMs, sino que en base a llamar a uno

o múltiples LLMs, se genera un esquema de trabajo que replica un agente "inteligente" capaz de comunicarse con datos y generar acciones mediante conectores usualmente llamados *plugins* o *herramientas*.

Para entender bien cómo funcionan estos sistemas basados en agentes, primero vamos a definir algunos de los conceptos básicos presentes en la mayoría de estas librerías. En general, un sistema basado en agentes se compone de al menos de 2 elementos: el agente y sus herramientas (ver Figura 1).

Herramientas

Las herramientas son conexiones a sistemas externos al LLM, como lo es una base de datos, una API o cualquier otro sistema que se pueda encapsular en una función que pueda entregar su resultado en forma de texto plano legible para el LLM. Este concepto es muy potente, ya



Fuente: Elaboración propia.

Figura 2. Fintual Copiloto en simple: posee 2 agentes, uno enfocado en edición y estilo, y otro enfocado en contenido. El agente enfocado en contenido posee acceso a una serie de herramientas enfocadas en análisis, asesoría y educación apalancando las APIs internas y públicas de Fintual, acceso a servicios externos y bases de datos documentales para búsqueda semántica.

que cualquier acción, mientras pueda ser expresada como una función, puede ser escrita como una herramienta que recibe y entrega texto. El listado de herramientas posibles crece cada día, desde conexiones con tus herramientas de gestión para manipular correos o presentaciones, hasta la conexión a otros tipos de modelos de inteligencia artificial (como es el caso de HuggingFace Agents) donde puedes invocar el análisis o creación de imágenes, audio, etc.

Agentes

Por otra parte, un agente no es más que un uso inteligente de *prompts* donde, en vez de pedir a un LLM responder directamente una pregunta, se le pide que

escriba una estrategia para resolverla en base a una lista de recursos (*tools* o *plugins*), definiendo cuáles usaría y qué argumentos le entregaría. Estas instrucciones luego son ejecutadas y sus resultados se le devuelven al LLM para que genere una respuesta final.

Para ilustrar esto, pensemos en una solicitud simple donde necesito conocer la multiplicación de 2 números. Uno podría directamente pasar a un LLM la consulta "Cuál es la raíz cúbica de pi?", lo cual va a depender de la capacidad aritmética implícita del modelo y que puede ser inexacta.

Por el contrario, un agente con acceso a una calculadora daría un contexto donde se pasaría al LLM un *prompt* como éste:

Eres un asistente y tienes acceso a una herramienta "Calculadora" la cual se invoca con el siguiente formato:

- tool: 'Calculadora'
- input: sintaxis de la operación

Usa esta herramienta para responder la siguiente pregunta:

- ¿Cuál es la raíz cúbica de pi?

Esta manera de formular el mensaje fuerza al LLM a no buscar la respuesta directamente, sino a generar un comando explícito con la herramienta a usar y el argumento necesario para generar la respuesta.



Figura 3. Ejemplos de algunas conversaciones en Fintual Copiloto (sic).

Este es un ejemplo excesivamente simple, pero refleja la potencia de este esquema, donde el LLM podría orquestar y combinar el uso de una multitud de herramientas, como bases de datos, documentos, APIs o herramientas de cálculo para generar respuestas factualmente correctas o acciones a pedido del usuario.

Agentes en la asesoría de inversiones: Fintual Copiloto

¿Qué es Fintual?

Fintual es una plataforma digital de ahorro e inversión con presencia en Chile y México que, a través de la tecnología y automatización de procesos, busca simplificar y dar acceso al sector financie-

Esta manera de formular el mensaje fuerza al LLM a no buscar la respuesta directamente, sino a generar un comando explícito con la herramienta a usar.

ro reduciendo barreras como las altas comisiones y montos mínimos. Fintual nace el 2016 de la mano de sus 4 fundadores y hoy cuenta con más de 100 mil usuarios activos y administra más de 700 millones de dólares.

El desafío de la asesoría financiera y el nacimiento de Copiloto

En la misión de Fintual de acercar el mundo de las inversiones y el ahorro a las personas, resulta crítico ser capaz de entregar ayuda oportuna y simple para que sus usuarios entiendan y confíen en los servicios que se entregan. En esta línea, tradicionalmente el mun-

do de las inversiones se ha basado en tener ejércitos de vendedores (más conocidos como asesores, captadores, etc.) los cuales, haciendo un trabajo 50% asesoría 50% venta, buscan responder dudas, planificar una estrategia de ahorro y recomendar productos para invertir. Esta dependencia en la asesoría 1:1 incrementa los costos de administración enormemente, reduciendo el retorno final que perciben las personas y alejando la inclusión financiera de grupos de bajo patrimonio.

Por esta razón, Fintual desde sus inicios nunca ha contado con fuerza de ventas. Más bien tiene un buen sistema de soporte (donde todos los empleados de



Es ante este desafío donde en Fintual nace la idea de apalancar todo su contenido financiero y herramientas internas de análisis e información mediante el uso de LLMs y agentes para desarrollar un “Copiloto de Inversiones”.



Fintual dedicamos tiempo a resolver dudas) y constantemente genera contenido financiero simple para el público general (a través de la revista Fintualist). Desde estos dos flancos ha logrado enseñar e introducir al buen hábito de invertir a miles de personas, donde en más del 75% de los usuarios Fintual es su primera inversión.

Pese a los esfuerzos anteriores, existe un factor de individualización y cercanía que un post de inversiones nunca va a tener, ni una disponibilidad 24/7 inclusive si toda la empresa se dedicara al soporte. Es ante este desafío donde en Fintual nace la idea de apalancar todo su contenido financiero y herramientas internas de análisis e información mediante el uso de LLMs y agentes para desarrollar un “Copiloto de Inversiones”, aka Fintual Copiloto.

Copiloto: un asesor 24/7

Fintual Copiloto (ver Figuras 2 y 3) se trata de un servicio de asesoría en el que con sólo enviar un mensaje al WhatsApp de Fintual podrás conversar con nuestro agente de inversiones (por ahora sólo disponible en México). Sus objetivos principales son ayudar a organizar las finanzas personales de los clientes, informar sobre los productos de Fintual y resolver dudas sobre el mercado, economía y las inversiones en general.

Visión a futuro: de vuelta al quiosco

En vista de esta evolución de modelos, desde sus orígenes intentando resol-

ver simples tareas de procesamiento de texto, hasta convertirse en los potentes LLMs actuales capaces de obedecer órdenes y operar como agentes conversacionales, ¿cómo se ve el futuro relativo a nuestra relación con servicios digitales?

Una analogía útil para ver lo que se viene es la siguiente. El paradigma actual de diseño y desarrollo, y aplicaciones, se parece mucho a una tienda por departamentos o supermercado; los productos (información o acciones) que necesitan los usuarios son ubicados dentro de una aplicación la cual, mediante la navegación vía menús y botones, buscan encontrar lo que necesitan, similar a como tú navegas con tu carrito de compras entre pasillos, leyendo la señalética para encontrar lo que necesitas.

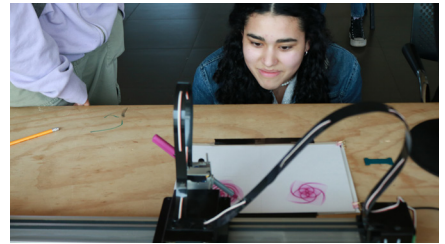
En el futuro, la capacidad de conversar hace posible el que estos mismos activos, al ser conectados a agentes, puedan ser encontrados y accionados simplemente pidiéndolos. Es más, el usuario ya no requiere saber a priori cuál es el activo específicamente que necesita, sino que en base a la conversación con su asesor, llega a definir qué es lo que más le conviene. Este comportamiento lo conocemos desde hace mucho tiempo, ya que es como opera cualquier tienda de la esquina o quiosco, donde quien conoce a cabalidad los productos es el administrador y es él quien recomienda y pone a disposición lo que necesita el usuario.

En conclusión, los LLMs son herramientas revolucionarias pero lejos de ser perfectas. Sin embargo, su integración en sistemas está permitiendo una nueva forma de interactuar de forma natural y desestructurada con datos y aplicaciones. Y si bien todavía estamos en sus primeras iteraciones, con un poco de optimismo es posible que pasemos de un presente de menús y botones, a un futuro como el de las películas, donde a la tecnología se le habla. ■



Niñas Pro descifró el código:

La programación no tiene género



Niñas
PRO

Niñas Pro comenzó el 2016 como una actividad universitaria realizada por un grupo de estudiantes mujeres de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile, para aumentar el interés de niñas por la programación. Hoy está constituida como una corporación sin fines de lucro y es una red de más de 80 voluntarias distribuidas en varias regiones del país con el apoyo de diferentes universidades y empresas interesadas en disminuir la brecha de género en STEM.

 <https://ninaspro.cl/>



RESUMEN. A lo largo de la historia los estereotipos negativos, sociales, familiares e incluso al interior de los colegios, sobre las habilidades de las mujeres en la ciencias, matemáticas y tecnología han generado una baja presencia femenina en las denominadas carreras STEM, tanto es así que para el 2020 sólo el 20% de las matrículas de estas carreras fueron ocupadas por mujeres.

Para combatir esta injusticia social, que deben enfrentar millones de mujeres en el mundo, la corporación Niñas Pro decidió impartir una serie de talleres gratuitos para que niñas y adolescentes de enseñanza media se sumerjan en el mundo de la programación. En siete años, más de 2.000 estudiantes han pasado por sus aulas con la intención de aprender y de generar una comunidad que permita avanzar en paridad de oportunidades en profesiones que históricamente han sido dominadas por hombres.



Alumnas del Curso Anual de Programación impartido por Niñas Pro a estudiantes de enseñanza media.

Un informe publicado por el Ministerio de la Mujer y Equidad de Género confirmó que en la actualidad el 53% de la matrícula total para ingresar a la educación superior, corresponde a mujeres. Se trata de una cifra histórica, que se ha incrementado con los años, y que refleja la importancia de las estudiantes en el sistema educacional chileno. Sin embargo, esas cifras no se han visto reflejadas en las denominadas carreras STEM (Ciencia, Tecnología, Ingeniería y Matemáticas, por su sigla en inglés) donde se estima que sólo una de cada cuatro mujeres opta por estas áreas [1].

Lamentablemente, el bajo porcentaje de mujeres matriculadas en este tipo de carreras no es un fenómeno exclusivamente chileno. Dentro de los países pertenecientes a la Organización para la Cooperación y el Desarrollo Económicos

(OCDE) la cifra de estudiantes egresadas de pregrado de programas STEM bordea el 13%, siendo Alemania el que tiene el porcentaje más alto con 19%. Chile cierra el listado con un 7% [2].

Al comparar las matrículas de carreras STEM con otras áreas del conocimiento, notamos que la mayor presencia de estudiantes mujeres se registra en Salud (77%) y Educación (76%). Por el contrario, los menores ingresos se mantienen en las carreras ligadas a las Fuerzas Armadas y de Orden, con sólo el 25% de su matrícula total ocupadas por mujeres, y las ya mencionadas carreras ligadas a la ingeniería, computación, electrónica y construcción, que sólo mantienen el 20% de sus estudiantes mujeres [3].

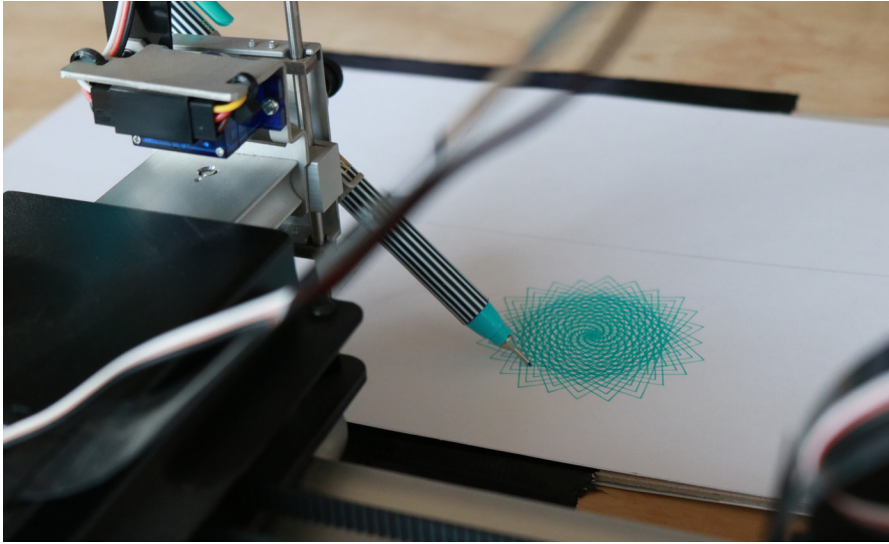
Esta preocupante brecha de género tiende a aumentar en la medida que avanza el nivel educativo de las estudiantes. Así lo reveló el informe de 2021 de la Agencia Nacional de Investigación y Desarrollo que estableció que las mujeres ocuparon, para ese año, el

51% de las matrículas de magíster y el 43% en doctorados, cifras que otra vez van a la baja en el caso de carreras de STEM, ya que en ese ámbito la matrícula alcanzó el 29% en magíster y 37% en doctorados [4].

Sin duda estamos frente a un escenario desigual, que por distintas razones mantiene a las mujeres alejadas de las carreras de ciencias y tecnologías, impidiendo al mundo la posibilidad de generar innovación y nuevas perspectivas para abordar desafíos del futuro. No se trata de que las mujeres “sean malas para las matemáticas”, “no entiendan de programación” o “tengan otros intereses en la vida, distintos a las ciencias”, como muchas veces hemos escuchado en distintos grupos, si no que por años han sido postergadas, discriminadas y, por qué no decirlo, miradas en menos en un área que históricamente ha sido dominada por hombres. No es menor que sólo el 30% de las mujeres en el mundo sean investigadoras en ciencia y que el Premio Nobel sólo haya sido otorgado a 59 mujeres en toda su



Sin dudas estamos frente a un escenario desigual, que por distintas razones mantiene a las mujeres alejadas de las carreras de ciencias y tecnologías.



Taller de Arte Geométrico impartido por Niñas Pro en conjunto con Fintual. El taller fue impartido por Sofía Suazo, creativa de experiencias educativas de STEM en Niñas Pro y Artista Medial.

historia. Eso debe cambiar y estamos trabajando en ese proceso [5].

Con la intención de que todas las niñas de Chile vivan la ciencia y la tecnología sin estereotipos de género, en 2016 nació Niñas Pro, organización sin fines de lucro formada por estudiantes de Ingeniería de la Universidad de Chile, que tiene como objetivo empoderar a niñas y adolescentes a través de la enseñanza de la programación e incentivar en ellas vocaciones científicas y tecnológicas.

Fue una edición de “Grace Hopper Celebration”, la reunión de tecnólogas más importante del mundo, en la que mujeres tienen la oportunidad de aprender, relacionarse e inspirarse en la científica informática Grace Hopper, la que sirvió como inspiración para crear en Chile un nuevo proyecto destinado sólo

para estudiantes mujeres. Fue en esa instancia donde Jazmine Maldonado y Vanessa P. Araya, ambas en esos años estudiantes de postgrado en el DCC de la Universidad de Chile, decidieron darle vida a una idea que hace rato les rondaba la cabeza: enseñar programación a niñas de distintos puntos de la Región Metropolitana.

“En ese momento había un curso en la Universidad que entrenaba a niños para asistir a las Olimpiadas de Informática en Chile, un curso para escolares, pero participaban puros hombres. Sólo había una niña. Fue ahí cuando dijimos ‘hagamos uno para niñas’, porque nuestra teoría no era que no estuvieran interesadas o que no les gustara, sino que el espacio era un poco intimidante. Teníamos el ejemplo de una niña que había ido a unas clases y después se fue. Creemos

que eso pasaba porque habían muchos niños y por lo tanto no era tan atractivo para ellas participar en una actividad extra programática así, por lo que decidimos hacer uno sólo para niñas y ver cómo nos iba”, recuerda Jazmine.

Como todo proyecto que termina siendo próspero, el inicio no fue fácil. Había muchas “dificultades domésticas” que solucionar. Se necesitaba un espacio físico para impartir las clases y ninguna de las voluntarias era profesora, por lo que no había un método pedagógico para enseñar, menos a niñas de enseñanza media. Sin embargo, desde la misma Universidad de Chile vino la ayuda y se facilitó un laboratorio equipado con lo necesario para impartir una clase y un pequeño programa de contenido sería la guía para comenzar el proyecto. Pero faltaba lo más importante y quizás el punto más complejo: ¿cómo reclutar a las niñas?, ¿se interesarían en este taller a pesar de lo que dicen las estadísticas?

Las redes sociales fueron fundamentales para el proceso de “reclutamiento”. Una ficha publicada en Facebook y el contacto cara a cara con varios centros de alumnas permitió correr la voz de lo que sería “Niñas Pro” y el resultado fue la inscripción de 30 estudiantes de varios colegios de Santiago, entre los que destacaron, Carmela Carvajal, el Liceo 1 y algunos colegios técnicos que decidieron tomar las clases extra programáticas.

Derribando prejuicios del aula a la casa

El informe “Mujeres en carreras académicas en las áreas de Ciencias, Tecnologías, Ingenierías y Matemáticas (STEM)”, entregado a la Cámara de Diputados y Diputadas, en medio de la discusión del proyecto de ley que establece la paridad de género en la adjudicación de proyectos de investigación y becas, estableció que los estereotipos negati-



vos sobre las habilidades de las niñas en las matemáticas puede disminuir considerablemente el desempeño de las estudiantes en la pruebas de acceso a las universidades chilenas. Por lo que padres, madres y la docencia son esenciales y pueden ayudar a mejorar el rendimiento de las niñas a la hora de postular a una carrera [6].

“Cuando los administradores de pruebas le dicen a los estudiantes que las niñas y los niños son igualmente capaces en matemáticas, la diferencia en el rendimiento desaparece, lo que ilustra que los cambios en el entorno del aprendizaje pueden mejorar el rendimiento de las niñas en matemáticas”, asegura el informe, que además entrega una serie de prejuicios a los que se deben enfrentar las niñas que deciden elegir las carreras STEM para su vida profesional.

Entre los más descarnados destacan que: “Los padres y maestros subestiman las habilidades de las niñas en matemáticas”; “los profesores son más duros con ellas y las califican más bajo por el mismo trabajo y asumen que necesitan trabajar más para alcanzar el nivel de los niños” [6].

Este tipo de situaciones, sin duda, pone a las mujeres en inferioridad y discriminación frente a los hombres en la etapa escolar, lo que lamentablemente tiene consecuencias negativas, como que “las niñas son más duras consigo mismas y se autoevalúan más bajo que los niños con logros similares” y lo que es peor, “las niñas y mujeres jóvenes no ven ejemplos de mujeres científicas e ingenieras en los libros, los medios y la cultura popular”, asegura el documento [6].

Con estos mismos desafíos se encontró Niñas Pro en su desarrollo durante todo este tiempo. Más aún cuando para el segundo año se pasó de 30 estudiantes a 200. “Las niñas nos contaban que



De izquierda a derecha: Florencia Miranda, Jazmine Maldonado, Jocelyn Simmonds, Carolina Silva y Valentina Urzúa. Florencia y Jazmine fueron parte del directorio anterior de Niñas Pro, mientras que Jocelyn y Carolina son parte del directorio actual. Valentina es tutora de la sede Santiago.¹

en sus colegios hacían talleres de robótica, pero había que postular y a veces no las dejaban entrar. Entonces sentían que había ciertas barreras para estar en esos espacios. O muchos colegios sencillamente daban el curso de computación donde enseñaban a hacer un Power Point y no había mucho desarrollo de cosas novedosas, por lo que faltaba un espacio y cuando publicamos el taller funcionó bien”, comenta Jazmine Maldonado, quien asegura que Niñas Pro no tiene como objetivo central incentivar el estudio futuro de una carrera de programación, sino acompañar un proceso de descubrimiento y de experimentación de cara a la vida universitaria.

El crecimiento en el número de participantes llevó a la organización a expandir

la invitación a la Universidad Católica y a la Universidad de Los Andes, reclutar a más voluntarias y expandir los talleres. La “profesionalización” del proyecto era el próximo paso, y ya para el 2019 se creó de forma legal la corporación “Niñas Pro” (sigla que hace referencia a la palabra “Programadoras”) y se abrió una sede en Las Condes. Para el 2020 se abrieron sedes en Coquimbo y Concepción. La pandemia por COVID-19 no fue impedimento para continuar con el proyecto y las clases se hicieron tal como funcionaba todo el mundo, de manera virtual. Para 2023, el proyecto contaba con sedes en Santiago, Coquimbo y Concepción.

“Muchas niñas vuelven a tomar el curso porque además de aprender a programar encuentran a un grupo de amigas.

1. Ángeles Aldunate y Loraine Schlotterbeck también forman parte del directorio actual



Cuando los administradores de pruebas le dicen a los estudiantes que las niñas y los niños son igualmente capaces en matemáticas, la diferencia en el rendimiento desaparece.

Crean una identidad en torno a eso, en niñas que se interesan en las tecnologías y se arma un grupo de niñas de distintos colegios, pero que tienen ese interés en común”, comenta Jazmine.

Compartir el conocimiento

Otro de los aspectos positivos de la evolución de Niñas Pro fue la incorporación de decenas de voluntarias y voluntarios que decidieron poner su conocimiento en distintas áreas a disposición del proyecto y así ayudar al aprendizaje de las estudiantes. Este año (2023), contamos con 78 voluntarios y voluntarias, el 74% son mujeres, lo que confirma el compromiso con la participación femenina tanto en la enseñanza como en el aprendizaje de programación.

“Comencé en este voluntariado por mi propia experiencia de vida. Cuando estaba en enseñanza media no sabía qué estudiar. Si bien mi colegio tenía charlas vocacionales, no sabía qué hacer. Nunca me di cuenta de que mi área fuerte eran las matemáticas. A las niñas nunca se les incentiva a ir por áreas más duras, yo al menos nunca me sentí guiada hacia ese lado. Aunque hoy cada vez hay más mujeres en el área de la computación, no somos muchas. Por eso me motiva hacer un cambio en el mundo, mostrarle a las niñas y mujeres que lo podemos hacer todo”, comenta Victoria Guerriero, de 22 años, estudiante de ingeniería civil en computación y voluntaria de Niñas Pro, quien destaca el cambio de foco que impone la corporación en temas sociales, ya sea en educación como en otras problemáticas nacionales.

“Encuentro que somos una juventud activa y en acción. Muchos de los cambios que se están haciendo han sido transformados gracias a la juventud. Eso es porque mayoritariamente no estamos de acuerdo en cómo es el sistema hoy en día. Tenemos muchos temas distintos que se están abordando para cambiar la sociedad y hacerla mejor, como el problema medioambiental y la desigualdad. Son asuntos que están muy presentes con la juventud y nuestro desafío más grande es poder hacer un cambio para crear un mundo mejor”, sostiene Victoria.

Constancia, aprendizaje y ganas de pasarlo bien

Durante el desarrollo histórico de Niñas Pro se han generado programas de aprendizaje, enfocados en estudiantes que no tienen nociones básicas de programación, pero tienen la motivación de aprender en cursos que se pueden extender por varios meses.

Entre ellos destaca el curso anual de pensamiento computacional. En este taller se desarrolla el pensamiento computacional mediante la realización de actividades didácticas y resolución de problemas usando código. Actualmente el curso se dicta en dos versiones, una programando en C++ y otra en Python.

Además, en este curso, las alumnas aprenden sobre ciencia y tecnología a través de distintas charlas en las que se relacionan con mujeres que trabajan en el mundo STEM, con ellas pueden descubrir nuevas áreas de investigación y conocimiento en las que se pueden desenvolver en la industria.

Las alumnas que ingresan a este programa no necesitan tener conocimientos previos de programación, únicamente compromiso y ganas de aprender, ya que deben ser constantes en asistir a clases entre abril y octubre.

Al iniciar el curso se realiza una evaluación diagnóstica para que, en los casos de quienes dominen competencias básicas de programación, puedan profundizar su aprendizaje en un grupo avanzado mediante actividades de desarrollo de algoritmos y manejo de otras estructuras de datos.

Con este curso las alumnas pueden introducirse al mundo tecnológico, conociendo las posibles aplicaciones de la herramienta técnica de la programación, además de unirse a una comunidad de mujeres involucradas e interesadas en el mundo STEM.

Otro de los talleres emblemáticos de Niñas Pro es el curso de Arduino Quest, que se realiza en conjunto con el Instituto Milenio Fundamentos de los Datos y tiene como objetivo promover la creatividad, capacidad de autogestión del tiempo y colaboración entre las niñas participantes, de forma que puedan experimentar, expresar sus ideas y aprender haciendo por medio de la resolución de los desafíos y proyectos que comprende Arduino Quest.

La postulación es totalmente gratuita y se realiza en conjunto con el Instituto Milenio Fundamento de los Datos. El desafío consta de 3 etapas, las cuales no tienen ganadores o perdedores, sólo personas que pasan al siguiente desafío según cumplimiento de cada etapa anterior.

La primera etapa consta de una semana de clases en formato online donde se les enseña el manejo de la placa Arduino y de los distintos componentes principales para crear un circuito electrónico funcional. Luego, las alumnas trabajan con un simulador electrónico de Arduino llamado Tinkercad y realizan un



El desafío era enorme: había que aumentar la oferta de cursos y talleres, promover la participación de niñas de Santiago y regiones y sobre todo encantar a más empresas privadas para que se sumaran al proyecto.

proyecto en este simulador recibiendo apoyo de mentoras(es) vía online. Finalmente, el curso termina con la realización de un proyecto con un kit Arduino real.

El futuro son las STEM

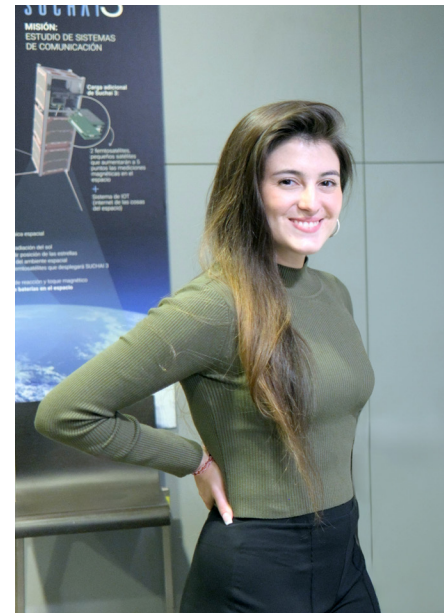
A principios de 2023, Niñas Pro dio un paso enorme hacia la profesionalización de su trabajo y decide nombrar a Camila Pinto, ingeniera civil industrial, candidata a magíster en Data Science, como la nueva directora ejecutiva de la entidad. El desafío era enorme: había que aumentar la oferta de cursos y talleres, promover la participación de niñas de Santiago y regiones y, sobre todo, encantar a más empresas privadas para que se sumaran al proyecto.

“Nos encontramos cada día con más personas que se enamoran de la iniciativa y quieren sumarse apoyando de una u otra forma, nuestro desafío actual es encontrar a los colaboradores clave que al sumarse con auspicios permitan la eje-

cución de varios proyectos que tenemos en tabla”, dice Camila, quien asegura que Niñas Pro no tiene como finalidad que las estudiantes opten al final del taller por una carrera universitaria ligada a la programación, pero sí han tenido casos positivos de niñas que tras los cursos han optado por carreras STEM.

“A todas las niñas que están prontas a escoger una carrera de educación superior las invitamos a aprovechar este espacio de crecimiento, aprendizaje y exploración. En nuestros cursos acercamos a las alumnas a distintas aplicaciones de la programación en diversos rubros. Escuchan constantemente testimonios de mujeres profesionales que usan estas herramientas en su día a día y tienen un acercamiento al campo laboral al que se podrían enfrentar”, comenta la directora ejecutiva de Niñas Pro, quien, desde su propia experiencia como estudiante de ingeniería confirma las ventajas de estudiar una carrera ligada al mundo STEM.

“Las áreas STEM son cada día más importantes para el desarrollo de la socie-



Camila Pinto, directora ejecutiva de Niñas Pro.

dad, demandan la mayor cantidad de profesionales y técnicos, además son algunas de las disciplinas que ofrecen los mejores salarios en el espacio laboral. En estas carreras se desarrolla el pensamiento crítico, habilidad fundamental para ser un aporte activo en la sociedad. También las carreras del mundo STEM no encasillan a un trabajo en particular y te abren un abanico de posibilidades para desarrollarse profesionalmente”, sentenció. ■

REFERENCIAS

- [1] “Más Mujeres en Ciencias”, Ministerio de la Mujer y Equidad de Género https://minmujeryeg.gob.cl/?page_id=4080.
- [2] “¿A qué tipo de carreras aspiran los chicos y las chicas?”, www.pisa.oecd.org.
- [3] <https://www.oecd.org/pisa/pisaproducts/pisainfocus/PISA%20in%20Focus-n%C2%B014%20ESP.pdf>.
- [4] Consejo Nacional de Educación (www.cned.cl) “Matrículas sistema de educación superior” septiembre 2021.
- [5] “Radiografía de Género en Ciencia, Tecnología, Conocimiento e Innovación”, Ministerio de Ciencias 2022.
- [6] “Educación en STEM con perspectiva de género: empoderar a las niñas y las mujeres para los trabajos de hoy y de mañana”, Unesco.
- [7] “Mujeres en carreras académicas en las áreas de Ciencias, Tecnologías, Ingenierías y Matemáticas (STEM)”, Biblioteca Congreso Nacional de Chile.



El Doctorado en Computación de la Universidad de Chile



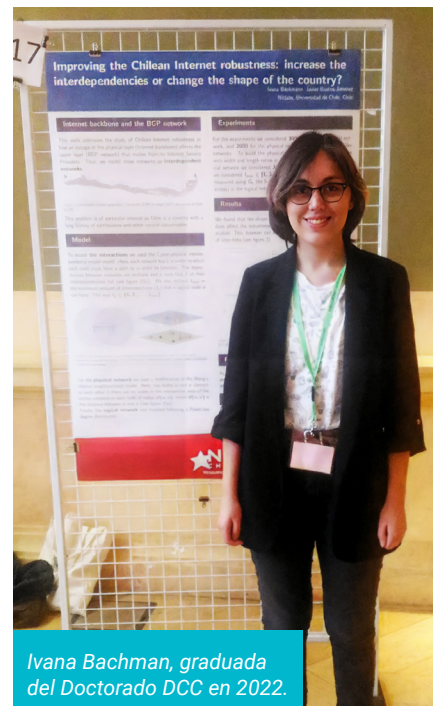
GONZALO NAVARRO

Profesor Titular del Departamento de Ciencias de la Computación de la Universidad de Chile y Coordinador del Doctorado en Computación.

✉ gnavarro@dcc.uchile.cl



Anelis Pereira, estudiante de Doctorado DCC.



Ivana Bachman, graduada del Doctorado DCC en 2022.

El Departamento de Ciencias de la Computación (DCC) imparte el programa de Doctorado en Computación de la Facultad de Ciencias Físicas y Matemáticas. El objetivo principal del programa es formar especialistas que posean un amplio dominio del área de la Computación y que sean capaces de hacer aportes de investigación originales a nivel internacional, los cuales signifiquen una contribución al avance de esta disciplina. Los graduados del programa estarán capacitados para desempeñar tanto cargos académicos en instituciones de educación superior como cargos de investigación y desarrollo en instituciones y empresas científico-tecnológicas, tanto en Chile como en el extranjero.

El programa cuenta con más de 30 estudiantes de varios países, graduando un promedio de 5 estudiantes por año. Los estudiantes interactúan e investigan junto al cuerpo académico del DCC, todos prestigiosos científicos, en un ambiente de colaboración y cordialidad. Además son estimulados a que conozcan el desarrollo de la investigación internacional en su área, mediante la inclusión de sus

trabajos en conferencias internacionales o con pasantías en centros de excelencia mundial. El DCC recibe anualmente la visita de una veintena de investigadores extranjeros, teniendo los estudiantes la posibilidad de interactuar con ellos. Regularmente se realizan intercambios de estudiantes con diversas universidades de Europa y Norteamérica.

El requisito general de admisión del programa es poseer el grado de Licenciado en Ciencias mención Computación, o de algún título o grado que pueda considerarse equivalente en nivel y contenido. Los postulantes de nivel excepcional que no posean esta formación podrán ser admitidos condicionalmente a realizar una nivelación.

El plan de estudios contempla 4 años de trabajo, divididos en tres etapas:

1. La etapa de iniciación (2 semestres), que consiste en tomar cursos de formación académica y de introducción a la investigación. Esto incluye un semestre en el cual se desarrolla una investigación acotada en un área es-

pecífica. La etapa de iniciación concluye con la presentación del examen de calificación.

2. La etapa de desarrollo (3 semestres), que consiste en iniciar el desarrollo del trabajo de tesis, con una dirección bien definida. En el primer mes de iniciar esta etapa, el estudiante entrega una propuesta inicial de tesis. Esta etapa culmina con la presentación del examen de propuesta doctoral.
3. La etapa de consolidación (3 semestres), que consiste en la profundización del trabajo de tesis, trabajando en los resultados y validación de la tesis. Esta etapa termina con la defensa de tesis doctoral y la obtención del grado de Doctor en Computación.

Los cursos a tomar en la primera etapa se dividen en obligatorios, básicos, y electivos. Los cursos obligatorios, Investigación en Ciencias de la Computación e Introducción a la Tesis, introducen al estudiante al mundo de la investigación. En el primero se ven metodologías de investigación, y en el segundo se desarro-



lla una pequeña investigación acotada con un profesor guía que usualmente termina siendo el director de la tesis. Estos cursos pueden saltarse si el estudiante trae una experiencia de investigación previa, como por ejemplo un Magíster.

Los cursos básicos buscan dar al estudiante una cobertura más amplia en las áreas generales de computación. Deben elegirse tres cursos entre Teoría de la Computación, Algoritmos, Lenguajes de Programación, Bases de Datos, Sistemas Operativos e Ingeniería de Software. Algunos de estos cursos pueden convalidarse con cursos previos del mismo nivel que el estudiante haya tomado en su carrera de pregrado, y se convalidan todos en caso de provenir de la Licenciatura en Ciencias de la Computación de la FCFM.

Los cursos avanzados son tres electivos que pueden escogerse de la oferta de cursos avanzados de cada semestre, incluyendo cursos dictados en otros Departamentos y otras universidades. También pueden convalidarse con pasantías o asistencia a escuelas temáticas en el extranjero.

El claustro del programa determina los profesores que pueden guiar tesis. Otros profesores pueden coGUIAR con un profesor del claustro. En este momento el claustro está formado por 16 profesores jornada completa del DCC, internacionalmente reconocidos en sus áreas de investigación. El claustro ofrece temas de tesis en las áreas de inteligencia artificial, teoría de la computación, ciencia e ingeniería de datos, lenguajes de programación, ingeniería de software, ciencia e ingeniería computacional, y computación centrada en las personas. Se puede ver un registro de las tesis realizadas en https://www.dcc.uchile.cl/postgrado/registro_tesis/.

La ANID ofrece becas para cursar el programa, que incluyen el arancel completo y un estipendio mensual que permite dedicarse a los estudios en forma ex-

Acreditación

7
años

Acreditado hasta el
23 de septiembre de 2024



clusiva, además de financiamiento para materiales y asistencia a congresos y estadías en el extranjero. Los académicos del claustro también pueden tener acceso a proyectos de investigación que permiten financiar becas y viajes para estudiantes de doctorado. Finalmente, el programa complementa estas becas con otras propias provenientes de convenios con NIC Chile, y ofrece asimismo rebajas de arancel y ayudas de viaje adicionales.

El DCC ofrece amplios espacios de trabajo compartido y oficinas para los estudiantes, así como salas de esparcimiento con la infraestructura adecuada. Los estudiantes también tienen disponibles los espacios de la Facultad, incluyendo su infraestructura deportiva. Cada alumno tiene acceso a un computador propio para su trabajo, así como acceso a la red inalámbrica, escáneres e impresoras láser. Para ciertos proyectos, existe la posibilidad de acceso a *clusters* de computadores de alta velocidad. Finalmente, la

biblioteca de la Facultad ofrece acceso y préstamo de libros físicos, así como un gran número de publicaciones digitales. El DCC provee además suscripción a la ACM Digital Library, IEEE/IET Xplore Electronic Library, y SpringerLink.

La mayoría de los graduados del programa se desempeña en empresas que ofrecen importantes desafíos tecnológicos o como profesores en universidades de todo el mundo. Otros, más recientemente graduados, se encuentran haciendo estadías postdoctorales.

En el último proceso la CNA acreditó el programa por 7 años hasta el 23 de septiembre de 2024.

Más información y proceso de postulación están disponibles en <https://www.dcc.uchile.cl/postgrado/doctorado/>. En la sección Estudiantes DCC de la Revista, también publicamos regularmente la experiencia de los graduados recientes del programa. ■



Estudiantes DCC



En esta sección de la Revista estudiantes recientemente graduados del Departamento de Ciencias de la Computación (Universidad de Chile) nos cuentan, junto a sus profesores guías, sobre sus trabajos de memoria y/o tesis.



Tesis de doctorado

Spatio-temporal Textual Data Modeling

Estudiante: Juglar Díaz

Profesores guías: Bárbara Poblete y Felipe Bravo Márquez



Me gradué de Ciencia de la Computación en la Universidad de Oriente en Santiago, Cuba. Después de graduarme estuve unos años vinculado a un centro de investigación asociado a la universidad donde investigaba en temas de minería de textos, y desarrollaba soluciones que luego eran implementadas en aplicaciones prácticas. Al mismo tiempo trabajaba como docente en materias de inteligencia artificial en la misma universidad. Mi primer contacto con el DCC de la Universidad de Chile fue a través de compañeros de pregrado que fueron aceptados en el programa de doctorado. Lo que me motivó finalmente a postular, fue el trabajo de Bárbara Poblete en temas de minería de datos en redes sociales. Bárbara Poblete y Felipe Bravo Márquez, que se incorporó luego al DCC, terminaron siendo mis profesores guías.

Mi tesis de doctorado “Spatio-temporal Textual Data Modeling” se enfocó en la representación conjunta de texto, tiempo, y espacio. La popularidad y el uso de redes sociales en dispositivos móviles con GPS, proporciona una fuente de textos enriquecidos con contexto espacio-temporal. Otros dominios, como consultas a motores de búsqueda y descripciones de incidentes delictivos, son también fuentes de textos para los que se conoce cuándo y dónde fueron generados. Texto, tiempo y espacio tienen diferentes formas de representación, por lo que no es trivial desarrollar un modelo que los represente de forma conjunta. La representación conjunta de texto, tiempo y espacio se ha basado en técnicas que ignoran la estructura secuencial de los textos además de propiedades de tiempo y espacio como

vecindad y jerarquía. Esto puede limitar la expresividad de un modelo para representar ciertos patrones. En la tesis presentamos dos nuevos modelos basados en redes neuronales para recuperación de información multimodal y modelado de lenguaje condicionado espacio-temporalmente. Los modelos propuestos presentan aplicaciones prácticas en la recuperación en texto-espacio-tiempo y en la caracterización de zonas urbanas.

Durante el doctorado, el método de trabajo fue encuentros semanales donde actualizábamos cómo íbamos en la investigación o la publicación que estuviéramos trabajando en ese momento y decidíamos cuáles eran los próximos pasos a seguir. Trabajar con Bárbara y Felipe fue una experiencia muy buena y de mucho aprendizaje constante sobre cómo se desarrolla un proyecto de investigación, cómo se escribe un artículo científico, y cómo se muestran los resultados a la comunidad científica. En el caso de la investigación en minería de datos, existe mucho entusiasmo por los nuevos avances en técnicas de aprendizaje automático basadas en redes neuronales, lo que puede guiar a un estudiante de doctorado hacia una filosofía enfocada en modelos. Si tuviera que resumir el mayor aprendizaje de mi investigación durante el doctorado, diría que es enfocarse en los datos y permitir que los datos y la información que descubramos en los datos sea lo que guíe el trabajo de investigación. Esta filosofía la llevo conmigo en mi nueva etapa profesional donde me desempeño como Científico de Datos Principal en la división MACH del Banco de Crédito e Inversiones (BCI).



Tesis de doctorado

Engineering Interaction Requirements of Mobile Applications that Support People-Driven Collaborative Processes

Estudiante: Maximiliano Canché

Profesores guías: Sergio Ochoa y Daniel Perovich



Estudié Licenciatura en Ciencias de la Computación en la Universidad Autónoma de Yucatán, México. Entre el último par de años de mis estudios y un tiempo después de egresar, trabajé en la industria y la administración de un centro de cómputo para después realizar un Magíster en Ciencias en Tecnología Informática en el Instituto Tecnológico y de Estudios Superiores de Monterrey, México. Después de ello trabajé en la docencia en mi universidad de formación y en 2016, por interés en la investigación, decidí iniciar el doctorado en el Departamento de Ciencias de la Computación de la Universidad de Chile.

Mi tesis de doctorado, titulada “Engineering Interaction Requirements of Mobile Applications that Support People-Driven Collaborative Processes”, se enmarca en el área de ingeniería de requerimientos de sistemas colaborativos. Dicho trabajo está centrado en la identificación y validación de los requerimientos de interacciones para el diseño de aplicaciones que apoyan procesos colaborativos dirigidos por personas (PDCP, por sus siglas en inglés), cuyo flujo de trabajo no está definido al momento de su diseño ya que dependen en la práctica de variables del contexto (tales como una lista de prioridades de las tareas, disponibilidad de las personas, etc.). Un estudio inicial realizado en este trabajo, muestra que los lenguajes y notaciones usados para crear modelos visuales de los escenarios de interacción de procesos colaborativos dirigidos por personas, resultan limitados para abordar los desafíos involucrados en esta actividad de modelado. En consecuencia, proponemos un lenguaje denominado CIMoL (Computer-Mediated Interactions Modeling Language) para modelar escenarios de interacción mediados por computadora involucrados en PDCPs. Este lenguaje, junto con una herramienta de generación automática de prototipos (denominada CIMoL Modeler), permite a los ingenieros de software y *stakeholders* identificar y validar los servicios de interacción (e.g. audio y videoconferencia, envío de mensajes de texto, pizarrón electrónico, etc.) que requieren los

participantes en un PDCP con el fin de lograr un entendimiento compartido sobre el alcance del sistema en desarrollo.

Uno de los principales desafíos a los cuales me enfrenté al iniciar el doctorado, fue volver a adquirir el ritmo de estudiante de pregrado. Entre las clases, tareas, proyectos, exámenes, desarrollo del tema de tesis desde el inicio de los estudios, me surgieron complicaciones, principalmente en los dos primeros semestres del doctorado. Afortunadamente, el adaptarme de nuevo a tal ritmo me facilitó superar otros desafíos implícitos en los estudios doctorales.

El trabajo con mis profesores guías, Sergio Ochoa y Daniel Perovich fue excelente, las capacidades de ambos se evidencian en la dirección de todos los productos generados en este trabajo de tesis. Primero inicié con Sergio, cuyas ideas, disposición y conocimientos, me ayudaron a definir mi tema de investigación. Posteriormente, Daniel nos apoyó para mejorar la propuesta inicial hasta ser lo que ahora finalizamos. El camino no fue fácil, sin embargo, el apoyo de ambos fue determinante para poder lograr terminar el doctorado exitosamente. En ese camino tuve la oportunidad de crear amistad con gente del Departamento, además de vínculos con investigadores de Chile y de otros países, asistir a diversas conferencias internacionales, publicar varios artículos de investigación derivados del trabajo de tesis, asistir a una escuela de verano y realizar una pasantía de investigación.

Actualmente me encuentro trabajando en México en mi universidad de origen, la Universidad Autónoma de Yucatán, como académico en la carrera de Ingeniería de Software. Adicional a la docencia, realizo actividades de tutoría, gestión, e investigación, lo cual incluye dirección de tesis de estudiantes de Licenciatura en la que aplico muchos de los conocimientos y habilidades adquiridos durante los estudios doctorales.



Tesis de doctorado

Socio-Technical Analysis of The Robot Operating System Ecosystem for Fostering Participation in Knowledge Sharing Activities



Estudiante: Pablo Estefó

Profesores guías: Jocelyn Simmonds, Romain Robbes y Johan Fabry

En mis últimos años de pregrado en el DCC comencé a adentrarme en el mundo de la investigación. Junto con el profesor Alexandre Bergel, trabajamos en una herramienta visual para la refactorización de tests unitarios. Este trabajo fue seleccionado en la Student Research Competition enmarcada en la Conferencia Internacional de Ingeniería de Software (ICSE), la más importante del área donde obtuve el primer lugar en la categoría pregrado. Este hito me llevó a realizar una pasantía de investigación en la École des Mines de Douai, al norte de Francia. Allí tuve mi primer acercamiento a la robótica. A mi retorno de la pasantía, recibí la invitación del profesor Johan Fabry para realizar el doctorado en el tema de Ingeniería de Software para Robótica.

Mi trabajo de doctorado se titula "Socio-Technical Analysis of The Robot Operating System Ecosystem for Fostering Participation in Knowledge Sharing Activities" (en español, "Análisis Sociotécnico del Ecosistema de Robot Operating System para fomentar la participación en actividades de intercambio de conocimientos"). El desarrollo de software para robots es una tarea compleja con desafíos particulares que la diferencian de, por ejemplo, el desarrollo de software para aplicaciones móviles u otros contextos. Mi trabajo de tesis comprende principalmente un estudio exploratorio de la experiencia de desarrollo de los desarrolladores de software para robots. A través de un estudio de campo recolectamos, analizamos y categorizamos evidencia que nos permitió resumir en seis cuellos de botella de contribución en el ecosistema el Robot Operating System (ROS). Apoyándonos en la literatura y contrastándola con la evidencia recolectada, propusimos seis recomendaciones que abordan uno o más de estos cuellos de botella.

Finalmente, implementé un recomendador de expertos para la plataforma web de preguntas y respuestas (Q&A) que usa esa comunidad: ROS Answers. El problema clásico de estos recomendadores es encontrar el usuario más idóneo para responder una pregunta dada. En mi caso le di un pequeño giro a la pregunta estableciendo el siguiente problema: encontrar un usuario calificado para responder la pregunta, pero con un nivel de participación menor. Esto último con el objetivo de distribuir mejor la carga de trabajo y fomentar la participación de usuarios menos activos. Adapté un sistema de recomendación exist

ente que se basa fuertemente en los "tags" o palabras usadas para describir preguntas. Luego de tres pruebas pude concluir que si bien recomienda al experto, no fomenta la participación de usuarios menos activos.

La principal contribución de mi doctorado fue la evidencia de aspectos sociotécnicos del ecosistema ROS, recomendaciones basadas en la literatura y la evidencia encontrada para abordar los cuellos de botella de contribución en el ecosistema de ROS y, finalmente, un enfoque novedoso para distribuir la carga de trabajo en actividades de intercambio de conocimiento en el ecosistema de ROS.

El principal desafío que tuve al comienzo del doctorado, fue que no tenía un problema definido a tratar en el mismo. Junto con esto, el estudio de ecosistemas de software estaba recién comenzando y había muy pocos estudios de la actividad de desarrollo de aplicaciones para robótica desde una perspectiva de ingeniería de software. Esto motivó un estudio de campo para explorar y evidenciar la experiencia de los desarrolladores de software para robots.

En mi camino en el doctorado, tuve la inusual experiencia de estar bajo la tutela de tres profesores guías. El profesor Johan Fabry, quien dirigió mi primer año como estudiante de doctorado, tuvo que volver a su país natal por razones personales. Luego, trabajé guiado por el profesor Romain Robbes, quien aportó una perspectiva diferente a los estudios que estaba llevando hasta ese entonces. El profesor Robbes se trasladó a la Universidad Libre de Bolzano-Bozen en el norte de Italia. La segunda mitad del doctorado lo realicé con la profesora Jocelyn Simmonds. Trabajar con ella fue una gran escuela, aprendí mucho de metodologías de trabajo académico, así como también conté con alguien que me alentó a continuar mis estudios encontrando valor en cada aporte realizado. Finalmente puedo decir que tuve la suerte de aprender de tres académicos con vocación formativa.

Actualmente estoy trabajando en Cornershop by Uber. Mi plan es continuar desarrollándome en la industria aportando con mis habilidades desarrolladas en el doctorado.



Tesis de magister



Business Intelligence en la producción con un enfoque estratégico, táctico y operativo para la gestión de proyectos informáticos

Estudiante: Eduardo Sánchez Villegas

Profesor guía: Hugo Beltrán Alejos

La tesis se enfoca en la importancia de la implementación de Business Intelligence en la producción de proyectos informáticos, en particular para la gestión y control de iniciativas empresariales.

El objetivo principal es desarrollar un sistema de gestión de información para supervisar la construcción de proyectos en la organización. Como resultado, se definió y construyó un sistema de paneles de gestión que contiene varios KPIs (Key Performance Indicators) para medir y mejorar los índices de productividad. Este sistema se desarrolló mediante una gestión ágil de proyectos, realizando entregas tempranas para probar y mejorar el producto final de manera evolutiva.

Los paneles de gestión registran las horas de trabajo, el presupuesto, los recursos consumidos y los entregables de los proyectos, permitiendo un monitoreo constante. Los resultados de la evaluación muestran que el sistema proporciona visibilidad del estado de los proyectos a la dirección de la empresa y facilita la toma de decisiones para mejorar los indicadores. Durante el desarrollo del proyecto se hizo un seguimiento de las métricas para validar los objetivos. Como resultado final, se adoptaron prácticas como el registro de horas, que es esencial para cualquier indicador de gestión definido.

Tesis de magister



A Framework for Training and Evaluating Incremental Word Embeddings from Text Data Streams

Estudiante: Gabriel Iturra Bocaz

Profesor guía: Felipe Bravo Márquez

Los *Word Embeddings* (WE) han adquirido una relevancia indispensable en diversas tareas de Procesamiento de Lenguaje Natural (NLP) y Recuperación de Información (IR), tales como clasificación de documentos, ranking y respuesta a preguntas. Sin embargo, estos modelos presentan una limitación significativa: son entrenados en entornos estáticos, lo que significa que no pueden actualizar sus conceptos en el tiempo. Esta rigidez dificulta su capacidad para adaptarse a los patrones de lenguaje en constante evolución que surgen en fuentes como redes sociales y la Web, como nuevos hashtags o nombres de marca. Para enfrentar este desafío, se han desarrollado algoritmos de Incremental WE que permiten la actualización dinámi-

ca de las representaciones de palabras a nuevos patrones de lenguaje y flujos de datos de texto.

A pesar de los avances en los algoritmos de Incremental WE, aún enfrentamos problemas importantes. Estos algoritmos suelen encontrarse en repositorios aislados, sin una interfaz común para su utilización. Además, debido al enfoque continuo o de "*streaming*" en su aprendizaje, no existe una estandarización o metodología clara para evaluar la calidad de los WE creados a partir de estos algoritmos.

Frente a esta problemática, hemos desarrollado RiverText, una herramienta innovadora que busca reunir las diferentes técnicas de Incremental WE en una interfaz común y de fácil uso para la comunidad científica. Adicionalmente, RiverText propone una metodología para evaluar la calidad de los WE entrenados a partir de flujos continuos de texto, que consiste en aplicar una serie de evaluaciones periódicas utilizando tareas intrínsecas de NLP.

Los resultados de este trabajo se presentaron en la 46ª Conferencia Internacional SIGIR sobre Investigación y Desarrollo en Recuperación de Información, llevada a cabo en julio de 2023 en Taipei, Taiwán. Durante esta instancia, el trabajo fue reconocido como una contribución novedosa y aplicable, que aborda un problema real y abre la puerta a nuevas interacciones entre las disciplinas de Incremental Learning, NLP e IR.



Memoria de pregrado

Implementación de Leapfrog Triejoin sobre Estructuras de Datos Compactas



Estudiante: Daniela Campos
Profesores guías: Gonzalo Navarro y Diego Arroyuelo

En esta memoria se abordó el problema de resolver eficientemente consultas en bases de datos de grafos, en particular Basic Graph Patterns (BGPs), que son las consultas más frecuentes.

En las bases de datos de grafos, los nodos representan entidades y las aristas entre ellos, con labels, representan propiedades. Los datos se pueden ver como un gran conjunto de triples $(s, p, o) = (\text{sujeto, predicado, objeto})$, por ejemplo $(\text{Ana, jefa-de, Bruno})$ establece una arista con label “jefa-de” que va de “Ana” a “Bruno”. Un BGP consiste en un conjunto de triples donde algunas posiciones son variables, que debe encontrarse en el grafo de todas las formas posibles. Por ejemplo, (Ana, jefa-de, X) , $(\text{Ana, estudió-en, Y})$, $(\text{X, estudió-en, Y})$ encuentra subordinados X de Ana que estudiaron en su misma universidad Y. Resolver estas consultas es muy costoso, y equivale a realizar *multi-joins* (y *selects*) en bases de datos relacionales.

La *worst-case optimality (wco)* es un concepto reciente que alude a un algoritmo cuyo costo es óptimo “en el peor caso”

porque es similar al tamaño del *output* en alguna base de datos. Se ha demostrado que los clásicos planes de consultas *multi-joins* haciendo *joins* de a pares no son *wco*. A pesar de ello, pocos sistemas incorporan los algoritmos *wco* para resolver sus BGPs. La razón es que éstos requieren de mucho espacio. Por ejemplo, el Leapfrog Triejoin (LTJ), el algoritmo *wco* más popular, requiere de 6 copias de los índices para funcionar, donde cada copia es un *trie* que almacena todos los triples del grafo en distinto orden $((s, p, o), (o, p, s), \text{etc.})$.

En esta memoria se buscó abordar este problema mediante la implementación de los *tries* utilizando estructuras de datos compactas. Concretamente, se utilizó una representación sucinta de árboles generales que utiliza dos bits por nodo y permite navegar hacia cualquier hijo en tiempo constante. Los identificadores de los nodos se almacenaron como vectores compactos. Sobre esta representación se implementó el algoritmo LTJ, que realiza búsquedas sobre los nodos a medida que baja en el árbol.

El resultado es una representación que ocupa 4 veces menos espacio que Jena-LTJ, un índice clásico *wco* que representa los 6 *tries*. Es, además, mucho más rápido. Hay estructuras aún más compactas, como el Ring, que ocupa 30% del espacio de la nueva estructura, pero es unas 15 veces más lento en la mediana. Esta nueva estructura hace posible incorporar algoritmos *wco* en implementaciones reales que manejen bases de datos de grafos masivas. El resultado de esta memoria se incluyó en un artículo que está en evaluación en una conferencia internacional de primer nivel.

Esta memoria se realizó en el marco del Instituto Milenio Fundamentos de los Datos (IMFD).



Memoria de pregrado



Hacia un primer modelo de implicancias de diseño para videojuegos en realidad virtual considerando estrategias de inmersión e involucramiento de jugadores

Estudiante: Fabiola Rivera Salgado

Profesor guía: Francisco J. Gutiérrez

En estos últimos años la realidad virtual (RV) se ha posicionado como una herramienta emergente en diversos dominios, transformando no sólo la forma en que interactuamos con el mundo digital, sino que también en cómo se diseñan experiencias significativas para los usuarios. En el área de videojuegos, los diseñadores han aprovechado las oportunidades de interacción que ofrece el hardware y el software para crear experiencias llamativas que transporten a los jugadores al reino virtual. Gracias a la RV, los jugadores pueden explorar entornos desde nuevas perspectivas, interactuar con personajes y objetos de forma realista y experimentar una mayor sensación de presencia e inmersión en los ambientes virtuales.

La unión de la RV y los videojuegos abre un sinfín de oportunidades, pero presenta un desafío único: la ausencia de puntos de referencia de diseño establecidos. En esta línea, aún no se tiene vasto conocimiento acerca de cómo diseñar experiencias que tengan el concepto de inmersión como principal factor a considerar. Actualmente, en la literatura se han explorado alternativas del uso de esta tecnología en los campos de la medicina y educación, pero no se han realizado estudios centrados en medir la experiencia de los usuarios en un contexto de videojuegos. Es por esta razón que se ve la necesidad de definir un marco teórico adecuado para la medición del fenómeno de experiencia de usuarios con un enfoque en inmersión para videojuegos de RV.

En este trabajo de título se exploraron estrategias de inmersión e involucramiento para videojuegos que utilizan RV como principal mecanismo de interacción con el jugador. Para ello, se inició con una búsqueda en la literatura, cuyo conocimiento fue consolidado en una primera instancia del metamodelo para experiencias de juego inmersivas en RV. En esta línea, con la finalidad de dar soporte al modelo, se realizó un estudio empírico, en el cual los participantes interactuaron con videojuegos de RV y posteriormente respondieron distintos cuestionarios para evaluar los diferentes aspectos de la experiencia. Finalmente, a partir del análisis de los resultados del estudio empírico se abstrajeron implicancias de diseño, las cuales pueden ser utilizadas por diseñadores y programadores para mejorar la experiencia de usuario en términos de inmersión.



dcc

CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE



fcfm

ESCUELA DE POSTGRADO
Y EDUCACIÓN CONTINUA
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

Desarrollo Profesional

+
+
+
+

BOOTCAMPS

Programa de Educación Continua
Departamento de Ciencias de la Computación
Universidad de Chile

Reinvéntate profesionalmente y alcanza nuevas oportunidades laborales, con formación práctica de calidad y alta intensidad brindada por expertas y expertos de la industria. En estos bootcamps elaborarás un proyecto, siendo parte de un equipo multidisciplinario.

- + Metodologías ágiles
- + Diseño UX/UI
- + Desarrollo frontend
- + Desarrollo backend
- + Desarrollo de aplicaciones móviles

100%
vía streaming

Más información

www.dcc.uchile.cl/educacion_continua/bootcamps



MAGÍSTER EN TECNOLOGÍAS DE LA INFORMACIÓN

Departamento de Ciencias
de la Computación

Universidad de Chile

Acreditación

4
años

Acreditado por Acredita CI
hasta el 6 de septiembre
de 2027



ÁREAS DE ESPECIALIZACIÓN:

- Gestión y Desarrollo de Proyectos de Tecnologías de la Información
- Ingeniería de Software
- Ciencia e Ingeniería de Datos
- Inteligencia Artificial

INFORMACIÓN Y POSTULACIONES EN:

<https://www.dcc.uchile.cl/postgrado/mti/>

CONTACTO:

Departamento de Ciencias de la Computación
FCFM, Universidad de Chile



Beauchef #851, Edificio Norte, tercer piso,
oficinas 326-327. Santiago.



Correo:
capacita@dcc.uchile.cl



Teléfonos:
+56 2 2978 4965
+56 2 2978 4976
+56 9 3871 9957 *WhatsApp*

REVISTA DEL DEPARTAMENTO DE CIENCIAS DE LA
COMPUTACIÓN DE LA UNIVERSIDAD DE CHILE

Bits
DE CIENCIA