

La otra clonación

Jamie Drake

30 de julio de 2010

Cuando escuchamos o leemos la palabra clonación, generalmente la asociamos a la obtención de una copia genética de una entidad biológica. Sin embargo, existe otro tipo de clonación, una técnica que surge en el área de inteligencia artificial cuyo fin es copiar las habilidades de una persona para que una computadora pueda reproducirlas. A esa técnica se le conoce como clonación de comportamiento. En este artículo se describirá su origen, en qué consiste y sus aplicaciones.

1. Había una vez ...

Las ideas sobre máquinas que aprenden, que poseen inteligencia tomaron impulso en los años sesenta. Uno de los precursores fue Donald Michie (1923 - 2007), doctor en genética quien sentó las bases de muchas de las técnicas de aprendizaje automático usadas en la actualidad. Fue en 1993 cuando nace la técnica a la que Donald Michie llamó clonación de comportamiento. Pero, antes de continuar, conozcamos más sobre el origen de lo que ahora conocemos como aprendizaje automático (*Machine Learning*) y cómo de un hecho aparentemente negativo, se derivaron los conocimientos que han generado la tecnología actual.

Vientos de guerra. Eran finales de los años treinta y el avance de la guerra crecía día a día. En Marzo de 1938, Hitler ordenó la ocupación de Austria, y en Mayo del mismo año ordenó la ocupación de Checoslovaquia. La amenaza de guerra avanzaba sobre Europa. El gobierno británico contaba con un centro que se dedicaba a la inteligencia en las comunicaciones. Este centro era la Escuela de Cifrado y Codificación con sede en Londres y necesitaba un sitio seguro. El gobierno Británico inició la búsqueda... y encontró Bletchley Park.

Bletchley Park es una propiedad de aproximadamente 60 acres que se ubica 80 km al noroeste de Londres. Al centro se construyó una mansión de estilos arquitectónicos variados. Bletchley Park contaba con las condiciones buscadas, vías de comunicación y teletipo; era el lugar adecuado. Bletchley Park sería el lugar donde las claves y códigos de diversos países del Eje serían descifrados. Y para Agosto de 1939 los decodificadores de código empezaron a llegar, entre ellos, Alan Turing (1912 - 1954).



Figura 1: Bletchley Park

Y mientras eso sucedía, en 1943 el joven Michie recibió una beca para Estudios Clásicos en Balliol College (Oxford). En ese mismo año, un amigo de la familia, oficial de la oficina de

guerra, le contó sobre los cursos de japonés para oficiales de inteligencia. A Michie le interesaba contribuir con la campaña bélica y convencido de que eso le garantizaría un lugar en la universidad una vez que la guerra terminara, intentó inscribirse. El curso iba a ser impartido en la ciudad de Bedford, a 15 millas de Bletchley; era primavera y Michie se dirigió hacia allá para presentarse a una entrevista. Sin embargo, le comunicaron que el curso ya estaba lleno y el siguiente iniciaría en el otoño. El oficial lo vio muy desanimado y le planteó otra opción que le podía interesar: un curso de criptografía que iniciaría el Lunes siguiente. Las cosas no salieron de acuerdo a sus planes y no imaginaba lo que le esperaba.

Aunque Michie no tenía experiencia en máquinas ni en matemáticas, muy pronto sus profesores descubrieron que poseía un impresionante talento para las materias. Seis semanas después de iniciar el curso, llegó un oficial de Bletchley quien buscaba reclutar a alguien y por los resultados mostrados durante el curso, Michie fue el elegido.

Así, a poco más de un mes de haber llegado, Michie fue asignado a una sección en Bletchley Park que se había establecido en Julio de 1942. En esa sección se trabajaba descifrando el tráfico de datos encriptados generado por el teletipo alemán al cual los británicos llamaban “Fish”. Un teletipo es una máquina de escribir que se usaba para enviar y recibir mensajes usando el radio y microondas como medio de transmisión. Los mensajes del teletipo se encriptaban con una máquina alemana codificadora, la Lorenz SZ40/42. Tanto a la máquina como a su tráfico de datos se les llamaba “Tunny”. Michie inició su trabajo bajo la dirección del Mayor Ralph Tester quien trabajaba con “Tunny” usando métodos manuales.

Un problema crítico era que los alemanes transmitían cientos de mensajes al día por lo que de no elaborar una estrategia distinta, los métodos manuales de desciframiento no serían suficientes. Se necesitaba con urgencia automatizar el proceso. Y fue a Max Newman, un matemático teórico a quien se le asignó la tarea de automatizar la decodificación de los mensajes. Newman necesitaba un asistente; alguien que estuviera perfectamente familiarizado con los métodos manuales, que además tuviera aptitudes para probabilidad, estadística y lógica booleana. Esa persona no fue otra que Donald Michie. Al grupo se le unió Tommy Flowers, ingeniero que por su gran capacidad, había impresionado a Alan Turing quien lo presentó a Newman. El resultado del trabajo de este equipo fue una de las primeras computadoras electrónicas, digitales, programables a la que llamaron Colossus. Su diseño es obra de Flowers y se convirtió en una de las principales herramientas para decodificar mensajes. La computadora estaba compuesta por 1,500 tubos de vacío.

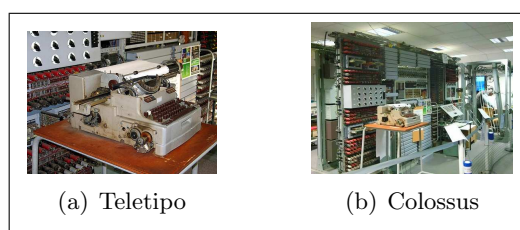


Figura 2: Teletipo y Colossus

Donald Michie hizo importantes contribuciones. Sus mejoras a la computadora Colossus redujeron significativamente el tiempo necesario para interceptar y decodificar los mensajes, pasando de semanas a unas cuantas horas, un logro más allá de lo esperado.

Durante la estancia en Bletchley, Michie y Turing se hicieron amigos. Compartían juegos de ajedrez y charlaban sobre las posibilidades de construir una computadora que jugara ajedrez,

como parte de un un proyecto mayor para la creación de una inteligencia artificial. Esos momentos de esparcimiento fueron definitorios para el futuro de la inteligencia artificial. En 1946 Michie dejó Bletchley; las computadoras Colossus -diez para entonces- fueron destruidas y el trabajo se mantuvo en secreto hasta los años setenta, cuando se empezó a liberar información.

Después de la guerra. Al terminar la guerra, Donald Michie aceptó la beca en Estudios Clásicos que le había sido otorgada pero pronto encontró esos estudios aburridos. Optó entonces por estudiar anatomía humana y fisiología (1949) y posteriormente obtuvo un doctorado en genética de mamíferos (1953). Hizo aportaciones también en biología molecular. Sin embargo, a principios de los sesenta su atención regresó a aquellas charlas con Turing durante su estancia en Bletchley. En particular, le interesaba investigar si las computadoras podían ser programadas para aprender de la experiencia [5]. ¿Qué habría pasado si el curso de japonés no se llena?

Michie siempre estuvo al tanto del trabajo que se hacía en el Laboratorio Nacional de Física en donde trabajaban con ideas sobre juegos automatizados similares a las máquinas para jugar sobre las que Turing y él platicaban en Bletchley. Aunque Michie carecía en ese tiempo de acceso a computadoras, ese no fue un obstáculo para explorar sus ideas. Michie desarrollaba sus sistemas en papel y utilizaba otros recursos. En 1960 diseñó MENACE (Matchbox Educable Noughts and Crosses Engine), usando cajas de cerillos (sí!, cajas de cerillos!) y cuentas de cristal de distintos colores ó semillas. MENACE era capaz de aprender por la experiencia, fue uno de los primeros experimentos en aprendizaje por refuerzo y aprendió a jugar lo que conocemos como gato (tic-tac-toe, noughts and crosses). MENACE utilizaba el conceptualmente simple algoritmo de propósito general BOXES el cual podía descubrir también estrategias de control para el problema del péndulo y el carro. Actualmente nos quejamos cuando no contamos con acceso a internet; sin embargo, cuando se tiene tal pasión por hacer las cosas, no hay obstáculo válido.



Figura 3: MENACE

MENACE causó tal interés que Michie fue invitado a Stanford a implementarlo en una computadora IBM. A su regreso a Inglaterra, persuadió al Consejo de Investigación y Ciencia a fundar la investigación en aprendizaje automático. El programa se estableció en 1965 en la Universidad de Edinburgo, donde Michie fue su primer director. En 1967, Michie fue designado Profesor de aprendizaje automático y la unidad cambió su nombre a Departamento de Aprendizaje Automático y Percepción. Y así, las ideas surgidas en esas amenas pláticas en Bletchley estaban convirtiéndose en realidad.

El siguiente paso de Michie fue el desarrollo de técnicas para extracción inductiva de conceptos a partir de ejemplos. Fue así como en 1973 su equipo construyó a Freddy, el primer robot que era capaz de aprender al mostrarle cómo ensamblar un objeto. Freddy usaba visión para reconocer componentes. En 1983 desarrolló Expert-Ease, un programa que razonaba y podía generar explicaciones. El programa extraía reglas de humanos expertos.

El objetivo de Michie era lograr que las computadoras mostraran inteligencia y para ello, se requería el aprendizaje de habilidades. Ya no se trataba solamente de aprender reglas de juego sino de la adquisición de habilidades que estaban más allá de lo que una persona podía describir. Hacer que una computadora adquiera una habilidad no es algo sencillo; a continuación veremos por qué.

2. ¿Qué es una habilidad?

Cuando queremos aprender algo, primero aprendemos lo básico, practicamos, adquirimos nuevo conocimiento y a medida que el conocimiento y la práctica aumenta, nuestra habilidad crece y nos hacemos expertos. La adquisición de una habilidad consta de dos componentes: el cognitivo y el subcognitivo. La parte cognitiva es el cómo-hacerlo, es adquirir el conocimiento sobre lo que se quiere aprender. Por ejemplo, si se está aprendiendo un nuevo idioma, primero se aprende gramática, pronunciación básica y expresiones simples. Conforme se avanza, se va adquiriendo nuevo conocimiento y practicando, incrementando el dominio del lenguaje. En el componente cognitivo cuando a la persona se le enseña algo nuevo, piensa en ello, lo expresa en sus propias palabras, observa cómo la nueva información se ajusta a otras cosas que la persona ya conoce; la persona es consciente de lo que aprende.

Por otro lado, el componente subcognitivo es la modificación de la habilidad que ocurre con la práctica y se da de manera automática, sin que la persona sea consciente. Cuando la persona practica, perfecciona la habilidad, y puede llegar a ser un experto. Sin embargo, si el conocimiento recibido fue equivocado, la habilidad se perfeccionará pero siempre con un resultado defectuoso. Por ejemplo, si en el aprendizaje de un nuevo idioma se enseñó mal la pronunciación, la práctica se hace de acuerdo a esa mala pronunciación. Sólo adquiriendo conocimiento nuevo que sustituya al equivocado y practicando puede corregir el error. En este caso, los órganos que intervienen en la pronunciación tienen que ajustarse gradualmente al nuevo conocimiento adquirido lo que va más allá del alcance de quien pronuncia. No se le puede ordenar explícitamente a los músculos cómo moverse. Esta es la parte subcognitiva, lo que conscientemente no puede modificarse ni explicarse.

La pregunta era: ¿Cómo hacer que una computadora extrayera o aprendiera una habilidad? Se necesitaba de un mecanismo capaz de adquirir el conocimiento a nivel subcognitivo. Con este objetivo, surgió la clonación de comportamiento, en la cual, el aprendizaje automático se usa para construir una descripción simbólica donde la introspección por el humano falla porque la tarea de realiza de forma subconsciente.

3. Clonación de comportamiento

Esta técnica se originó en parte, motivada por las críticas a la Inteligencia Artificial en el sentido de que no se daba importancia a las habilidades de bajo nivel y sólo se enfocaba a los procesos cognitivos de alto nivel [10]. El término **clonación de comportamiento**, a la que nos referiremos como **clonación**, fue acuñado por Donald Michie en 1992 y tiene por objetivo extraer conocimiento explícito de habilidades de bajo nivel. La **clonación** busca hacer explícita una habilidad y que ese conocimiento sirva para construir controladores que reproduzcan esa habilidad. Al ser difícil programar manualmente ese tipo de habilidades, el aprendizaje automático es de gran ayuda para construir estos sistemas de control.

Mediante la clonación, las actividades subcognitivas requeridas para realizar una habilidad de alto nivel pueden modelarse de tal forma que la habilidad sea explicable y que el modelo sea operacional. Ser explicable significa que la salida del programa de aprendizaje debe poder ser leída y entendida por un humano.

La técnica básica consiste en obtener ejemplos o trazas de las variables involucradas en el proceso cuando un operador o sistema ejecuta la tarea. Los ejemplos se dan como entrada a algoritmos de aprendizaje automático que pueden construir modelos, por ejemplo reglas o árboles, que al ejecutarse producirán comportamientos similares a quien generó los ejemplos. Los modelos resultantes pueden ser incorporados como programas controladores. A estos programas se les llama “clones” porque reproducen la habilidad de una persona o sistema.

Una de las ventajas de esta técnica es que no requiere un modelo matemático de control tradicional pues no siempre se cuenta con la información suficiente o el modelo exacto del proceso. La persona que entrena o instruye, únicamente necesita ejecutar la tarea o la acción que se quiere enseñar; no tiene que comprender sus fundamentos teóricos. En este contexto, los “clones” son representaciones simbólicas de comportamientos de bajo nivel.

Los principales objetivos de esta técnica son: (i) producir “clones” que puedan realizar la tarea de control y (ii) producir “clones” que hagan explícita la habilidad, que describan lo que el operador hace [3]. El primer objetivo es importante porque producir un clon que sirva como controlador facilita la programación al no tenerse que codificar explícitamente la tarea. El segundo objetivo permite tener información explícita sobre la estrategia de control llevada a cabo por el operador.

La formulación original de la clonación es la recuperación de la estrategia de control a partir de trazas y es un mapeo directo de estados a acciones. Una traza está formada generalmente por pares (Estado, Clase) donde Estado es un vector de atributos (x_1, x_2, \dots) y Clase es la acción efectuada por el operador en ese estado.

Las técnicas de aprendizaje que se han usado más frecuentemente en clonación son los árboles de decisión [13], aunque otras técnicas que reconstruyan funciones a partir de ejemplos pueden ser usadas, como las redes neuronales.

Más allá de la clonación básica. El enfoque básico de clonación consiste en obtener trazas, aplicar un algoritmo y obtener un modelo. Sin embargo, este proceso no es tan directo cuando se tiene una tarea que se compone a su vez de sub-tareas y metas.

Conforme las habilidades a aprender se han ido haciendo más complicadas han surgido otros enfoques, por ejemplo, la descomposición, el manejo de metas y jerarquías. El tipo de los datos contenidos en las trazas también es importante. Las tareas de control generalmente contienen atributos numéricos y aunque es común usar la discretización, existen técnicas que se aplican a este tipo de datos. Otro aspecto importante a determinar es la representación del Estado; si va a ser mediante un vector de atributos fijo o si se va a usar aprendizaje relacional. La clonación se ha combinado también con otras técnicas de aprendizaje como por ejemplo, aprendizaje por refuerzo. El desarrollo continúa y cada vez los retos son mayores.

4. Aplicaciones de la clonación

¿Acaso no sería muy interesante que se pudiera entrenar a los dispositivos que usamos para que se comporten de acuerdo a nuestras necesidades? Por ejemplo, que el despertador, después de un tiempo de usarlo, sea capaz de activar o desactivar su alarma de acuerdo al entrenamiento adquirido. Y además, que el auto sea capaz de conducirse de forma autónoma, con un estilo similar al de su dueño. Estas aplicaciones no están lejos de ser una realidad, si no es que

algunas ya existen. Los experimentos realizados con clonación han sido diversos, motivados por esa realidad futura. Veamos a continuación algunos ejemplos:

El poste y el carro. Es uno de los ejemplos clásicos de control y consiste en un carro que se desliza sobre un riel de longitud fija. Un poste está asegurado al carro de modo que sólo puede balancearse en una dimensión. El carro se mueve únicamente al aplicarle una fuerza de magnitud fija hacia la izquierda o derecha. La tarea del aprendizaje es construir una estrategia de control que mantenga el poste en equilibrio y sin que el carro golpee los extremos del riel. En este problema se mostró por primera vez el aprendizaje automático por imitación con entrenamiento de un humano [9, 15]. Se han realizado también experimentos para tutores inteligentes [12], y en México, se exploró el control utilizando un enfoque bayesiano [7].

Aprendiendo a volar. Existen diversos trabajos sobre este dominio y se han hecho experimentos tanto para aviones como para helicópteros. Una de las principales motivaciones de esta aplicación ha sido el tener un piloto automático entrenado con las estrategias de vuelo de pilotos experimentados. El objetivo es que al obtener trazas de pilotos expertos, se aprenda un modelo de control para que el simulador pueda realizar un vuelo de forma automática de forma parecida a como lo hace el piloto. En uno de los trabajos se entrenó al piloto automático con las siguientes tareas: despegar, volar a una distancia y altitud determinada, regresar y aterrizar [14]. Se obtuvieron trazas de pilotos expertos, las cuales se procesaron con algoritmos de inducción de árboles. Los modelos resultantes se podían ejecutar en el simulador. Sorpresivamente, se observó que los mejores pilotos no generaron buenas trazas de aprendizaje puesto que cometían pocos errores. Las trazas de los pilotos expertos no registraron tantas situaciones de riesgo como para que los algoritmos de aprendizaje pudieran inducir modelos útiles. Se generaron mejores controladores cuando los pilotos tuvieron que corregir sus acciones. Este experimento fue aplicable al entrenamiento de pilotos.

Conforme la investigación avanza, se van agregando complicaciones a los experimentos, de modo que los modelos que se aprendan sean más realistas. Así, en experimentos posteriores se aprendió a realizar maniobras de turbulencia usando una descomposición jerárquica [8]. En México, se realizaron experimentos combinando la clonación con aprendizaje por refuerzo y una representación relacional [11].

En experimentos más recientes [4], se utilizó un helicóptero miniatura XCell para aprender a seguir una trayectoria deseada proporcionando un pequeño número de demostraciones. En todas las pruebas realizadas, el helicóptero autónomo lo hizo mejor que el piloto.

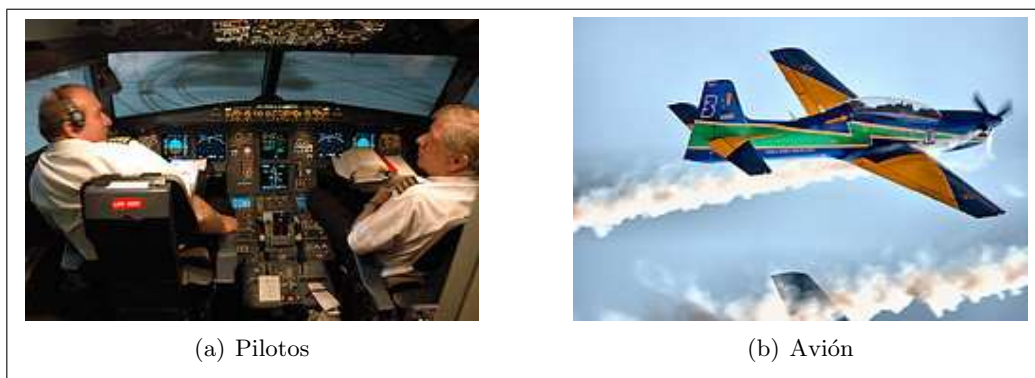


Figura 4: Avion y Pilotos



Figura 5: Helicóptero

Robosoccer. Esta aplicación [1] surgió motivada por RoboCup, una iniciativa cuyo objetivo es fomentar la investigación en inteligencia artificial. RoboCup usa el juego de soccer como dominio de prueba y aunque la idea es desarrollar un equipo de robots humanoides completamente autónomo que pueda ganar al equipo humano campeón del mundo, existen diversas categorías. Una de ellas es la liga en simulación que usa un software llamado Robosoccer. Los agentes de Robosoccer son generalmente programados manualmente por lo que el objetivo de esta aplicación es que los agentes aprendan de forma automática a jugar de forma similar a como lo hace un humano experto.

Para lograrlo, se hizo una interfaz para que una persona jugara Robosoccer como si fuera un video-juego. La persona jugó varios encuentros durante los cuales se registraron las entradas y las salidas. Las posibles acciones durante los encuentros eran: girar a la izquierda, correr rápido o lento, patear la pelota, patear a gol. La entrada es lo que la persona ve en el campo y la salida es la acción que la persona ejecutó en cada situación que se le presentó. Posteriormente, mediante algoritmos de aprendizaje se extrae un modelo, el cual es usado para controlar el agente. Los comportamientos que se aprendieron fueron: buscar la pelota, llevar la pelota hacia la meta y anotar en presencia de oponentes. Los agentes con el comportamiento clonado fueron capaces de enfrentar oponentes y anotar goles de forma parecida a la persona que jugó los encuentros de entrenamiento. Sin embargo, los autores del trabajo consideraron que el juego de los agentes era demasiado reactivo y no consideraba aspectos que el humano si toma en cuenta. Se pretende incorporar aprendizaje de mayor nivel como por ejemplo, pasar la pelota, empujar la pelota, driblear y poder modelar el aprendizaje a nivel de entrenador.

Robótica. Uno de los sueños de los humanos ha sido la creación robots dotados de inteligencia que faciliten la realización de tareas cotidianas. Ejemplos de este sueño abundan: Robotina, la empleada doméstica de los supersónicos; R2-D2 y C-3PO (la guerra de las galaxias); y por supuesto, Optimus Prime (Transformers) quien es la envidia de toda persona que trabaja o ha trabajado con robots.

La robótica presenta grandes retos y también grandes dificultades, empezando por la di-



Figura 6: Robosoccer

versidad de sensores que un robot puede tener y sus respectivas limitaciones. Por el contrario, los humanos tenemos un modelo estándar consistente en dos ojos, una nariz, una boca, un par de brazos y piernas que funcionan sin que nos pongamos a pensar cómo hacerlo. En este contexto, existen diversas técnicas de aprendizaje que se han aplicado a robots entre las cuales está también la clonación.

En la vida diaria, es común que una persona camine de un lugar a otro o busque algún objeto. Por esa razón, la navegación y la persecución de un objeto son habilidades robóticas sobre las que se ha hecho énfasis. En [6] se muestran experimentos en los cuales se guió a un robot con un joystick para que aprendiera a desplazarse sin chocar y a perseguir un vehículo de juguete color verde. Se registraron trazas consistentes en la información de los sensores (cámara y sonares) que se procesaron con algoritmos de inducción de árboles y redes neuronales. El robot capturó el estilo de control de la persona que lo guió, en este caso, aprendió dos estilos: temerario y lento/estable.

En México se han realizado trabajos sobre clonación para un robot de servicio. En el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) se desarrolla desde el 2006 una arquitectura para robots de servicio que ha sido probada en diversos concursos. El primer robot de prueba -Markovito- aprendió a navegar sin chocar, a salir de trampas y a seguir objetos móviles [16, 2] mediante clonación y programación lógica inductiva. En un trabajo más reciente, el robot aprendió a navegar y seguir un objeto combinando clonación con aprendizaje por refuerzo, generando políticas de acciones continuas [17].

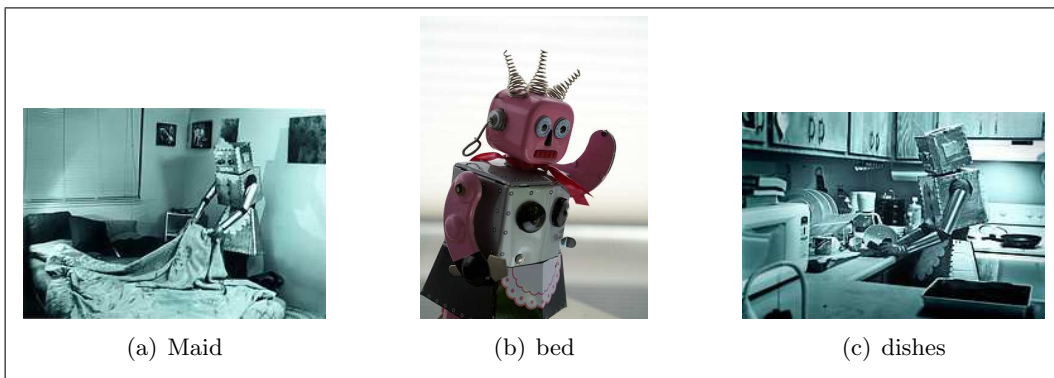


Figura 7: Robot de servicio

5. Beneficios y desventajas

Beneficios:

- La clonación tiene dos funciones: aprendizaje automático de controladores y descripción explícita de la estrategia seguida por el experto.
- No es necesario conocer aspectos internos del sistema para poder inducir un controlador. El usuario se enfoca en obtener los ejemplos que serán proporcionados como entrada. Este aspecto facilita la programación y reduce el tiempo para obtener resultados.
- El aprendizaje automático de sistemas de control ayuda a entender mejor las habilidades subcognitivas que son inaccesibles por introspección. Pueden descubrirse causas de fallas que a simple vista no se notan.
- En áreas como tutores inteligentes o entrenamiento, la clonación facilita la extracción de modelos del aprendiz para que puedan analizarse y mejorar el desempeño.
- Observar los resultados es divertido. Ver cómo el sistema hace las cosas de forma parecida a quien lo entrenó y estar a la expectativa de lo que hará a cada paso es emocionante.

Desventaja:

- Obtener ejemplos que sean útiles puede ser laborioso. No necesariamente lo que el humano observa es lo que los sensores perciben.

6. Distribución en el mundo

En el mapa se muestran los lugares donde han surgido trabajos representativos en clonación; no es un recuento exhaustivo. México figura entre los lugares donde se trabaja en esa área.

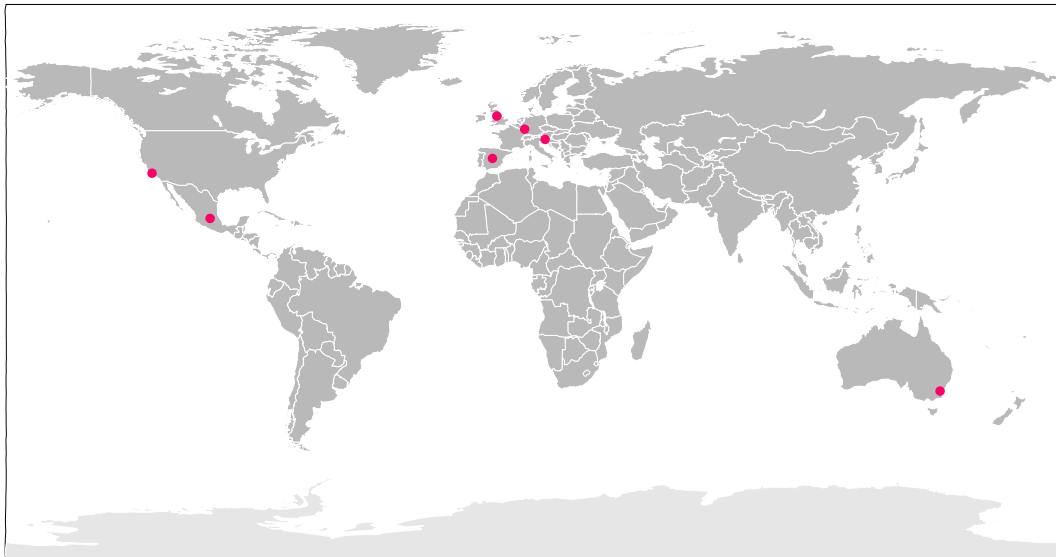


Figura 8: Clonación de comportamiento en el mundo

7. Conclusiones

Cuando Donald Michie decidió ir a inscribirse al curso de japonés nunca imaginó lo que estaba por iniciar. Desde ese momento, y gracias a una mala información, la historia del aprendizaje automático continuó escribiéndose de manera insospechada. La clonación de comportamiento ha sido una técnica que ha contribuido al avance de la inteligencia artificial y su desarrollo aun no termina. Falta mucho por hacer.

Referencias

- [1] Ricardo Aler, Jose M. Valls, David Camacho, y Alberto Lopez. Programming robosoccer agents by modeling human behavior. *Expert Systems with Applications*, 36(2):1850–1859, 2009.
- [2] Héctor Hugo Avilés, L. Enrique Sucar, Eduardo F. Morales, Blanca A. Vargas, Josué Sánchez, y Elva Corona. Markovito: A flexible and general service robot. En *Studies in Computational Intelligence*, volume 177, pages 401–423. Springer Berlin / Heidelberg, January 2009.
- [3] Ivan Bratko. Modelling operator’s skill by machine learning. *22nd International Conference Information Technology Interfaces*, pages 23–30, June 2000.
- [4] Adam Coates, Pieter Abbeel, y Andrew Y. Ng. Learning for control from multiple demonstrations. En *ICML ’08: Proceedings of the 25th international conference on Machine learning*, pages 144–151, New York, NY, USA, 2008. ACM.
- [5] B. Jack Copeland. *Colossus: The secrets of Bletchley Park’s code-breaking computers*. Oxford University Press, Inc., New York, NY, USA, 2010.
- [6] Claire D’Este, Mark O’Sullivan, y Nicholas Hannah. Behavioural cloning and robot control. En *Robotics and Applications*, pages 179–182, 2003.
- [7] Julio Freyre-Gonzalez y Enrique Sucar. Clonacion conductista de habilidades de control humano a maquina: un enfoque bayesiano. En *Proceedings of the IBERAMIA/SBIA 2000 Workshops*, pages 40–47, Sao Paulo, Brasil, 2000.
- [8] Andrew Isaac y Claude Sammut. Goal-directed learning to fly. En *In Proceedings of the Twentieth International Conference on Machine Learning*, pages 258–265. AAAI Press, 2003.
- [9] D. Michie y R. Camacho. Building symbolic representations of intuitive real-time skills from performance data. En *Machine Intelligence 13*, pages 385–418. Oxford University Press, 1994.
- [10] Donald Michie y Claude Sammut. Behavioural clones and cognitive skill models. *Machine Intelligence*, 14:395–404, 1995.
- [11] Eduardo F. Morales y Claude Sammut. Learning to fly by combining reinforcement learning with behavioural cloning. En *In ICML*, page 76. ACM Press, 2004.

- [12] Rafael Morales y Helen Pain. Modelling of novices' control skills with machine learning. En *In: UM99 – Proceedings of the Seventh International Conference on User Modeling*, pages 159–168. In: Kay, 1999.
- [13] J. R. Quinlan. Induction of decision trees. En Jude W. Shavlik y Thomas G. Dietterich, editors, *Readings in Machine Learning*. Morgan Kaufmann, 1990. Originally published in *Machine Learning* 1:81–106, 1986.
- [14] Claude Sammut, Scott Hurst, Dana Kedzier, y Donald Michie. Learning to fly. En *In Proceedings of the Ninth International Conference on Machine Learning*, pages 385–393. Morgan Kaufmann, 1992.
- [15] Dorian Suc y Ivan Bratko. Skill modelling through symbolic reconstruction of operator's trajectories. En *Automated systems based on human skill : preprints of the 6th IFAC Symposium*, pages 617–624, 1999.
- [16] Blanca Vargas y Eduardo F. Morales. Learning navigation teleo-reactive programs using behavioural cloning. *IEEE International Conference on Mechatronics (ICM)*, 2009.
- [17] Julio H. Zaragoza y Eduardo F. Morales. Relational reinforcement learning with continuous actions by combining behavioural cloning and locally weighted regression. *Journal of Intelligent Learning Systems and Applications*, 2:69–79, 2010.