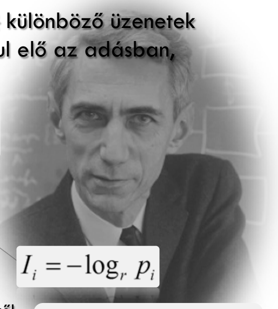
Információ, Kódolás, Tömörítés

Berke József – Szabó Rita

2013

Az Információ Mérése



Legyen m pozitív egészre $\{a_1, a_2, ..., a_m\}$ különböző üzenetek halmaza. Ha az a_i üzenetet k_i-szer fordul elő az adásban, akkor…

- a_i (gyakorisága) = k_i
- a_i relatív gyakorisága: $p_i = \frac{k_i}{n}$

A jel információtartalma: $\sum_{i=1}^{m} p_i = 1.$

Az a_i üzenet egyedi információtartalma: $I_i = -\log_r p_i$

…ahol r egy egynél nagyobb valós szám az információ egysége, r = 2 esetén beszélünk bitekről

Az üzenetek átlagos információtartama: $H = \sum_{i=1}^{m} p_i \, ld\left(\frac{1}{p_i}\right)$

Entrópia

Claude E. Shannon (1948)

Az Információ Mérése

PÉLDA: Vegyünk egy hat oldalú dobókockát. Mekkora az átlagos előfordulási valószínűsége az egyes dobásoknak?

Ekkor az egyes dobások előfordulási valószínűsége:

$$p_1 = p_2 = p_3 = p_4 = p_5 = p_6 = \frac{1}{6}$$

$$H = \sum_{i=1}^{m} p_i \log_2\left(\frac{1}{p_i}\right) = -\sum_{i=1}^{6} \frac{1}{6} * \log_2 \frac{1}{1/6}$$

$$= -6\left(\frac{1}{6} * \log_2\left(\frac{1}{6}\right)^{-1}\right) = \log_2 6 \approx 2,6$$

Ha a jelkészlet minden eleme azonos valószínűséggel fordul elő, akkor az entrópia értéke maximális.

$$H_{max} = \log_2 n$$

$$H = \sum_{i=1}^{m} p_i \log_2\left(\frac{1}{p_i}\right)$$

m - üzenetek száma
i - az üzenet sorszáma
p_i - az i-edik üzenet előfordulási valószínűsége

Entrópia Általános Matematikai Definíciója

Rényi Alfréd (1961)

$$H_\alpha(X) = \frac{1}{1-\alpha}\log\left(\sum_{i=1}^{n} p_i^\alpha\right)$$

ahol $\alpha \geq 0$ és $\alpha \neq 1$

H=0

H=1

Mi a kódolás ?

A kód egy olyan leképezési eljárás, amely egy jelkészletet egy másik jelkészletbe visz át, s a kódolás ennek az eljárásnak a végrehajtása.

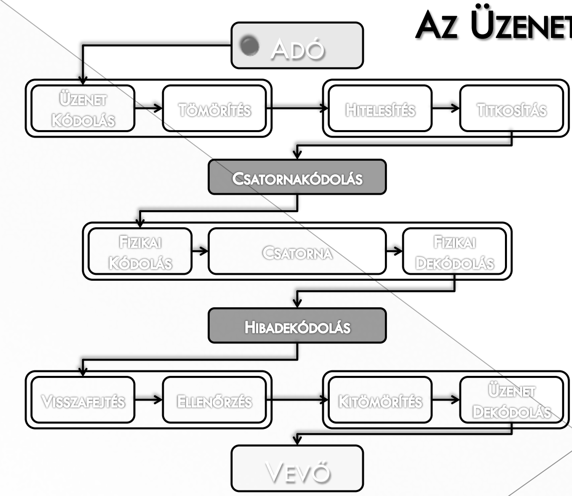
A kódolással ellentétes eljárás a dekódolás.

A kódolás céljai

Az üzenet…

- zavarérzékenységének csökkentése,
- és olyan jelekké alakítása, amely gazdaságosan tárolható és továbbítható,
- esetleges titkosítása.

Az üzenet útja



Kódelmélet alapfogalmak

Tekintsünk egy úgynevezett kimeneti ábécét. Jelöljük így:

(Ami lehet akár a magyar ábécé betűinek halmaza is.)

$$A = \{a_1, a_2, \dots, a_n\}$$

Betű szerinti kódolás

A kódolásnak azt a formáját, amelynél a kimeneti ábécé minden betűjének egy bináris sorozatot feleltetünk meg. A bináris sorozatot jelöljük így:

$$K = \{k_1, k_2, \dots k_n\}$$

A K halmazt bináris kódnak, elemeit kódszavaknak, a kódszavak tetszőleges sorozatát pedig kódolt közlésnek nevezzük.

Betű szerinti kódolás

PÉLDA: Legyen a kimeneti ábécé:

$$A = \{á, d, k, l, o, ó, s\}$$

A bináris kód pedig:

$$K = \{000, 001, 010, 011, 100, 101, 110\}$$

Ebben az esetben a ‘kódolás’ sorozatnak (elsődleges közlésnek) a betű szerinti kódolás eljárás függvénye:

$$f : A^* \to K^*$$

A így kapott kódolt közlés: $010\ 101\ 001\ 100\ 011\ 000\ 110$

A dekódolást az eljárási függvény inverzével végezhetjük el:

$$f^{-1} : K^* \to A^*$$

Felbonthatósági feltételek

Felbonthatóság: A kódhalmazt felbonthatónak nevezzük, ha tetszőleges bináris sorozat legfeljebb egyféleképpen bontható kódszavak szorzatára.

Prefix kód: Ha a kódhalmaz egyetlen kódszava sem valódi prefixuma (kezdőszelete) egy másik kódszónak.

Minden prefix kód felbontható is!

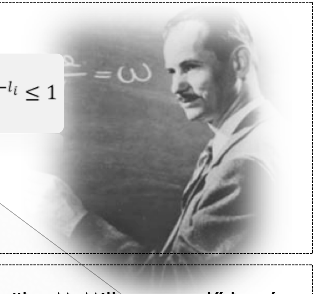
Példák: A felírt példák felbonthatók-e? Amennyiben felbonthatók prefixek-e?

K_1={110, 111, 001}	K_2={10, 100, 1000}	K_3={10, 101, 010}
Felbontható, mert minden kódszó 3bit hosszú, így a tetszőleges üzenetet hármasával kell felbontani. Prefix.	Felbontható, mert az 1-esek jelzik a kódszó elejét. Nem prefix, mert a k_2 elem első két bitje megegyezik a k_1 elemmel és a k_3 elem első 3 bitje megegyezik a k_2 elemmel.	Nem felbontható, mert az 101010 sorozatot felbonthatjuk 10|10|10 és 101|010 alakban is. Nem prefix mert a k_2 elem első 2 bitje megegyezik a k_1 elemmel.

Felbonthatósági feltételek

McMillan - egyenlőtlenség:
Ha K kód felbontható, akkor: $\sum_{i=1}^{n} 2^{-l_i} \leq 1$

Nem szükséges feltétele a felbonthatóságnak, megadható olyan nem felbontható kód, ami eleget tesz az egyenlőtlenségnek!



Kraft - egyenlőtlenség: Ha teljesül a McMillan egyenlőtlenség, akkor létezik olyan $K=\{a_1, a_2, \ldots, a_n\}$ prefix kód, melyben a kódszavak hossza rendre $l_1, l_2, \ldots, l_n$.

Optimális kódok

Tételezzünk fel egy jelforrást, amely a jeleket egymás után véletlenszerűen bocsájtja ki. Jelöljük így: $F = \{a_1, a_2, \ldots, a_n\}$

Átlagos kódhossz

Változó hosszúságú kódok esetén is meghatározhatjuk az átlagos kódhosszt.

$$L(K) = \sum_{i=1}^{n} p_i\, l_i$$

Optimális kód

A K^0 felbontható kódot optimálisnak nevezzük, ha tetszőleges K felbontható kódnak az F mellett számított L(K) átlagos kódhossza nem kisebb, mint $L(K^0)$.

$$L(K) \geq L(K^0)$$

Tetszőleges F forráshoz mindig létezik optimális prefix kód!

Optimális kódok

Tételezzünk fel egy jelforrást, amely a jeleket egymás után véletlenszerűen bocsájtja ki. Jelöljük így: $F = \{a_1, a_2, \ldots, a_n\}$

Shannon – tétele zajmentes csatornákra

Egy F jelforráshoz tartozó tetszőleges K felbontható kódra igaz, hogy:

$$L(K) \geq H(F)$$

Az F forráshoz tartozó K^0 optimális kód átlagos kódhosszára igaz az alábbi egyenlőtlenség:

$$L(K^0) \leq H(F) + 1$$

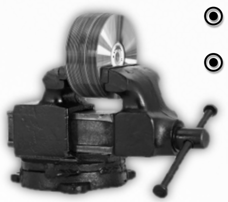
A kód hatásfoka

Egy F forráshoz tartozó K felbontható kód hatásfoka:

$$\eta_{FK} = \frac{H(F)}{L(K)}$$

A Tömörítés Céljai

A képi adatokat …



- egzaktul,
- minimális veszteséggel,
- és minél kisebb adat-mennyiséggel

…lehessen ábrázolni.

Analóg Tömörítési Eljárás

Promlémái:
- információveszteség
- jelhűség csökkenése
- zaj növekedése

- Szögmoduláció
  - Fázismoduláció (PM)
  - Frekvenciamoduláció (FM)
- Amplitúdómoduláció (AM)
  - Egy oldalsávos moduláció (SSB, vagy SSB-AM), módosított változata az egy oldalsávos, elnyomott vivőjű moduláció (SSB-SC)
  - Vestigial-oldalsávos moduláció (VSB, vagy VSB-AM)
- Szigma-delta moduláció (∑Δ)

A Tömörítési Arány

$$\text{Tömörítési arány} = \frac{\text{Tömörítetlen fájl mérete}}{\text{Tömörített fájl mérete}}$$

Példa: Ha az eredeti tömörítetlen képem mérete 600 kByte, a tömörített képfájl mérete 20 kByte, akkor a tömörítési arány:

$$\text{Tömörítési arány} = \frac{600}{20} = \frac{30}{1} = 30:1$$

Redundanciák I.

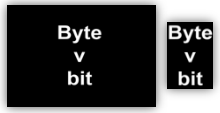
Az adattömörítés *(data compression)* kifejezés azt a folyamatot jelöli, mely bizonyos digitális információt reprezentáló adatok mennyiségét csökkenti.

Adat ≠ Információ

Arra az adathalmazra, mely nem a lehető legkisebb mennyiségű adattal jellemzi az információt azt mondjuk, hogy redundáns.

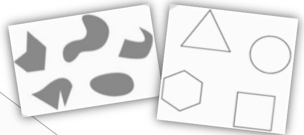
- Csökkentésével növelni lehet az átvitel sebességét.
- Növelésével javítani lehet az átviteli biztonságot.

Redundanciák II.



- **Kódolási redundancia**

  Pl.: vegyünk egy képet, melyen a képpontértékek ábrázolása 1 byte-on történik, ugyanakkor az ábrázolt képen összesen kétféle képpontérték található: 0 és 255



- **Képi redundancia**

  A kép többféle olyan belső összefüggéssel, szabályszerűséggel is rendelkezhet, melynek kihasználása esetén az ábrázolásához kevesebb adat is elegendő.

- **Pszichovizuális redundancia**

  Számos esetben a képen olyan információ is található, amely az emberi látás számára nem hordoz információt vagy felesleges, így nem vesz részt az emberi agy által végzett képérzet létrehozásában.

Eljárások csoportosítása

Az adathűség alapján megkülönböztetünk:

Veszteséges	Veszteségmentes
• Nem teszi lehetővé az eredeti képtartalom maradéktalan helyreállítását. • A leggyakoribb elvárás a látvány változatlansága.	• A bemeneti és kimeneti kép összes képpontja azonos. • A folyamat során nincs információvesztés.

A képfeldolgozási eljárások alapján:

Pixel alapú	Prediktív	Transzformáció alapú	Egyéb
PCM	Delta	Zóna	Hibrid
Futamhossz	Vonali DPCM	Szintrevágás	Kéttónusú
Bit-sík	2D DPCM	Multi-dimenziós	Szín alapú
	Interframe	Adaptív	Vektor

Veszteségmentes tömörítés

I. Változó hosszúságú kódolás

/VLC - Variable Length Coding/

A tömörítés kizárólag a kódolási redundancia csökkentésén alapul.

A szokásos esetekben elérhető a 2:1 tömörítési arány; nagyon jó esetben - pl. szkennerrel beolvasott kétszintes szöveges oldalak esetén - a 30:1 arány is elérhető.

Huffman kódolás — Aritmetikai kódolás

I. Változó hosszúságú kódolás

a) Huffman kódolás

Előnyei:
- a legmagasabb tömörítési arányt ez adja
- visszafejtése egyértelmű

Hátrányai:
- dinamikus blokk esetén nagy a számításigénye

- legismertebb módszer
- pixel alapú
- képi, kódolási és pszichovizuális redundanciára épül

Minden egyes adathoz olyan kódot rendel, melynek hossza (bitszáma) fordítottan arányos az adathalmazban való előfordulásainak számával(relatív gyakoriságával).

bemenő kép → relatív gyakoriságok számítása → gyakoriságok rendezése → gyakoriságok összevonása → kódok hozzárendelése → kimenő kép

I. Változó hosszúságú kódolás

a) Huffman kódolás

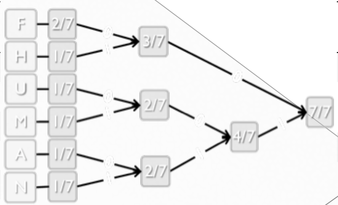
Példa: Kódoljuk a szöveget: **HUFFMAN**

I. Lépés: Adjuk meg a szimbólumok (betűk) előfordulási gyakoriságát és előfordulási valószínűségét.

II. Lépés: A legkisebb előfordulási valószínűségtől (alulról felfelé) kettős csoportokba vonjuk a gyakoriságokat. Ezt folytatjuk amíg az összes szimbólum sorra nem kerül. A kisebb gyakorisághoz 1-et, a nagyobbhoz 0-át rendelünk.

III. Lépés: A kifejtés irányát visszakövetve (jobbról balra) leolvassuk a szimbólumokhoz tartozó utat.

Szim.	Gyak.	Valósz.
H	1	1/7
U	1	1/7
F	2	2/7
M	1	1/7
A	1	1/7
N	1	1/7



Szim.	Kód
H	01
U	100
F	00
M	101
A	110

I. Változó hosszúságú kódolás

Előnyei:
- könnyen programozható
- nagy tömörítési arány érhető el
- az adatokhoz nemcsak egész számmal megadható hosszúságú kód tartozhat

b) Aritmetikai kódolás

bemenő kép
↓
relatív gyakoriságok számítása
↓
kódszavak hosszának meghatározása
↓
kódok kiosztása
↓
kimenő kép

- statisztikai alapú tömörítések használatakor és nagyméretű blokkok esetén használt módszer
- egy valószínűséget egy intervallummal ábrázolunk
  (forrás modellezés)

Egy nagy bit-folyamot állít elő, melyben a bemenő adatok és a kimenő adatok részletei közt nincs egy-egy értelmű megfeleltetés. Az eljárás a bemenő adatok sorozatának egy részletéhez rendel kódot.

II. Bit-sík kódolás

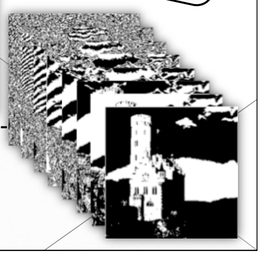
- pixel alapú
- kódolási redundanciára épül
- képpontértékek közti korrelációt is kihasználja
- 1.5:1 - 2:1 tömörítési arányt tesz lehetővé

Előnyei:
- hatékony kódtábla használat

Hátrányai:
- érzékeny a csatorna hibák okozta zajra

bemenő kép
↓
képsíkra bontás
↓
VLC kódolás
↓
kimenő kép

/8bit, grayscale/



IV. Előrebecsléses/prediktív/kódolás

A kép szomszédos részletei között nagymértékű a korreláció, azaz a kép adott részlete alapján annak közeli környezete becsülhető.

$$K(n) = I'(n) + d'(n)$$

bemenő kép
↓
kódolás típusának eldöntése
↓
kontúr kódok meghatározása
↓
kimenő kép

DPCM: Differenciális Impulzus Kód Moduláció

$I(n)$ – a bemenő képpontértékek,
$I'(n)$ – a kódolás előtti kódolt képpontértékek,
$d(n)$ – az előző két függvény előjel nélküli különbsége $|I(n)-I'(n)|$,
$d'(n)$ – a $d(n)$ függvény értékének kettes számrendszerben ábrázolt minimális bitjeinek a száma.

IV. Előrebecsléses/prediktív/Kódolás

DPCM: Differenciális Impulzus Kód Moduláció

Példa: Vegyünk egy olyan képet, ahol az egymás után következő szomszédos pontok intenzitásértéke rendre a következő: **100, 102, 103, 130, 130, 133, 128, 129**

n	I(n)	I'(n)	d(n)	d'(n)	K(n)	Δ(n)
0	100	-	-	-	100	0
1	102	100	2	1	101	1
2	103	101	2	1	102	1
3	130	102	28	5	107	23

$$K(n) = I'(n) + d'(n)$$

$$\Delta(n) = I(n) - K(n) = d(n) - d'(n)$$

$K(n)_1 = 100 + 1 = 101$

$K(n)_2 = 101 + 1 = 102$

Be:100, 102, 103, 130, 130, 133, 128, 129
Ki: 100, 101, 102, 107, 112, 117, 121, 124