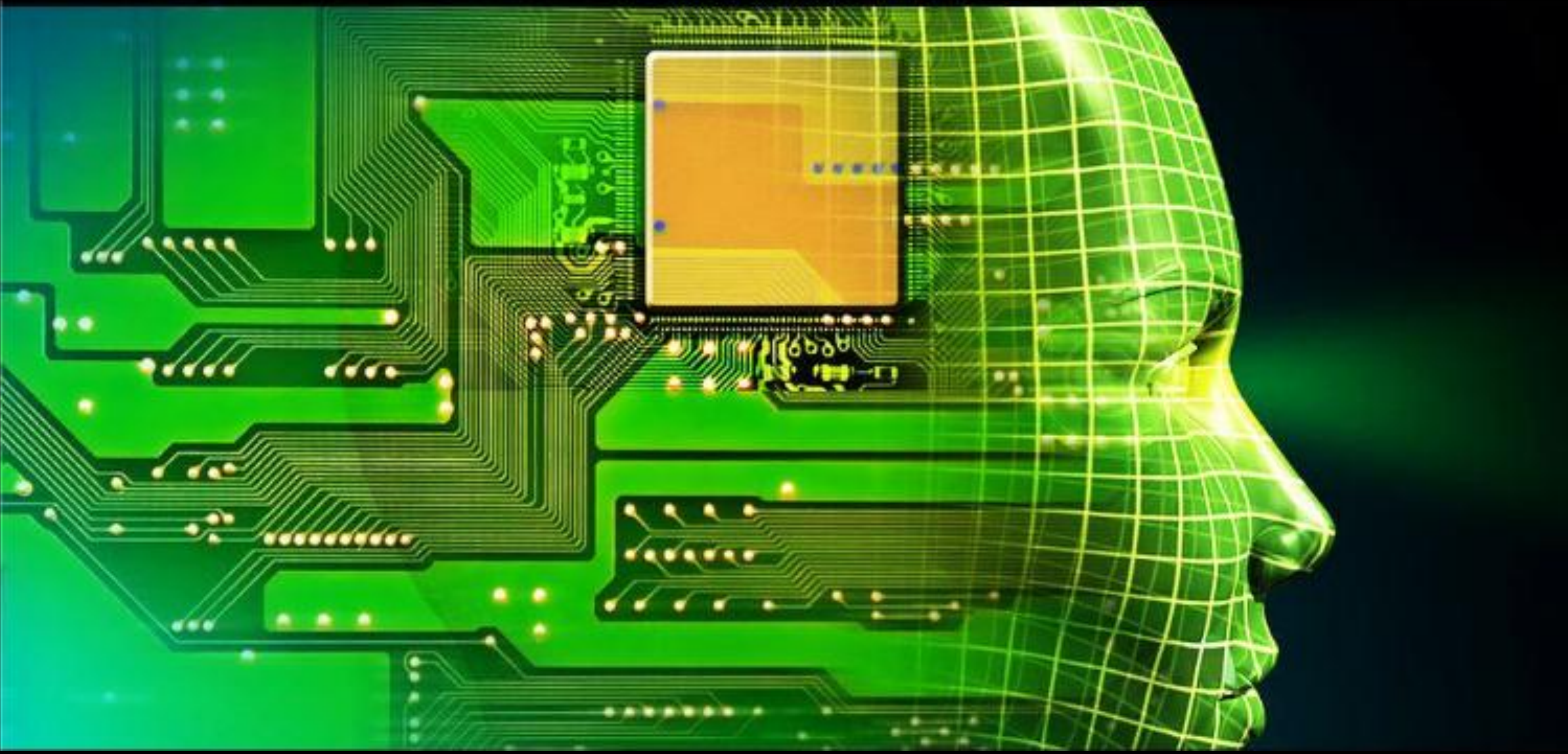
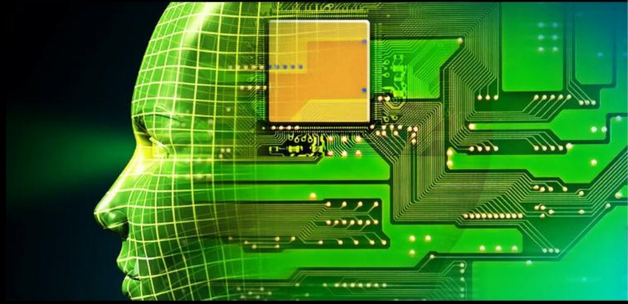
Искусственный интеллект: современный подход



Знания и рассуждения в условиях неопределённости

Логический подход



$$\forall p\ \text{Симптом}(p, \textit{зубная боль}) \Rightarrow \text{Болезнь}(p, \textit{дупло})$$

$$\forall p\ \text{Симптом}(p, \textit{зубная боль}) \Rightarrow$$
$$\text{Болезнь}(p, \textit{дупло}) \vee \text{Болезнь}(p, \textit{десна}) \vee \text{Болезнь}(p, \textit{нарыв}) \vee \ldots$$

$$\forall p\ \text{Болезнь}(p, \textit{дупло}) \Rightarrow \text{Симптом}(p, \textit{зубная боль})$$

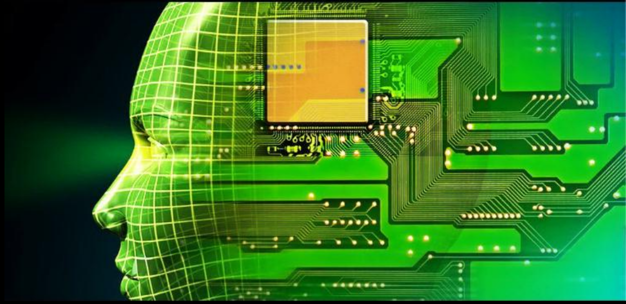
Причины неудачи

Лень

Причины неудачи

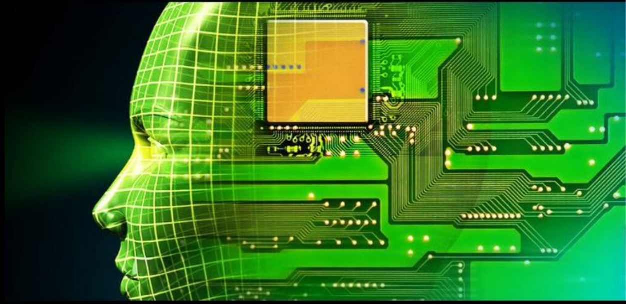
Теоретическое невежество

Причины неудачи



Практическое невежество

Итог



степень

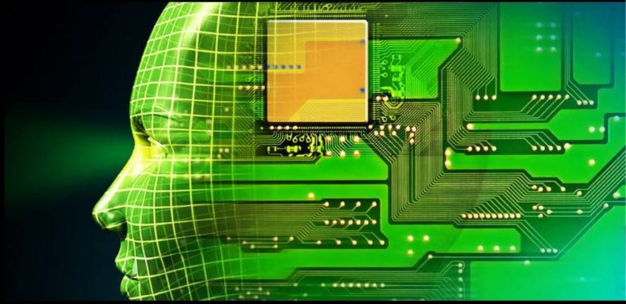
0 1

уверенности

Стандартный аппарат: теория вероятностей

(*суммарный учёт неопределённости, возникающей из-за лени и невежества*)

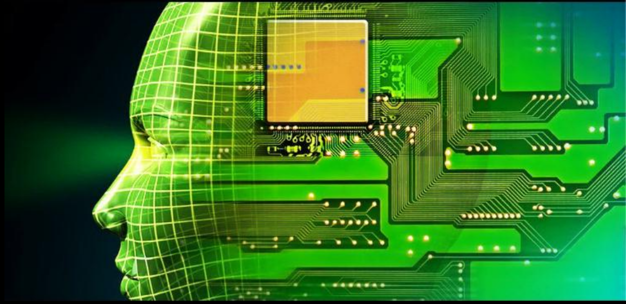
Базовые понятия теории вероятностей



Я предполагаю, что вы знаете, помните и любите следующие понятия:

- Вероятность
- Независимость событий
- Условная вероятность
- Случайная величина
- Распределение случайной величины
- Правило Байеса

Пример вероятностного вывода



	toothache		*¬toothache*	
	catch	*¬catch*	*catch*	*¬catch*
cavity	0.108	0.012	0.072	0.008
¬cavity	0.016	0.064	0.144	0.576

$$P(cavity \mid toothache) = \frac{P(cavity \land toothache)}{P(toothache)}$$

$$0.108 + 0.012$$

$$\mathbf{P}(Cavity \mid toothache) = \alpha\, \mathbf{P}(Cavity, toothache)\}$$

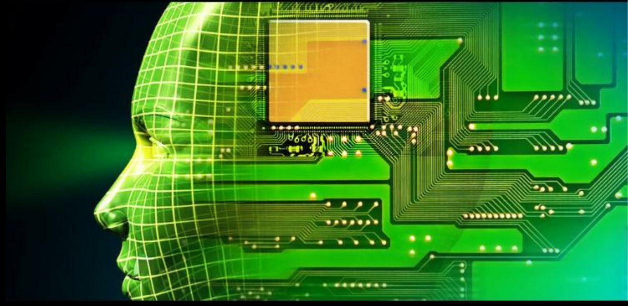
$$= \alpha\, [\mathbf{P}(Cavity, toothache, catch) + \mathbf{P}(Cavity, toothache, \neg catch)]$$

$$= \alpha\, [<0.108, 0.016> + <0.012, 0.064>] = \alpha <0.12, 0.08> = <0.6, 0.4>$$

$$P(\neg cavity \mid toothache) = \frac{}{P(toothache)}$$

$$= \frac{0.016+0.064}{0.108 + 0.012 + 0.016 + 0.064} = 0.4$$

Вероятностный вывод

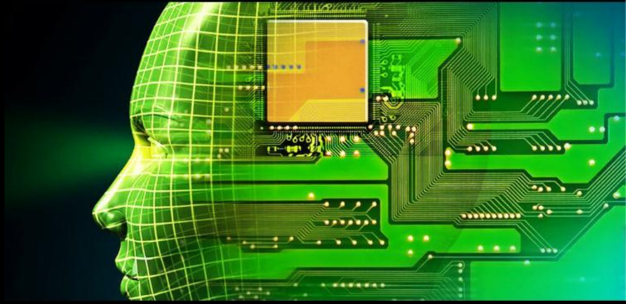


Обозначения:

- query variables X_i
- evidence variables $E_1, E_2, \ldots, E_m$
- Nonevidence (hidden) variables $Y_1, Y_2, \ldots, Y_m$

$$P(X|e)=?$$

Алгоритм вероятностного вывода



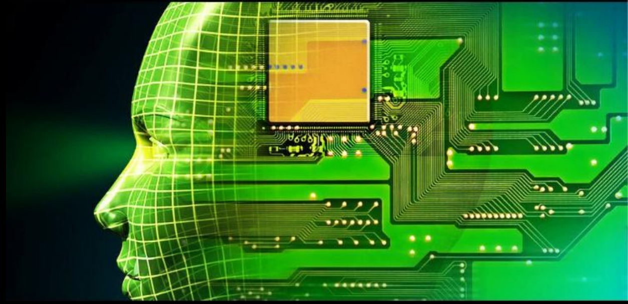
$$\mathbf{P}(X|\mathbf{e}) = \alpha\,\mathbf{P}(X,\mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X,\mathbf{e},\mathbf{y})$$

Анализ для *n* булевских переменных

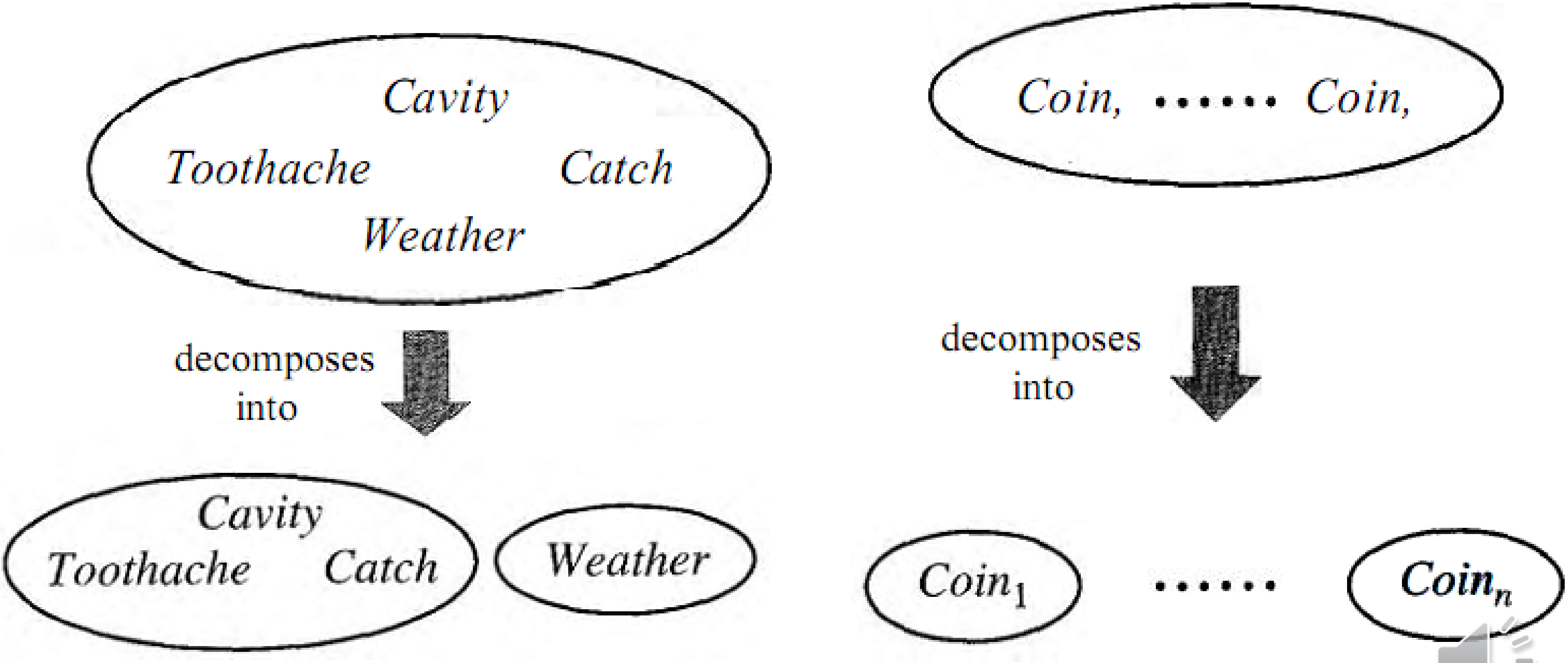
Сложность по памяти: $O(2^n)$

Сложность по времени: $O(2^n)$

Независимость



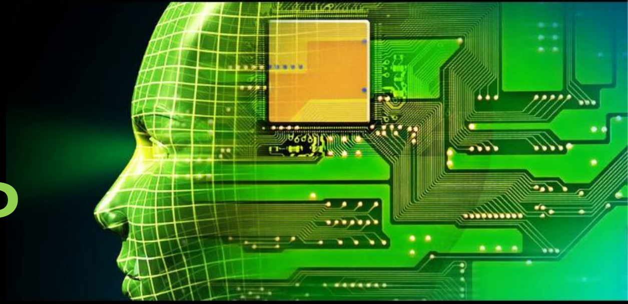
$$\mathbf{P}(Toothache, Catch, Cavity, Weather) = \mathbf{P}(Toothache, Catch, Cavity)\mathbf{P}(Weather)$$



(a)

(b)

Условная независимость



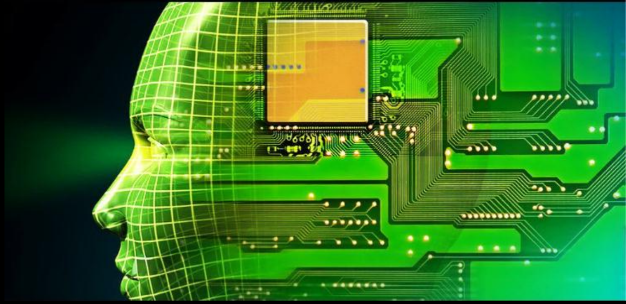
$$\mathbf{P}(Cavity \mid toothache \wedge catch) = \alpha \, \langle 0.108, 0.016 \rangle \approx \langle 0.871, 0.129 \rangle$$

$$\mathbf{P}(Cavity \mid toothache \wedge catch) = \alpha \, \mathbf{P}(toothache \wedge catch \mid Cavity) \, \mathbf{P}(Cavity)$$

$$\mathbf{P}(toothache \wedge catch \mid Cavity) = \mathbf{P}(toothache \mid Cavity) \, \mathbf{P}(catch \mid Cavity)$$

$$\begin{aligned} \mathbf{P}(Toothache, Catch, Cavity) &= \mathbf{P}(Toothache, Catch \mid Cavity) \, \mathbf{P}(Cavity) \\ &= \mathbf{P}(Toothache \mid Cavity) \, \mathbf{P}(Catch \mid Cavity) \, \mathbf{P}(Cavity) \end{aligned}$$

Размер представления



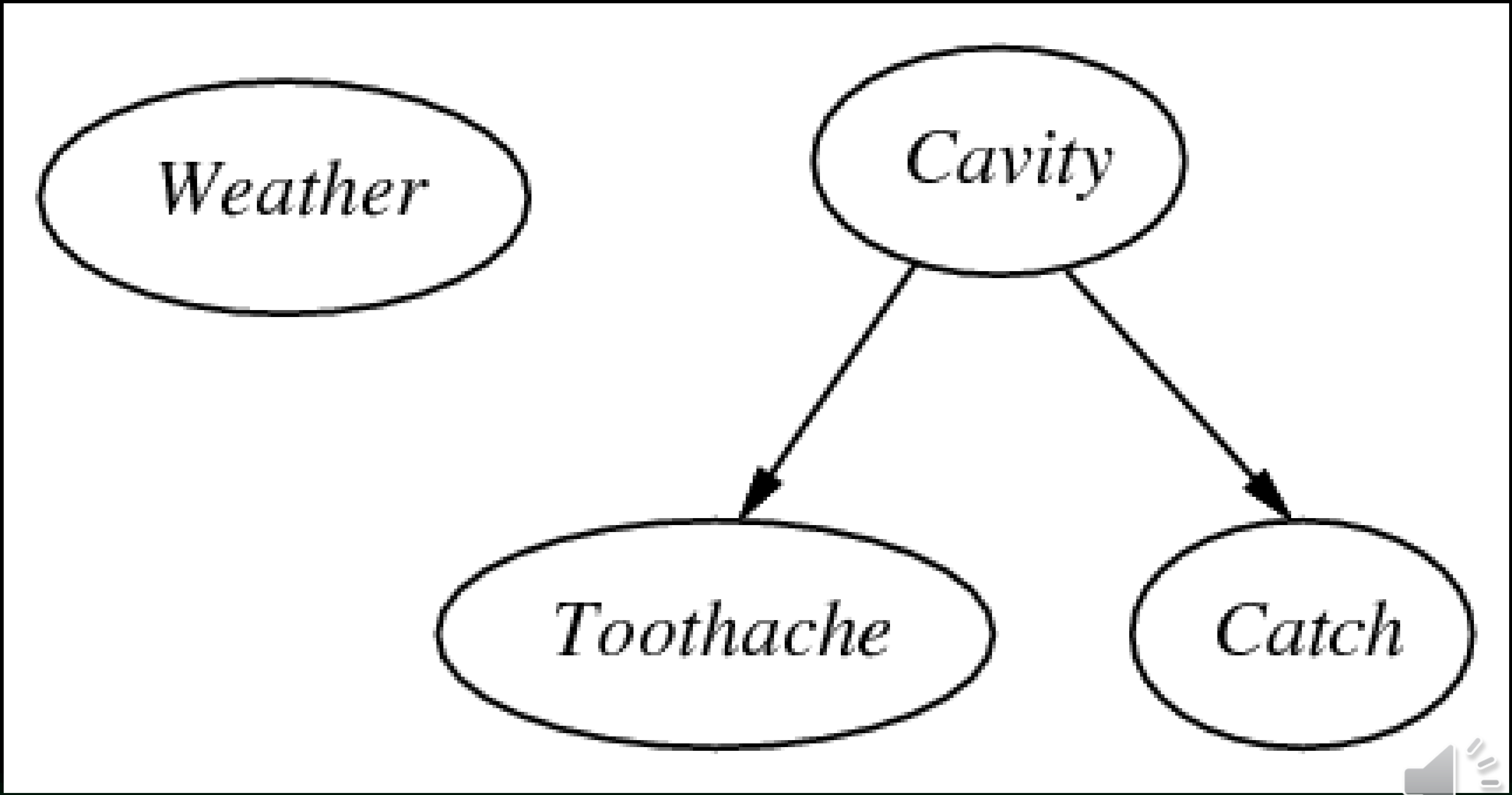
n условно независимых симптомов

Сложность по памяти: $O(n)$

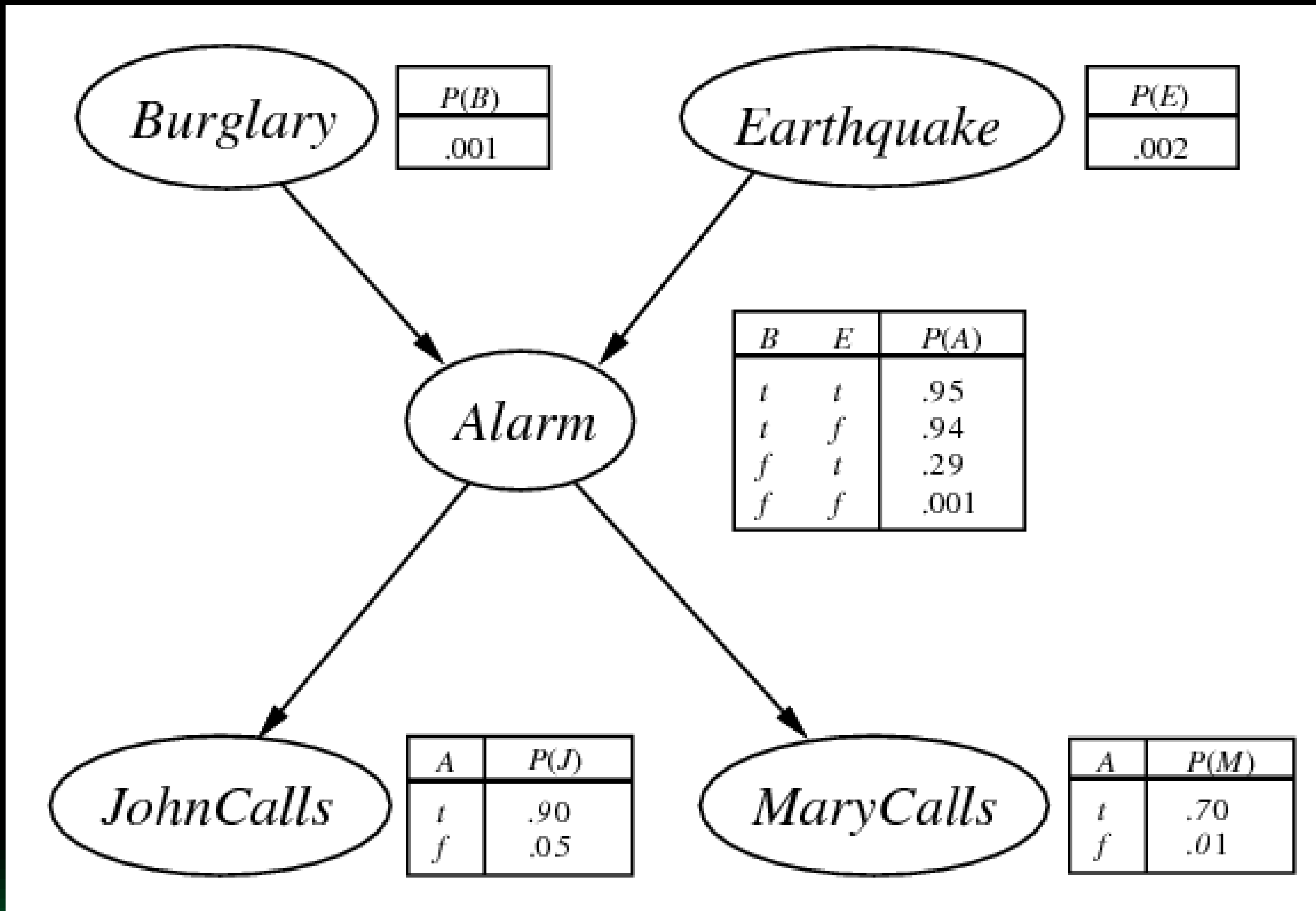
Наивная байесовская модель:

$$\mathbf{P}(Cause, Effect_1, \ldots, Effect_n) = \mathbf{P}(Cause) \prod_i \mathbf{P}(Effect_i \mid Cause)$$

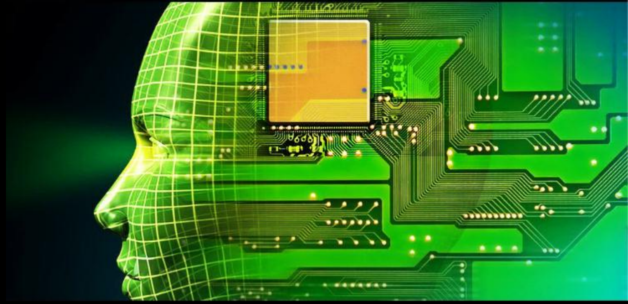
Байесовская сеть



Ещё одна байесовская сеть



Семантика байесовской сети

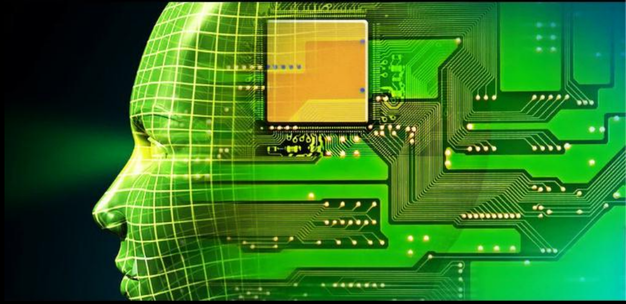


$$P(x_1, \dots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

Пример:

$$P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$$
$$= P(j|a)P(m|a)P(a|\neg b \; A \; \neg e)P(\neg b)P(\neg e)$$
$$= 0.90 \; x \; 0.70 \; \text{x} \; 0.001 \; \text{x} \; 0.999 \; \text{x} \; 0.998 = 0.0006'2.$$

Конструирование байесовской сети



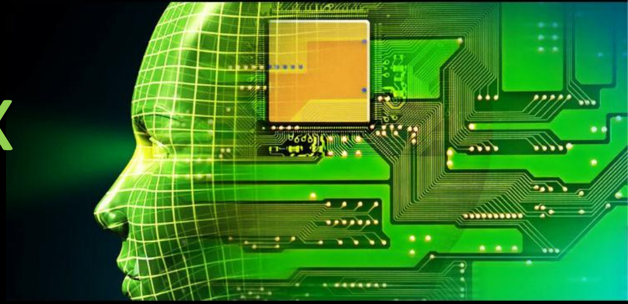
$$P(x_1, \dots, x_n) = P(x_n | x_{n-1}, \dots, x_1) P(x_{n-1}, \dots, x_1)$$

$$P(x_1, \dots, x_n) = P(x_n | x_{n-1}, \dots, x_1) P(x_{n-1} | x_{n-2}, \dots, x_1) \dots P(x_2 | x_1) P(x_1) =$$

$$= \prod_{i=1}^{n} P(x_i | x_{i-1}, \dots, x_1) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

$$P(X_i | X_{i-1}, \dots, X_1) = P(X_i | parents(X_i))$$

Компактность байесовских сетей



Пусть:

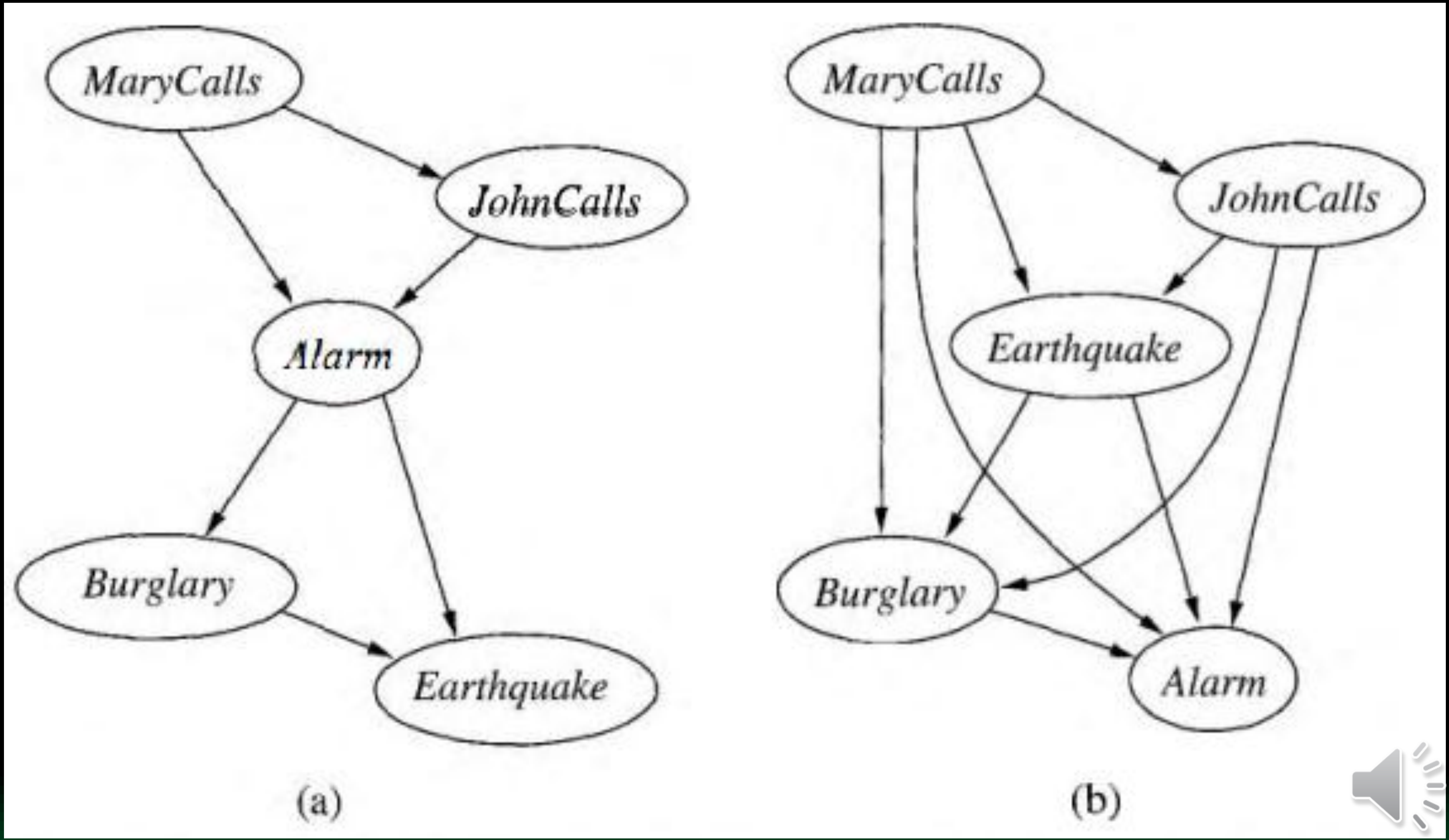
Количество входящих рёбер не больше *k*

n булевых переменных

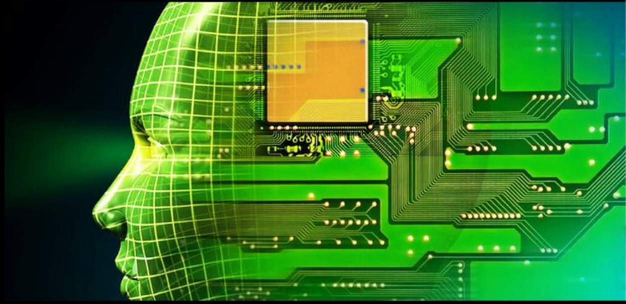
Тогда:

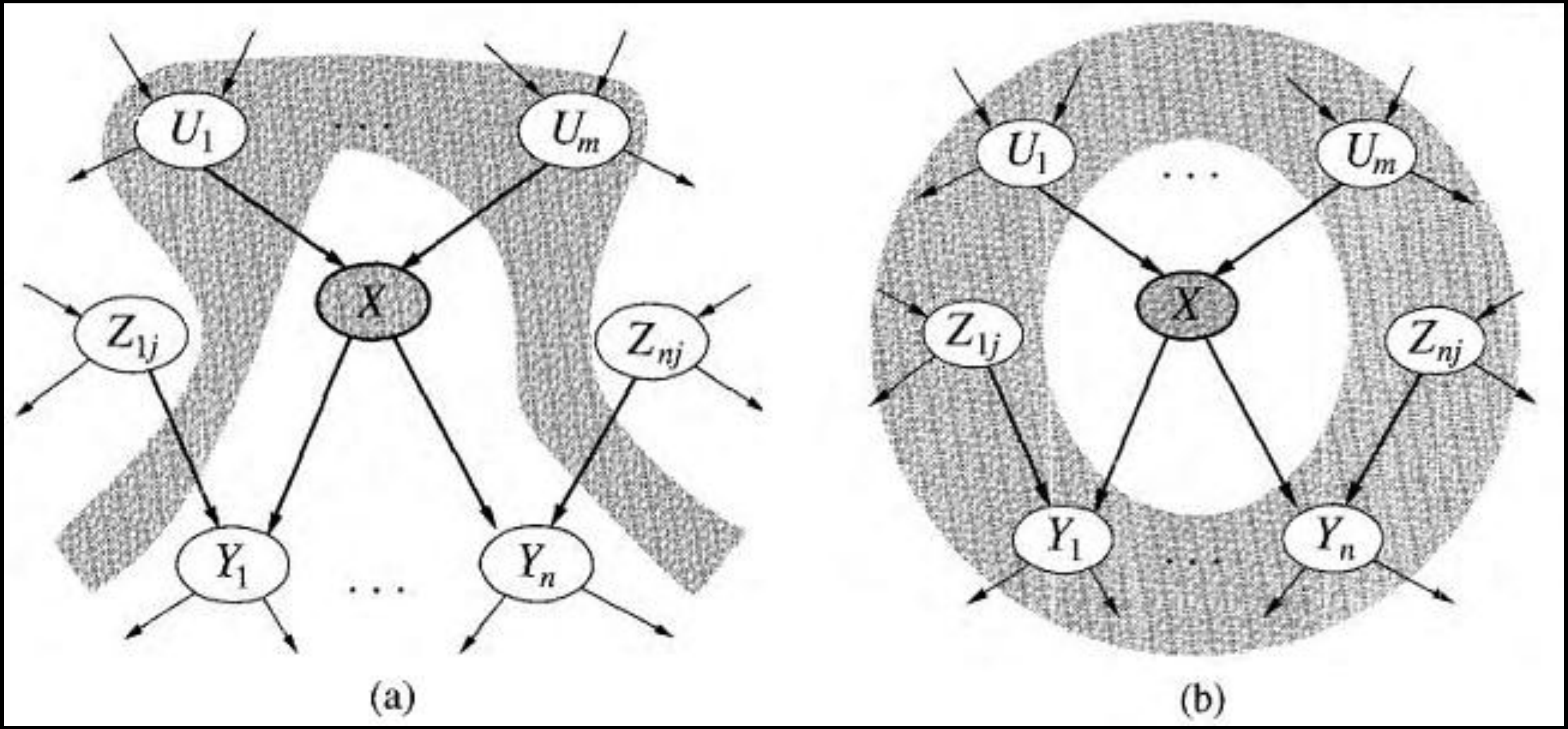
Размер представления: $n2^k$ чисел

Порядок добавления узлов

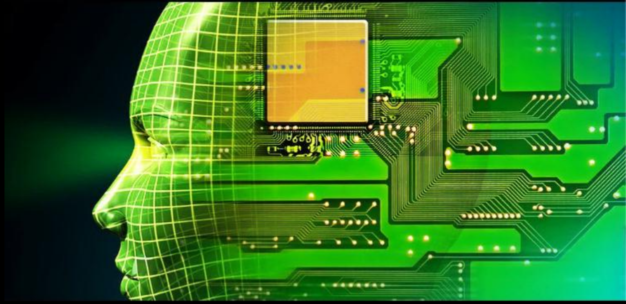


Топологическая семантика байесовской сети





Вероятностный вывод

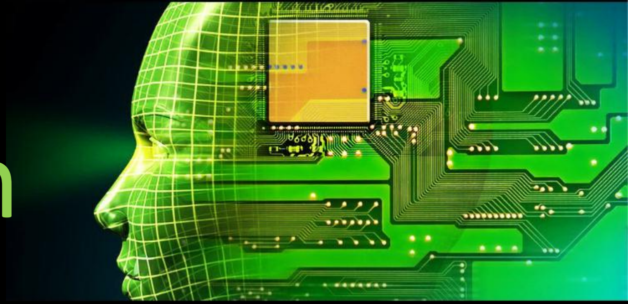


Обозначения:

- query variables X_i
- evidence variables $E_1, E_2, \ldots, E_m$
- Nonevidence (hidden) variables $Y_1, Y_2, \ldots, Y_m$

$$P(X|e)=?$$

Inference by enumeration



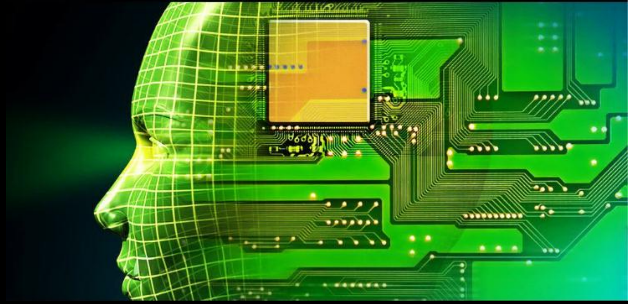
$$\mathbf{P}(X|\mathbf{e}) = \alpha\,\mathbf{P}(X,\mathrm{e}) = \alpha \sum_{\mathrm{Y}} \mathbf{P}(X,\mathrm{e},\mathrm{y})$$

Пример:

$$\mathbf{P}(B|j,m) = \alpha\,\mathbf{P}(B,j,m) = \alpha \sum_{e}\sum_{a} \mathbf{P}(B,e,a,j,m)$$

$$P(b|j,m) = \alpha \sum_{e}\sum_{a} P(b)P(e)P(a|b,e)P(j|a)P(m|a)$$

Сложность алгоритма



Сложность для сети с *n* булевыми переменными:

$$O(n2^n)$$

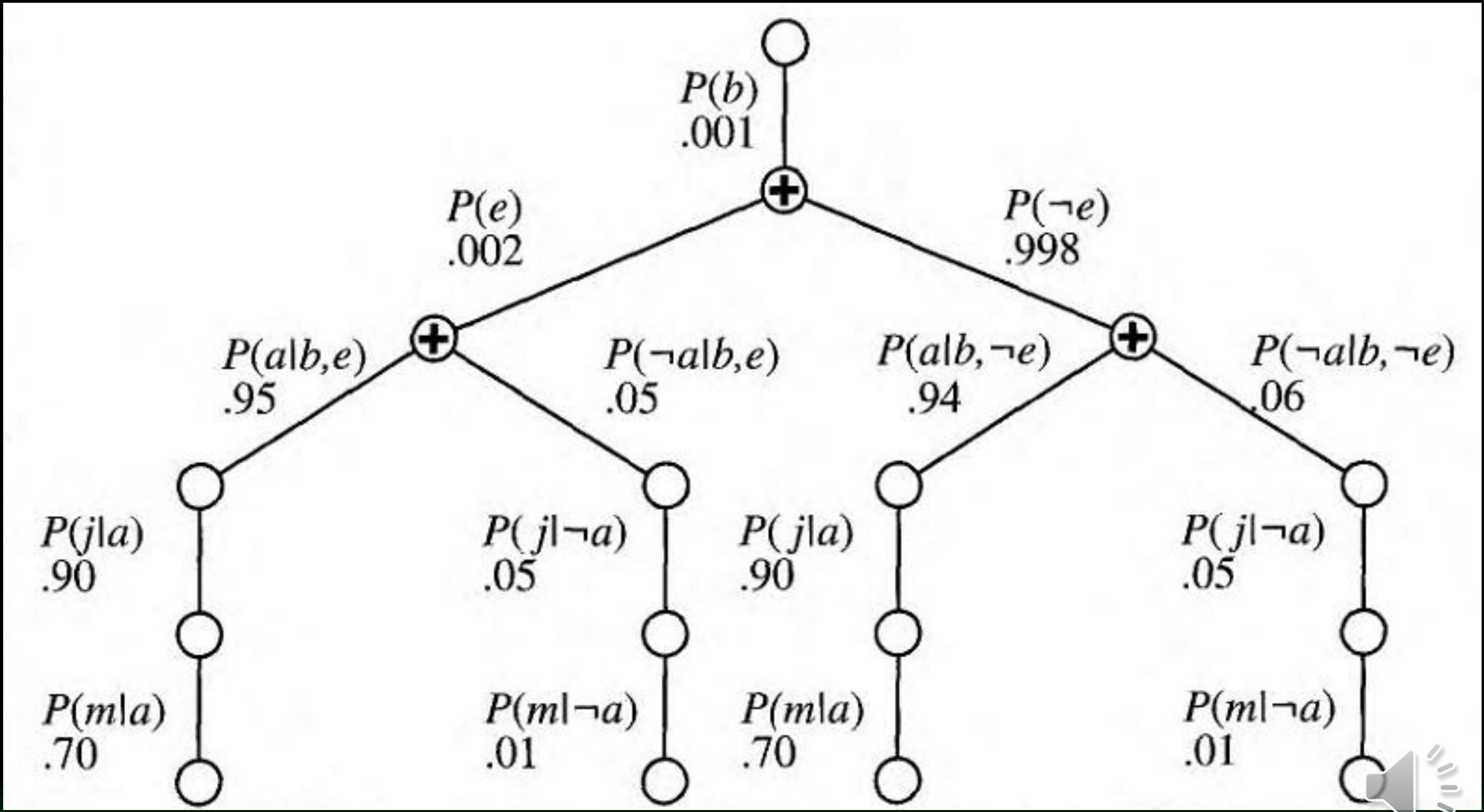
Оптимизация:

$$P(b|j,m) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e) P(j|a) P(m|a)$$

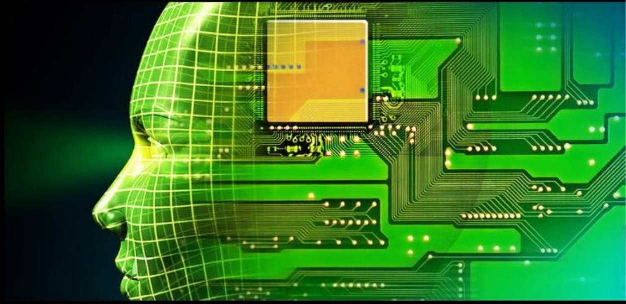
Сложность по памяти: $O(n)$

Сложность по времени: $O(2^n)$

Процесс вычислений в алгоритме



The variable elimination algorithm



$$\mathbf{P}(B|j,m) = \alpha \underbrace{\mathbf{P}(B)}_{B} \sum_e \underbrace{P(e)}_{E} \sum_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M}$$

Идея алгоритма:

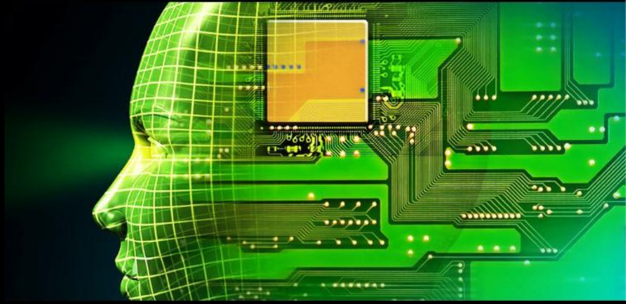
Динамическое программирование

Pointwise product

$$\mathbf{f}(X_1 \ldots X_j, Y_1 \ldots Y_k, Z_1 \ldots Z_l) = \mathbf{f}_1(X_1 \ldots X_j, Y_1 \ldots Y_k)\, \mathbf{f}_2(Y_1 \ldots Y_k, Z_1 \ldots Z_l)$$

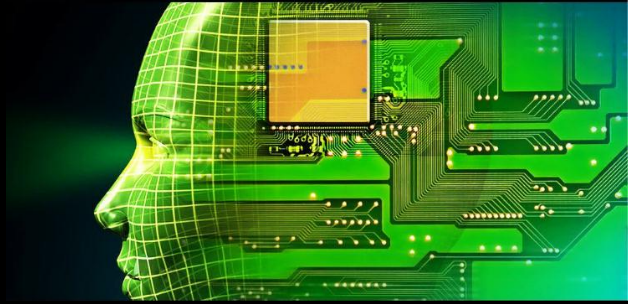
A	B	$\mathbf{f}_1(A,B)$	B	C	$\mathbf{f}_2(B,C)$	A	B	C	$\mathbf{f}_3(A,B,C)$
T	T	.3	T	T	.2	T	T	T	$.3 \times .2$
T	F	.7	T	F	.8	T	T	F	$.3 \times .8$
F	T	.9	F	T	.6	T	F	T	$.7 \times .6$
F	F	.1	F	F	.4	T	F	F	$.7 \times .4$
						F	T	T	$.9 \times .2$
						F	T	F	$.9 \times .8$
						F	F	T	$.1 \times .6$
						F	F	F	$.1 \times .4$

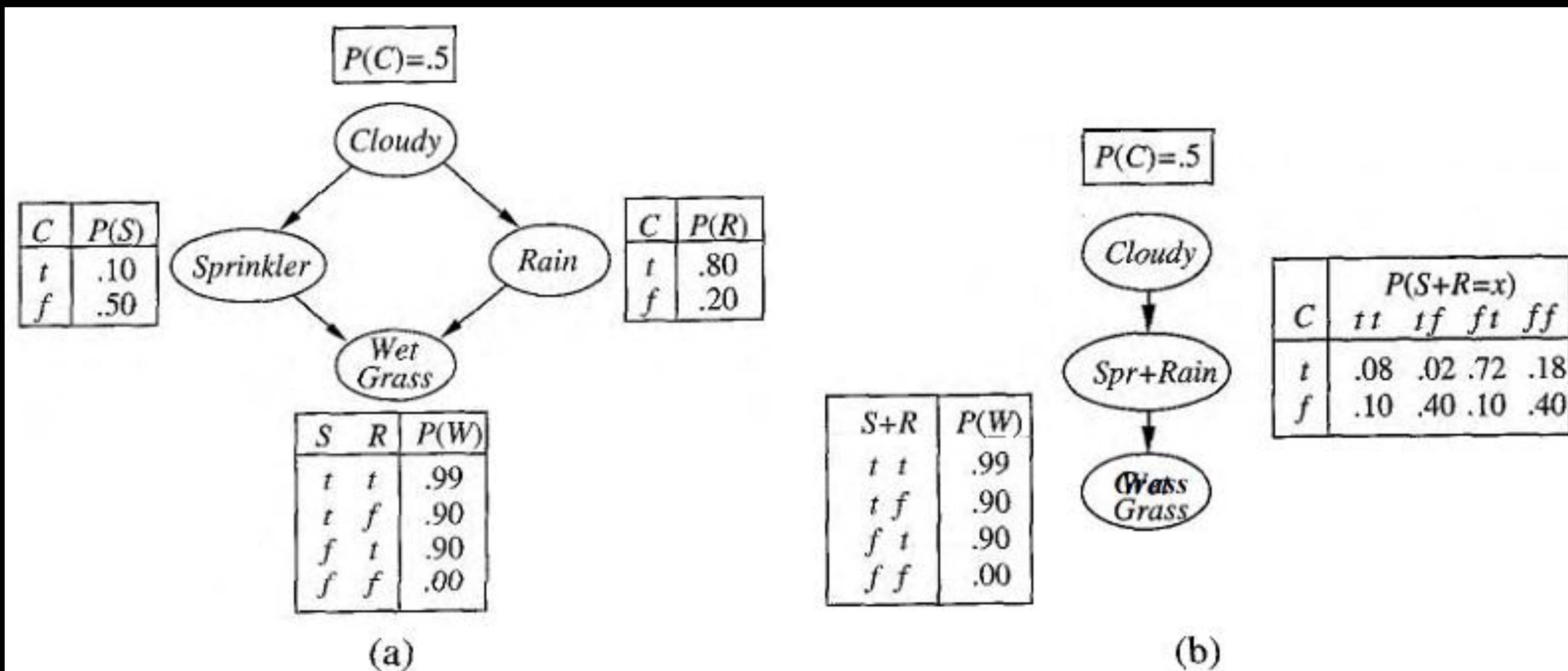
Сложность алгоритма

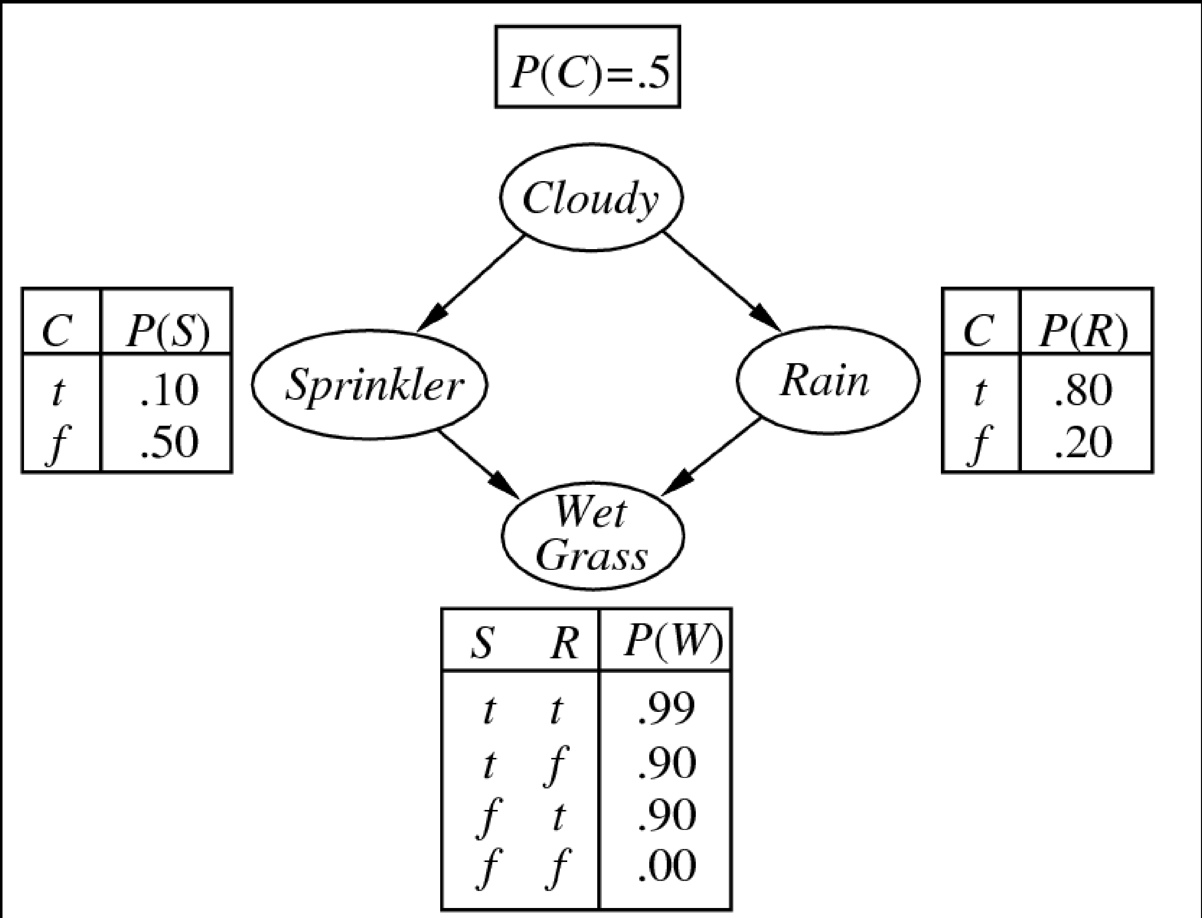


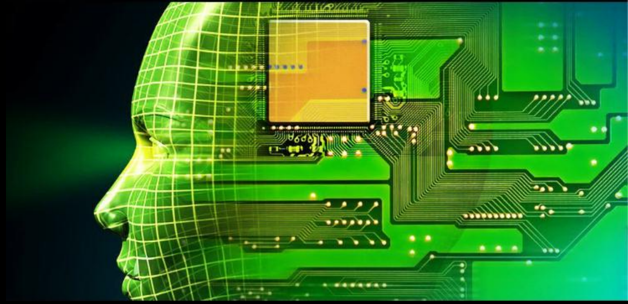
Тип сети	Сложность (по времени и по памяти)
Singly connected network (polytree)	Линейная
Multiply connected network	Экспоненциальная

Алгоритмы кластеризации





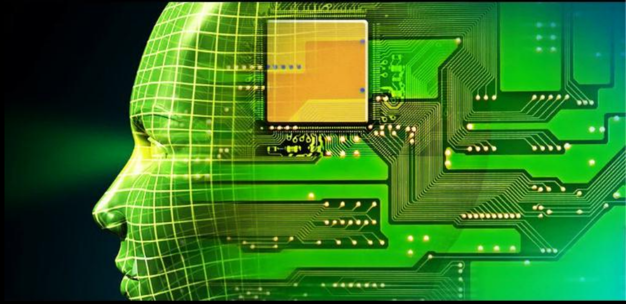




Direct sampling

P (*Cloudy*)	(0.5, 0.5)	**true**
P (*Sprinkler* | *Cloudy* = true)	(0.1, 0.9)	**false**
P (*Rain* | *Cloudy* = true)	(0.8, 0.2)	**true**
P (*WetGrass* | *Sprinkler* = false, *Rain* = true)	(0.9, 0.1)	**true**

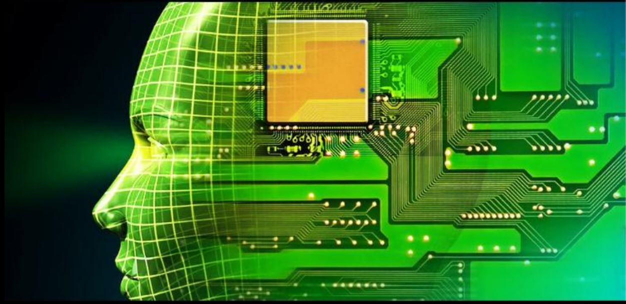
Почему это работает



$$S_{PS}(x_1 \ldots x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

$$S_{PS}(x_1 \ldots x_n) = P(x_1 \ldots x_n)$$

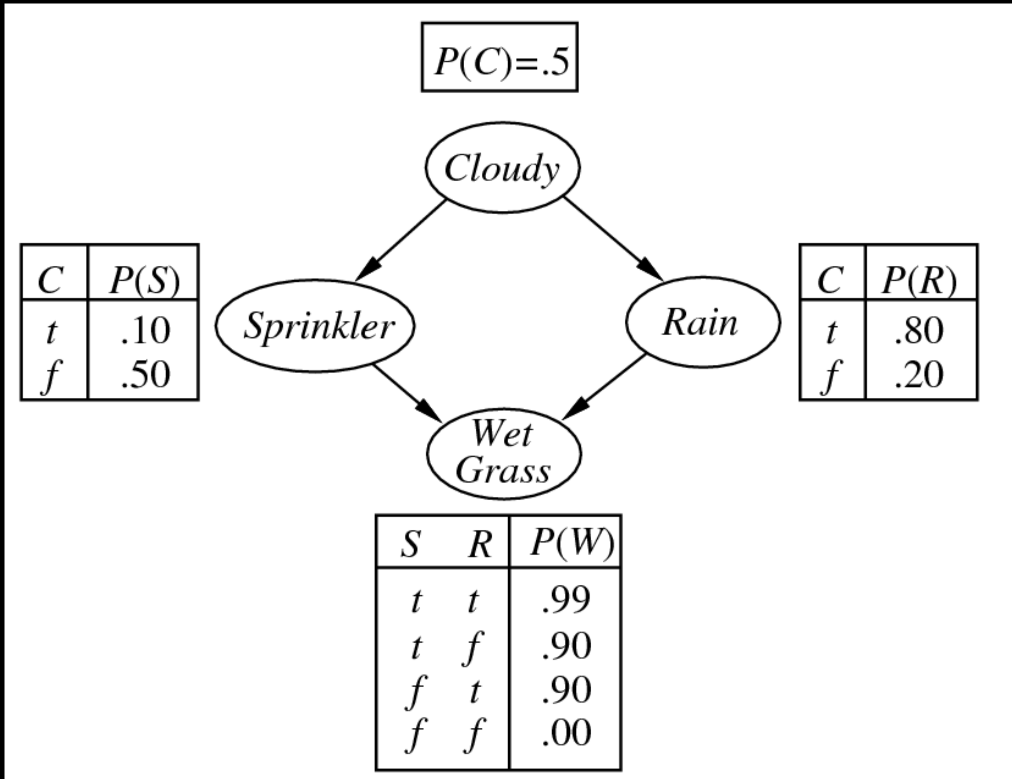
$$P(x_1, \ldots, x_m) \approx N_{PS}(x_1, \ldots, x_m)/N$$

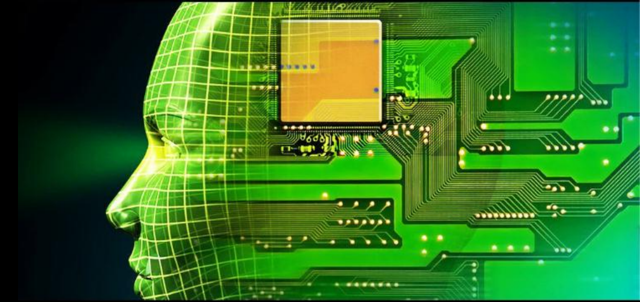


Rejection sampling

$$P(X|e) = \frac{P(X,e)}{P(e)}$$

Исключаем выборки, в которых нет события *e*



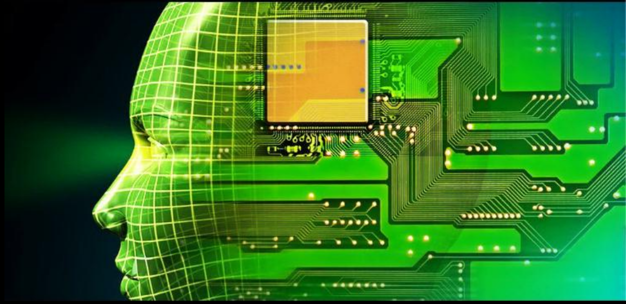


Likelihood weighting

$$P(Rain \mid Sprinkler = true, WetGrass = true) = ?$$

1. $P\,(Cloudy) = (0.5, 0.5) \rightarrow \boldsymbol{true}$
2. $w = w * P\,(Sprinkler = true \mid Cloudy = true) = 0.1$
3. $P\,(Rain \mid Cloudy = true) = (0.8, 0.2) \rightarrow \boldsymbol{true}$
4. $w = w * P\,(WetGrass = true \mid Sprinkler = true, Rain = true) = 0.099$

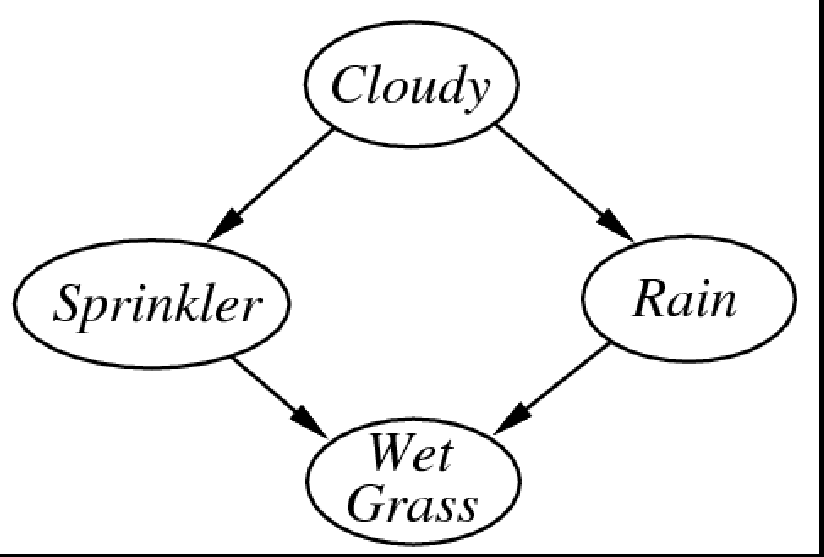
Почему это работает



$$S_{WS}(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{l} P(z_i | parents(Z_i))$$

$$w(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{m} P(e_i | parents(E_i))$$

$$\begin{aligned} S_{WS}(\mathbf{z}, \mathbf{e}) w(\mathbf{z}, \mathbf{e}) &= \prod_{i=1}^{l} P(z_i | parents(Z_i)) \prod_{i=1}^{m} P(e_i | parents(E_i)) \\ &= P(\mathbf{z}, \mathbf{e}) \,, \end{aligned}$$



Markov chain Monte Carlo (MCMC) algorithm

$$P(Rain \mid Sprinkler = true, WetGrass = true) = ?$$

$$Cloudy \rightarrow \boldsymbol{true}, \qquad Rain \rightarrow \boldsymbol{false}$$

$$[\boldsymbol{true}, \boldsymbol{true}, \boldsymbol{false}, \boldsymbol{true}]$$

$$P(Cloudy \mid Sprinkler = true, Rain = false) \rightarrow Cloudy = \boldsymbol{false}$$

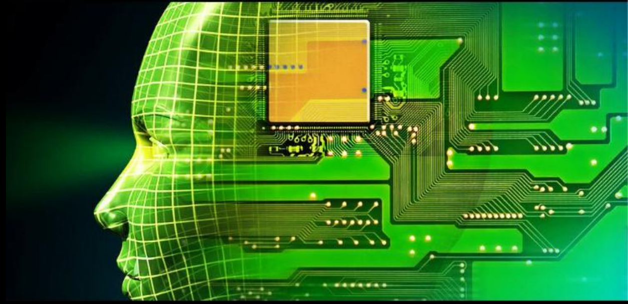
$$[\boldsymbol{false}, \boldsymbol{true}, \boldsymbol{false}, \boldsymbol{true}]$$

$$P(Rain \mid Cloudy = false, Sprinkler = true, WetGrass = true) \rightarrow Rain = \boldsymbol{true}$$

$$[\boldsymbol{false}, \boldsymbol{true}, \boldsymbol{true}, \boldsymbol{true}]$$

.....

Почему это работает - 1



$$\pi_{t+1}(x') = \sum_x \pi_t(x) q(x \to x'), \qquad \text{где}$$

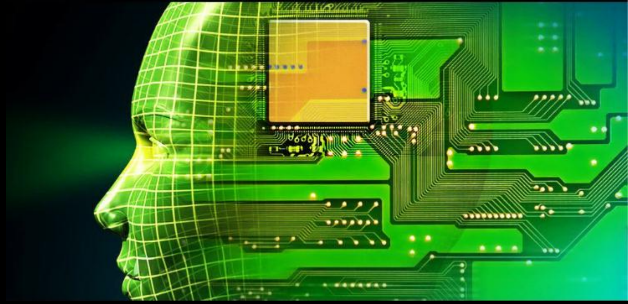
$\pi_t(x)$ – вероятность находится в состоянии x в момент времени t

$q(x \to x')$ – вероятность перехода из состояния x в состояние x'

Равновесное распределение вероятностей: $\pi_{t+1} = \pi_t$

$$\forall x' \quad \pi(x') = \sum_x \pi(x) q(x \to x')$$

Почему это работает - 2



Property of ***detailed balance:***

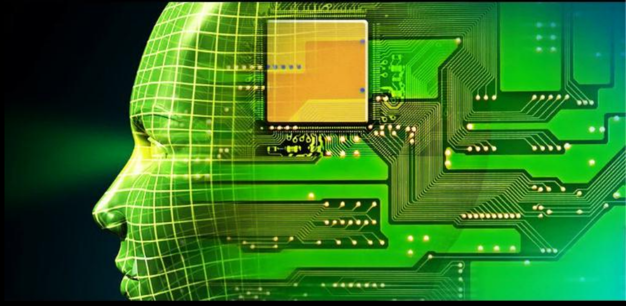
$$\forall x, x' \quad \pi(x) q(x \to x') = \pi(x') q(x' \to x)$$

Тогда $\pi(x)$ - равновесное распределение вероятностей:

$$\forall x' \quad \sum_x \pi(x) q(x \to x') = \sum_x \pi(x') q(x' \to x) =$$

$$= \pi(x') \sum_x q(x' \to x) = \pi(x')$$

Почему это работает - 3



Gibbs sampler:

$$q(\mathbf{x} \rightarrow x') = q((x_i, \overline{\mathbf{x}_i}) \rightarrow (x'_i, \overline{\mathbf{x}_i})) = P(x'_i|\overline{\mathbf{x}_i}, \mathbf{e})$$

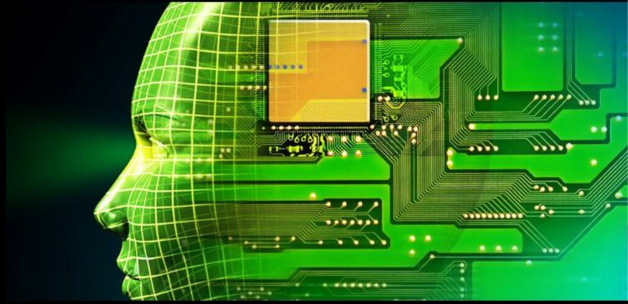
X_i - переменная для генерации

$\bar{X}_i$ - все остальные скрытые переменные

В этом случае равновесной будет апостериорная вероятность:

$$\begin{aligned}
\pi(\mathbf{x})q(\mathbf{x} \rightarrow x') &= P(\mathbf{x}|\mathbf{e})P(x'_i|\overline{\mathbf{x}_i}, e) = P(x_i, \overline{\mathbf{x}_i}|\mathbf{e})P(x'_i|\overline{\mathbf{x}_i}, \mathbf{e}) \\
&= P(x_i|\overline{\mathbf{x}_i}, e)P(\overline{\mathbf{x}_i}|\mathbf{e})P(x'_i|\overline{\mathbf{x}_i}, e) \qquad \text{(using the chain rule on the first term)} \\
&= P(x_i|\overline{\mathbf{x}_i}, e)P(x'_i, \overline{\mathbf{x}_i}|\mathbf{e}) \qquad \text{(using the chain rule backwards)} \\
&= \pi(\mathbf{x}')q(\mathbf{x}' \rightarrow x)\,.
\end{aligned}$$

Почему это работает - 4



Markov blanket:

- Родители
- Дети
- Родители детей

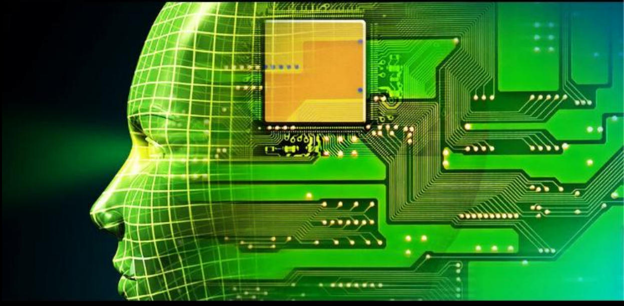
Факт №1: $$P(x_i' | \overline{\mathbf{x}_i}, \mathbf{e}) = P(x_i' | mb(X_i))$$

Факт №2:

$$P(x_i' | mb(X_i)) = \alpha P(x_i' | parents(X_i)) \times \prod_{Y_j \in Children(X_i)} P(y_j | parents(Y_j))$$

Temporal model

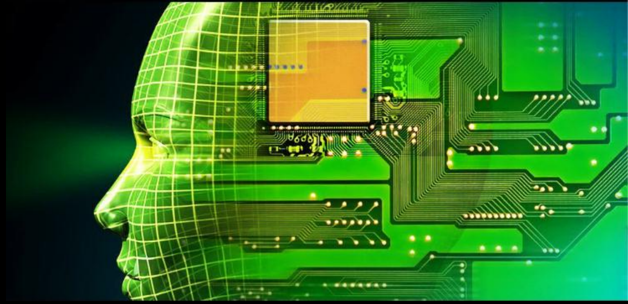
Обозначения



- X_t – множество ненаблюдаемых случайных величин в момент времени t
- E_t – множество наблюдаемых случайных величин в момент времени t

$$X_{a:b} = \bigcup_{i=a}^{b} X_i$$

Предположения



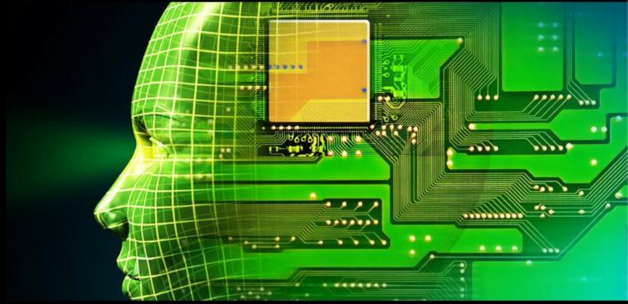
Процесс:

- Стационарный
- Марковский:

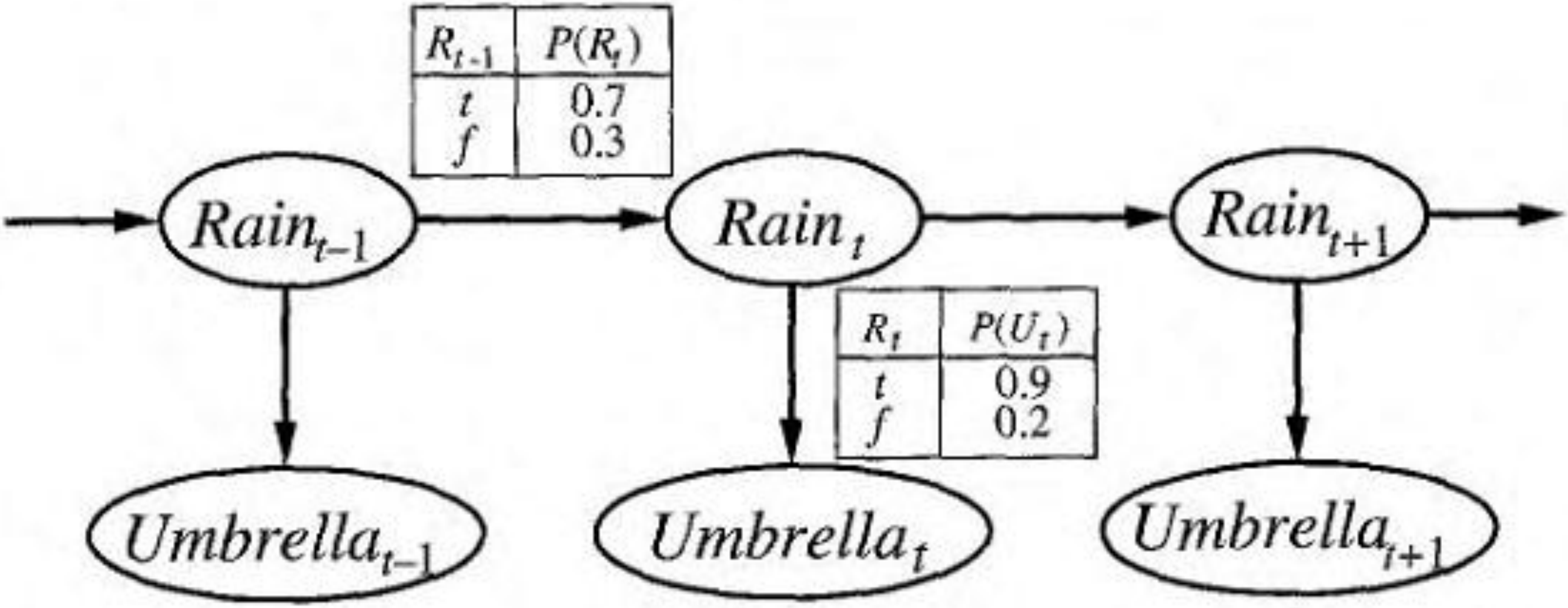
$$P(X_t \mid X_{0:t-1}) = P(X_t \mid X_{t-1})$$

Модель наблюдения:

$$P(E_t \mid X_{0:t}, E_{0:t-1}) = P(E_t \mid X_t)$$

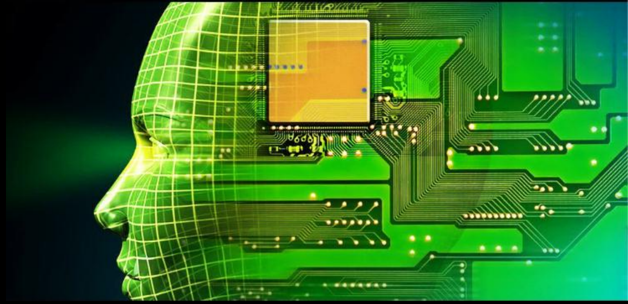


Байесовская сеть



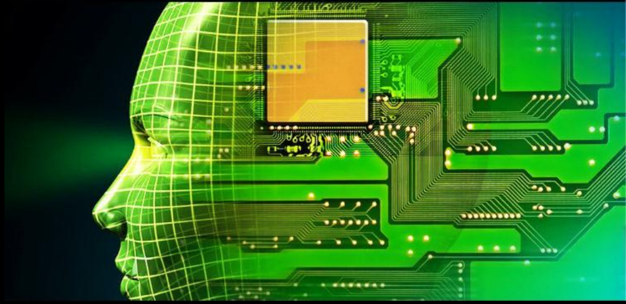
$$\mathbf{P}(\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_t, \mathbf{E}_1, \ldots, \mathbf{E}_t) = \mathbf{P}(\mathbf{X}_0) \prod_{i=1}^{t} \mathbf{P}(\mathbf{X}_i | \mathbf{X}_{i-1}) \mathbf{P}(\mathbf{E}_i | \mathbf{X}_i)$$

Inference in Temporal Models



Название задачи	Найти
Filtering (monitoring)	$P(X_t \mid e_{1:t})$
Prediction	$P(X_{t+k} \mid e_{1:t})$
Smoothing (hindsight)	$P(X_k \mid e_{1:t})$
Most likely explanation	$P(x_{1:t} \mid e_{1:t})$

Filtering and prediction



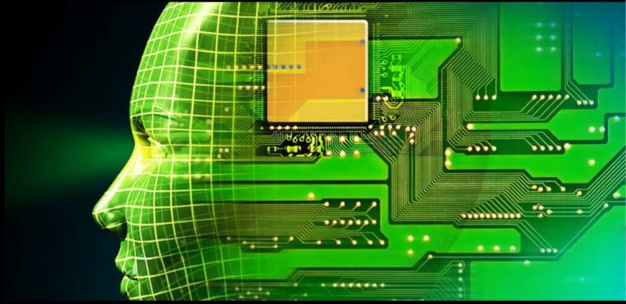
Рекурсивное вычисление Filtering:

$$
\begin{aligned}
\mathbf{P}(\mathbf{X}_{t+1}|\mathbf{e}_{1:t+1}) &= \mathbf{P}(\mathbf{X}_{t+1}|\mathbf{e}_{1:t}, \mathbf{e}_{t+1}) && \text{(dividing up the evidence)} \\
&= \alpha \mathbf{P}(\mathbf{e}_{t+1}|\mathbf{X}_{t+1}, \mathbf{e}_{1:t}) \mathbf{P}(\mathbf{X}_{t+1}|\mathbf{e}_{1:t}) && \text{(using Bayes' rule)} \\
&= a \mathbf{P}(\mathbf{e}_{t+1}|\mathbf{X}_{t+1}) \mathbf{P}(\mathbf{X}_{t+1}|\mathbf{e}_{1:t}) && \text{(by the Markov property of evidence)}
\end{aligned}
$$

Развёртываем Prediction:

$$
\begin{aligned}
\mathbf{P}(\mathbf{X}_{t+1}|\mathbf{e}_{1:t+1}) &= \alpha \mathbf{P}(\mathbf{e}_{t+1}|\mathbf{X}_{t+1}) \sum_{\mathbf{x}_t} \mathbf{P}(\mathbf{X}_{t+1}|\mathbf{x}_t, \mathbf{e}_{1:t}) P(\mathbf{x}_t|\mathbf{e}_{1:t}) \\
&= a \mathbf{P}(\mathbf{e}_{t+1}|\mathbf{X}_{t+1}) \sum_{\mathbf{x}_t} \mathbf{P}(\mathbf{X}_{t+1}|\mathbf{x}_t) P(\mathbf{x}_t|\mathbf{e}_{1:t}) && \text{(using the Markov property).}
\end{aligned}
$$

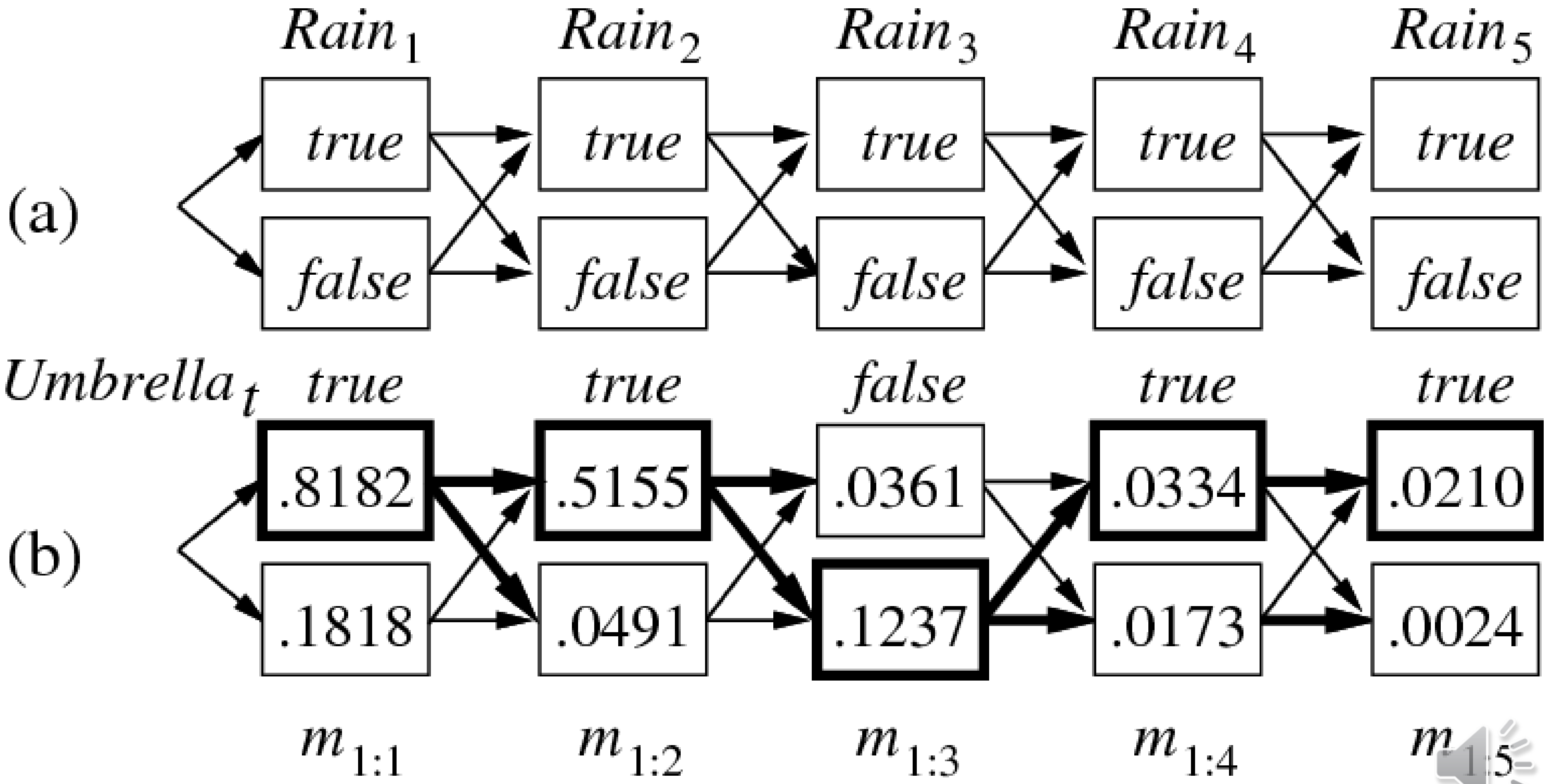
Smoothing



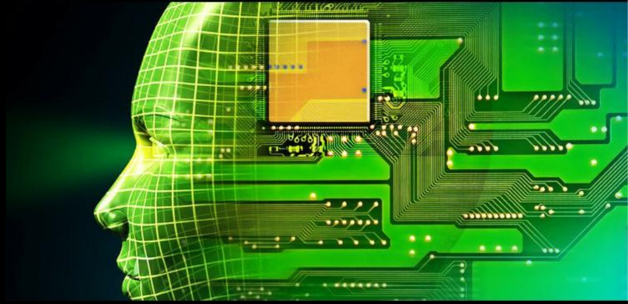
$$
\begin{aligned}
\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:t}) &= \mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k},\mathbf{e}_{k+1:t}) \\
&= \alpha\,\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k})\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k,\mathbf{e}_{1:k}) \quad \text{(using Bayes' rule)} \\
&= \alpha\,\mathbf{P}(\mathbf{X}_k|\mathbf{e}_{1:k})\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) \quad \text{(using conditional independence)} \\
&= \alpha\,\mathbf{f}_{1:k}\mathbf{b}_{k+1:t}\,,
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k) &= \sum_{\mathbf{x}_{k+1}} \mathbf{P}(\mathbf{e}_{k+1:t}|\mathbf{X}_k,\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \quad \text{(conditioning on } \mathbf{X}_{k+1}\text{)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \quad \text{(by conditional independence)} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1},\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k) \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1}|\mathbf{x}_{k+1})P(\mathbf{e}_{k+2:t}|\mathbf{x}_{k+1})\mathbf{P}(\mathbf{x}_{k+1}|\mathbf{X}_k)\,, \qquad (15.7)
\end{aligned}
$$

The most likely sequence

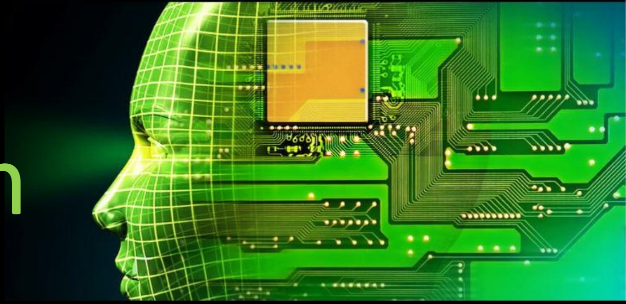


Сложность алгоритмов



Название задачи	Время	Память
Filtering (monitoring)	$O(t)$	$O(1)$
Prediction		
Smoothing (hindsight)	$O(t)$	
Most likely explanation	$O(t)$	$O(t)$

Forward-backward algorithm



Хочется узнать $P(X_k \mid e_{1:t})$ для всех *k*. Что делать?

Forward-backward algorithm:

1. Forward filtering от 1 до t

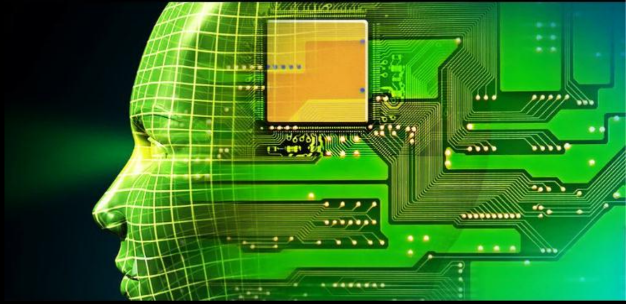
   (*запоминая результаты вычислений*)

2. Backward recursion от t до 1

Сложность во времени: $O(t)$

Сложность по памяти: $O(|f|t)$, где $|f|$ - размер forward message

Hidden Markov Model (HMM)



HMM: состояние – одна дискретная случайная величина

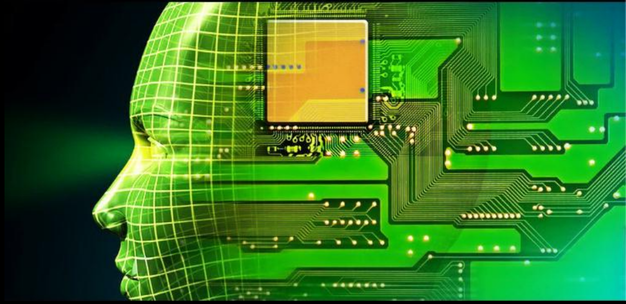
$P(X_t \mid X_{t-1}) \;\rightarrow\;$ матрица T размера $S \times S$:

$$T_{ij} = P\left(X_t = j \mid X_{t-1} = i\right)$$

O_t: на диагонали $P(e_t \mid X_t = i)$

Пример:

Элегантная запись формул



Forward equation: $$\mathbf{f}_{1:t+1} = \alpha \mathbf{O}_{t+1} \mathbf{T}^{\top} \mathbf{f}_{1:t}$$

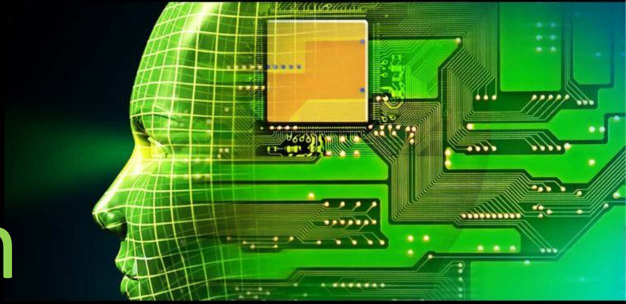
Backward equation: $$\mathbf{b}_{k+1:t} = \mathbf{T}\mathbf{O}_{k+1} \mathbf{b}_{k+2:t}$$

Forward-backward алгоритм

Сложность по времени: $O(S^2 t)$

Сложность по памяти: $O(St)$

Улучшение для forward-backward algorithm



На каждом шаге нужно знать и $f_{1:k}$ и $b_{k+1:t}$

Forward-backward algorithm:

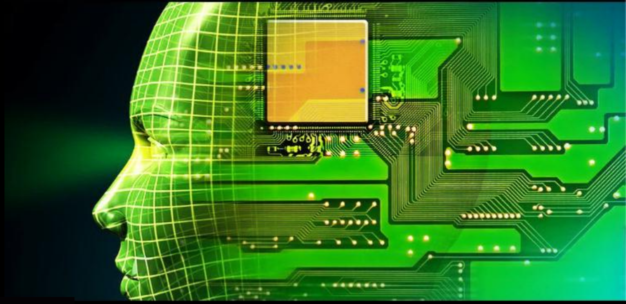
1. Forward filtering от 1 до *t*

   (не запоминая результатов вычислений)

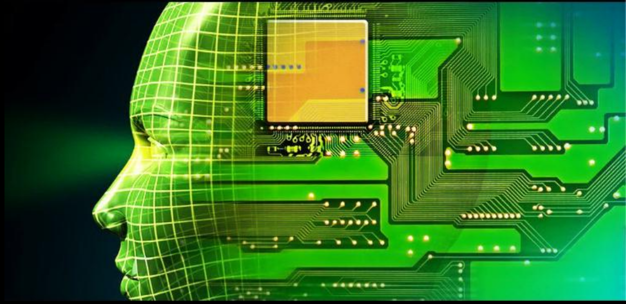
2. Backward recursion от *t* до 1

$$\mathbf{f}_{1:t} = \alpha' (\mathbf{T}^{\top})^{-1} \mathbf{O}_{t+1}^{-1} \mathbf{f}_{1:t+1}$$

Kalman Filters



Модель



Обновление состояния:

$$X_{t+\Delta} = X_t + \dot{X}\Delta$$

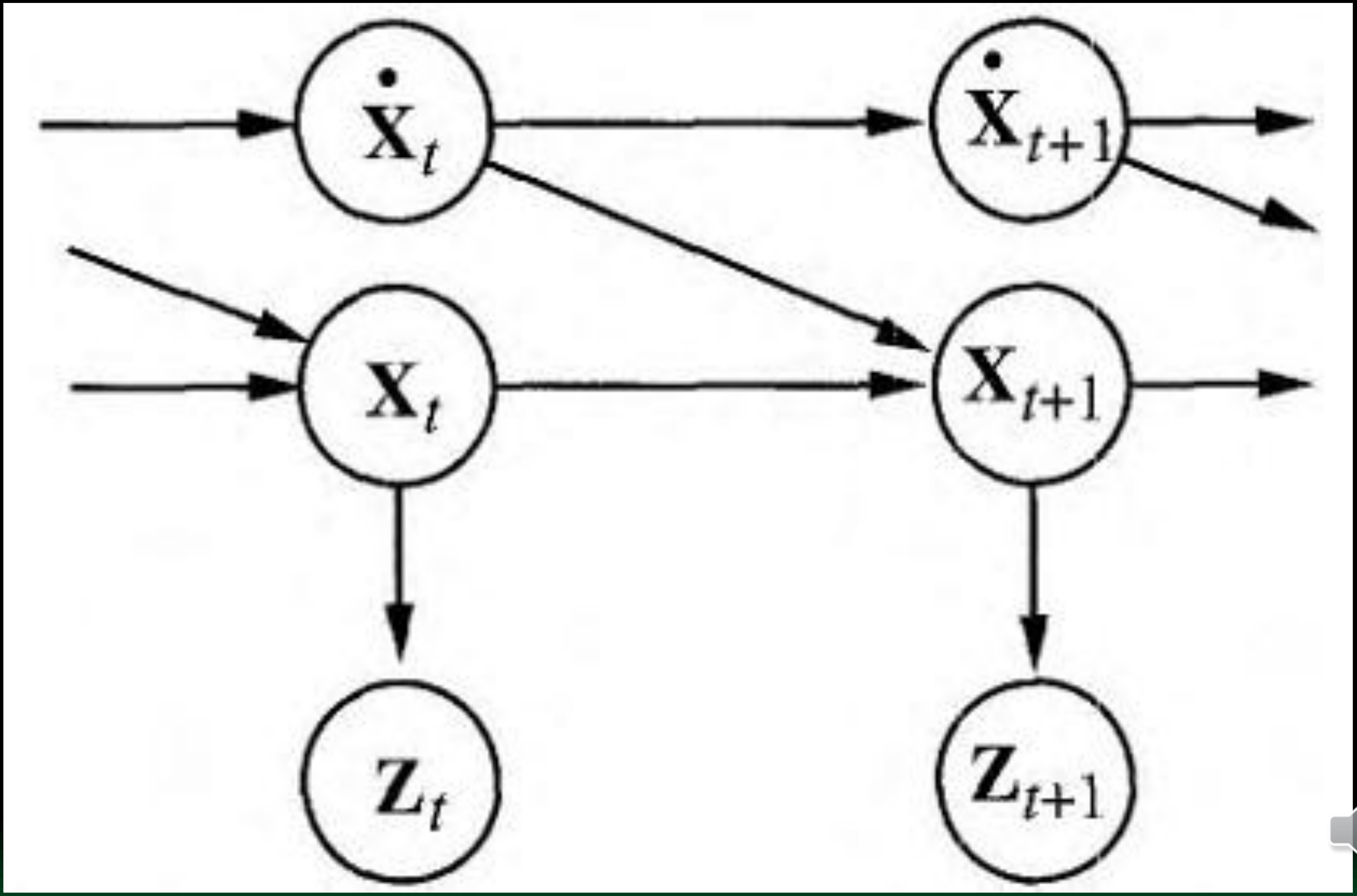
Аддитивный гауссовский шум:

$$X_{t+\Delta} = X_t + \dot{X}\Delta + N(0, \sigma)$$

$$P(X_{t+\Delta} = x_{t+\Delta} \mid X_t = x_t, \dot{X}_t = \dot{x}_t) =$$

$$= N(x_t + \dot{x}_t\Delta, \sigma)(x_{t+\Delta})$$

Байесовская сеть



Одномерный случай

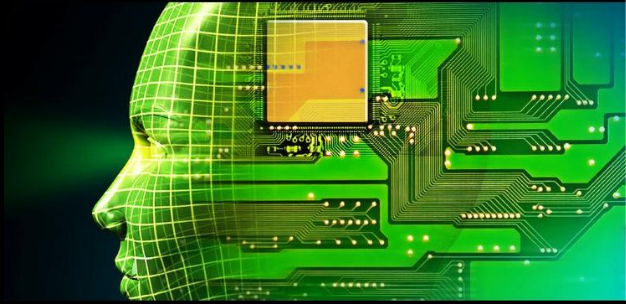
Все распределения нормальные:

$$P(x_0) = \alpha\, e^{-\frac{1}{2}\left(\frac{(x_0-\mu_0)^2}{\sigma_0^2}\right)}$$

$$P(x_{t+1}|x_t) = \alpha\, e^{-\frac{1}{2}\left(\frac{(x_{t+1}-x_t)^2}{\sigma_x^2}\right)}$$

$$P(z_t|x_t) = \alpha\, e^{-\frac{1}{2}\left(\frac{(z_t-x_t)^2}{\sigma_z^2}\right)}$$

Одномерный случай

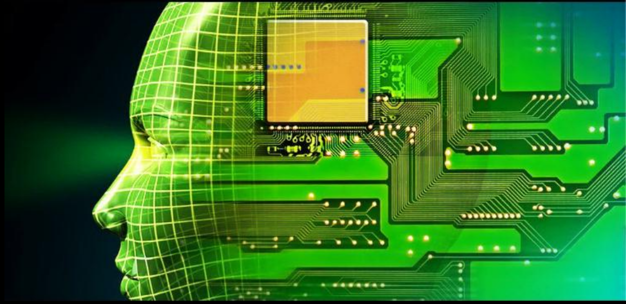


Filtering:

$$\mu_{t+1} = \frac{(\sigma_t^2+\sigma_x^2)z_{t+1}+\sigma_z^2\mu_t}{\sigma_t^2+\sigma_x^2+\sigma_z^2}$$

$$\sigma_{t+1}^2 = \frac{(\sigma_t^2+\sigma_x^2)\sigma_z^2}{\sigma_t^2+\sigma_x^2+\sigma_z^2}.$$

Общий случай



Все распределения нормальные:

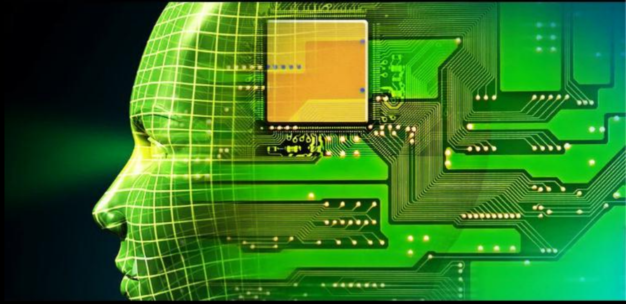
$$
\begin{aligned}
P(\mathbf{x}_{t+1}|\mathbf{x}_t) &= N(\mathbf{F}\mathbf{x}_t, \boldsymbol{\Sigma}_x)(\mathbf{x}_{t+1}) \\
P(\mathbf{z}_t|\mathbf{x}_t) &= N(\mathbf{H}\mathbf{x}_t, \boldsymbol{\Sigma}_z)(\mathbf{z}_t)\ ,
\end{aligned}
$$

Filtering:

$$
\begin{aligned}
\mu_{t+1} &= \mathbf{F}\mu_t + \mathbf{K}_{t+1}(\mathbf{z}_{t+1} - \mathbf{H}\mathbf{F}\mu_t) \\
\boldsymbol{\Sigma}_{t+1} &= (\mathbf{I} - \mathbf{K}_{t+1})(\mathbf{F}\boldsymbol{\Sigma}_t\mathbf{F}^\top + \boldsymbol{\Sigma}_x)
\end{aligned}
$$

$$
\mathbf{K}_{t+1} = (\mathbf{F}\boldsymbol{\Sigma}_t\mathbf{F}^\top + \boldsymbol{\Sigma}_x)\mathbf{H}^\top(\mathbf{H}(\mathbf{F}\boldsymbol{\Sigma}_t\mathbf{F}^\top + \boldsymbol{\Sigma}_x)\mathbf{H}^\top + \boldsymbol{\Sigma}_z)^{-1}
$$

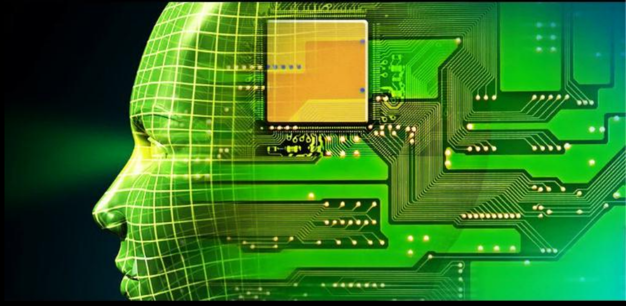
Трудности распознавания речи



- Шум в данных
  - Фоновый шум
  - Шум от оцифровки
- Одно слово произносится по-разному
- А различные слова могут звучать одинаково
- …

Проблема распознавания речи –
проблема вероятностного вывода!

Общая постановка задачи

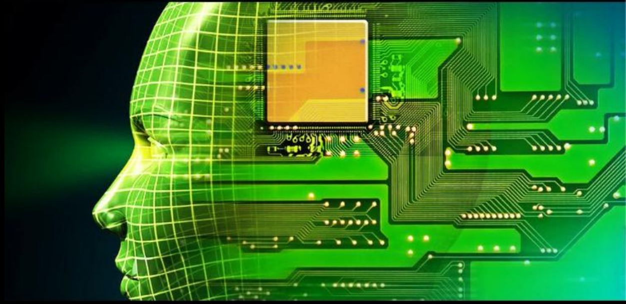


$$\max_{words} P(words \mid signal)$$

$$P(words \mid signal) =$$

$$= \alpha \underbrace{P(signal \mid words)}_{\text{акустическая модель}} \underbrace{P(words)}_{\text{языковая модель}}$$

Акустическая модель



Частота дискретизации:

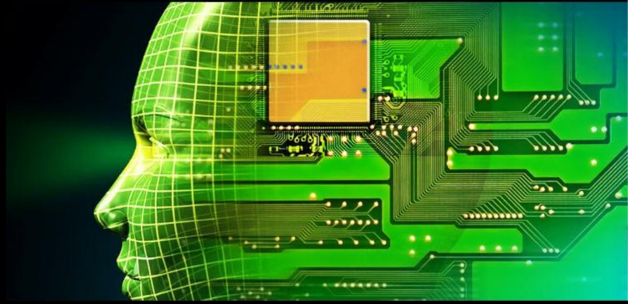
- Речь: 8-16 kHz
- Музыка: более 44 kHz

Коэффициент квантования:

8-12 бит

Представление речи: 500 Кб в минуту

Акустическая модель



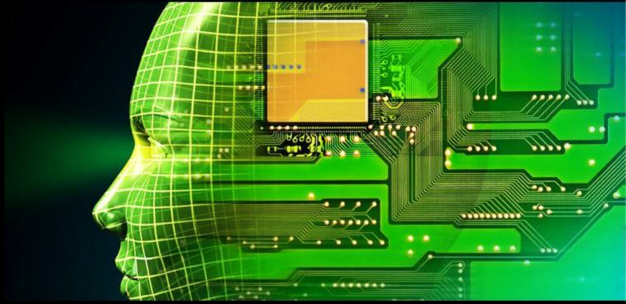
Свойства сигнала суммируются по *фреймам*

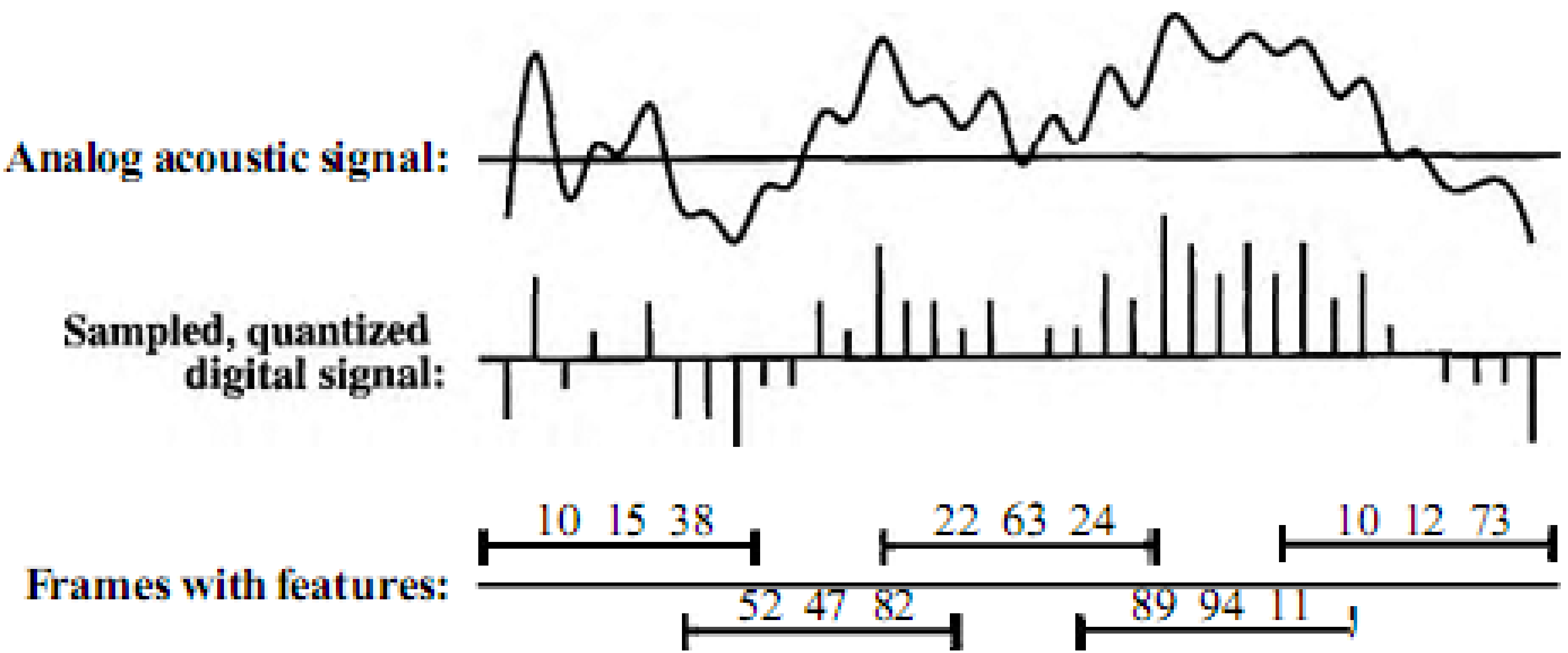
Длина фрейма: ≈ 10 мс

Features:

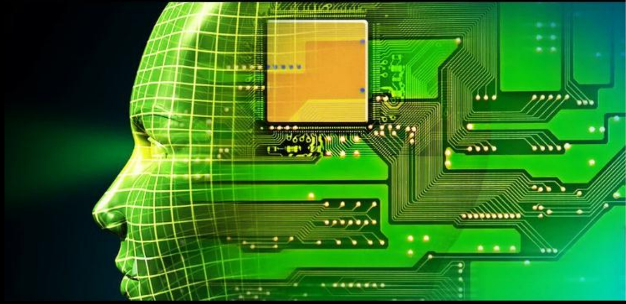
- Количество энергии в определённом частотном диапазоне
- Общее количество энергии во фрейме
- Отличие от предыдущего фрейма
- …

Пример




Analog acoustic signal:
Sampled, quantized digital signal:
10 15 38
22 63 24
10 12 73
Frames with features:
52 47 82
89 94 11

Акустическая модель



Количество возможных фреймов:

$$256^n$$

Проблема: как хранить $P(features \mid phone)$?

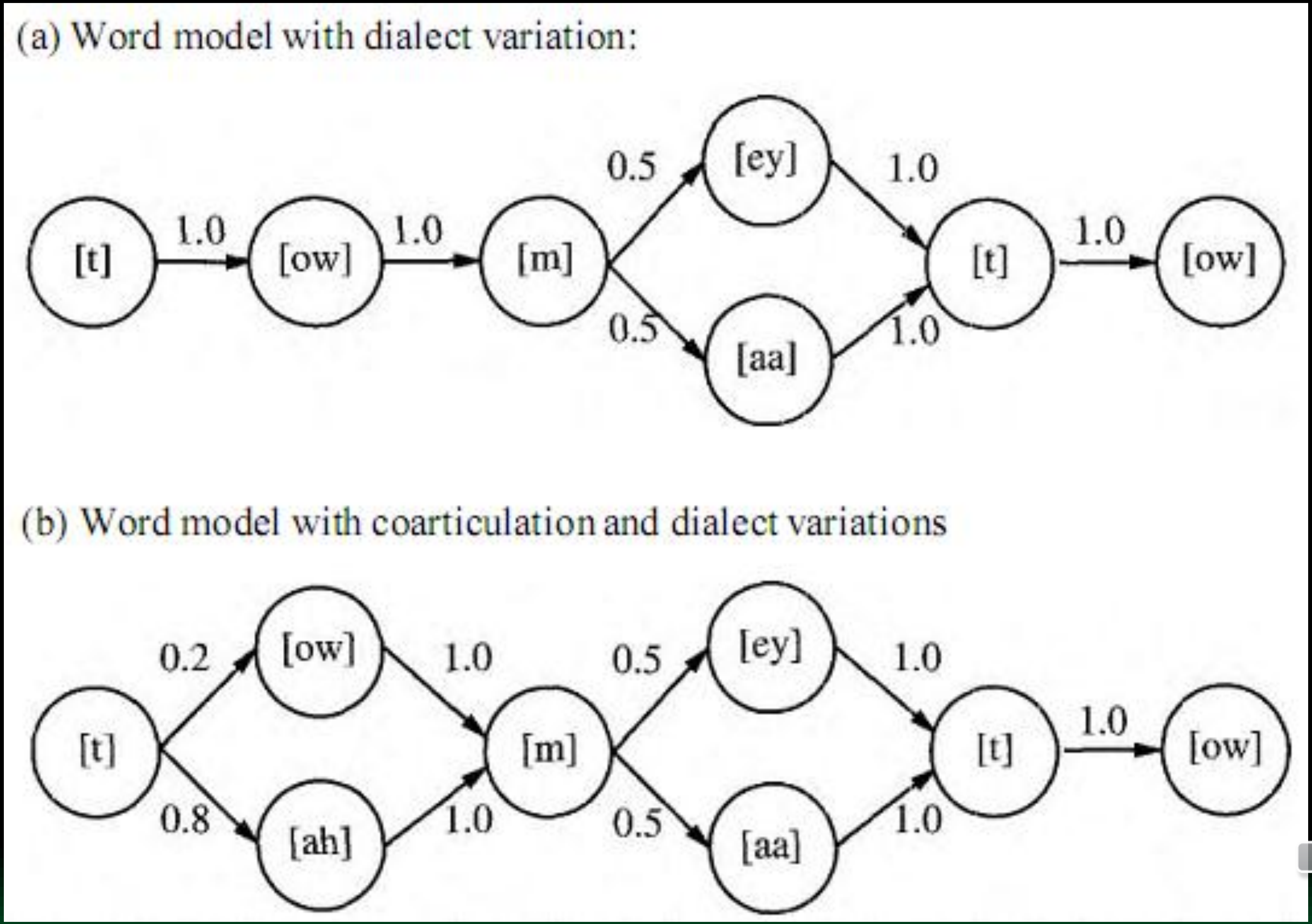
- Векторное квантование:

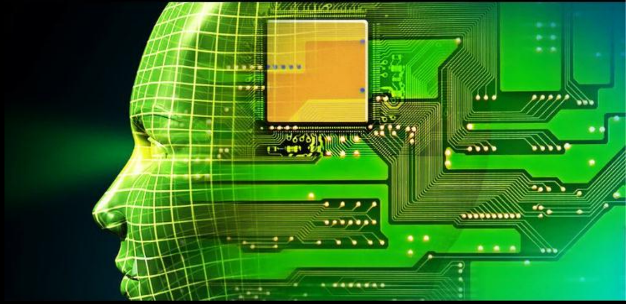
  VQ – vector quantization

- Параметризация распределения:

  Mixture of Gaussians: $O(kn^2)$ параметров

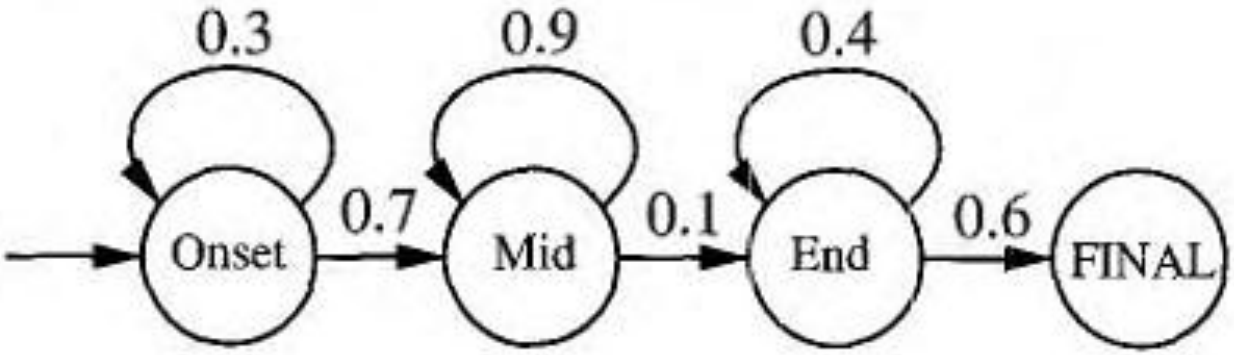
Модель произношения





Модель фонем

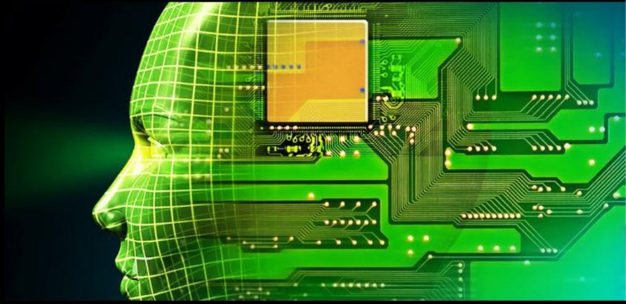
Phone HMM for [m]:



Output probabilities for the phone HMM:

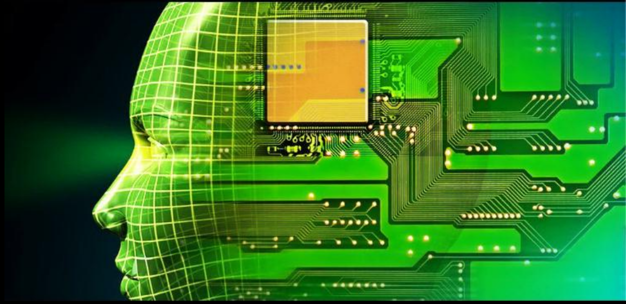
Onset:	Mid:	End:
C_1: 0.5	C_3: 0.2	C_4: 0.1
C_2: 0.2	C_4: 0.7	C_6: 0.5
C_3: 0.3	C_5: 0.1	C_7: 0.4

Распознавание изолированных слов



$$P(word \mid e_{1:t}) = \alpha P(e_{1:t} \mid word) P(word)$$

Предложения



$$P(w_1, \dots, w_n) =$$

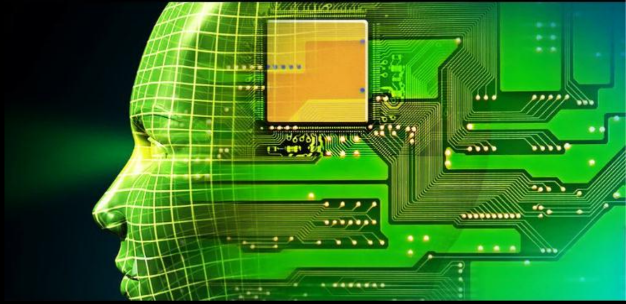
$$= P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1 w_2) \dots P(w_n \mid w_1 \dots w_{n-1})$$

$$= \prod_{i=1}^{n} P(w_i \mid w_1 \dots w_{i-1})$$

Bigram model:

$$P(w_i \mid w_1 \dots w_{i-1}) = P(w_i \mid w_{i-1})$$

Распознавание речи



1. Комбинированная HMM

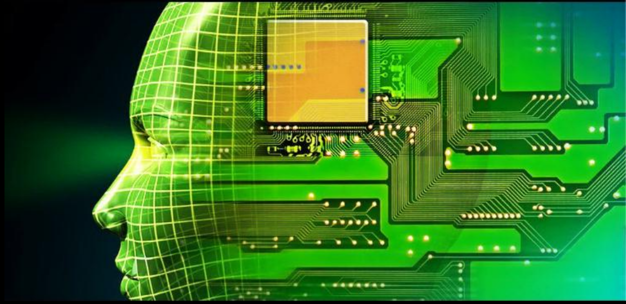
   Состояния: $[m]^{tomato}_{Onset}$

2. Алгоритм Витерби → наиболее вероятная последовательность состояний

3. Извлекаем последовательность слов

Но это *не* наиболее вероятная последовательность слов!

Декодер A*

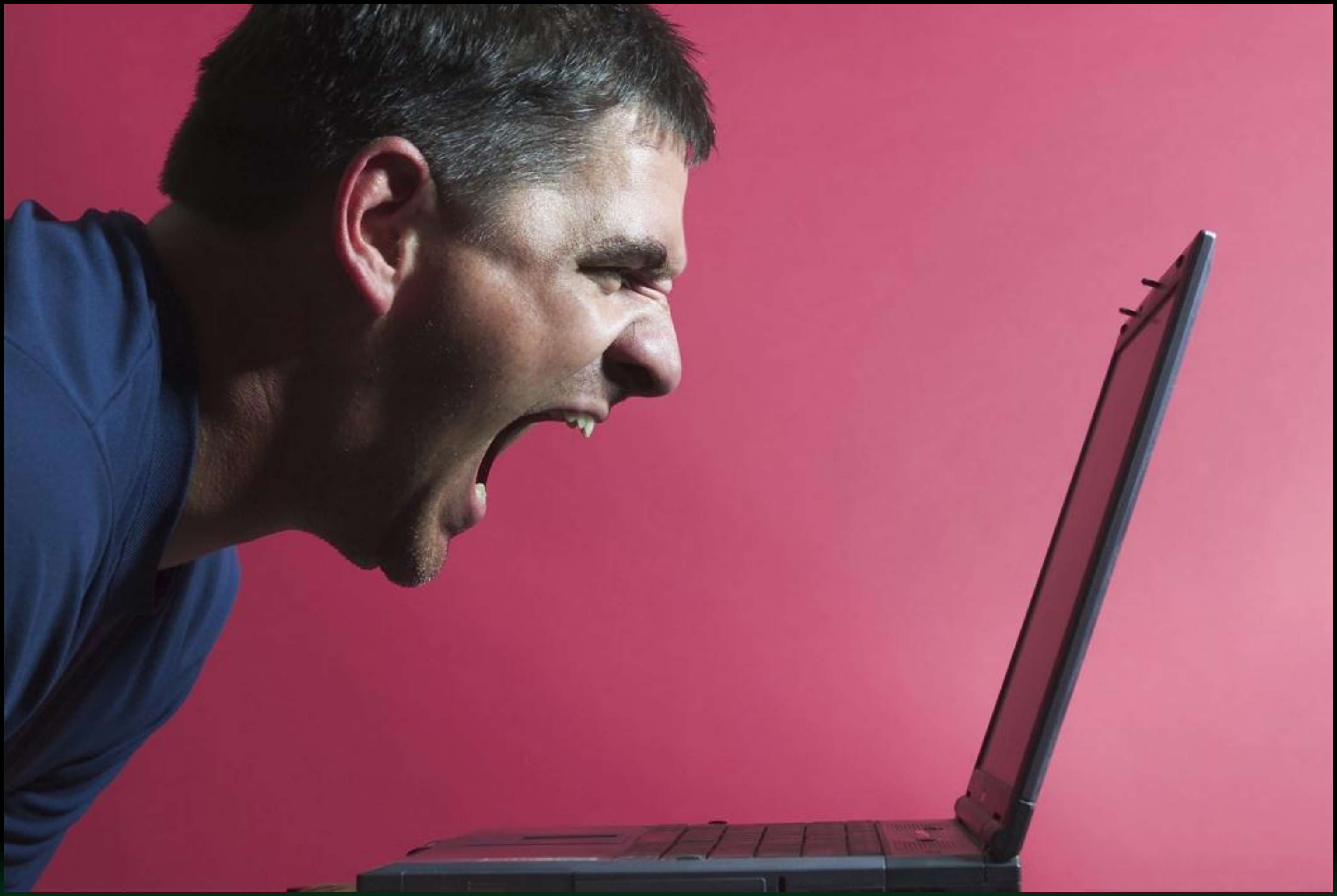


Ориентированный граф:

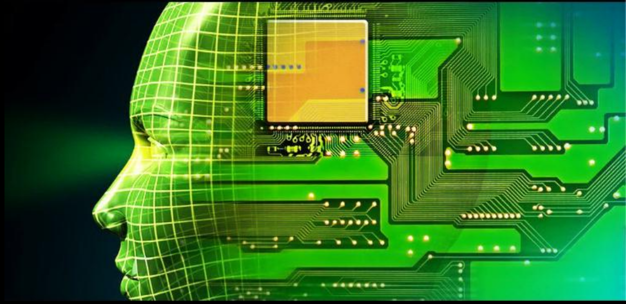
- Вершина: слово
- Ребро (w_1, w_2): слово w_2 может идти после слова w_1
- Стоимость ребра: $-\log P(w_2|w_1)$

$$Cost(w_1 \cdots w_n) = \sum_{i=1}^{n} -\log P(w_i|w_{i-1}) = -\log \prod_{i=1}^{n} P(w_i|w_{i-1})$$

Microsoft Speech Recognition



Контакты



Илья Лысенков

ilya.lysenkov@gmail.com

http://twitter.com/ilysenkov

Клуб Computer Science and Its Applications:

http://groups.google.com/group/computer-science-and-its-applications