Computación Blanda

**Soft Computing**

Autor: José Gilberto Vargas Cano

*IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia*

Correo-e: gilberto@utp.edu.co

***Resumen*— Este documento presenta un resumen de las líneas clásicas de la Computación Blanda: redes neuronales, lógica difusa, sistemas expertos, algoritmos genéticos y machine learning. El objetivo del documento es brindar una panorámica general de las temáticas, mostrando su relación con las técnicas de inteligencia artificial. La diferencia entre el paradigma de Inteligencia Artificial y la computación blanda está centrada en el mecanismo de inferencia utilizado y su aplicación a la solución de problemas tomados de lo cotidiano, de las teorías de conocimiento y de su relación con ciencias afines.**

***Palabras clave—* sistemas, redes, inteligencia artificial, software, computación, investigación, industria, genético, aprendizaje.**

***Abstract*— This document presents a summary of the classic lines of Soft Computing: neural networks, fuzzy logic, expert systems, genetic algorithms and machine learning. The objective of the document is to provide a general overview of the topics, showing their relationship with artificial intelligence techniques. The difference between the Artificial Intelligence paradigm and soft computing is centered on the inference mechanism used and its application to the solution of problems taken from everyday life, from knowledge theories and their relationship with related sciences.**

***Key Word*— systems, networks, artificial intelligence, software, computing, research, industry, genetic, learning.**

1. **INTRODUCCIÓN**

La temática de la Computación Blanda se encuentra enmarcada en el paradigma de la Inteligencia Artificial. La diferencia con dicho paradigma radica en que la Computación Blanda está centrada en la aplicación pragmática de las teorías de la Inteligencia Artificial a la solución de problemas complejos en diversos campos del conocimiento.

Las líneas derivadas de la Computación Blanda, se configuran en las siguientes tendencias: a) Redes Neuronales Artificiales, b) Lógica Difusa, c) Sistemas Expertos, d) Algoritmos Genéticos, e) Deep Learning (Machine Learning).

En los siguientes apartados se presenta un resumen de dichas tendencias.

* 1. **REDES NEURONALES**

Las Redes Neuronales Artificiales (RNA) están inspiradas en la biología, esto significa que están formadas por elementos que se comportan de manera análoga a las neuronas (en las funciones más elementales) y están organizadas de una forma similar a la del cerebro, pero las analogías no son muchas más. Las características fundamentales de las RNA son:

• Aprenden de la experiencia: Las RNA pueden modificar su comportamiento como respuesta a su entorno. Dado un conjunto de entradas (quizá con las salidas deseadas), las RNA se ajustan para producir respuestas consistentes. Una amplia variedad de algoritmos de entrenamiento se han desarrollado, cada uno con sus propias ventajas e inconvenientes.

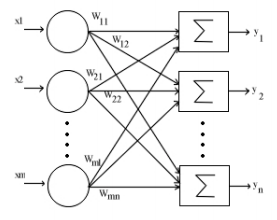
• Generalizan de ejemplos anteriores a los ejemplos nuevos: Una vez que la RNA esté entrenada, la respuesta de la red puede ser, hasta un cierto punto, insensible a pequeñas variaciones en las entradas, lo que las hace idóneas para el reconocimiento de patrones.

• Abstracción de la esencia de las entradas: Algunas RNA son capaces de abstraer información de un conjunto de entradas. Por ejemplo, en el caso de reconocimiento de patrones, una red puede ser entrenada en una secuencia de patrones distorsionados de una letra. Una vez que la red sea correctamente entrenada será capaz de producir un resultado correcto ante una entrada distorsionada, lo que significa que ha sido capaz de aprender algo que nunca había visto.

**Redes de capa simple**

A pesar de que una sola neurona puede realizar modelos simples de funciones, su mayor productividad viene dada cuando se organizan en redes. La red más simple es la formada por un conjunto de perceptrones a los que entra un patrón de entradas y proporcionan la salida correspondiente. Por cada perceptrón que tengamos en la red vamos a tener una salida, que se hallará como se hacía con un perceptrón solo, haciendo el sumatorio de todas las entradas multiplicadas por los pesos. Al representar gráficamente una red, se añade una "capa" inicial que no es contabilizada a efectos de computación, solamente sirve para distribuir las entradas entre los perceptrones. La denominaremos la capa 0.

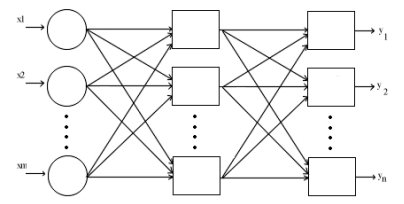
De esta manera, la representación gráfica de una red de capa simple sería la siguiente:



**Redes multicapa**

Las redes multicapa se forman por un conjunto de redes de capa simple en cascada unidas por pesos, donde la salida de una capa es la entrada de la siguiente capa. Generalmente son capaces de aprender funciones que una red de capa simple no puede aprender, por lo que ofrecen mejores capacidades computacionales. Para que este incremento en poder computacional sea tal, tiene que existir una función de activación no lineal entre las capas, por lo que generalmente se utilizará una función de activación sigmoidea en detrimento de la lineal o umbral.

Para calcular la salida de una red multicapa se debe hacer de la misma manera que en las redes de capa simple, teniendo en cuenta que las salidas de una capa son las entradas de la siguiente capa.:

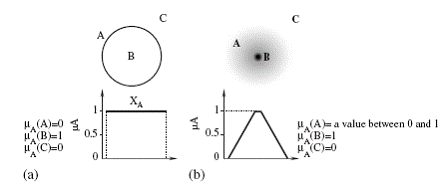


**Redes recurrentes:** Las redes consideradas hasta ahora no tienen conexiones entre pesos de la salida de una capa a la entrada de la misma capa o anteriores. Las redes que poseen esta característica son conocidas como redes recurrentes. Las redes recurrentes no tienen memoria, es decir, la salida solamente está determinada por las entradas y los pesos. Las capas recurrentes redireccionan previas salidas a entradas. Su salida es determinada por su entrada y sus salidas previas, por lo que se puede asemejar a la memoria a corto plazo de los seres humanos.

* 1. **LÓGICA DIFUSA**

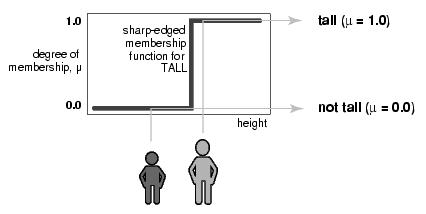
La forma en que la gente piensa es, inherentemente, difusa. La forma en que percibimos el mundo está cambiando continuamente y no siempre se puede definir en términos de sentencias verdaderas o falsas. Consideremos como ejemplo el conjunto de vasos del mundo, que pueden estar vacíos o llenos de agua. Ahora tomemos un vaso vacío y comencemos a echar agua poco a poco, ¿en qué momento decidimos que el vaso pasa de estar vacío a estar lleno? Evidentemente, hay dos situaciones extremas que reconocemos sin ninguna duda, la primera cuando el vaso está completamente vacío, sin una sola gota de agua en su interior, y la segunda cuando está completamente lleno, cuando no cabe ni una sola gota más en él, pero una gota antes de estar completamente lleno, ¿diríamos que es falso que el vaso está lleno?, observa que para afirmar su condición, en la frase anterior no solo he usado el término **lleno**, sino que he añadido un modificador diciendo **completamente lleno**. Si a un vaso lleno de agua le quito una gota de agua... ¿dejo de considerarlo lleno y automáticamente pertenece para mí a otra clasificación?

Las definiciones de **vaso completamente vacío** y **vaso completamente lleno** son demasiado estrictas como para que resulten interesantes en un razonamiento en el que se consideran operaciones de llenado y vaciado de vasos, y entre los términos de **lleno** y **vacío** hay un área que no está claramente definida de pertenencia a ninguno de esos extremos. En el lenguaje natural que usamos en el mundo real hemos cubierto esta imprecisión por medio de una jerarquía de términos intermedios junto con modificadores que permiten cubrir un espectro más grande de áreas usando un número limitado de ellos, y podemos hablar de lleno, medio lleno, completamente lleno, casi lleno, etc. Matemáticamente, los conceptos de **sí/no**, **verdadero/falso** están representados por medio del concepto clásico de conjunto, pero necesitamos extenderlo para poder representar este tipo de información más difusa.

Un conjunto difuso permite a sus elementos tener un grado de pertenencia. Si el valor 1 se asigna a los elementos que están completamente en el conjunto, y 0 a los que están completamente fuera, entonces los objetos que están parcialmente en el conjunto tendrán un valor de pertenencia estrictamente entre 0 y 1. Por tanto, si un vaso completamente lleno tiene un grado de pertenencia a **los vasos llenos** de valor 1, y un vaso completamente vacío un grado de pertenencia a **los vasos llenos**de valor 0, entonces al añadir una gota a este último, su grado de pertenencia **a los vasos llenos** sería ligeramente superior a 0.

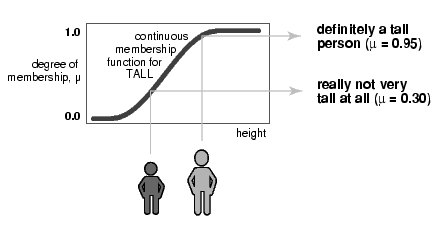
**Definición Matemática de Conjuntos Difusos**

Matemáticamente, un conjunto es una colección de objetos que verifican alguna propiedad, de forma que un objeto o bien pertenece al conjunto, o no pertenece. Por ejemplo, supongamos que decimos que una persona es alta si su altura está por encima de 180cm, algo que puede ser representado gráficamente de la siguiente forma:



La función anterior describe la pertenencia al **conjunto de los altos**, o estás dentro o estás fuera. Estas funciones **nítidas** funcionan muy bien con las operaciones matemáticas clásicas, pero no funcionan tan bien describiendo el mundo real. Por una parte, no hace distinción entre individuos que midan 181cm y los que miden 215cm, aunque hay una clara distinción entre ellos. El otro problema es la diferencia entre una persona que mida 180cm y otra de 181cm, apenas 1cm de diferencia entre ellos y el primero de ellos no está en el conjunto de los altos, y el segundo sí.

La aproximación de los conjuntos difusos al conjunto de los altos proporciona una representación mucho mejor sobre la propiedad **ser alto** de una persona. El conjunto se define por medio de una función continua que puede tomar valores intermedios entre los extremos 0 y 1.



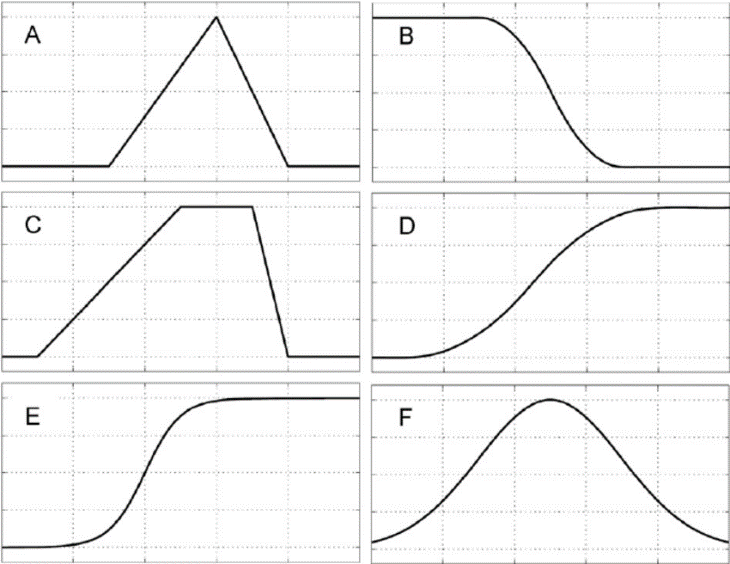
La función de pertenencia define el conjunto difuso para los posibles valores de altura (mostrados en el eje horizontal) y proporciona el grado de pertenencia de la altura al conjunto difuso (mostrado en el eje vertical con valores entre 0 y 1). De esta forma, el grado de pertenencia de la primera persona es 0.3 y, por tanto, no es muy alto, mientras que el segundo tiene un grado de pertenencia de 0.95 y, definitivamente, es alto.

Los **conjuntos difusos** fueron propuestos inicialmente por Lofti A. Zadeh en su artículo de 1965 titulado *"Fuzzy Sets"*. Este artículo establece los fundamentos de la lógica difusa que se deduce de la definición de conjunto difuso y sus propiedades. Esta definición es:

*Sea*XX*un conjunto clásico. Un conjunto difuso,*AA*, en*XX*viene caracterizado por la función de pertenencia*fA(x)fA(x)*, que asocia a cada punto*x∈Xx∈X*un número real del intervalo*[0,1][0,1]*, donde los valores de*fA(x)fA(x)*representan el "grado de pertenencia" de*xx*en*AA*, de forma que, cuanto más cerca esté el valor de*fA(x)fA(x)*a*11*, mayor es el grado de pertenencia de*xx*a*AA*.*

Esta definición de conjunto difuso extiende de alguna forma la definición clásica de conjunto, que sería el caso particular en el que fA(x)∈{0,1}fA(x)∈{0,1}. Pero para trabajar con ellos también tendremos que extender las operaciones clásicas entre conjuntos (unión, intersección, etc.), lo que nos llevará a extender las operaciones lógicas binarias habituales (conjunción, disyunción, etc.).

Obsérvese que no indicamos cómo ha de ser la función de pertenencia, ya que eso dependerá de las características propias del conjunto real que se quiere representar, pero suelen usarse algunas funciones clásicas comunes como las que se muestran a continuación:



* 1. **SISTEMAS EXPERTOS**

Típicamente, un sistema experto se basa en un amplio cuerpo de conocimientos sobre un área problemática específica. En general, este conocimiento se organiza como un conjunto de reglas que permita que el sistema extraiga conclusiones a partir de datos o premisas dadas, capacitándolo, en consecuencia, para ofrecer un consejo inteligente o tomar decisiones inteligentes. Este enfoque del diseño de sistemas basado en el conocimiento representa un cambio evolutivo en la ciencia de las computadoras, con consecuencias revolucionarias. Sustituye a la tradicional fórmula de:

DATOS + ALGORITMO = PROGRAMA

Por una nueva arquitectura centrada alrededor de una base de conocimientos y un motor de inferencias, de modo que: CONOCIMIENTOS + INFERENCIAS =SISTEMA EXPERTO

Esta fórmula es, obviamente similar, pero con un enfoque lo suficientemente diferente como para tener profundas implicaciones. Para tener un sistema experto resulta útil la siguiente lista de verificación de características típicas:

— Un sistema experto se limita a un campo de pericia relativamente delimitado.

— Debe ser capaz de razonar con datos inciertos y reglas no fiables.

— Debe ser capaz de explicar su cadena de razonamiento de una forma exhaustiva.

— Los hechos y los mecanismos de inferencia son "separables".

— El conocimiento no está codificado como parte de los procedimientos deductivos.

— Está diseñado para crecer por incrementos.

— Típicamente se basa en reglas.

— Su salida es un consejo o sugerencia, no tablas de cifras ni gráficos. La palabra clave es "conocimiento". Está claro que el objetivo de un sistema inteligente para resolver problemas es omitir la búsqueda ciega o aleatoria. Para hacerlo, un sistema de computadora ha de explotar la misma ventaja que tiene el experto humano en relación al novato, es decir, la pericia o el conocimiento organizado: el conocimiento de hechos, de reglas de inferencia y de estrategias de solución.

Un sistema experto totalmente viable tiene cuatro componentes esenciales:

1. La base de conocimientos.

2. El motor de inferencia.

3. El módulo de adquisición de conocimientos.

4. La interfaz explicativa. Los cuatro módulos son críticos. Un sistema basado en el conocimiento puede carecer de alguno de ellos, pero un auténtico sistema experto no puede carecer de ninguno. Hablaremos de esos cuatro módulos de uno en uno y explicaremos cómo trabajan todos juntos.

* 1. **ALGORITMOS GENÉTICOS**

El algoritmo genético es una técnica de búsqueda basada en la teoría de la evolución de Darwin, que ha cobrado tremenda popularidad en todo el mundo durante los últimos años. Se presentarán aquí los conceptos básicos que se requieren para abordarla, así como unos sencillos ejemplos que permitan a los lectores comprender cómo aplicarla al problema de su elección.

En los últimos años, la comunidad científica internacional ha mostrado un creciente interés en una nueva técnica de búsqueda basada en la teoría de la evolución y que se conoce como el algoritmo genético. Esta técnica se basa en los mecanismos de selección que utiliza la naturaleza, de acuerdo a los cuales los individuos más aptos de una población son los que sobreviven, al adaptarse más fácilmente a los cambios que se producen en su entorno. Hoy en día se sabe que estos cambios se efectúan en los genes de un individuo (unidad básica de codificación de cada uno de los atributos de un ser vivo), y que sus atributos más deseables (i.e., los que le permiten adaptarse mejor a su entorno) se transmiten a sus descendientes cuando éste se reproduce sexualmente.

Un investigador de la Universidad de Michigan llamado John Holland era consciente de la importancia de la selección natural, y a fines de los 60s desarrolló una técnica que permitió incorporarla a un programa. Su objetivo era lograr que las computadoras aprendieran por sí mismas. A la técnica que inventó Holland se le llamó originalmente "planes reproductivos", pero se hizo popular bajo el nombre "algoritmo genético" tras la publicación de su libro en 1975.

Una definición bastante completa de un algoritmo genético es la propuesta por John Koza:

"Es un algoritmo matemático altamente paralelo que transforma un conjunto de objetos matemáticos individuales con respecto al tiempo usando operaciones modeladas de acuerdo al principio Darwiniano de reproducción y supervivencia del más apto, y tras haberse presentado de forma natural una serie de operaciones genéticas de entre las que destaca la recombinación sexual. Cada uno de estos objetos matemáticos suele ser una cadena de caracteres (letras o números) de longitud fija que se ajusta al modelo de las cadenas de cromosomas, y se les asocia con una cierta función matemática que refleja su aptitud. "

**Definición**

Los Algoritmos Genéticos (AGs) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acorde con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin. Por imitación de este proceso, los Algoritmos Genéticos son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de las mismas.

Un algoritmo genético consiste en una función matemática o una rutina de software que toma como entradas a los ejemplares y retorna como salidas cuales de ellos deben generar descendencia para la nueva generación.

Versiones más complejas de algoritmos genéticos generan un ciclo iterativo que directamente toma a la especie (el total de los ejemplares) y crea una nueva generación que reemplaza a la antigua una cantidad de veces determinada por su propio diseño. Una de sus características principales es la de ir perfeccionando su propia heurística en el proceso de ejecución, por lo que no requiere largos períodos de entrenamiento especializado por parte del ser humano, principal defecto de otros métodos para solucionar problemas, como los Sistemas Expertos.

**Problemática**

Los principios básicos de los Algoritmos Genéticos fueron establecidos por Holland, y se encuentran bien descritos en varios textos . Goldberg, Davis, Michalewicz, Reeves.

En la naturaleza los individuos de una población compiten entre sí en la búsqueda de recursos tales como comida, agua y refugio. Incluso los miembros de una misma especie compiten a menudo en la búsqueda de un compañero. Aquellos individuos que tienen más éxito en sobrevivir y en atraer compañeros tienen mayor probabilidad de generar un gran número de descendientes. Por el contrario individuos poco dotados producirán un menor número de descendientes. Esto significa que los genes de los individuos mejor adaptados se propagarán en sucesivas generaciones hacia un número de individuos creciente. La combinación de buenas características provenientes de diferentes ancestros, puede a veces producir descendientes "superindividuos", cuya adaptación es mucho mayor que la de cualquiera de sus ancestros. De esta manera, las especies evolucionan logrando unas características cada vez mejor adaptadas al entorno en el que viven.

Los Algoritmos Genéticos usan una analogía directa con el comportamiento natural. Trabajan con una población de individuos, cada uno de los cuales representa una solución factible a un problema dado. A cada individuo se le asigna un valor ó puntuación, relacionado con la bondad de dicha solución. En la naturaleza esto equivaldría al grado de efectividad de un organismo para competir por unos determinados recursos. Cuanto mayor sea la adaptación de un individuo al problema, mayor será la probabilidad de que el mismo sea seleccionado para reproducirse, cruzando su material genético con otro individuo seleccionado de igual forma. Este cruce producirá nuevos individuos . descendientes de los anteriores . los cuales comparten algunas de las características de sus padres. Cuanto menor sea la adaptación de un individuo, menor será la probabilidad de que dicho individuo sea seleccionado para la reproducción, y por tanto de que su material genético se propague en sucesivas generaciones.

De esta manera se produce una nueva población de posibles soluciones, la cual reemplaza a la anterior y verifica la interesante propiedad de que contiene una mayor proporción de buenas características en comparación con la población anterior. Así a lo largo de las generaciones las buenas características se propagan a través de la población. Favoreciendo el cruce de los individuos mejor adaptados, van siendo exploradas las áreas más prometedoras del espacio de búsqueda. Si el Algoritmo Genético ha sido bien diseñado, la, población convergerá hacia una solución óptima del problema.

**Ventajas y Desventajas**

* No necesitan conocimientos específicos sobre el problema que intentan resolver.
* Operan de forma simultánea con varias soluciones, en vez de trabajar de forma secuencial como las técnicas tradicionales.
* Cuando se usan para problemas de optimización maximizar una función objetivo- resultan menos afectados por los máximos locales (falsas soluciones) que las técnicas tradicionales.
* Resulta sumamente fácil ejecutarlos en las modernas arquitecturas masivamente paralelas.
* Usan operadores probabilísticos, en vez de los típicos operadores determinísticos de las otras técnicas.
* Pueden tardar mucho en converger, o no converger en absoluto, dependiendo en cierta medida de los parámetros que se utilicen tamaño de la población, número de generaciones, etc.-.
* Pueden converger prematuramente debido a una serie de problemas de diversa índole.

**Limitaciones**

El poder de los Algoritmos Genéticos proviene del hecho de que se trata de una técnica robusta, y pueden tratar con éxito una gran variedad de problemas provenientes de diferentes áreas, incluyendo aquellos en los que otros métodos encuentran dificultades. Si bien no se garantiza que el Algoritmo Genético encuentre la solución óptima, del problema, existe evidencia empírica de que se encuentran soluciones de un nivel aceptable, en un tiempo competitivo con el resto de algoritmos de optimización combinatoria. En el caso de que existan técnicas especializadas para resolver un determinado problema, lo más probable es que superen al Algoritmo Genético, tanto en rapidez como en eficacia. El gran campo de aplicación de los Algoritmos Genéticos se relaciona con aquellos problemas para los cuales no existen técnicas especializadas. Incluso en el caso en que dichas técnicas existan, y funcionen bien, pueden efectuarse mejoras de las mismas hibridándolas con los Algoritmos Genéticos.

* 1. **DEEP LEARNING**

Deep Learning o aprendizaje profundo es una técnica dentro del machine learning basado en arquitecturas neuronales. Un modelo basado en deep learning puede aprender a realizar tareas de clasificación directamente a partir de imágenes, texto o sonido, etc. Sin necesidad de intervención humana para la selección de características. Esto se puede considera la principal ventaja del deep learning, llamada “feature discovering”. Pueden, además, poseer una precisión que supera a la capacidad del ser humano.

El aprendizaje profundo es un subconjunto de aprendizaje automático en Inteligencia Artificial (AI) que tiene redes capaces de aprender sin supervisión a partir de datos que no están estructurados ni etiquetados. También conocido como Deep Neural Learning o Deep Neural Network. Aquí es donde reside la gran diferencia respecto al Machine Learning.

Las Redes Neuronales Artificiales se inspiran en nuestra comprensión de la biología de nuestros cerebros, todas esas interconexiones entre neuronas. Pero, a diferencia de un cerebro biológico donde cualquier neurona se puede conectar a cualquier otra neurona dentro de una cierta distancia física, estas redes neuronales artificiales tienen capas discretas, conexiones y direcciones de propagación de datos.

Pueden, por ejemplo, tomar una imagen, cortarla en un grupo de teselas que se ingresan en la primera capa de la red neuronal. En la primera capa, las neuronas individuales pasan los datos a una segunda capa. La segunda capa de neuronas hace su tarea, y así sucesivamente, hasta que se produce la última capa y producción final. Cada neurona asigna una ponderación a su entrada: qué tan correcta o incorrecta es en relación con la tarea que se realiza. El resultado final se determina luego por el total de esas ponderaciones.

Por ejemplo, imaginemos de nuevo la señal de Stop. Los atributos de una imagen de una señal de stop son cortados y «examinados» por las neuronas: su forma octogonal, el color rojo de su motor de bomberos, sus letras distintivas, el tamaño de su señal de tráfico y su movimiento o falta de ella. La tarea de la red neuronal es concluir si esto es una señal de Stop o no. Se trata de un «vector de probabilidad», realmente una suposición altamente educada, basada en la ponderación.

El Deep Learning ha permitido muchas aplicaciones prácticas de Machine Learning y, por extensión, el campo general de Inteligencia Artifical. El aprendizaje profundo desglosa las tareas de manera que hace que todo tipo de asistencia en máquinas parezca posible, incluso probable. Los automóviles sin conductor, una mejor atención médica preventiva, incluso mejores recomendaciones de películas, son claros ejemplos de aplicación. La Inteligencia es el presente y el futuro. Con la ayuda de Deep Learning, la Inteligencia Artificial puede llegar a ese estado de ciencia ficción que tanto tiempo hemos imaginado con Star Wars y Terminator.

En la última década, las empresas están utilizando el aprendizaje profundo para resolver los desafíos a nivel empresarial. Desde la detección de rostros (Face ID) hasta recomendaciones de productos, segmentación de clientes, reorganización de dígitos, traducción automática, inteligencia de negocios, Internet de las cosas, seguridad de redes, etc. El uso del deep learning y machine learning han transformado por completo el mundo en el que vivimos hoy.

REFERENCIAS

Referencias en la Web:

[1]

<https://computerhoy.com/reportajes/tecnologia/inteligencia-artificial-469917>

[2]

<http://www2.ulpgc.es/hege/almacen/download/38/38584/practica_ia_2.pdf>

[3]

<https://sites.google.com/site/inteligenciascarol/algoritmo-genetico>

[4]

<https://itelligent.es/es/diferencias-entre-deep-learning-machine-learning-inteligencia-artificial/#:~:text=El%20aprendizaje%20profundo%20es%20un,Learning%20o%20Deep%20Neural%20Network.>