Lecture 4: Short Description of Big Data

Xiaotie Deng

AIMS Lab
Department of Computer Science
Shanghai Jiaotong University

October 16, 2017

1 Kolmogorovcomplexity

Kolmogorovcomplexity

Universal Turing Machine

- A Turing Machine has a transitional map for (state,symbol) to (state, symbol-written, move) with a head which reads/writes on/to the current tape.
- Universal Turing Machine
    - INPUT tape: program and input data
    - a standard set of operation rules.
    - Output: written on the tape.

Universal Turing Machine

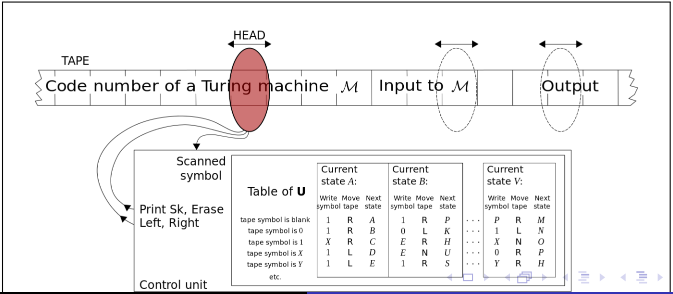
HEAD
TAPE
Code number of a Turing machine $\mathcal{M}$
Input to $\mathcal{M}$
Output
Scanned symbol
Print Sk, Erase Left, Right
Control unit

Table of **U**	Current state A:			Current state B:				Current state V:		
	Write symbol	Move tape	Next state	Write symbol	Move tape	Next state		Write symbol	Move tape	Next state
tape symbol is blank	1	R	A	1	R	P	. . .	P	R	M
tape symbol is 0	1	R	B	0	L	K	. . .	1	L	N
tape symbol is 1	X	R	C	E	R	H	. . .	X	N	O
tape symbol is X	1	L	D	E	N	U	. . .	0	R	P
tape symbol is Y	1	L	E	1	R	S	. . .	Y	R	H
etc.										

UTM Description of a datum x

- Input y and Output x.
  - Let $x = T(p, y)$ be the output of the UTM on program p and data y.
- Kolmogorov complexity
  - The shortest such $p = H(x|y)$ is called the conditional complexity of x with respect to y.
  - $H(x) = H(x|\emptyset)$ is called the complexity of x, denoted by x^* here.
- Invariance Theorem: The Kolmogorov complexity is independent of the Universal Turing Machine we use, up to an additive constant.
- Reference: (https://cs.uwaterloo.ca/ mli/cs882-kc.html)

Fundamentals of Kolmogorov complexity

- Invariance: Given any description language L, the optimal description language is at least as efficient as L, with some constant overhead.
- Key idea of the proof:
  - The turing machine is written in a constant size program not related to the input size.
  - However, program size may dependent to the input size in general for networked computers.
- Unboundable Kolmogorov complexity: $\forall n \exists x : K(x) \geq n$: otherwise, $\exists n \textit{ such that } \forall x \; K(x) < n$. There are an infinite number of such strings. But this contradicts the fact that we only have a finite number of programs with a size less than n. They can generate only a finite number of strings.

Kolmogorov complexity is not computable

- Suppose it is computable by ComputeKolmComplexity(s) with a 1M bytes program.
- Create the following program: Compute a string
  - while $i > 0$ do for each string $s : |s| = i + +$ run
  - if ComputeKolmComplexity(s) $>$ 2M bytes
  - return s.
- The program also outputs something as $K(x)$ is unbounded.
- s is output by the above program of length no more than $1M + 1000$ bytes.
- but the program outputs s only if its requires $>$ 2M bytes by any program.
- A contradiction.

Assignment II (first part)

Do three problems from the followings: Compute the Kolmogorov Complexities of the following numbers: more specifically, compute its n-th bid for all n.

- $H(1/3)$,
- $H(\pi)$,
- $H(e)$,
- $H(r)$: r is the foot of the equation $x^5 - 5x^2 + 1$,
- (a, b) where $x = a$ and $y = b$ are the root for the set of simultaneous equations: $x^3 + x * y - 5$ and $x + 4x^2 * y + y^3 - 10 = 0$.
- Prove that for any number x, $H(x)$ always exists.
- Give an upper bound on $H(x)$.