INTRODUCERE ÎN COMPRESIA DATELOR

Compresia datelor se ocupă cu reprezentarea informației într-o formă compactă. Acest lucru se realizează prin identificarea şi extragerea redundanței din date. Datele pot fi caractere dintr-un fişier text, numere ce reprezintă eşantioane ale unor semnale vocale sau de imagine etc.

Exemple de compresie [2]:

- codul Morse, care alocă literelor ce apar mai des în cuvinte mai puține semne (linie, punct) şi celor mai rare, mai multe semne;

- alfabetul Braille, care foloseşte tablouri de puncte de dimensiuni 2x3, în care punctele sunt distribuite în funcție de probabilitățile de apariție a literelor în cuvinte.

În aceste exemple de compresie s-a folosit *structura statistică a cuvintelor,* dar mai există şi alte caracteristici ale semnalelor ce pot fi folosite în scopul realizării compresiei. Astfel, în timpul vorbirii, construcția fizică a cavității bucale dictează tipul de sunete ce se produc, adică mecanica producerii vorbirii impune structura vorbirii, prin urmare, în loc de a transmite vorbirea însăşi, s-ar putea transmite informații cu privire la conformația cavității bucale, care ar putea fi folosite la recepție pentru a sintetiza vorbirea. În acest sens, o cantitate adecvată de informație privind conformația cavității bucale poate fi reprezentată mai compact decât numerele ce reprezintă eşantioanele semnalului vocal.

În compresie mai pot fi folosite caracteristicile utilizatorului datelor. De multe ori, când se transmit semnale sonore sau vizuale, acestea sunt destinate percepției umane, dar oamenii au posibilități de percepție limitate. De exemplu, oamenii nu pot auzi frecvențe foarte înalte, pe care, însă, unele animale le pot percepe. Aşadar, nu are rost a transmite informații ce nu pot fi percepute, astfel încât informațiile irelevante nu mai trebuie transmise.

Cu toate că în ultimul timp capacitatea de stocare şi transmitere a informației a crescut mult (de exemplu compact discul şi fibra optică), aceasta nu răspunde pe deplin cerințelor practice, motiv care justifică necesitatea compresiei fişierelor de dimensiuni mari ce trebuie stocate sau transmise. În cazul stocării, pentru a creşte volumul de date ce poate fi stocat, cum este cazul fişierelor grafice, se folosesc metode care comprimă datele la stocare şi le decomprimă la redare. În cazul transmiterii, lățimea de bandă a semnalelor digitale este foarte mare, în comparație cu a canalelor disponibile, astfel încât compresia datelor la emițător este absolut necesară pentru a putea folosi canalul. La recepție are loc operația de decompresie. De exemplu, semnalul TV de înaltă definiție, HDTV, necesită pentru transmiterea fără compresie aproximativ 884 Mbiți/secundă. Pentru a transmite această cantitate mare de informație, ar fi necesar un canal cu banda de cca. 220 MHz. Folosind compresia, este necesar a transmite mai puțin de 20 Mbiți /sec, care, împreună cu semnalul audio, pot fi transmişi într-o bandă de 6 MHz.

Tehnici de compresie şi decompresie

Tehnicile sau algoritmii de compresie şi de decompresie cuprind doi algoritmi: unul care furnizează reprezentarea X_c pe mai puțini biți a

semnalului de intrare X şi algoritmul de reconstrucție care operează asupra lui X_c, pentru a produce semnalul reconstruit $\hat{X}$.

Schemele de compresie se împart în două clase:

- compresie fără pierderi, când $X = \hat{X}$;

- compresie cu pierderi, când $X \neq \hat{X}$. În acest caz se realizează o compresie mai mare.

Compresia fără pierderi

După cum arată numele, aceasta nu implică pierderea de informație, datele inițiale fiind refăcute exact din cele compresate. Această compresie se foloseşte de obicei pentru semnale discrete ca text, date generate de calculator, unele tipuri de informație video, deoarece refacerea exactă este esențială în cazul textelor (sensul comunicării), unor imagini (imagistică pentru diagnostic), numere (comunicări bancare) etc.

Compresia cu pierderi

Compresia cu pierderi conduce la o rată de compresie superioară cu prețul pierderii de informație, datele care au fost supuse compresiei neputând fi refăcute exact. Această compresie este mai eficientă când se aplică semnalelor imagistice sau sonore, caz în care pierderea de informație din afara percepției vizuale sau auditive umane poate fi tolerată. Multe tehnici de compresie cu pierderi pot fi ajustate la diferite nivele de calitate. Evident, creşterea acurateței semnalului refăcut se obține cu prețul unei compresii mai reduse. Volumul compresiei depinde atât de mărimea redundanței sursei, cât şi de eficiența reducerii acesteia.

Evaluarea compresiei

Un algoritm de compresie poate fi evaluat în funcție de:

- necesarul de memorie pentru implementarea algoritmului;
- viteza algoritmului pe o anumită maşină;
- raportul de compresie;
- calitatea reconstrucției.

De obicei, ultimele două criterii sunt esențiale în adoptarea algoritmului de compresie.

Uzual, performanțele compresiei pot fi exprimate prin raportul de compresie şi rata de compresie.

Raportul de compresie este raportul dintre numărul de biți necesar reprezentării datelor înainte şi după compresie.

Rata de compresie reprezintă numărul mediu de biți necesar reprezentării unui eşantion.

În compresia cu pierderi, versiunea reconstruită diferă de varianta originală şi, pentru a determina eficiența algoritmului de compresie, trebuie să existe mijloace de apreciere a diferenței dintre semnalul original şi cel reconstruit. Diferența dintre semnalul original şi cel reconstruit se numeşte *distorsiune*. Tehnicile de compresie cu pierderi se folosesc de obicei pentru compresia semnalelor analogice, care se mai numesc forme de undă, motiv pentru care compresia semnalelor analogice mai poartă denumirea de *codarea formelor de undă.* În cazul codării semnalelor video şi sonore destinatarul este, de obicei, omul, al cărui răspuns este dificil de apreciat, motiv pentru care se folosesc măsuri aproximative de determinare a calității reconstrucției.

Alți termeni folosiți în aprecierea diferenței dintre semnalul original şi cel refăcut este *fidelitatea* şi *calitatea*. Acestea sunt cu atât

mai ridicate, cu cât diferența dintre versiunea reconstruită şi cea originală a sursei este mai mică.

Modelare şi codare

Cerințele reconstrucției sunt cele ce impun dacă compresia este cu sau fără pierderi, schema exactă depinzând de un număr de factori. Unii din cei mai importanți sunt impuşi de caracteristicile datelor destinate compresiei, de exemplu, o tehnică de compresie poate fi eficientă pentru compresia unui text, dar total ineficientă pentru imagini. Fiecare aplicație prezintă particularități specifice.

Dezvoltarea algoritmilor de compresie pentru o varietate de date cuprinde două faze:

- prima fază se referă de obicei la *modelare,* când se încearcă extragerea informației despre orice redundanță din date şi descrierea acesteia sub forma unui model;

- a doua fază este *codarea*.

Diferența dintre date şi model se numeşte *secvență reziduală.*

În continuare sunt prezentate câteva exemple de modelare a datelor.

Exemplul 1.

Fie secvența de eşantioane

$\{x_n\} = \{9,11,11,11,14,13,15,17,16,17,20,21\}$, pentru $n = \overline{1,12}$.

Dacă se doreşte a se transmite sau stoca reprezentarea binară a acestor numere, ar fi necesari 5 biți/eşantion. Un model sugerat de reprezentarea eşantioanelor secvenței, care ar necesita mai puțini biți, ar putea fi o dreaptă, descrisă de ecuația:

$$\hat{x}_n = n + 8; \qquad n = 1, 2, \ldots$$

În Fig. 1 s-au reprezentat eşantioanele secvenței.

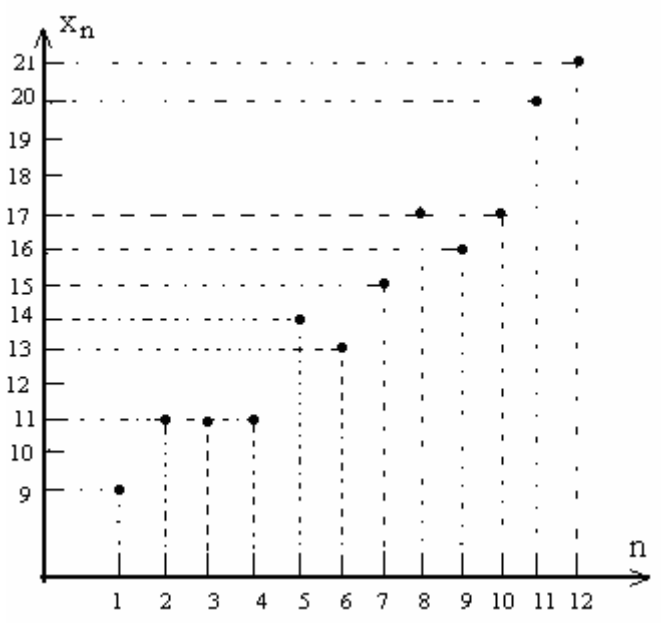

Fig. 1

Diferența dintre date şi model, adică eroarea reziduală, este dată de secvența:

$$e_n = x_n - \hat{x}_n = \{0, 1, 0, -1,\ 1, -1, 0, 1, -1, -1, 1, 1\}$$

Secvența reziduală conține numai trei valori, numerele $\{-1, 0, 1\}$, care pot fi codate cu 2 biți, de exemplu:

$$-1 \rightarrow 00$$
$$0 \rightarrow 01$$
$$1 \rightarrow 10$$

şi, prin urmare, se obține compresia prin transmiterea sau stocarea parametrilor modelului şi secvența reziduală.

Exemplul 2.

Fie secvența $x_n = \{27, 28, 29, 28, 26, 27, 29, 28, 30, 32, 34, 36, 38\}$, cu reprezentarea din Fig. 2.

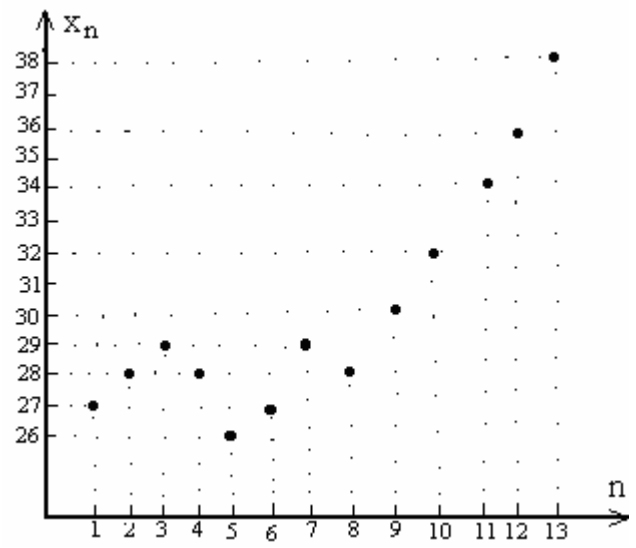

Fig. 2

Se observă că secvența nu urmează o lege simplă, ca în cazul precedent, în schimb, fiecare valoare este apropiată de precedenta. Se presupune că se transmite prima valoare, apoi se transmite diferența față de valoarea precedentă.

Astfel, secvența transmisă ar fi: 1, 1, -1, -2, 1, 2, -1, 2, 2, 2, 2, 2. Pentru transmiterea acestei secvențe sunt necesare 4 numere {-1, 1, -2, 2}, care pot fi codate cu 2 biți, după cum urmează:

$-1 \rightarrow 00$

$1 \rightarrow 01$

$-2 \rightarrow 10$

$2 \rightarrow 11$

La fel ca în cazul precedent, numărul valorilor distincte a fost redus, fiecare număr fiind reprezentat pe un număr mai mic de biți, efectuându-se astfel compresia. Decodorul adaugă fiecare valoare recepționată la valoarea precedentă decodată pentru a obține secvența reconstruită.

Tehnicile care folosesc valorile trecute ale secvenței pentru a estima valoarea curentă şi apoi codează eroarea de predicție sau reziduală se numesc *predictive.*

În cazul redundanței de tip statistic, sursele generează mesaje cu diferite probabilități, caz în care este avantajos a asigna coduri binare de lungimi diferite, diferitelor simboluri.

Exemplul 3.

Fie secvența de mesaje m a r e a ␣ e ␣ t a a a r e ␣ t a a a a r e ␣ m a r e, tipică pentru toate secvențele generate de sursă. Se observă că secvența este formată din şase mesaje diferite. Pentru a reprezenta şase mesaje, sunt necesari 3 biți/mesaj. În loc de a proceda astfel, se poate folosi codul din Tabelul 1.

Tabelul 1

a	1
e	000
r	001
␣	010
m	0110
t	0111

Se observă că s-a atribuit cuvântul cu un singur bit mesajului care apare cel mai des si cel mai lung cuvânt mesajului care apare cel mai rar. Cu acest cod rezultă 75 biți pentru întreaga secvență. Cum sunt

27 mesaje în secvență, rezultă că sunt atribuiți 2,7 biți/mesaj. Aceasta înseamnă un raport de compresie 1,08. În cazul textelor, împreună cu redundanța statistică, există redundanță în forma cuvintelor care se repetă mai des. Avantajul acestui tip de redundanța constă în posibilitatea construirii unei liste de cuvinte şi apoi să se reprezinte poziția lor în listă. Acest tip de compresie se numeşte *tehnică de dicționar*. Adesea, structura sau redundanța datelor devine mai evidentă, dacă se consideră grupuri de mesaje.

Preliminarii matematice pentru compresia fară pierderi

Shannon a definit informația ataşată unui eveniment A, care se produce cu probabilitatea $p(A)$, ca fiind [87]

$$i(A) = -\log_x p(A) \tag{1}$$

Considerații asupra bazei logaritmului vor fi făcute ulterior. Cu alte cuvinte, cu cât probabilitatea unui eveniment este mai mică, cu atât informația adusă de acesta este mai mare, şi invers. Caracterul subiectiv al unui eveniment este greu măsurabil, motiv pentru care informația asociată unui eveniment va fi măsurată matematic cu relația (1), ea fiind eliberată de caracterul său semantic.

O altă proprietate a acestei definiții matematice a informației este că informația obținută la producerea a doua evenimente independente este suma informațiilor obținute la producerea evenimentelor individuale.

$$i(AB) = i(A) + i(B) \tag{2}$$

Unitatea de măsură a informației depinde de baza logaritmului.

Pentru

baza 2 $\rightarrow$ unitatea de măsură se numeşte bit;

baza e $\rightarrow$ unitatea de măsură se numeşte nat;

baza 10 $\rightarrow$ unitatea de măsură se numeşte hartley.

De obicei se foloseşte baza 2, care nu este disponibilă în calculator. Pentru a folosi baza *e* cu care operează calculatoarele, se ține cont că

$$\log_2 x = \frac{\ln x}{\ln 2}.$$

Dacă se dispune de o mulțime de evenimente independente A_i, care reprezintă mulțimile realizărilor unui eveniment S, astfel încât

$$\bigcup A_i = S, \tag{3}$$

atunci informația medie asociată evenimentului este dată de

$$H = \sum_i p(A_i) i(A_i) = -\sum_i p(A_i) \log_x p(A_i) \tag{4}$$

care se numeşte *entropia* asociată experimentului.

Shannon a arătat că dacă experimentul este o sursă care furnizează mesajele A_i dintr-o mulțime A, atunci entropia este o măsură a numărului mediu de simboluri binare necesare codării ieşirii sursei.

De asemenea, Shannon a arătat că un algoritm de compresie fără pierderi optim poate coda mesajele sursei cu un număr mediu de biți, cel puțin egal cu entropia sursei [87].

Mulțimea mesajelor, numite şi mesaje, defineşte *alfabetul sursei.* Pentru o sursă S cu alfabetul $A = \{1, 2, \ldots m\}$ care generează secvența $\{x_1, x_2, \ldots, x_n\}$, entropia este [1], [34]:

$$H(S) = \lim_{n\to\infty} \frac{1}{n} G_n \tag{5}$$

unde

$$G_n = -\sum_{i_1=1}^{m} \sum_{i_2=1}^{m} \ldots \sum_{i_n=1}^{m} p(x_1 = i_1, x_2 = i_2, \ldots, x_n = i_n) \log p(x_1 = i_1, \ldots, x_n = i_n)$$

Dacă fiecare element din secvență este independent şi identic distribuit, (i.i.d), rezultă atunci

$$G_n = -n\sum_{i_1=1}^{m} p(x_1 = i_1)\log p(x_1 = i_1) \tag{6}$$

Modelul statistic i.i.d. înseamnă că variabilele aleatoare sunt independente şi sunt caracterizate de aceeaşi lege de repartiție.

În acest caz entropia devine:

$$H(S) = -\sum_{i_1=1}^{m} p(x_1 = i_1)\log p(x_1 = i_1) = -\sum_{i_1=1}^{n} p(x_i)\log p(x_i) \tag{7}$$

Pentru multe surse relațiile (5) şi (7) nu sunt identice. Pentru a face deosebire între ele, cantitatea calculată cu (7) se numeşte *entropia de ordinul întâi* a sursei, iar (5) *entropia sursei*.

În general, nu este posibil a cunoaşte entropia unei surse fizice, aceasta trebuind estimată. Estimarea entropiei depinde de presupunerile asupra statisticii sursei.

Exemplul 4.

Fie secvența de eşantioane:

{1, 2, 3, 2, 3, 4, 5, 4, 5, 6, 7, 8, 9, 8, 9, 10}

Pentru această secvență probabilitățile de apariție pentru fiecare număr sunt

$$p(1) = p(2) = p(7) = p(10) = \frac{1}{16}$$

$$p(2) = p(3) = p(4) = p(5) = p(8) = p(9) = \frac{2}{16}$$

Presupunând secvența i.i.d., entropia sa este aceeaşi cu entropia de ordinul 1, dată de relația (7).

$$H(S) = -\sum_{1}^{16} p(i) \log p(i) = 3,25 \text{ biți}$$

Aceasta înseamnă că cea mai bună schemă de codare pentru această secvență o poate coda la 3,25 biți/mesaj. Dacă se presupune că există corelație între mesajele sursei şi aceasta se înlătură, reținând numai diferența între valorile eşantioanelor vecine, se ajunge la secvența reziduală:

{1, 1, 1, -1, 1, 1, 1,-1, 1, 1, 1, 1, 1, –1, 1}

Această secvență este constituită numai din două valori 1 şi –1, cu probabilitățile:

$$p(1) = \frac{13}{16} \quad \text{şi} \quad p(-1) = \frac{3}{16}$$

În acest caz entropia este 0,7 biți /mesaj. Evident, cunoscând numai această secvență nu se va putea reconstrui originalul. Receptorul trebuie să cunoască procesul prin care a fost generată secvența reziduală din cea originală. Procesul depinde de presupunerile asupra redundanței sau structurii secvenței. Aceste presupuneri determină *modelul* pentru secvență.

În exemplul considerat modelul este

$$x_n = x_{n-1} + r_n$$

unde x_n este al *n*-lea element al secvenței originale şi r_n, al *n*-lea element al secvenței reziduale. Acest model este *static*, deoarece rămâne invariant, oricare ar fi *n*. Modelul care se modifică odată cu *n* se numeşte *adaptiv*.

Exemplul 5.

Fie următoarea secvență:

1, 2, 1, 2, 3, 3, 3, 3, 1, 2, 3, 3, 3, 3, 1, 2, 3, 3, 1, 2.

Evident, în aceste date există o anumită redundanță. Dacă se consideră fiecare mesaj separat, este dificil de extras redundanța din date. Pentru acest caz,

$$p(1) = p(2) = \frac{1}{4}; \quad p(3) = \frac{1}{2};$$

rezultă $H(S)$=1,5 biți/mesaj.

Secvența conține 20 de mesaje, deci în total sunt necesari 1,5 x 20 = 30 de biți pentru a reprezenta secvența.

În continuare se consideră secvența observată în blocuri de lungime 2. Astfel, se observă că sunt numai două mesaje 12 şi 33 cu probabilitățile:

$$p(12) = \frac{1}{2} \; ; \quad p(33) = \frac{1}{2};$$

rezultând $H(S)$=1 bit /mesaj

Cum în secvență sunt 10 astfel de grupe de două mesaje, rezultă un necesar de 10 biți pentru a reprezenta întreaga secvență. Teoria spune că întotdeauna se poate extrage redundanța din date, prin considerarea blocurilor de date din ce în ce mai mari, dar în practică există limitări la această abordare.

Pentru a evita aceste limitări, se va încerca obținerea unui model cât mai exact pentru date şi să se codeze sursa conform modelului.

Modele

Un model bun pentru date conduce la algoritmi de compresie eficienți. Pentru a dezvolta algoritmi care efectuează operații matematice asupra datelor, acestea trebuie modelate matematic.

Modele fizice

Dacă se cunoaşte ceva despre mecanismul de generare a datelor, se poate folosi această informație pentru construirea modelului. De exemplu, în aplicațiile referitoare la vorbire, cunoaşterea mecanismului de producere a vorbirii poate fi folosit la construirea unui model matematic pentru procesul vorbirii eşantionate.

Modele probabilistice

Cel mai simplu model statistic pentru sursă este de a presupune că fiecare mesaj furnizat de sursă este independent de celelalte şi fiecare se produce cu aceeaşi probabilitate. Acesta este numit *model de ignoranță* si ar putea fi util când nu se cunoaşte nimic despre sursă.

Următorul pas în creşterea complexității modelului este de a păstra presupunerea asupra independenței, dar de a înlătura presupunerea de probabilitate egală pentru mesaje. În acest caz se alocă fiecărui mesaj o probabilitate în funcție de frecvența de furnizare a mesajului respectiv.

Pentru o sursă care generează mesaje dintr-un alfabet

$$S = \{s_1, ..., s_M\}$$

se poate folosi *modelul de probabilitate*

$$P = \{p(s_1), p(s_2), ..., p(s_M)\}$$

Cu acest model se poate calcula entropia sursei cu relația (7) şi pot fi construite coduri eficiente pentru reprezentarea mesajelor din *S*, adică se poate folosi un număr mediu minim de biți pentru fiecare mesaj. Dacă se renunță la presupunerea de independență, se pune problema găsirii unui mod de a descrie dependența datelor, aşa cum este în cazul modelelor Markov.

Modele Markov

Unul dintre cele mai răspândite moduri de reprezentare a dependenței între date este folosirea modelului Markov [49]. Pentru modelarea datelor, în compresia fără pierderi se foloseşte un model particular numit *lanț Markov discret*.

Fie $\{x_n\}$ secvența observată. Aceasta secvență se zice că este un model Markov de ordin *k*, dacă:

$$p(x_n \mid x_{n-1},...x_{n-k},..) = p(x_n \mid x_{n-1},....x_{n-k}), \tag{8}$$

adică simbolul x_n depinde numai de ultimele *k* mesaje $x_{n-1},...,x_{n-k}$.

Mulțimile $\{x_{i-1},...,x_{i-k}\}$, $i = n, n-1,...$, reprezintă *stările procesului*. Dacă mărimea alfabetului sursei este *l*, numărul de stări este l^k. Cel mai des folosit model Markov este cel de ordinul 1, pentru care:

$$p(x_n \mid x_{n-1}, x_{n-2}....) = p(x_n \mid x_{n-1}) \tag{9}$$

Relațiile (8) şi (9) indică existența dependenței între eşantioane, dar modul în care a fost introdusă această dependență nu este explicit. Se pot dezvolta diferite modele de ordinul 1, în funcție de presupunerile asupra modului de introducere a dependenței între eşantioane.

Dacă se presupune că dependența a fost introdusă liniar, secvența de date ar putea fi văzută ca ieşirea unui filtru liniar excitat cu zgomot alb. Ieşirea acestui filtru este dată de ecuația cu diferențe [36]:

$$x_n = \rho x_{n-1} + w_n \tag{10}$$

unde w_n este zgomot alb. Acest model se foloseşte frecvent la dezvoltarea algoritmilor pentru vorbire şi imagine.

Folosirea modelului Markov nu necesită prezumția de liniaritate. De exemplu, fie o imagine binară, care are numai două feluri de pixeli - albi şi negri. Se ştie că apariția unui pixel alb la observarea următoare

depinde în oarecare măsură dacă pixelul curent este alb sau negru. Prin urmare, se poate modela succesiunea de pixeli cu un lanț Markov.

Definind stările S_a şi S_n corespunzătoare cazului când pixelul curent este alb, respectiv negru, probabilitățile de tranziție $p(a/n)$ şi $p(n/a)$ şi probabilitățile fiecărei stări $p(S_a)$ şi $p(S_n)$, modelul Markov poate fi reprezentat ca în Fig. 3.

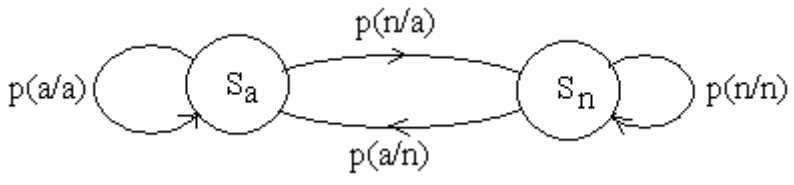

Fig. 3

$$p(a/a) = 1 - p(n/a)$$

În general, entropia unui proces cu $M = l^k$ stări finite, S_i, este valoarea medie a entropiei fiecărei stări:

$$H = \sum_{i=1}^{M} p(S_i) H(S_i); \quad M = l^k \tag{11}$$

$$H(S_i) = -\sum_{j=1}^{l} p(S_j \left| S_i) \log p(S_j \right| S_i); \tag{12}$$

unde $p(S_j \left| S_i\right.)$ este probabilitatea de trecere din starea *i* în starea *j*.

Folosirea modelelor Markov în compresia textelor

Modelele Markov sunt utile în compresia textelor, deoarece probabiltatea de apariție a unei litere este influențată de precedentele. Fie cuvântul "*ornitoring*". Se presupune că s-a procesat deja *ornitorin* şi

urmează a se coda următoarea literă. Dacă nu se ține seama de context şi se tratează litera ca o surpriză, probabilitatea să apară litera *g* este relativ scăzută. Dacă se foloseşte un model Markov de ordinul I, probabilitatea să urmeze *g* creşte considerabil. Cu creşterea ordinului modelului Markov (de la "n" la "in", la "rin" ş.a.m.d.) probabilitatea literei *g* devine mai mare, ceea ce conduce la o entropie scăzută.

Model de surse compuse

În multe aplicații nu este potrivit a se folosi un singur model pentru a descrie sursa, caz în care se poate defini o *sursă compusă*, care poate fi văzută ca o combinație de diverse surse, din care una este activă la un moment dat. Fiecare din acestea este descrisă de un model propriu.