

Реферат

по Разпознаване на образи

*2. Цифрово представяне на изображенията.
Трансформация на Фурие и теорема на Шенон.
Шум в изображенията.
Косинус трансформация*

*15. Марковски вериги и вероятностни автомати.
Скрити Марковски модели заразпознаване на говорни сигнали.
Понятие за кепстър на говорен сигнал.
Приложение при
разпознаване на ръкопис.*

Александра Михалева, M23571

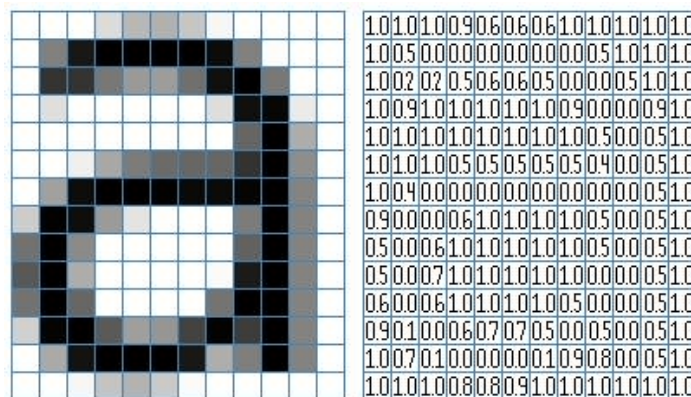
2. Цифрово представяне на изображенията. Трансформация на Фурие и теорема на Шенон. Шум в изображенията. Косинус трансформация

Цифрово представяне на изображенията

Цифровото изображение е числово (оибкновено двоично) представяне на двумерно изображение. Съществуват различни видове цифрово представяне на изображенията, от които по-важни са:

- Растерно

При растерното представяне изображението представлява grid от елементи наречени пиксели. И на всеки елемент е присвоена цифрова стойност, която съответства на цвета в съответния пиксел.



Фигура 1. Растерно представяне

- Векторно

При векторното представяне изображението е описано, чрез представяне на съдържанието му в математически вид - дефиниции на големина, позиция, форма и т.н.

Трансформация на Фурие

Трансформацията на Фурие служи за преобразуване на изображения от обектното пространство в честотното. Тя се основава на свойството, че всяка периодична функция може да бъде представена чрез безкрайна сума от синуси и косинуси, с различна честота:

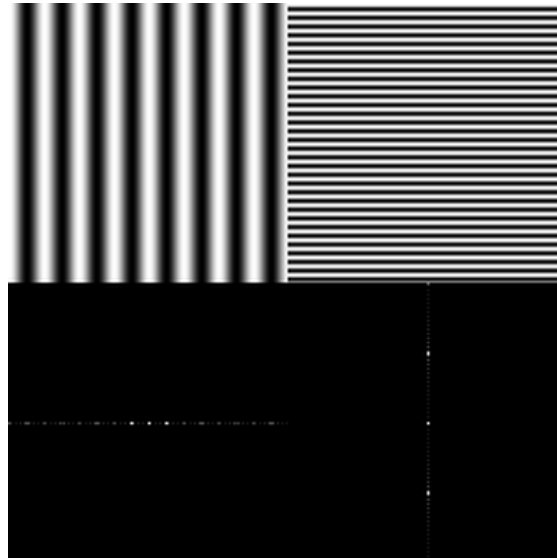
$$f(t) = a_0 + \sum_{k=1}^{\infty} a_k \sin(k\omega_0 t) + \sum_{k=1}^{\infty} b_k \cos(k\omega_0 t)$$

Честотата a_0 се нарича основна.

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) dt$$

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(k\omega_0 t) dt$$

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(k\omega_0 t) dt$$



Фигура 2. FT

Continuous Fourier Transform (FT)

Правя трансформация на Фурие:

$$F(f(x)) = F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx$$

Обратна трансформация на Фурие:

$$F^{-1}(F(u)) = f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

Където:

$$e^{\pm j\theta} = \cos(\theta) \pm j\sin(\theta)$$

Свойства на Фурие трансформацията:

- Линейност

$$f(x) + g(x) \leftrightarrow F(u) + G(u)$$

$$af(x) \leftrightarrow aF(u)$$

Това означава, че когато събираме/изваждаме 2 сигнала събираме/изваждаме техните спектри.

- Скалиране

$$f(ax) \leftrightarrow \frac{1}{|a|} F\left(\frac{u}{a}\right)$$

Това означава, че когато функцията се разпростира в посока x , спектърът се свива в същата посока.

- Отместване във времето

$$f(x - x_0) \leftrightarrow (e^{-j2\pi u x_0}) F(u)$$

- Честотно отместване

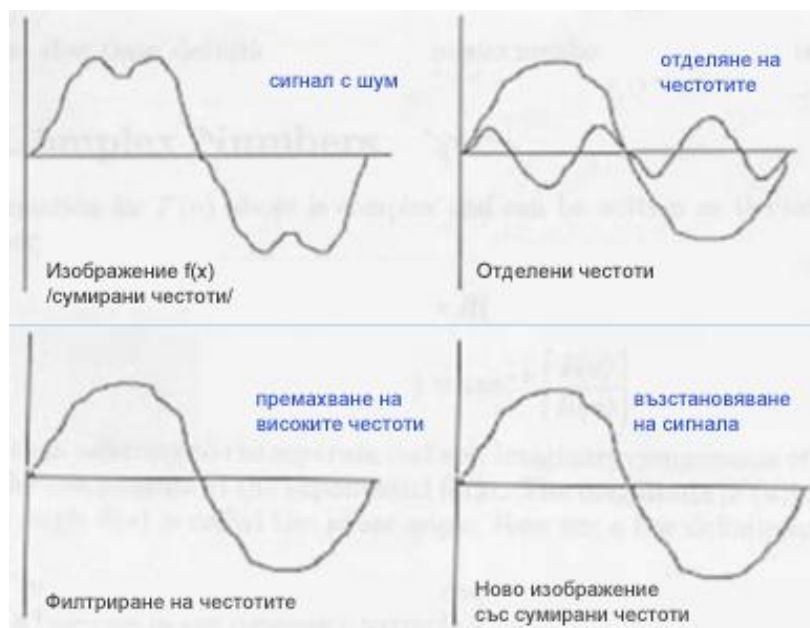
$$(e^{-j2\pi x x_0}) f(x) \leftrightarrow F(u - u_0)$$

Фурие трансформацията е полезна, защото:

- прави значително по-лесно премахването на различни честоти
- извършването на операциите в честотното пространство е много по-бързо отколкото в обектното

Пример:

За да премахнем някои честоти просто задаваме стойност 0 на съответстващият $F(u)$ коефициент.



Фигура 3. Премахване на честотите

Преминаваме през 3 етапа:

1. Правим фурие трансформация на входното изображение $f(x)$
2. Премахваме нежеланите честоти $D(F(f(x)))$
3. Конвертираме обратно сигнала $\hat{f}(x) = F^{-1}(D(F(f(x))))$

Двумерна Фурие трансформация

Права трансформация:

$$F(f(x, y)) = F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy$$

Обратна трансформация:

$$F^{-1}(F(u, v)) = f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du dv$$

Дискретна Фурие трансформация

Дискретната трансформация на Фурие (DFT) е дискретизирана FT и поради тази причина не съдържа всички честоти, формиращи изображението, а само достатъчно голям екземпляр от тях, който напълно да описва обектното пространство. Обикновено размерът на изображението в обектното пространство е равен на този в честотното пространство. За изображение с размери $M \times N$ двумерната дискретна Фурие трансформация изглежда по следния начин:

Права трансформация:

$$F(u, v) = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(x, y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})}, \quad u = 0 \div M-1; v = 0 \div N-1$$

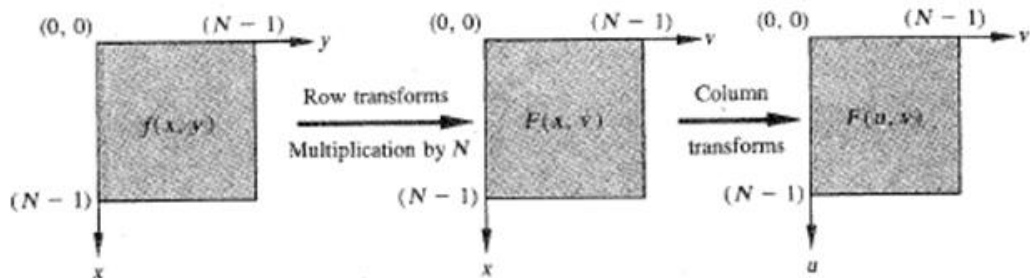
Обратна трансформация:

$$f(x, y) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} F(u, v) e^{j2\pi(\frac{ux}{M} + \frac{vy}{N})}, \quad x = 0 \div M-1; y = 0 \div N-1$$

Свойства:

- Разделимост – двумерната DFT можем да представим като 2 едномерни DFT

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi\left(\frac{ux}{N}\right)} F(x, v)$$



Фигура 4.

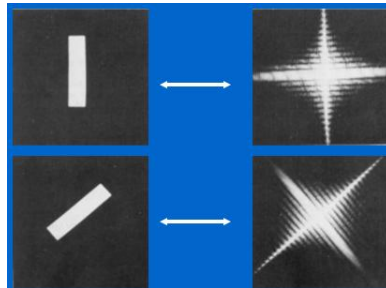
- Периодичност – правата и обратна DFT са периодични с период N

$$F(u, v) = F(u + N, v) = F(u, v + N) = F(u + N, v + N)$$

- Симетричност

$$F(u, v) = F^*(-u, -v) \rightarrow |F(u, v)| = F(-u, -v)$$

- Транслиране – трансляция в обектното пространство е еквивалентна на трансляция в честотното
- Ротиране – ротация с ъгъл θ в обектното пространство е равностойна на ротация с ъгъл θ в честотното



Фигура 5. Ротация в обектното и честотното пространство

- Събиране/Изваждане

$$F(f(x, y) + g(x, y)) = F(f(x, y)) + F(g(x, y))$$

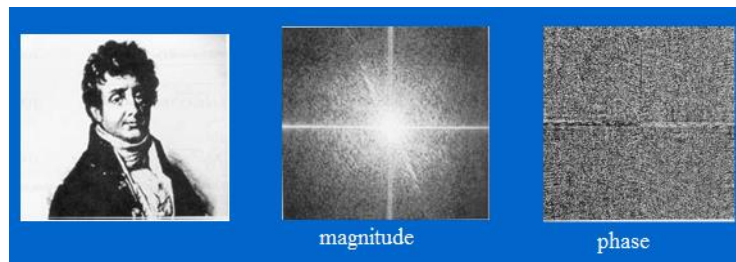
Но

$$F(f(x, y)g(x, y)) \neq F(f(x, y))F(g(x, y))$$

- Скалиране

$$af(x, y) \leftrightarrow aF(u, v)$$

При FT можем да получим две изображения – на магнитута и на фазата. Като по-голямата част от значещата информация се запазва в магнитута:



Фигура 6. Изображения на магнитута и фазата при FT

Въпреки това, за да реконструираме обратно входното изображение са ни необходими и двете изображения.



Фигура 7. Реконструкция на изображение от магнитута и от фазата

Теорема на Шенон (за дискретизацията)

За всяка функция $f(t)$ ($t \in [0, T]$), съществува минимален интервал Δt такъв че $\frac{T}{\Delta t} = N$ и ако изберем по-малък от този интервал $\Delta \hat{t} < \Delta t$ то $f(t)$ не може да се възстанови в рамките на грешката ε .

Шум в изображенията.

По принцип е желателно яркостта на изображението да бъде еднаква, с изключение на случаите, когато тя се променя, за да формира образ. Има фактори обаче, които произвеждат различия в яркостта, които са случайни и нежелателни, защото влошават качеството на изображението.

Затова един от основните проблеми в обработката на изображения е премахването на шума. То може да стане както в обектното, така и в честотното пространство.

Обработка в обектното пространство

Премахването на шум в изображение в обектното пространство може да стане чрез конволюция с премахваща шума маска, пример за каквато са следните:

$$H = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

$$H = \frac{1}{10} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

$$H = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

Шумът в обектното пространство може да бъде премахнат и чрез медианен филтър, Гаусов филтър и др.

Обработка в честотното пространство

Премахването на шума в честотното пространство се извършва, чрез премахване на високите честоти по описаната по-горе схема за премахване на нежелани честоти.

Косинус трансформация

Дискретната косинус трансформация (DCT) е подобна на дискретната Фурие трансформация. Тя се извършва върху дискретен брой точки и представя функцията като сума от синусоиди с различна честота и амплитуда. Разликата е, че DCT използва само косинус функции. Косинус трансформацията има 4 различни типа DCT-I, DCT-II, DCT-III, DCT-IV, като най-разпространен е DCT-II:

Права трансформация:

$$F(u, v) = \frac{2 c(u) c(v)}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \cos\left(\frac{2m+1}{2N} u\pi\right) \cos\left(\frac{2n+1}{2N} v\pi\right),$$

$$u = 0 \div N-1, v = 0 \div N-1$$

Обратна трансформация:

$$f(m, n) = \frac{2}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} c(u) c(v) F(u, v) \cos\left(\frac{2m+1}{2N} u\pi\right) \cos\left(\frac{2n+1}{2N} v\pi\right),$$

$$m = 0 \div N-1, n = 0 \div N-1$$

Разлики между DCT и DFT.

- DFT съдържа реалната и комплексната част на фурие трансформацията. DCT съдържа само реалната.
- DFT е по-трудна за изчисление от DFT, заради експоненциалната част

15. Марковски вериги и вероятностни автомати. Скрити Марковски модели за разпознаване на говорни сигнали. Понятие за кепстър на говорен сигнал. Приложение при разпознаване на ръкопис.

Марковски вериги и вероятностни автомати

Марковска верига в теорията на вероятностите, е процес, който приема стойности от дискретно множество, наречено пространство на състоянията, като стойността му се изменя във фиксирани моменти от времето и зависи само от предходното състояние.

Нека имаме множество от състояния $S = \{s_1, s_2, \dots, s_r\}$. Процесът започва от едно от тях и ни дава възможност да преминаваме от едно състояние в друго. Всяко преминаване в ново състояние се нарича стъпка. От състояние s_i можем да преминем в състояние s_j с вероятност p_{ij} без тази вероятност да зависи от предишните състояния. Вероятността p_{ij} наричаме вероятност на прехода и за нея е изпълнено следното условие:

$$\sum_j p_{ij} = 1$$

Удобен начин за представяне на марковските вериги са вероятностните автомати.

Вероятностен автомат наричаме недетерминистичен краен автомат $(Q, \Sigma, \delta, q_0, F)$, разширен с две вероятности:

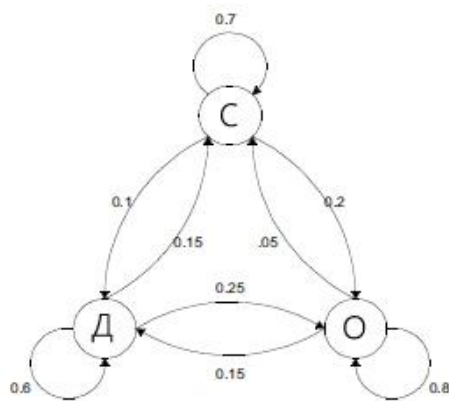
- P – вероятността да се извърши даден преход
- q_0 – вероятността да се намираме в началното състояние

Нека разгледаме следният пример:

		Утре		
		Слънчево (C)	Облачно (O)	Дъждовно (Д)
Днес	Слънчево (C)	.7	.2	.1
	Облачно (O)	.05	.8	.15
	Дъждовно (Д)	.15	.25	.6

Фигура 1.

Целта ни е да можем да прогнозираме какво ще бъде времето утре, по това какво е то днес. Като имаме 3 възможности - времето може да бъде слънчево (C), облачно (O) или дъждовно (Д). На фиг.1 са представени вероятностите, които се базират само на това какво е времето предишния ден. Т.е е изпълнено свойството на Марков и можем да представим горната таблица като Марковска верига със състояния $S = \{C, O, Д\}$ и вероятности на прехода – стойностите от таблицата. Веригата би изглеждала по следния начин представена с автомат на вероятностите:

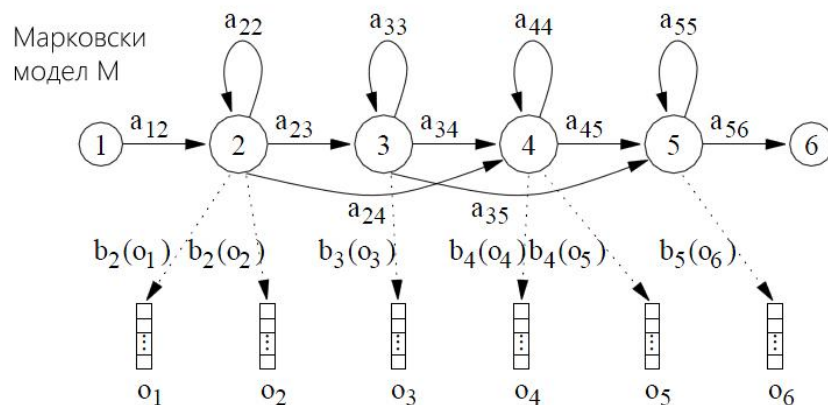


Фигура 2.

Скрит Марковски модел

Скрит Марковски модел наричаме модел, който изпълнява свойството на Марковски (условното разпределение на вероятността на бъдещите състояния на процеса при зададени текущо и минали състояния зависи само от текущото състояние и не зависи от миналите). За простота може да считаме, че скрит марковски модел е марковска верига с неявни (скрити) състояния.

Нека разгледаме следния пример:



Фигура 3.

Тук $b_k(o_i)$ е вероятността да получим изход o_i

Матрицата на вероятността на преходите изглежда по следния начин:

$$A = \begin{bmatrix} 0 & a_{12} & 0 & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & a_{24} & 0 & 0 \\ 0 & 0 & a_{33} & a_{34} & a_{35} & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} & 0 \\ 0 & 0 & 0 & 0 & a_{55} & a_{56} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Фигура 4.

В примера за всяко състояние имаме 3 възможности за преход:

- $a_{i,i}$ – да останем в същото състояние
- $a_{i,i+1}$ – да отидем в следващото
- $a_{i,i+2}$ – да прескочим едно състояние

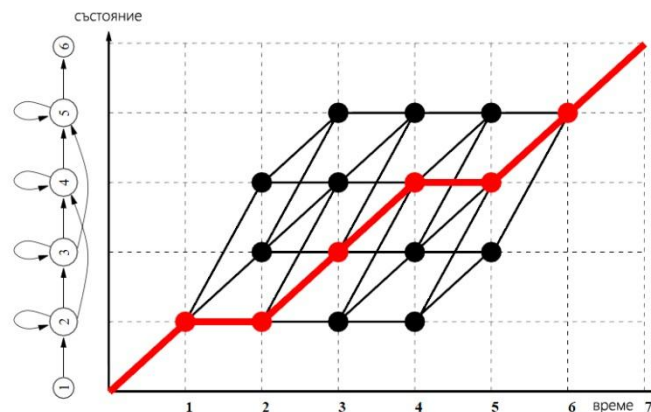
Такива модели, които минават само в следващи състояния наричаме ляво-десни. Повечето модели използвани за разпознаване на реч са ляво-десни.

Поредица от състояния, през които преминаваме последователно в модела наричаме път в модела. В повечето случаи тези поредици са невидими отвън и затова моделите се наричат скрити.

Вероятност за HMM да генерира поредица O

Един от основните въпроси при скритите марковски модели е как да изчислим вероятността даден модел да генерира поредица O. Дефинираме вероятността за генериране на O от пътя X:

$$P(O, X|M) = a_{x(o)x(1)} \prod_{t=1}^T b_{x(t)}(O_t) a_{x(t)x(t+1)}$$



Фигура 5.

В нашият пример имаме:

$$P(O, X|M) = a_{12}b(o_1)a_{22}b_{22}(o_2)a_{23}b_3(o_3)$$

Имаме две възможности да дефинираме вероятност за генериране на поредица O:

- Baum-Welch вероятност:

$$P(O|M) = \sum_{\{X\}} P(O, X|M)$$

- Viterbi вероятност:

$$P^*(O|M) = \max_{\{X\}} P(O, X|M)$$

Обучение на параметрите:

Досега предполагаме, че параметрите на модела са ни известни. Обикновено обаче те не са и затова трябва да проведем обучение. Нека илюстрираме процеса като намерим параметрите на модела M_i за думата w_i от обучаващото изречение. Това става в 2 стъпки:

1. Изчисляваме грубо параметрите.

$$\mu = \frac{1}{T} \sum_{t=1}^T o(t)$$

Пълна ковариационна матрица:

$$\Sigma = \frac{1}{T} \sum_{t=1}^T (o(t) - \mu)(o(t) - \mu)^T$$

Диагонална ковариационна матрица:

$$\sigma = \frac{1}{T} \sum_{t=1}^T (o(t) - \mu)^2$$

2. Итерации

а) присвояваме векторите на състояния

б) преизчисляваме параметрите базирани на горното присвояване с помощта на окупационната вероятност (дефинирана по-долу)

$$\mu_j = \frac{\sum_{t=1}^T L_j(t) o(t)}{\sum_{t=1}^T L_j(t)}$$

Пълна ковариационна матрица:

$$\Sigma_j = \frac{\sum_{t=1}^T L_j(t) (o(t) - \mu_j)(o(t) - \mu_j)^T}{\sum_{t=1}^T L_j(t)}$$

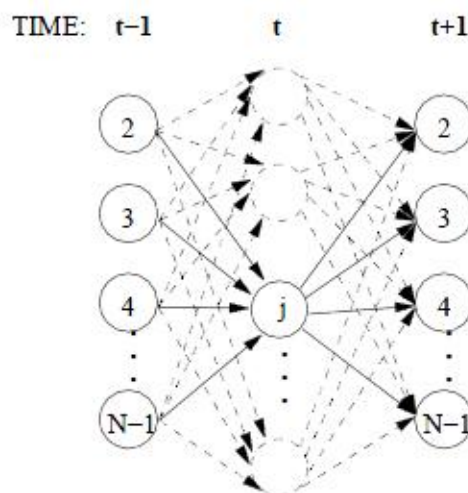
Диагонална ковариационна матрица:

$$\sigma_j = \frac{\sum_{t=1}^T L_j(t) (o(t) - \mu_j)^2}{\sum_{t=1}^T L_j(t)}$$

Оттук можем да получим формули за вероятностите на прехода a_{ij}

Окупационна вероятност – FW-BW алгоритъм

Основен проблем за намирането на параметрите на скрит марковски модел е намирането на окупационната вероятност $(L_j(t))$, т.е. вероятността да бъдем в състояние j в момент t . $(L_j(t))$



Фигура 6.

Един от начините да направим това е като изчислим сумата от вероятностите по всички пътища, пресичащи състояние j в момент t (непрекъснатите линии на фигура 6): $P(O, x(t) = j | M)$. Трябва да бъдем сигурни, че:

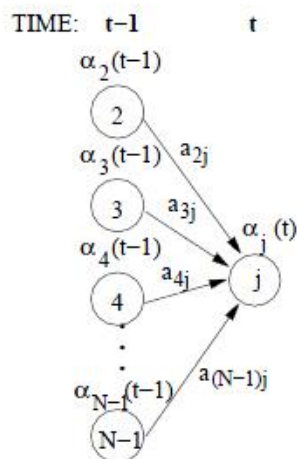
$$\sum_{j=1}^N L_j(t) = 1$$

Затова нормализираме и получаваме:

$$L_j(t) = \frac{P(O, x(t) = j | M)}{P(O | M)}$$

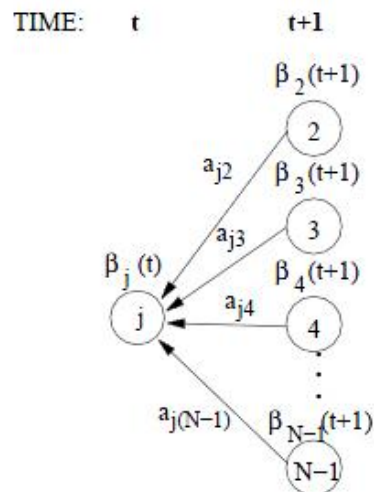
Много приложимо за изчисляването на $L_j(t)$ е първо да изчислим поотделно:

- вероятностите на всички пътища свършващи в състояние j в момент t (частична входна вероятност)



Фигура 7.

- вероятностите на всички пътища започващи от състояние j в момент t (частична изходна вероятност)



Фигура 8.

Тогава

$$L_j(t) = \frac{\alpha_j(t)\beta_j(t)}{P(O|M)}$$

Алгоритъм за обучение на параметри:

1. Създаване на акумулатор за всеки вектор/ матрица
2. Изчисляване на частичната входна/изходна вероятност $\alpha_j(t)$ и $\beta_j(t)$
3. Изчисляване на окупационната вероятност $L_j(t)$
4. Преизчисление на параметрите μ_j и σ_j
5. Ако $|P_{BIV}(M') - P_{BIV}(M)| < \epsilon$
GO TO 2
Иначе
Край

Скрити Марковски модели за разпознаване на говорни сигнали

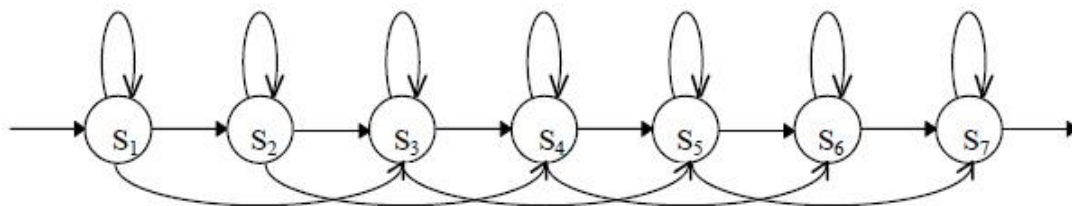
Едно от основните приложения на скритите марковски модели е разпознаването на говорни сигнали.

Нека имаме речник с N думи: $\omega_1 \dots \omega_N$. Всяка от тях е моделирана със скрит марковски модел $M_1, \dots, M_N$. Целта ни е да определим кой модел ще генерира O с най-голяма точност.

Понятие за кепстър на говорен сигнал

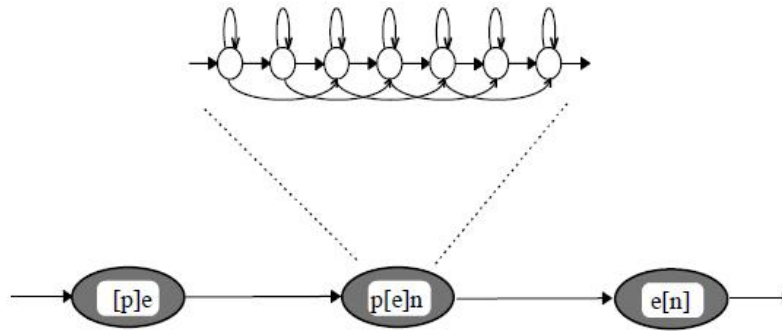
Скрити Марковски модели за разпознаване на ръкописен текст

Напоследък широко популярност добива използването на скрити марковски модели в разпознаването на ръкописен текст. Ще покажем нагледно как изглеждат скритите марковски модели за разпознаване на буква, текст и изречение.



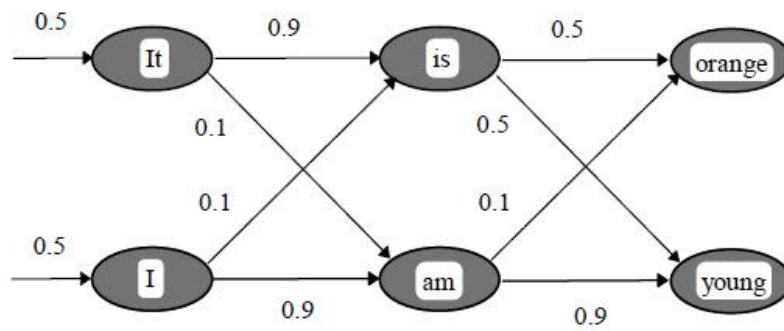
Фигура 9.

Фигура 9 показва ляво-десен HMM със 7 състояния за разпознаване на буква. (Като числото 7 е избрано експериментално с цел да се максимизира точността на разпознаването)



Фигура 10.

Фигура 10 показва моделиране на думата "pen" състояща се от 3 HMM със 7 състояния за триграфа: „[p]e“, „p[e]n“ и „e[n]“.



Фигура 11.

Фигура 11 показва моделиране на изречения посредством HMM.