Véges mintázatok információtartalma és entrópiája kombinatorikai szemszögből

Zsolt Pocze

January 22, 2025

Abstract

Különböző típusú mintázatok információtartalmának és entrópiájának egységes, a hagyományos információ- és entrópiafogalmakkal kompatibilis kombinatorikai meghatározása, túllépve az ergodikus Markov-folyamatokra értelmezhető Shannon-információ korlátain. Különböző típusú véges mintázatok információtartalmát hasonlítjuk össze és ebből meghatározzuk az infromációmennyiség általános tulajdonságait, és ezek segítségével definiálunk Kolmogorov-komplexitás és a tömörítési algoritmusokon alapuló normalizált információbecslési módszereket. A kombinatorikai nézőpont alapján újradefiniáljuk az entrópia fogalmát a hagyományos entrópiával aszimptotikusan kompatibilis módon.

1 Bevezetés

Az anyag jellemzője a mintázata, amit tág értelemben értelmezünk, mint az elemi részek elrendeződését. Ez magába foglalja az egyes részek közötti kapcsolatokat is. A fizikai valóságban minden véges dimenziós mintázat valamilyen pontossággal modellezhető egy dimenziós véges sorozatként, ezért az információt és az entrópiát véges sorozatokkal (mintázatokkal) összefüggésben vizsgáljuk. Jelölje X^*a véges mintázatok halmazát, ahol X halmaz a mintázatok értékkészlete vagy más néven alaphalmaza. Egy adott $A \in X^*$, $A = (x_1, x_2, ..., x_n)$ mintázat esetén jelölje $n = |A|$ a mintázat hosszát, $k = |X|$ az értékkészlet elemszámát, $f(x)$, $x \in X$ pedig az x elem előfordulási számát a mintázatban.

Az információ nem más, mint a bináris döntések száma [6], amellyel egy mintázat egyértelműen meghatározható, vagyis kombinatorikailag a döntések száma, amivel az adott mintázat kiválasztható az összes lehetséges mintázat és az üres mintázat közül. Az információ alapegysége a bináris döntés, amelynek *bit* a mértékegysége. A gyakorlatban a a döntések száma csak egész számot vehet fel, de ha folytonos függvényt használunk, nem kapunk mindig egész értékeket, ami a döntések elméleti száma. A matematikai számítások egyszerűsítése érdekében a döntések számán a továbbiakban mindig az elméleti döntésszámot értjük. Ezért és más okokból is az információmennyiség meghatározása mindig közelítő meghatározás.

Definíció 1. Egy véges $A \in X^*$ mintázat információja legyen a mintázat egyértelmű meghatározásához szükséges bináris döntések minimális száma, jelölése legyen $I(A)$, ahol $I : X^* \to \mathbb{R}^+$ függvény és

$$I(A) = min\{n | A\ reprodukálható\ (d_1, d_2, ..., d_n)\ elemi\ döntéssorozattal\}\ \square \tag{1}$$

Ez a definíció általános, mivel nem függ semmilyen konkrét rendszertől, tisztán elméleti, mivel minden implicit információ explicit módon beépül, és filozófiailag kevéssé vitatható, mert a minimális döntések száma az információtartalom legáltalánosabb mértéke. Ugyanakkor a definícióban található fogalmak nincsenek pontosan meghatározva ahhoz, hogy a gyakorlatban vagy akár elméletben alkalmazható

legyen: nem rögzítettük, mit nevezünk pontosan elemi döntésnek és reprodukálhatóságnak és az sem, hogy egy mintázat létrehozásának a leírását hogyan bontjuk elemi döntésekre.

A fenti információ definíció egy speciális esete a Kolmogorov-komplexitás [4], amely egy univerzális Turing-géppel határozza meg, hogy hány döntés szükséges a mintázatok leírásához:

Definíció 2. Legyen U egy rögzített univerzális Turing-gép. Ekkor egy véges $A \in X^*$ mintázat Kolmogorov-komplexitása $I_K(A)$ az alábbi módon adható meg:

$$I_K(A) \ = \ min \mid B \mid : \ U(B) = A \tag{2}$$

ahol $\mid B \mid$ a bináris program (bitsorozat) hosszát jelöli, és a minimumot azon B programok között keressük, amelyek bemenetként véve a U univerzális Turing-gépen pontosan A-t állítanak elő kimenetként. □

A Kolmogorov-komplexitás sajnos általánosságban nem kiszámítható [1]. Az információtartalom bizonyos határesetekben azonban nagyon pontosan, explicit módon meghatározható: ilyen például a számok, a konstans mintázatok az egyenletes eloszlású véletlen mintázatok és bizonyos jól meghatározott statisztikai jellemzőkkel rendelkező mintázatok, mint pl. az ergodikus Markov-folyamatokkal előállítható mintázatok.

Ha az információ meghatározásához elméleti és gyakorlati téren is pontosabb módszereket keresünk, mindenképpen érdemes először a határeseteket és az említett speciális eseteket vizsgálni.

2 Konstans mintázat információja

Konstans és véges $A \in \{a\}^*$ mintázat esetén a mintázat egyes elemeinek a meghatározásához nincs szükség információra, mivel egyetlen elemet ismétlünk. Egyedül a mintázat hossza, azaz n hordoz információt, melynek meghatározásához maximum $\lceil log_2 n \rceil$ döntésre (információra) van szükség, mert minden döntés megfelezi a lehetőségeket. Az egyszerűség és a matematikai kezelhetőség kedvéért használjuk a $log_2 n$ elméleti közelítést.

Az egész számok információjából kiindulva bármilyen azonos elemekből álló mintázat információtartalma kiszámolható, ha az információt az összes lehetséges részmintázatból való kiválasztásként értelmezzük, beleértve a nulla hosszúságú mintázatot.

1	()
2	a
3	aa
...	...
$n+1$	$aaa...a$

Table 1: Az n hosszúságú konstans mintázat esetén az információ meghatározását az $n+1$ elem közül történő kiválasztásra egyszerűsíthetjük, ahol () jelöli az üres mintázatot.

A mintázat elemeiből összerakható összes lehetséges mintázat számának logaritmusa adja a konkrét mintázat információtartalmát, ekkor az A mintázat információja:

$$I_{const}(A) = log_2(n+1) \tag{3}$$

Az n helyett az $n+1$ azért praktikusabb, mert így az üres mintázat információtartalma is értelmezve van és figyelembe vesszük, hogy az üres mintázat, mint lehetőség is hordoz információt. Könnyű belátni,

hogy a véges mintázatok közül a konstans mintázatok információtartalma a legalacsonyabb, mert a nem konstans mintázatokban a különböző elemek miatt több döntés szükséges a mintázat egyértelmű meghatározásához, ami növeli az információtartalmat.

Azért sem használjuk az $log_2(n)$ képletet, mert akkor egységnyi hosszúságú mintázatok esetén a szubadditivitás nem teljesül. A szubadditivitás feltétele $I_{rand}(ab) \leq I_{rand}(a) + I_{rand}(b)$. Ha a $log_2 n$ képletet használnánk, akkor $n = 1$ hosszúságú mintázatok konkatenációja esetén a szubadditivitás nem teljesülne: $log_2(2) \nleq log_2(1) + log_2(1)$. A $log_2(n+1)$ képlet esetén viszont a szubadditivitás minden $n \geq 0$ esetén teljesül:

n_1	n_2	Szubadditivitás
0	0	$log_2 1 \leq log_2 1 + log_2 1$
0	1	$log_2 2 \leq log_2 1 + log_2 2$
1	1	$log_2 3 \leq log_2 2 + log_2 2$
1	2	$log_2 4 \leq log_2 2 + log_2 3$
2	2	$log_2 5 \leq log_2 3 + log_2 3$

Table 2: A szubadditivitás teljesülése különböző hosszúságú egyenletes eloszlású véletlen sorozatok esetén.

3 Egyenletes eloszlású véletlen mintázat információja

Az egyenletes eloszlású véges mintázat $A \in X^*$ úgy állítható elő, hogy a mintázat minden eleme egy $log_2(k)$ bites független döntés eredménye. Figyelembe véve, hogy az üres mintázatot is számolva $n+1$ különböző hosszúságú mintázat közül választhatunk, a mintázat információtartalma:

$$I_{rand}(A) = log_2 \sum_{i=0}^{n} k^i \tag{4}$$

Ha a $k = 1$, vagyis az A konstans mintázat, akkor a képlet a konstans mintázat képletére egyszerűsíthető:

$$I_{rand}(A) = log_2 \sum_{i=0}^{n} 1^i = log_2(n+1)$$

Az $I_{rand}(A) = log_2\left(\sum_{i=0}^{n} k^i\right) = log_2\left(\frac{k^{n+1}-1}{k-1}\right)$ komplexitása $O(n \cdot log_2(k))$, tehát elegendően nagy n és k esetén $I_{rand}(A) \approx n \cdot log_2(k)$ képlettel is számítható. A $n \cdot log_2(k)$ képletet használva azonban a konstans mintázatok esetéhez hasonlóan nem teljesülne a szubadditivitás: $2 \cdot log_2(2) \nleq 1 \cdot log_2(1) + 1 \cdot log_2(1)$, és a konstans mintázatok esetén sem kapnánk megfelelő képletet.

4 Ergodikus Markov-folyamattal előállítható mintázat információja

Legyen $A \in X^*$ mintázat, amely ergodikus Markov-folyamattal előállítható, és legyenek $f_{rel}(x_i) = \frac{f(x_i)}{n}$, $x_i \in X$, $i = 1, ..., k$ az egyes értékek relatív gyakoriságai a mintázatban. Shannon eredeti $I_{Shannon}(A) = \sum_{i=1}^{k} f_{rel}(x_i) log_2 \frac{1}{f_{rel}(x_i)}$ képlete [6] nem lenne kompatibilis az egyenletes eloszlású mintázatok és a konstans mintázatok képleteivel, azért módosítani kell. A mintázat információja Shannon képletét módosítva:

$$I_{mark}(A) = log_2 \sum_{i=0}^{n} \prod_{x \in X} f_{rel}(x)^{-i \cdot f_{rel}(x)} \tag{5}$$

Ha a $k = 1$, vagyis az A konstans mintázat, azaz $f_{rel}(x) = 1$, $x \in X$ akkor a képlet a konstans mintázat képletére egyszerűsíthető:

$$I_{mark}(A) = log_2 \sum_{i=0}^{n} \prod_{x \in X} 1^{-i} = log_2(n+1)$$

Egyenletes eloszlású folyamattal előállítható mintázat esetén, ahol $f_{rel}(x_i) = \frac{1}{k}$, $x_i \in X$, $i = 1, ..., k$, vagyis az értékek relatív gyakoriságai azonosak, a képlet az egyenletes eloszlású mintázat információ képletére egyszerűsödik:

$$I_{mark}(A) = log_2 \sum_{i=0}^{n} \prod_{x \in X} \left(\frac{1}{k}\right)^{-i \cdot \frac{1}{k}} = log_2 \sum_{i=0}^{n} \prod_{x \in X} k^{i \cdot \frac{1}{k}} = log_2 \sum_{i=0}^{n} k^i$$

Megmutatható, hogy $lim_{n \to \infty} I_{mark}(A) = I_{Shannon}$. Legyen $c = \prod_{x \in X} f_{rel}(x)^{-f_{rel}(x)}$. Ha $n \to \infty$, akkor $I_{mark}(A) = log_2 \sum_{i=0}^{n} c^i \approx log_2 \left(\frac{c^{n+1}-1}{c-1}\right) \approx n \cdot log_2 c$. Ebből következik, hogy $I_{mark}(A) \approx n \cdot log_2 \prod_{x \in X} f_{rel}(x)^{-f_{rel}(x)}$, ami a logaritmus tulajdonságai alapján a $I_{mark}(A) \approx n \cdot \sum_{x \in X} log_2 \left(f_{rel}(x)^{-f_{rel}(x)}\right)$ alakra hozható, ami tovább alakítva $I_{mark}(A) \approx n \cdot \sum_{x \in X} f_{rel}(x) \cdot log_2 \left(\frac{1}{f_{rel}(x)}\right) = I_{Shannon}(A)$.

Állítás 1. *Ergodikus Markov-folyamatokkal előállítható véges mintázatok $I_{mark}(A)$ értéke akkor maximális, ha az értékek relatív gyakoriságai egyenlőek, azaz $f_{rel}(x) = \frac{1}{k}, \forall x \in X$* □.

Az információmérési képlet átírható logaritmus segítségével $I_{mark}(A) = log_2(\sum_{i=0}^{n} 2^{-i \sum_{x \in X} f_{rel}(x) log_2 f_{rel}(x)})$ alakba. Mivel a $-log_2 x$ függvény konvex, ezért alkalmazhatjuk a Jensen-egyenlőtlenséget: $\sum_{x \in X} f_{rel}(x) log_2 f_{rel}(x) \leq log_2(\sum_{x \in X} f_{rel}(x) \cdot 1)$, ami nem más, mint $\sum_{x \in X} f_{rel}(x) log_2 f_{rel}(x) \leq 0$. Az egyenlőség akkor áll fenn, ha az összes $f_{rel}(x)$ azonos, azaz $f_{rel}(x) = \frac{1}{k}, \forall x \in X$. Tehát az ergodikus Markov-folyamatokkal előállítható véges mintázatok információtartalma pontosan akkor maximális, ha minden érték azonos gyakorisággal fordul elő a mintázatban, és ebből következik, hogy az egyenletes eloszlású véletlen mintázatok rendelkeznek a maximális információmennyiséggel és az információtartalmuk $log_2 \sum_{i=0}^{n} k^i$.

Shannon az ergodikus Markov-folyamatokra határozta meg az információt [6], de fontos tudni, hogy a gyakorlatban a mintázatok jelentős része nem hozható létre ergodikus Markov-folyamattal, ezért Shannon képlete ezekben az esetekben nem használható információ- és entrópiamérésre. Az összes lehetséges véges mintázat között csak egy viszonylag kis rész az, amely ergodikus Markov-folyamattal létrehozható. Ennek oka, hogy az ergodikus Markov-folyamatok által generált mintázatoknak meg kell felelniük bizonyos statisztikai tulajdonságoknak és átmeneti valószínűségeknek. Shannon módszerénél általánosabb megoldást kínál Kolmogorov [4]. A Kolmogorov-komplexitás A Shannon-információval ellentétben minden létező véges mintázat esetén értelmezhető.

5 Általános mintázatok információtartalma

5.1 Az információ általános tulajdonságai

A speciális mintázatok információjából következtethetünk az információ általános tulajdonságaira. [3]. Könnyen belátható a következő állítás:

Állítás 2. *Legyen $A \in X^*$ mintázat, legyen $B \in X^*$ egy konstans mintázat, $C \in X^*$ egy véletlenszerű mintázat és $|A| = |B| = |C|$. Ekkor igaz a következő egyenlőtlenség:*

$$I_{const}(B) \leq I(A) \leq I_K(A) \leq I_{mark}(A) \leq I_{rand}(C) \; \square$$

A random mintázat I_{rand} információtartalma a legnagyobb és a konstans mintázaté I_{const} a legkisebb. A Kolmogorov-komplexitás a Turing-gépekre épül, ezért nem minden esetben képes olyan rövid leírást adni egy véges mintázatra, amelyet Turing-gép nélkül, más módszerrel adhatnánk: I_K nagyon jól közelíti az információtartalmat, de lehet nála nagyobb. Az ergodikus Markov-sorozatokra optimalizált I_{mark} módosított Shannon-információ nem ergodikus és nem Markov-folyamatok esetén az információtartalmat felülbecsüli, és a kevésbé véletlenszerű mintázatok esetén magasabb értéket ad.

Állítás 3. *Az információ általános tulajdonságai:*

1. **Normalizálás:** $log_2(n+1) \leq I(A) \leq log_2 \sum_{i=0}^{n} k^i$, bármely $A \in X^n$ és $n \in \mathbb{N}^+$ esetén.
2. **Szubadditivitás:** $I(AB) \leq I(A) + I(B)$, bármely $A, B \in X^*$.
3. **Reverzibilitás:** $|I(A) - I(A^R)| \leq c$, valamely $c \in \mathbb{R}_0^+$ esetén, ahol $A^R[i] = A[n-i]$, $\forall i \in \{1, ..., n\}$, bármely $A \in X^*$ esetén.
4. **Monotonitás:** $I(A) \leq I(B)$, bármely $A, B \in X^*$ esetén, ha A részmintázata B-nek.
5. **Redundancia:** $|I(A^r) - (I(A) + log_2(r))| < c$, valamely $c \in \mathbb{R}_0^+$ esetén, ahol A^r az A mintázatot tartalmazza r-szer.$\square$

A normalizálás tulajdonsága következik a konstans mintázat $I_{const}(A) = log_2(n+1)$ és az egyenletes eloszlású véletlen mintázat $I_{rand}(A) = log_2 \sum_{i=0}^{n} k^i$ információjából és az 1. állításból.

A szubadditivitás könnyen belátható a konstans mintázat esetén: $log_2(n+m+1) \leq log_2(n+1) + log_2(m+1)$, ami átalakítva $log_2(n+m+1) \leq log_2(n \cdot m + n + m + 1)$, ez minden esetben telesül. Ergodikus Markov-folyamatok esetén legyen $C = \prod_{x \in X} f_{rel}(x)^{f(x)}$, ekkor az egyenlőtlenség $log_2 \sum_{i=0}^{n+m} C^{-i} \leq log_2 \sum_{i=0}^{n} C^{-i} + log_2 \sum_{i=0}^{m} C^{-i}$,ami átalakítva $\sum_{i=0}^{n+m} C^{-i} \leq \left(\sum_{i=0}^{n} C^{-i}\right)\left(\sum_{i=0}^{m} C^{-i}\right)$. A jobb oldali összeget átalakítva $\sum_{i=0}^{n+m} C^{-i} \leq \sum_{i=0}^{n} \sum_{j=0}^{m} C^{-(i+j)} = \sum_{i=0}^{n+m} \left(\sum_{k,l;k+l=i} C^{-i}\right)$. Minden k-hoz legalább egy (k,l) pár létezik, mely teljesíti a feltételeket. Ez azt jelenti, hogy a belső összegek legalább egyszer tartalmazzák a C^{-k} tagot, ezért az egyenlőtlenség teljesül.

A reverzibilitás azt jelenti, hogy mindegy, melyik oldalról kezdjük el olvasni a mintázatot, az nem befolyásolja az információtartalmát, ami triviális, mert a mintázatot az értelmező könnyedén megfordíthatja. A monotonitás szintén triviális a konstans mintázatok és az ergodikus Markov-folyamattal előállítható mintázatok esetén egyaránt.

Legyen $A^r = AA...A$ redundáns, nr hosszúságú mintázat. Ekkor $I_{const}(A^r) = log_2(nr+1) = log_2(n+1) + log_2 r + log_2\left(\frac{nr+1}{(n+1)r}\right)$. A $log_2\left(\frac{nr+1}{(n+1)r}\right)$ kifejezés értéke n és r növekedésével 0-hoz közelít, így korlátos, ezért mindig van olyan $c \in \mathbb{R}_0$, hogy $|I(A^r) - (I(A) + log_2(r))| < c$. Véletlenszerű és ergodikus Markov-folyamatok esetén és általános esetben is intuitív módon belátható az összefüggés.

Definíció 3. legyen az $A \in X^*$ véges mintázat minimális információja az alábbi $I_{min} : \mathrm{X}^* \to \mathbb{R}^+$függvény:

$$I_{min}(A) = log_2(n+1)$$

maximális információja pedig legyen az alábbi $I_{max} : \mathrm{X}^* \to \mathbb{R}^+$függvény:

$$I_{max}(A) = log_2 \sum_{i=0}^{n} k^i \; \square$$

5.2 Információ számítása Kolmogorov-komplexitás alapján

Az információ az 1. definícióban ismertetett általános meghatározása szorosan kapcsolódik a 2. definícióban meghatározott Kolmogorov-komplexitáshoz [4], amely egy adott univerzális gépen a mintázatokat előállító legrövidebb bináris programkódok hosszaként határozza meg a mintázatok információtartalmát.

A Kolmogorov-komplexitás esetén az univerzális gép rögzítése biztosítja, hogy a különböző mintázatok információja összehasonlítható legyen, az univerzális gépek eredményei között ugyanis lehet konstans eltérés. Az eltérések hosszabb mintázat esetén elhanyagolhatók, rövid mintázatoknál viszont jelentősek lehetnek. Az információ és a Kolmogorov-komplexitás közti kapcsolatot a $K(A) = I(A) + c$ képlettel jellemezhetjük [2], ahol c a K kiszámításához használt univerzális gépre jellemző konstans érték. Mivel az $I_{min}(A)$ minimális információt pontosan ismerjük, így a konstans eltérést kiküszöbölve meghatározható az információ Kolmogorov-komplexitás alapján történő mérése:

Definíció 4. Az $A \in X^*$ mintázat Kolmogorov-komplexitással mért információja legyen:

$$I_K(A) = K(A) - K_{min}(A) + I_{min}(A) \tag{6}$$

ahol $K_{min}(A)$ az n hosszúságú konstans mintázat Kolmogorov-komplexitása:

$$K_{min}(A) = K(\{a\}^n) \; \square$$

5.3 Információ számítása tömörítési algoritmus alapján

A Kolmogorov-komplexitás, vagyis az információ pontos meghatározása azonban általános mintázatok esetén elméletileg lehetetlen, csak közelíteni lehet, és erre a legjobbak a veszteségmentes tömörítési algoritmusok. [5] Az ezekkel tömörített mintázatok a nagy információsűrűség miatt közel véletlenszerűek. Ha a betömörített mintázat információtartalmát megmérjük véletlenszerű mintázatot feltételezve, akkor megkapjuk a közelítő információtartalmát az eredeti mintázatnak. A tömörítési algoritmusokra jellemző, hogy a tömörített kódba gyakran a kitömörítéshez szükséges algoritmust és más adatokat is beleírnak, ami kisebb mintázatok tömörítésekor arányaiban nagy többletinformációt jelent, ezért az eredményül kapott információt normálni kell.

Definíció 5. A $C : X^* \to X^*$függvényt tömörítésnek nevezzük, ha:

1. C injektív, azaz ha $A, B \in X^*$ és $C(A) = C(B)$, akkor $A = B$.
2. Minden $A \in X^*$ esetén $\mid C(A) \mid \leq \mid A \mid$.
3. Létezik legalább egy $A \in X^*$, amelyre $\mid C(A) \mid < \mid A \mid$. $\square$

A tömörítő függvényt az egyszerűség kedvéért úgy definiáltuk, hogy a tömörítetlen és a tömörített mintázatoknak azonos legyen az értékkészlete.

Állítás 4. *Ha $C : X^* \to X^*$ tömörítés, akkor $I(A) \leq C(A)$ bármely $A \in X^*$ esetén.*

Definíció 6. Legyen $A \in X^n$ tetszőleges mintázat, $C : X^* \to X^*$ tetszőleges tömörítési algoritmus, akkor az A mintázat C tömörítési algoritmussal mért információja legyen:

$$I_C(A) = \frac{I_{max}C(A) - I^C_{min}(A)}{I^C_{max}(A) - I^C_{min}(A)} \cdot I_{max}(A) + I_{min}(A)$$

ahol

$$I^C_{min}(A) = \min_{A \in X^n} I_{max}(C(A))$$
$$I^C_{max}(A) = \max_{A \in X^n} I_{max}(C(A)) \ \square$$

Az I^C_{min} és I^C_{max}meghatározása a definíció alapján a gyakorlatban látszólag körülményes, de ha figyelembe vesszük, hogy a tömörített mintázatok a nagy információsűrűség miatt közel véletlenszerűek, és ezért Markov-folyamattal jól modellezhetők, akkor alkalmazhatjuk a következő közelítést:

$$I^C_{min}(A) = I_{mark}(C(B))$$
$$I^C_{max}(A) = I_{mark}(C(D))$$

ahol $B \in X^n$ tetszőleges konstans mintázat és $D \in X^n$ tetszőleges egyenletes eloszlású véletlen mintázat.

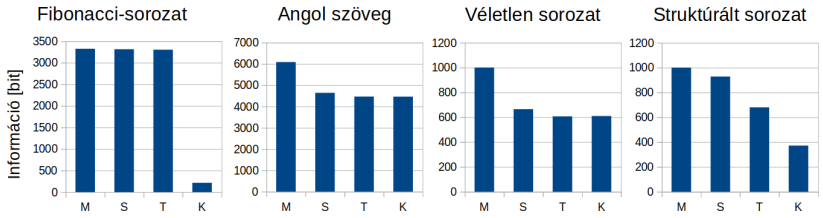


Figure 1: Az ábrán különböző értékkészletű, 1000 karakter hosszúságú mintázatok (FÜGGELÉK I.) információértékeinek az összehasonlítása látható. M: a maximális információmennyiséget jelöli, ami az adott hosszúságú és adott értékkészletű mintázat esetében lehetséges. S: a mintázat módosított Shannon-információja. T: a mintázat GZip tömörítési algoritmussal mért információja. K: a mintázat közelítő Kolmogorov-komplexitása. A véletlen mintázat egy bizonyos fokú redundanciával rendelkező véletlen bináris mintázat, a struktúrált mintázat pedig egy 40x25 méretű bináris karaktermátrix, amelyen az 1-es szimbólumok koncentrikus körökben helyezkednek el. Látszik, hogy a Fibonacci-sorozat információtartalmát a látszólagos véletlenszerűsége miatt még a tömörítési algoritmus sem tudta meghatározni, míg a Kolmogorov-komplexitása alacsony információtartalmat mutat. Az angol szöveg és a véletlen mintázat esetén a Shannon-féle módszer és a tömörítési algoritmus egyaránt jó eredményt adott. A struktúrált szöveg esetén viszont a tömörítési algoritmus látványosan jobban közelíti a valós információtartalmat, mint a Shannon-féle képlet, amely a véletlenszerű mintázatokra lett kitalálva. (A használt algoritmuosok a FÜGGELÉK II-IV-ben olvashatók.)

A különböző információmérési módszerek különböző struktúrák esetén eltérő hatékonyságúak, ezért nagyobb pontosság érhető el, ha többféle módszerrel mért információ eredményeinek a minimumát vesszük.

Definíció 7. Ha $(I_1, I_2, ..., I_m)$ információmérési módszerek, akkor $A \in X^n$ mintázat I_m információmérési módszerekkel mért információja:

$$I_m(A) = \min_{i=1,...,m} I_i(A) \ \square \tag{7}$$

6 Véges mintázatok entrópiája

Az entrópia az információval ellentétben a mintázatnak egy átlagos jellemzőjét jelenti, az egy elem meghatározásához szükséges átlagos információmennyiséget. A legtöbb esetben az entrópiát - tévesen - a Shannon-entrópiával azonosítják [7], amely ergodikus Markov-folyamatok esetén közelíti csak jól az elemenkénti átlagos információtartalmat. A Kolomogorov-komplexitásból számolt entrópia jobb közelítést ad és általánosabb, ezért az entrópiát célszerűbb az információtartalom alapján definiálni, ahol az információtartalom mérésének módszere nem meghatározott.

Ha $A \in X^*$ konstans sorozat, $X = \{a\}$, és () jelöli az üres sorozatot, az entrópia az üres mintázatot is figyelembe véve, kombinatorikai szempontból a következő képpen értelmezhető:

n	Mintázat	Entrópia
0	()	$log_2(1)$
1	()(a)	$log_2(2)/2$
2	()$(a)(a)$	$log_2(3)/3$

Table 3: Konstans mintázat entrópiája.

Általánosságban az entrópiát ebben az értelmezésben az alábbi módon definiálhatjuk.

Definíció 8. Egy véges $A \in X^*$ mintázat $H_C : X^* \to \mathbb{R}^+$entrópiája legyen a mintázat elemeinek átlagos információtartalma, azaz

$$H_C(A) = \frac{I(A)}{n+1} \tag{8}$$

ahol I az információ. □

Az $n+1$ a nevezőben lehetővé teszi a képlet üres mintázatokon való értelmezését. Konstans $A \in X^*$ mintázat esetén, ha $|X| = 1$, az entrópia $H(A) = \frac{log_2(n+1)}{n+1}$, ami azt jelenti, hogy n növekedésével az entrópia aszimptotikusan közelít a nullához.

Ergodikus Markov-folyamatok esetén az entrópia n növekedésével a Shannon-entrópiához konvergál:

$$lim_{n\to\infty} H_C(A) = H(A)$$

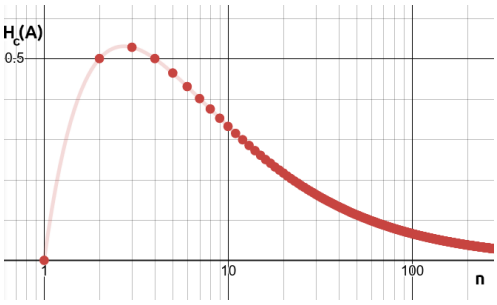


Figure 2: Konstans mintázat entrópiája n függvényében. Shannon eredeti jelforrásokra definiált entrópiaértelmezésével ez összhangban van. Hétköznapi értelmezésben ésszerű az a feltevés, hogy a jelforrás ha nem bocsát ki jelet, az nem meglepő, alapállapotnak tekinthető, ezért az entrópiája nulla. Ha kibocsát egyetlen jelet és elhallgat, az meglepetést okoz. Ha két azonos jelet bocsát ki, kicsit nagyobb a meglepetés, ha azonban a kibocsátott azonos jelek sorozata egyre hosszabb lesz, egyre kevésbé lesz érdekes.

7 Összefoglalás

Ez a tanulmány egységes szemléletet kínál a véges mintázatok információ- és entrópmértékeire, túlmutatva a hagyományos Shannon-megközelítésen. Az ergodikus Markov-folyamatokra épülő Shannon-entrópia és a Kolmogorov-komplexitás jellegű általánosabb eljárások összevetésével szélesebb perspektívát nyújt a különböző szerkezetű minták információtartalmának mérésében. Bemutatja a konstans, véletlen és Markov-folyamatok által generált minták információjának alapfogalmait, valamint olyan általános tulajdonságokat, mint a szubadditivitás és a redundancia. Míg a hagyományos módszerek gyakran pontatlan becsléseket adnak rövid minták esetén, ez a keret, kiegészítve gyakorlati, tömörítési technikákkal, nagyon rövid szekvenciákra is elfogadható eredményt szolgáltat, és hidat képez az elméleti megfontolások és a valós alkalmazások között. Az itt bemutatott egységes megközelítés tisztázza a különböző entrópiafogalmak alkalmazhatóságát különféle adatstruktúrák esetében, miközben világos példákkal, formális bizonyításokkal és újszerű felismerésekkel szolgál a matematikusok, informatikusok, illetve a magas szintű adat- és információelmélet iránt érdeklődők számára – akár a jól ismert, akár a kevésbé feltárt területeken.

References

[1] Gregory J. Chaitin. On the length of programs for computing finite binary sequences. *J. ACM*, 13(4):547–569, October 1966.

[2] Gregory J. Chaitin. A theory of program size formally identical to information theory. *Journal of the ACM (JACM)*, 22(3):329–340, 1974.

[3] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

[4] A. N. Kolmogorov. On tables of random numbers. *Mathematical Reviews*, 1963.

[5] Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer, 2nd edition, 1997.

[6] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 1948.

[7] Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, 1949.

FÜGGELÉK I.

Az 1. ábrán látható összehasonlításhoz használt 1000 karakter hosszúságú mintázatok.

Fibonacci-sorozat

0 1 2 3 5 8 1 3 2 1 3 4 5 5 8 9 1 4 4 2 3 3 3 7 7 6 1 0 9 8 7 1 5 9 7 2 5 8 4 4 1 8 1 6 7 6 5 1 0 9 4 6 1 7 7 1 1 2 8 6 5 7 4 6 3 6 8 7 5 0 2 5 1 2 1 3 9 3 1 9 6 4 1 8 3
1 7 8 1 1 5 1 4 2 2 9 8 3 2 0 4 0 1 3 4 6 2 6 9 2 1 7 8 3 0 9 3 5 2 4 5 7 8 5 7 0 2 8 8 7 9 2 2 7 4 6 5 1 4 9 3 0 3 5 2 2 4 1 5 7 8 1 7 3 9 0 8 8 1 6 9 6 3 2 4 5 9 8 6 1
0 2 3 3 4 1 5 5 1 6 5 5 8 0 1 4 1 2 6 7 9 1 4 2 9 6 4 3 3 4 9 4 4 3 7 7 0 1 4 0 8 7 3 3 1 1 3 4 9 0 3 1 7 0 1 8 3 6 3 1 1 9 0 3 2 9 7 1 2 1 5 0 7 3 4 8 0 7 5 2 6 9 7 6 7
7 7 8 7 4 2 0 4 9 1 2 5 8 6 2 6 9 0 2 5 2 0 3 6 5 0 1 1 0 7 4 3 2 9 5 1 2 8 0 0 9 9 5 3 3 1 6 2 9 1 1 7 3 8 6 2 6 7 5 7 1 2 7 2 1 3 9 5 8 3 8 6 2 4 4 5 2 2 5 8 5 1 4 3 3
7 1 7 3 6 5 4 3 5 2 9 6 1 6 2 5 9 1 2 8 6 7 2 9 8 7 9 9 5 6 7 2 2 0 2 6 0 4 1 1 5 4 8 0 0 8 7 5 5 9 2 0 2 5 0 4 7 3 0 7 8 1 9 6 1 4 0 5 2 7 3 9 5 3 7 8 8 1 6 5 5 7 4 7 0
3 1 9 8 4 2 1 0 6 1 0 2 0 9 8 5 7 7 2 3 1 7 1 6 7 6 8 0 1 7 7 5 6 5 2 7 7 7 7 8 9 0 0 3 5 2 8 8 4 4 9 4 5 5 7 0 2 1 2 8 5 3 7 2 7 2 3 4 6 0 2 4 8 1 4 1 1 1 7 6 6 9 0 3 0
4 6 0 9 9 4 1 9 0 3 9 2 4 9 0 7 0 9 1 3 5 3 0 8 0 6 1 5 2 1 1 7 0 1 2 9 4 9 8 4 5 4 0 1 1 8 7 9 2 6 4 8 0 6 5 1 5 5 3 3 0 4 9 3 9 3 1 3 0 4 9 6 9 5 4 4 9 2 8 6 5 7 2 1 1
1 4 8 5 0 7 7 9 7 8 0 5 0 3 4 1 6 4 5 4 6 2 2 9 0 6 7 0 7 5 5 2 7 9 3 9 7 0 0 8 8 4 7 5 7 8 9 4 4 3 9 4 3 2 3 7 9 1 4 6 4 1 4 4 7 2 3 3 4 0 2 4 6 7 6 2 2 1 2 3 4 1 6 7 2
8 3 4 8 4 6 7 6 8 5 3 7 8 8 9 0 6 2 3 7 3 1 4 3 9 0 6 6 1 3 0 5 7 9 0 7 2 1 6 1 1 5 9 1 9 9 1 9 4 8 5 3 0 9 4 7 5 5 4 9 7 1 6 0 5 0 0 6 4 3 8 1 6 3 6 7 0 8 8 2 5 9 6 9 5
4 9 6 9 1 1 1 2 2 5 8 5 4 2 0 1 9 6 1 4 0 7 2 7 4 8 9 6 7 3 6 7 9 8 9 1 6 3 7 6 3 8 6 1 2 2 5 8 1 1 0 0 0 8 7 7 7 8 3 6 6 1 0 1 9 3 1 1 7 7 9 9 7 9 4 1 6 0 0 4 7 1 4 1 8
9 2 8 8 0 0 6 7 1 9 4 3 7 0 8 1 6 1 2 0 4 6 6 0 0 4 6 6 1 0 3 7 5 5 3 0 3 0 9 7 5 4 0 1 1 3 8 0 4 7 4 6 3 4 6 4 2 9 1 2 2 0 0 1 6 0 4 1 5 1 2 1 8 7 6 7 3 8 1 9 7 4 0 2 7
4 2 1 9 8 6 8 2 2 3 1 6 7 3 1 9 4 0 4 3 4 6 3 4 9 9 0 0 9 9 9 0 5 5 1 6 8 0 7 0 8 8 5 4 8 5 8 3 2 3 0 7 2 8 3 6 2 1 1 4 3 4 8 9 8

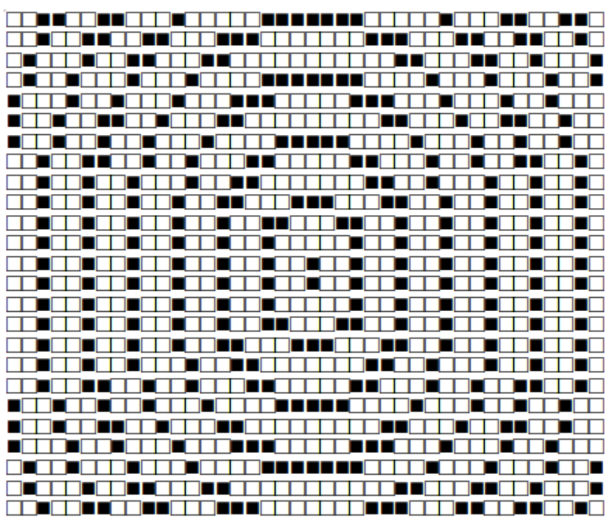
Angol szöveg

John Muir (/mjʊər/ MURE; April 21, 1838 - December 24, 1914),[1] also known as "John of the Mountains" and "Father of the National Parks",[2] was a Scottish-born American[3][4]:
42 naturalist, author, environmental philosopher, botanist, zoologist, glaciologist, and early advocate for the preservation of wilderness in the United States. His books,
letters and essays describing his adventures in nature, especially in the Sierra Nevada, have been read by millions. His activism helped to preserve the Yosemite Valley
and Sequoia National Park, and his example has served as an inspiration for the preservation of many other wilderness areas. The Sierra Club, which he co-founded, is a
prominent American conservation organization. In his later life, Muir devoted most of his time to his wife and the preservation of the Western forests. As part of the
campaign to make Yosemite a national park, Muir published two landmark articles on wilderness preservation in The Century Magazine, "The Treasure

Véletlen sorozat

1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 1 0 1
1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1
0 0 0 0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1
1 1 1 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 0 0 0 0 0 1 0 1 1 1 1 1 1 0 1 1 0 1 1 0 0 1 1 0 0 1 1 1 1 1 1 1 0 1 1 1
0 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1

Struktúrált sorozat



FÜGGELÉK II.

Minimális és maximális információmennyiség algoritmusai.

```
public class MinInfo{

        public double minInfo(Collection values){
                if (values == null) {
                        return 0;
                }
                if (values.isEmpty()) {
                        return 0;
                }
                if (values.size() == 1) {
                        return 1;
                }
                return Math.log(values.size() + 1) / Math.log(2);
        }

}

public class MaxInfo{

        public double maxInfo(Collection values){
                if (values == null) {
                        return 0;
                }
                if (values.isEmpty())
                {
                        return 0;
                }
                if (values.size() == 1) {
                        return 1;
                }
                Set atomicSet = new HashSet<>(values);
                int k = atomicSet.size();
                int n = values.size();
                double v = n * Math.log(k) / Math.log(2);
                if (v > 500) {
                        return v;
                }
                if (k == 1) {
                        return Math.log(n + 1) / Math.log(2);
                }
                return Math.log(
                        (Math.pow(k, n + 1) - 1) / (k - 1)) / Math.log(2);
        }

}
```

FÜGGELÉK III.

Mintázat módosított Shannon-információjának algoritmusa.

```
public class ModifiedShannonInfo{

        public double modifiedShannonInfo(Collection values) {
                if (values == null || values.isEmpty()) {
                return 0;
                }
                if (values.size() == 1) {
                        return 1;
                }
                Map<Object, Double> map = new HashMap<>();
                for (Object x : values) {
                        Double frequency = map.get(x);
                        if (frequency == null) {
                                map.put(x, 1.0);
                        } else {
                                map.put(x, frequency + 1);
                        }
                }
                int n = values.size();
                if (n > 100) {
                        return shannonInfo.value(values);
                }
                if (map.size() == 1) {
                        return Math.log(n + 1) / Math.log(2);
                }
                for (Object x : map.keySet()) {
                        map.put(x, map.get(x) / n);
                }
                double info = 0;
                for (int i = 0; i < n; i++) {
                        double p = 1;
                        for (Object x : map.keySet()) {
                                double f = map.get(x);
                                p *= Math.pow(f, -i * f);
                        }
                        info += p;
                }
                return Math.log(info) / Math.log(2);
        }

}
```

FÜGGELÉK IV.

Mintázat mintázat GZip tömörítési algoritmussal mért információjának algoritmusa.

```
public class GZipInfo{

        private final MinInfo minInfo = new MinInfo();
        private final MaxInfo maxInfo = new MaxInfo();

        public double gZipInfo(Collection values) {
                if (input == null || input.size() <= 1) {
                return 0;
                }

                byte[] values = ObjectUtils.serialize(input);

                double gzipInfo = ArrayUtils.toGZIP(values).length * 8;

                double min = minInfo.minInfo(input);
                double max = maxInfo.maxInfo(input);

                double minGzipInfo = ArrayUtils.toGZIP(
                        new byte[values.length]).length * 8;
                double maxGzipInfo = ArrayUtils.toGZIP(
                        generateRandomByteArray(values)).length * 8;

                if (originalMax == originalMin) {
                        return (newMin + newMax) / 2;
                }

                return newMin + ((gzipInfo - minGzipInfo)
                        / (maxGzipInfo - minGzipInfo)) * (max - min);
        }

}
```