21-5-2020

Tarea 3 Máquina de Turing

Martínez Coronel Brayan Yosafat

2CM3

Introducción

Como hemos visto a lo largo de la historia, grandes personajes han dejado aportaciones sumamente grandes para la sociedad, y, en especial, para nuestra ciencia, Alan Turing realizó grandes aportes en diversos campos, pero, hoy, hablaremos de su paper "*On Computable Numbers, with an Application to the Entscheidungsproblem",* pero, ¿Qué es el problema de decisión o *Entscheidungsproblem*?, bueno, debemos mencionar que el gran Hilbert, preguntado si una fórmula bien formada dada, es o no un teorema. Un problema que causaba demasiadas expectativas, ya que si tenía resultados positivos los matemáticos podían utilizar el cálculo recursivo para determinarlo en cosas de primer orden.

Sin embargo, de manera independiente, Alonzo Church y Alan Turing, uno con su cálculo Lambda y el otro con la Máquina de Turing determinaron que es recursivamente irresoluble, terminando 8 años de ardua investigación en la comunidad matemática (que en realidad fueron muchos más, porque el problema se remonta a tiempos de Leibniz, 200 años antes de que Hilbert lo formalizara).

Desarrollo

Aunque Alonzo también lo resolvió en 1936, los métodos son bastante diferentes, lo que dio mucha más credibilidad a ambos. Por efectos de tiempo, sólo hablaremos de Turing, en específico la máquina que lleva su nombre; al igual que Turing, es necesario hacer muchas aclaraciones y definiciones de su propio paper, ya que introduce muchas notaciones, que ciertamente son muy interesantes, pero sólo tomaremos lo necesario.

Turing nos habla de que una máquina circular, será aquella que pueda moverse a la derecha o a la izquierda y tenga la libertad de escribir un símbolo, en este caso, ese símbolo es 1 o 0, la máquina puede realizar un conjunto de comandos: L, se mueve a la izquierda, R, se mueve a la derecha, E, borra el símbolo que mira, Px, imprime el símbolo x, donde x pertenece al conjunto de símbolos permitidos en la máquina.

Sin embargo, elementos más ‘tangibles’ (en sentido figurado) son que la máquina tiene un observador, que tiene la capacidad (evidente) de observar una celda, esta celda pertenece a una cinta que contiene arreglos de celdas. Al tener una serie de comandos le llamamos a esto una configuración m, que básicamente, es ponerle nombre a esa secuencia en especial, esto abrevia demasiado todo el proceso de la Máquina de Turing, pero, para abreviarlo todavía más, Turing utiliza funciones, para poder pasar parámetros y poder llamar a otras funciones como parámetro de otra función.

Mediante una tabla, que Turing separa en dos partes, la configuración y el comportamiento, en la primera denotamos el nombre de una configuración, y en la segunda lo que hace y a qué configuración llama. Pero, de nuevo, hace otra abreviación y vuelve a renombrar las funciones como una letra, para luego también nombrarla en forma de números arábigos, que al escribirla nos da la forma estándar y la forma de números descriptores.

Después de todas estas definiciones, comienza con una explicación importante que las secuencias computables no son numerables, primero mediante el proceso diagonal, que toma un elemento enésimo y **toma por hecho que este elemento debe ser computable**, y luego toma el elemento siguiente, y dice que debe existir un número K tal que pase del n al n+1, pero para esto resulta que 1 debe ser par, lo cual es imposible, pero, Turing mismo dice que esto no es realmente aceptable, porque justamente se toma que el elemento n es computable, por lo que toma otro camino.

De hecho, al usar sus propias máquinas, en realidad está diciendo si tiene la capacidad de construir ese número, o sea, más allá de que el número puede ser numerable, desea saber si la máquina es capaz de tener un conjunto configuraciones que puedan justamente numerarla, pero, o sea, que si dadas unas instrucciones de una fórmula que se puede probar encontremos si es verdadera o falsa, aquí podemos decir que se parece a algo que se mencionó al comienzo.

De hecho, mediante este resultado, Turing continúa con el uso del cálculo funcional, y los resultados presentados por Gödel, además de los del propio Hilbert y Ackermann, desarrolla fórmulas bastante largas (que, de hecho, están abreviadas) mediante el uso de dos lemas, que usa una función hipotética, primero busca que sea probable sabiendo que el símbolo S1 aparece y luego por el opuesto de esto, que si la función se sabe es probable, entonces encontrar el símbolo S1. Pero, esto lleva a que el problema no puede ser solucionado.

Conclusiones

Pero ¿por qué es tan importante que este problema tuviera solución?, bueno estamos hablando de tener un algoritmo que pueda determinar si una fórmula dada tiene falsedad o es verdadera, es una máquina que podría determinar si son verdaderos los teoremas, siempre y cuando sean bien formulados. Sin embargo, Turing y Church encuentran que esto no es posible, un trabajo increíble por parte de ambos, de manera independiente, que nos demuestra (lamentablemente, porque de ser el contrario podríamos hacer grandes cosas) que esto no es posible. Pero, de manera colateral, resulta ser que la máquina de Turing describe de una manera muy formal el comportamiento de las computadoras actuales, estamos hablando de que Turing creó un sistema bastante formal para lograr grandes cosas totalmente abstractas y que pueden pasar a un proceso mecánico. Es decir, a 16 años de que cumpla un siglo su trabajo, estamos usando todavía el trabajo de Turing en un ramo que ya tiene independencia.

Prueba de Turing

Esta prueba consiste en preguntarnos si una computadora imaginable será capaz de realizar imitaciones al nivel que un evaluador no pueda determinar si se trata de una persona o una máquina. Para ejemplificar, imaginemos que estamos en una conversación en una aplicación de red social, como Messenger o WhatsApp, digamos que se encuentran 3 integrantes en cierta conversación, uno de ellos somos nosotros mismos, siendo el evaluador, mientras que los demás, uno es un humano y el otro un programa que puede realizar respuestas a preguntas genéricas. Al conversar, (suponiendo que el programa está bastante entrenado) resulta que podemos tener una conversación igual tanto como el humano que la computadora, en términos generales, de esto trata la prueba.

Tesis Church-Turing

Como hemos mencionado, Alonzo Church también realizó un trabajo, de hecho, antes que Turing, pero por un método bastante diferente: el cálculo Lambda; y como Turing lo menciona "el trabajo de Church resulta como un equivalente del presente, siendo que él usó el cálculo Lambda y yo las máquinas circulares libres", de hecho, eso es la tesis Church-Turing, que una función computable (que pertenece al trabajo de Church) es **equiValente** a una máquina de Turing. Esto es demasiado importante para nuestro ramo, porque estamos hablando de que "Todo algoritmo es equivalente a una Máquina de Turing", y aunque no es un teorema que se pueda demostrar formalmente, es aceptada prácticamente de manera universal (pero, no deja de ser Tesis).