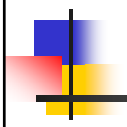

# Osnove informacij

TKO

## Informacija

Kaj je informacija ?

Na kaj vežemo informacijo ?

Kako je informacija merljiva ?

---

začetek informatike:
C.E. Shannon, Matematična teorija komunikacij, 1948

## Informacija o dogodku in verjetnost dogodka

Izhodišča:

- Informacija je vezana na verjetnost dogodka.
- Verjetnost dogodka merimo med 0 in 1:
  - p(neverjeten dogodek)=0
  - p(gotov dogodek)=1
- Gotovi dogodki ne nosijo informacije.
- Manj verjetni dogodki nosijo več informacije kot bolj verjetni dogodki.
- **Informacija o dogodku je obratno sorazmerna verjetnosti dogodka:**

$$I(A) \propto \frac{1}{p(A)}$$



## Merjenje informacije

- Verjetnost dveh neodvisnih dogodkov je enaka produktu verjetnosti dogodkov:
  - verjetnost dogodka A: p(A)
  - verjetnost dogodka B: p(B)
  - verjetnost dogodka A in B: p(A B)=p(A) p(B)
- Informacije o neodvisnih dogodkih se morajo seštevati.
- Naštete zahteve izpolnjuje **logaritemska mera**:

$$I(A) = \log_2 \frac{1}{p(A)} = -\log_2 p(A)$$

- Informacijo merimo v bitih: bit, kbit, Mbit, ...
- več bitov je združenih v besede: 8 bit =1 byte , 4 bit=nibble,..

## Povprečna infomacija izvora

- Izvor je generator množice N različnih znakov.
- Verjetnosti nastopanja posamičnih znakov lahko ugotovimo z merjenjem frekvenc dogodkov.
- Povprečna informacija je vsota informacij, ki so utežene z verjetnostjo dogodkov. Povprečno informacijo na izvoru imenujemo entropija izvora:

$$H = \sum_{i=1}^{N} p(A_i) I(A_i) = -\sum_{i=1}^{N} p(A_i) \log_2 p(A_i)$$

- Entropija je največja, če so vsi dogodki enako verjetni:

$$H_{\max} = \log_2(N)$$



## Met kocke

- Izvor je generator množice 6 različnih znakov.
  - K=1, K=2, K=3, K=4, K=5, K=6
- Vsi dogodki so enako verjetni: verjetnosti nastopanja posamičnih znakov so enake 1/6.
- Povprečna informacija pri metu kocke je približno 2,58 bita.

$$H = H_{\max} = \log_2(6) \approx 2,58$$

## Kodiranje znakov

Vsakemu znaku priredimo določeno lastno binarno kodo.

- Dolžina kode je lahko enaka za vse znake.
  - 7bit ASCII tabela znakov (128 znakov) " a b c "= 1100001 1100010 110011
- Če imajo kode znakov različne dolžine da krajše kode niso enake okrajšanim začetkom daljših kod: (npr: A=1011, B=10111).



## Morsejeva koda

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| A | .- | M | -- | Y | -.-- | 6 | -.... |
| B | -... | N | -. | Z | --.. | 7 | --... |
| C | -.-. | O | --- | Ä | .-.- | 8 | ---.. |
| D | -.. | P | .--. | Ö | ---. | 9 | ----. |
| E | . | Q | --.- | Ü | ..-- | . | .-.-.- |
| F | ..-. | R | .-. | Ch | ---- | , | --..-- |
| G | --. | S | ... | 0 | ----- | ? | ..--.. |
| H | .... | T | - | 1 | .---- | ! | --.-- |
| I | .. | U | ..- | 2 | ..--- | : | ---... |
| J | .--- | V | ...- | 3 | ...-- | " | .-..-. |
| K | -.- | W | .-- | 4 | ....- | ' | .----. |
| L | .-.. | X | -..- | 5 | ..... | = | -...- |

- Samuel Morse, električni telegraf, 1836
- Morsejeva koda ni binarna, saj poleg kratkega znaka . in dolgega znaka _ vsebuje še presledke različnih dolžin:
  - zelo kratek presledek med pikami in črtami v znaku, kratek presledek med znaki, presledek med besedami, dolg presledek med stavki.
  - brez presledkov kode zato ni mogoče dekodirati.
- . . . _ _ _ . . . ?

## Učinkovito brezizgubno kodiranje znakov

- Ravnanje "po občutku" :
  - Znakom ki pogosto nastopajo priredimo krajšo kodo.
  - Znakom ki redko nastopajo lahko priredimo daljšo kodo.

- Znanstvena utemeljitev: Znaki z veliko verjetnostjo nastopanja nosijo manj informacije, zato jih kodiramo z manj biti.

$$I(A) \propto \frac{1}{p(A)}$$



## Omejitve pri kodiranju ?

Dolžina kode je odvisna od števila znakov.

- Za kodiranje N znakov potrebujemo največ $\log_2(N)$ bitov.
- Povprečna dolžina kode je lahko tudi manjša !

- Če ne želimo izgubiti dela informacije, potem povprečna dolžina kode na izhodu kodirnika ne sme biti manjša od entropije izvora. Takšno kodiranje zato imenujemo tudi brezizgubno kodiranje. Primer brezizgubnega kodiranja je stiskanje datotek (zip).

- Pri kodiranju govora upoštevamo dejstvo, da vsa informacija ni enako pomembna (relevantna). Učinkoviti kodirniki za govor izločajo nepomemben del informacije, ki za poslušalca ni zaznavna. Izgubljeni nepomemben del informacije imenujemo irelevanca. Takšno kodiranje imenujemo izgubno kodiranje.

## Primer izvora

- Izvor je generator množice 4 različnih znakov: **a**,**b**,**c** in **d**
- Po štetju 1000 znakov ugotovimo, da :
  - znak **a** nastopa 500 krat ,
  - znak **b** nastopa 250 krat ,
  - znak **c** nastopa 125 krat in
  - znak **d** nastopa 125 krat.
- Verjetnosti nastopanja znakov so:
  p(a)=0.5, p(b)=0.25, p(c)=0.125 in p(d)=0.125
- Informacije o dogodkih so:
  I(a)=1, I(b)=2 , I(c)=3 in I(d)=3
- Povprečna informacija je enaka:
  H=0.5*1+0.25*2+0.125*3+0.125*3=1.75 [bit]
- Če bi vsi znaki nastopali z enako verjetnostjo, bi bila entropija izvora 2 bita.



## Redundanca

- Če ne poznamo verjetnosti nastopanja znakov, je povprečna dolžina kode odvisna samo od števila znakov:
  - štiri znake kodiramo z dvemi biti,
  - za kodiranje 128 znakov potrebujemo 7 bitov,
  - za kodiranje 1024 različnih znakov potrebujemo 10 bitov...
- Povprečna dolžina kode je pogosto daljša od povprečne informacije.
- Relativno število "odvečnih" bitov imenujemo redundanca.
- Uporabimo že navedeni primer izvora s štirimi znaki (a,b,c,d)
  - Izvor generira štiri znake, vsak znak kodiramo z dvemi biti.
  - Entropija izvora je enaka 1.75 bit.
  - Za 1000 znakov uporabimo 2000 bitov potrebujemo pa le 1750 bitov.
  - Razlika 250 bitov je "odvečna" informacija.
  - Vsak osmi bit je v povprečju odveč.
  - Redundanca je po definiciji enaka R=1-H/Hmax=1-1.75/2=0.125

## Entopijsko kodiranje izvora

- Za prej navedeni primer izvora lahko generiramo kodo, ki ima v povprečju manj kot dva bita na znak.
  - Verjetnosti nastopanja znakov so:
    p(a)=0.5, p(b)=0.25, p(c)=0.125 in p(d)=0.125
- Kode znakov (Huffmanove kode):
  - *a : 0*
  - *b : 10*
  - *c : 1 1 0*
  - *d: 1 1 1*
- Povprečna dolžina kode je enaka entropiji izvora:
  L=0.5*1+0.25*2+0.125*3+0.125*3=1.75 [bit]
- Zaporedje bitov lahko dekodiramo v niz znakov brez izgube informacije: *..01010111.. = ..a,b,b,d..*