Algorithmische Kolmogorovkomplexität und Invarianz

Mario Frank
Seminar Kolmogorovkomplexität

Universität Potsdam

November 13, 2009

Allgemeines

Definition

Der Informationsgehalt eines Objektes ist die Anzahl der Bits, die notwendig sind, um es verlustfrei zu beschreiben.
Der Informationsgehalt ist auch davon abhängig, welche Darstellung man verwendet.

Beispiel

Zahl 101010101010101010 $\triangleq$ *18 Bits*
Oder: $10^9 \triangleq$ *(10)*1001_2
,$h("1") = 11, h("0") = 00, h("(") = 10, h(")") = 01$
$\rightarrow 1011000111000011_2$, *16 Bit*
$\rightarrow$ *anderer Informationsgehalt.*

Universelle Beschreibungsmethode

S-Menge von Objekten,
n(x)-Standardaufzählung mit n $=$ f(p)
f-Beschreibungsmethode(partielle, rekursive Funktion)
Es gilt:
$\forall x \in S(C_f(x) = min\{l(p) : f(p) = n(x)\})$
$C_f(x) = \infty$, wenn es kein solches p gibt.

Betrachte:
Menge paarweise disjunkter Methoden $\{f_1, f_2, ..., f_r\}$
$\exists f \forall x \in S(C_f(x) = min\{C_{f_1}(x), C_{f_2}(x), ..., C_{f_r}(x)\}) + c$ mit
$c \leq log(r)$
c Bits von p - Identifikation von f_i

Bemerkung

$\exists c \forall x \in S(C_f(x) \leq C_g(x) + c) \Leftrightarrow f$ *minorisiert* g
(untere Abschätzung)
Hat eine Menge von Beschreibungsmethoden ein minimales Element, dann ist dieses Element eine untere Abschätzung für alle anderen Elemente.

Definition

Sei C eine Unterklasse der partiellen Funktionen über $\mathbb{N}^+$.
Eine Funktion f ist additiv optimal für C, wenn $f \in C$ und
$\forall g \in C(\exists c_{f,g} \forall x \in S(C_f(x) \leq C_g(x) + c_{f,g}))$.
Dabei gilt für alle additiv optimalen Funktionen f und g:
$|C_f(x) - C_g(x)| \leq c_{f,g}$
Durch Ersetzen von x durch die Tupelfunktion $< x, y >$ erhält man Funktionen mit zwei Parametern.

Beispiel

Sei D die Klasse von Beschreibungsmethoden, die alle partiellen Funktionen über $\mathbb{N}^+$ enthält. Jede additiv optimale Funktion aus D muss unbeschränkt sein. Betrachte eine unendliche Folge $x_1, x_2, \ldots$ mit $C_f(x_i) \geq i$ und eine Funktion $g(i) = x_i$.
Dann ist $C_g(x_i) = log(i) + O(1) << C_f(x_i)$
$\rightarrow$ f ist nicht additiv optimal
$\rightarrow$ es gibt kein minimales Element in $\{C_n : n \in D\}$
Aber: Die Klasse der partiell rekursiven Funktionen hat ein additiv optimales Element.

Der Invarianz-Satz

Betrachten wir ein Objekt x aus dem Ergebnisraum S mit dem Index n(x) und die Klasse der Beschreibungsmethoden $\{\phi : \phi$ ist eine partiell rekursive Funktion $\}$.
Welche Unterschiede in der Komplexität gibt es in Abhängigkeit vom Alphabet?
Antwort:Die Alphabetgröße hat keinen gravierenden Einfluß auf die Gültigkeit von Aussagen.
Die Aussagen sind invariant bis auf einen logarithmischen multiplikativen Faktor in Abhängigkeit von der Alphabetgröße.

Lemma

Es gibt eine additiv optimale, universelle, partiell rekursive Funktion.

Beweis

Sei ϕ_0 *die von* U *berechnete Funktion, wobei* U *als Eingabe das totale Programm* $< n, p >= \underbrace{11...1}_{l(n) mal} 0np$ *fordert.* U *simuliert dann* T_n *mit* p *als Eingabe.*
$\Rightarrow \phi_0(< n, p >) = \phi_n(p)$
Ist $n = 0$*, dann ist* $\phi_0(< n, p >) = \phi_0(p)$
Berechnet also T_n ϕ_n*, dann gilt:*
$C_{\phi_0}(x) \leq C_{\phi_n}(x) + c_{\phi_n}$ *mit* $c_{\phi_n} = 2l(n) + 1$

Definition

Seien $x, y, p \in \mathbb{N}$. Jede partiell rekursive Funktion ϕ in Verbindung mit p und y, sodass $\phi(< y, p >) = x$ ist eine Beschreibung von x.
Dann ist $C_\phi(x|y) = min\{l(p) : \phi(< y, p >) = x\}$ beziehungsweise $C_\phi(x|y) = \infty$, wenn es kein solches p gibt.
Wir nennen p ein Programm, dass x durch ϕ berechnet, wenn y gegeben ist.

Der Invarianz-Satz

Es gibt eine additiv optimale,universelle,partiell rekursive Funktion ϕ_0 für die Klasse der partiell rekursiven Funktionen, die x berechnen, wenn y gegeben ist.
$\Rightarrow \forall \phi, x, y(C_{\phi_0}(x|y) \leq C_\phi(x|y) + c_\phi)$ *(inv)*

Beweis

Sei ϕ_0 die von U berechnete Funktion, sodass U mit Eingabe $< y, < n, p >>$ T_n mit der Eingabe $< y, p >$ simuliert.
Wenn also $T_n \phi_n$ berechnet, dann ist
$\phi_0(< y, < n, p >>) = \phi_n(< y, p >)$
$\Rightarrow \forall n(C_{\phi_0}(x|y) \leq C_{\phi_n}(x|y) + c_{\phi_n})$ *mit* $c_{\phi_n} = 2l(n) + 1$.

Für jedes Paar additiv optimaler Funktionen ψ, ψ' gibt es eine Konstante $c_{\psi,\psi'}$, sodass für alle x und y gilt:
$|C_\psi(x|y) - C_{\psi'}(x|y)| \leq c_{\psi,\psi'}$
Ansatz:

- $\phi_0 = \psi$ und $\phi = \psi'$ in (inv) einsetzen
- $\phi = \psi$ und $\phi_0 = \psi'$ in (inv) einsetzen
- Beide so entstandenen Ungleichungen kombinieren

Definition

Sei ϕ_0 eine additiv optimale Funktion.
Dann ist die bedingte Kolmogorovkomplexität $C(.|.)$definiert als
$C(x|y) = C_{\phi_0}(x|y)$
und die unbedingte Kolmogorovkomplexität $C(.)$ ist
$C(x) = C(x|\epsilon)$

ϕ_0 ist die Referenzfunktion für C. Die Turing-Maschine U, die ϕ_0 berechnet, nennen wir Referenzmaschine.

Beispiel

λ_i - *Lisp-Programm*, λ_P - *Fortran-Interpreter*

π_i - *Fortran-Programm*, π_L - *Lisp-Interpreter*

$$\Rightarrow C_{LISP}(x) \leq C_{FORTRAN}(x) + l(\lambda_P)$$

$$C_{FORTRAN}(x) \leq C_{LISP}(x) + l(\pi_L)$$

$$\Rightarrow |C_{LISP}(x) - C_{FORTRAN}(x)| \leq l(\lambda_P) + l(\pi_L)$$

Beispiel

Es gibt universelle, partiell rekursive Funktionen, die nicht additiv optimal sind und für die der Invarianz-Satz nicht gilt.
Sei ϕ *die Funktion, die von der universellen Turing-Maschine* U_ϕ*, sodass* U_ϕ *mit Eingabe* $< y, < n, pp >>$ T_n *mit Eingabe* $< y, p >$ *simuliert.*
Wenn T_n ϕ_n *berechnet, dann ist*
$\phi(< y, < n, pp >>) = \phi_n(< y, p >)$
$\Rightarrow \forall x, y, n(C_\phi(x|y) \geq 2C_{\phi_n}(x|y))$

Two-Part Codes

Die kürzeste effektive Beschreibung eines Objektes kann in einem zweiteiligen Code dargestellt werden.
Dabei ist der erste Teil die Beschreibung der passenden Turing-Maschine T und der zweite Teil die Beschreibung des Programms p, das, wenn es von der Turing-Maschine interpretiert wird, das Objekt x rekonstruiert.
Also ist:
$C(x) = min\{l(T) + l(p) : T(p) = x\} + O(1)$
mit l(T) = Länge der kodierten Turing-Maschine oder
$C(x) = min\{l(T) + C(x|T) : T \in \{T_0, T_1, ...\}\} + O(1)$

Der Turing-Maschinen-Teil entfernt die Regelmäßigkeiten aus x. Übrig bleiben Unregelmäßigkeiten/zufällige Aspekte, in Abhängigkeit von der Turing-Maschine.

Betrachten wir nun einen Teil aus der Gleichung von der letzten Folie:
$\min_T\{l(T) + C(x|T) : T \in \{T_0, T_1, ...\}\}$
Das T ist die Menge an in x enthaltener Information.

Obere Schranken

Satz

$\exists c \forall x, y(C(x) \leq l(x) + c) \wedge C(x|y) \leq C(x) + c)$

Beweis

1. *Sei T eine Turing-Maschine, die die Eingabe in die Ausgabe kopiert, sodass* $\forall x(C_T(x) = l(x))$
   $\Rightarrow C_T(x) \leq l(x) + c$
2. *Sei T die Turing-Maschine, die* $\forall y, z$ *aus* $< z, y >$ x *berechnet, wenn die Referenz-Maschine U* x *aus* $< z, \epsilon >$ *berechnet*
   $\Rightarrow C_T(x|y) = C(x)$
   $\Rightarrow \exists c(C(x|y) \leq C_T(x|y) + c) = C(x) + c$

Satz

Sei $A \subseteq \mathbb{N} \times \mathbb{N}$ *rekursiv aufzählbar und* $y \in \mathbb{N}$*.*
Angenommen, $Y = \{x : (x, y) \in A\}$ *ist endlich.*
$\Rightarrow \exists c \forall x \in Y(C(x|y) \leq l(d(Y)) + c)$

Beweis

Sei A *eine Aufzählung* $(x_1, y_1), (x_2, y_2), \ldots$ *durch* T *und eine Teilfolge* $(x_{i_1}, y_{i_1}), \ldots, (x_{i_k}, y_{i_k})$ *die Aufzählung der Elemente von* Y *mit* $k = d(Y)$*.*
Ist y *gegeben, dann kann man* T_y *erzeugen, sodass* $T_y(p) = x_{i_p}$ *mit* $1 \leq p \leq d(Y)$*.*
$\Rightarrow C(x|y) \leq C_{T_y}(x) + c \leq l(d(Y)) + c$*, wobei* c *nur von* A *abhängt.*

Invarianz der Kolmogorovkomplexität

$C(x)$ ist mit konstanter Abweichung, in Abhängigkeit von der Referenzfunktion invariant.
C_ϕ und C_ψ sind äquivalent $C_\phi \equiv C_\psi$, wenn
$\exists c \forall x(|C_\phi(x) - C_\psi(x)| \leq c)$

Äquivalenzklasse $[C_\phi] = \{C_\psi : C_\psi \equiv C_\phi\}$
Gibt es ein $c \geq 0$, sodass $\forall x(C_\phi(x) \leq C_\psi(x) + c)$:
$\Rightarrow$ partielle Ordnung mit einem minimalen Element $[C_{\phi_0}]$, sodass
$\forall C_\psi([C_{\phi_0}] \leq [C_\psi])$.

Betrachten wir Aufzählungen partiell rekursiver Funktionen $\phi_1, \phi_2, ...$ und $\psi_1, \psi_2, ...$, wobei ϕ die korrespondierende Aufzählung zu der im Invarianz-Satz ist.

Es gelte weiter $\psi_i = \phi_{f(i)}$ und $\phi_i = \psi_{g(i)}$ mit $i = 1, 2, ...$ Sind f und g partiell rekursiv, dann sind ϕ und ψ rekursiv isomorph und akzeptable Aufzählungen.
Sei $C(x) = C_\phi(x)$ und $C'(x) = C_\psi(x)$ gegeben.
$\Rightarrow \exists c \forall x (|C(x) - C'(x)| < c)$
$\Rightarrow$ C ist rekursiv invariant zwischen akzeptablen Aufzählungen.

Angenommen ϕ und ψ sind nicht rekursiv isomorph.
Sei $C(x) = C_\phi(x)$ und ψ wie folgt definiert:
$\psi = \{ {\psi_{2i}(1) := y_i \ (a) \atop \psi_{2i}(x) := \phi_i(x) \ \forall x > 1}$
(a) für manche y_i mit $C(y) \geq i^2$.

Außerdem ist $\psi_{2i+1} := \phi_i$.
$\Rightarrow \psi$ enthält alle partiell rekursiven Funktionen.

Sei nun $C(.)$ die Kolmogorovkomplexität in der ψ-Aufzählung.
$\Rightarrow C'(y_i) = C'_{\phi_{2i}}(y_i) + c_{\phi_{2i}}$

Nach obigen Definitionen ist $C'_{\phi_{2i}}(y_i) = 1$ und
$c_{\phi_{2i}} \leq 2log2i + O(1)$.

Andererseits: $C(y_i) \geq i^2$
$\Rightarrow |C'(y_i) - C(y_i)|$ ist nach oben unbeschränkt.

Konkrete Kolmogorovkomplexität

Bisher wurde bei der Invarianz mit der Unbestimmtheit "gleich bis auf eine Konstante c gearbeitet".
Eliminierung der Unbestimmtheit:

- Feste Definitionsmenge von Objekten
- Feste effektive Aufzählung von TMs
- Feste Wahl der additiv optimalen Funktionen

U' - Universelle Turing-Maschine
Für jedes x gibt es eine TM U', sodass $C_{U'}(x) = 0$

$\forall U' \forall x (C(x) \leq C_{U'}(x) + C(U'))$. $C(U')$ ist die Mindestlänge des Programmes p, sodass
$\forall q (U(pq) = U'(q))$
$\Rightarrow$ Ist $C_{U'}(x) = 0$, dann ist $C(U') \geq C(x)$ und die Beschreibung von U' enthält die Beschreibung von x.
$\Rightarrow$ Um die Komplexität eines langen x niedrig zu halten, muss auch U komplex sein.

Glossar

dispense - verallgemeinern, loslösen von
minorizes - ist untere Abschätzung für
pairing function -Tupelfunktion
Sample Space - Ergebnisraum
selfdelimiting - selbstbegrenzend
upper bounds - Obere Schranken
unbounded - unbeschränkt

Quellen:

- "An Introduction to Kolmogorov Complexity and its Applications", Ming Li, Paul Vitany, Third Edition, Springer Verlag
- Theoretische Informatik, Juraj Hromkovič, 3. Auflage, Teubner Verlag