

Конспект по Дискретной математике.

Чепелин В.А.

Содержание

1 Лекция 1.

1.1 Аксиоматическое вероятное пространство.

2 Лекция 2.

2.1 Случайная величина.

2.2 Мат. ожидание.

2.3 Незав. случайные величины

2.4 Дисперсия случайной величины.

3 Лекция 3.

3.1 Ковариация

3.2 Корреляция

3.3 Хвостовые неравенства

4 Лекция 4.

4.1 Введение в теорию информации.

4.2 Энтропия

5 Лекция 5.

5.1 Марковские цепи

6 Лекция 6.

7 Лекция 7.

7.1 Формальные языки

7.2 Описание языка.

7.3 Регулярные (автоматные языки)

7.4 Детерминированный Конечный Автомат (ДКА)

8 Лекция 8.

8.1 Недетерминированный конечный автомат (НКА).

8.2 ε - НКА

9 Лекция 9.

9.1 Теорема Клини.

9.2 Нерегулярность

10 Лекция 10.

10.1 Система линейных уравнений в регулярных выражениях.

10.2 Алгоритм анализа регулярных языков.

10.3 Алгоритм минимизации.

11 Лекция 11.

11.1 КС грамматика

12 Лекция 12.

12.1 Приведение грамматики к НФХ.

12.2 Алгоритм Коко-Янгера-Касами (КЯК или СҮК)

12.3 МП автомат

13 Информация о курсе

1 Лекция 1.

1.1 Аксиоматическое вероятное пространство.

Пусть у нас есть Ω - элементарные исходы и связанная с ним функция $p : \Omega \rightarrow [0, 1]$ - дискретная вероятностная мера (плотность вероятности) - функция, которая по элементарному исходу возвращает вероятность.

А также $\sum_{w \in \Omega} p(w) = 1$, а также $0 \leq p_i \leq 1$ А также мы считаем, что $|\Omega|$ не более чем счетно. Для множеств мощности континуума нам нужна более сложная теория.

Рассмотрим **примеры**:

1. Честная монета:

$$\Omega = \{0, 1\}. p(0) = p(1) = \frac{1}{2}.$$

2. Нечестная монета или распределение Бернулли:

$$\Omega = \{0, 1\}. p(0) = 1 - p(1) = q.$$

3. Честная игральная кость:

$$\Omega = \{1, 2, 3, 4, 5, 6\}. p(w) = \frac{1}{6}. p(w) = \frac{1}{52}$$

4. Колода карт:

$$\Omega = \{\langle c, r \rangle \mid 1 \leq c \leq 4, 1 \leq r \leq 15\}$$

5. Геометрическое распределение:

$$\Omega = \mathbb{N}, p(i) = \frac{1}{2^i}$$

Замечание. Не существует равномерного распределения на счетном множестве.

Событие — множество $A \subset \Omega$. $P(A) = \sum_{w \in A} p(w)$. (Иногда используют \Pr).

$P(A) = 1$ — достоверное событие.

$P(A) = 0$ — невозможное событие.

Рассмотрим примеры на честной игральной кости:

1. Только четные: $P(A) = \frac{3}{6} = \frac{1}{2}$.

2. Больше 4-ех: $P(A) = \frac{2}{6} = \frac{1}{3}$.

Замечание: нельзя с равной вероятностью выбрать случайное целое число.

Независимые события — A, B независимы, если $P(A \cap B) = P(A) \cdot P(B)$.

$\frac{P(A \cap B)}{P(B)} = \frac{P(A)}{P(\Omega)}$ — независимы (если произошло B , то вероятность не поменялась)

$P(A|B) = \frac{P(A \cap B)}{P(B)}$ — вероятность A при условии B — условная вероятность.

Произведение вероятностных пространств.

Пусть у нас есть Ω_1, p_1 , а также Ω_2, p_2 , тогда произведение вероятностных пространств:

$$\Omega = \Omega_1 \times \Omega_2$$

$$p(\langle w_1, w_2 \rangle) = p_1(w_1) \cdot p_2(w_2)$$

Утв. $\forall A \subset \Omega_1, B \subset \Omega_2$.

$A \times \Omega_2$ и $\Omega_1 \times B$ независимы.

Пусть у нас есть n событий: A_1, A_2, \dots, A_n .

Тогда обычно **независимость n событий** подразумевает:

1. A_i, A_j - независимы $\forall i, j, \quad i \neq j$
2. $\forall I \subset \{1, 2, 3, \dots, n\}: P(\bigcap_{i \in I} A_i) = \prod_{i \in I} P(A_i)$

Формула полной вероятности

$\Omega = A_1 \cup A_2 \cup \dots \cup A_n, \forall i \neq j: A_i \cap A_j = \emptyset$ — **полная система событий**.

Возьму B - какое-то событие.

$$P(B) = \sum_{i=1}^n P(B \cap A_i) = \sum_{i=1}^n P(B|A_i) \cdot P(A_i)$$

Пример: урна с шариками. Сначала выбираете урну, потом достаете шарик.

Формула Байеса.

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)} = \frac{P(B|A_i) \cdot P(A_i)}{\sum_{j=1}^n P(B|A_j) \cdot P(A_j)}$$

2 Лекция 2.

2.1 Случайная величина.

Случайная величина или численная характеристика каждого элементарного исхода — это отображение $\xi : \Omega \rightarrow \mathbb{R}$, которое сопоставляет каждому элементарному исходу какое-то число.

Пример:

1. $D = \{1, 2, \dots, 6\}$. Возьмем $\Omega = D^2$. Например, человек бросает два игральных кубика. Тогда, очевидно, $p(\langle i, j \rangle) = \frac{1}{36}$. И тогда он задает функцию случайной величины, например, как $\xi(\langle i, j \rangle) = i + j$.
2. Возьмем случайный граф G на n вершинах. $\xi(G) =$ количеству компонент связности. Или $\xi(G) =$ количеству ребер в этом графе.
3. Давайте кидать игральный кубик и сопоставим каждой выпадающей грани число, равное количеству точек на этой грани. То есть $\Omega = \{1, 2, \dots, 6\}$, $\xi(i) = i$.
4. $\Omega = \{1, 2, \dots, 6\}$; $E = \{2, 4, 6\}$. $x_E(w) = \begin{cases} 1, w \in E \\ 0, w \notin E \end{cases}$

Возьмем какие-то Ω, p, ξ :

$[\xi = i] = \{w | \xi(w) = i\} \subset \Omega$ — множество элементарных исходов, случайная величина которых равна i .

def: $f_\xi : \mathbb{R} \rightarrow \mathbb{R}$ — дискретная плотность вероятности ξ .

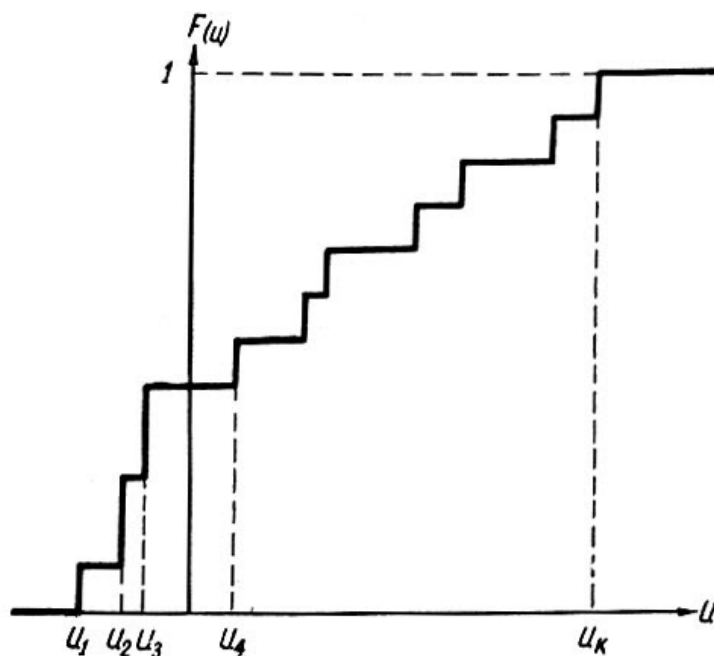
$$P([\xi = i]) = P(\xi = i) = f_\xi(i) = \sum_{w \in [\xi=i]} p(w)$$

Дискретная плотность вероятности — это функция, которая говорит нам, насколько вероятно каждое из этих отдельных значений, которые может принимать случайная величина. Другими словами, она присваивает вероятность каждому возможному исходу.

Немного поменяем и получим $[\xi \leq i] = \{w | \xi(w) \leq i\} \subset \Omega$.

$$P([\xi \leq i]) = P(\xi \leq i) = F_\xi(i)$$

def: $F_\xi : \mathbb{R} \rightarrow \mathbb{R}$ — функция распределения. У дискретной случайной величины функция распределения ступенчатая. Например:



2.2 Мат. ожидание.

Математическое ожидание — среднее значение случайной величины.

$$E_{\xi} = \sum_w p(w) \xi(w) = \sum_i i \cdot P(\xi = i).$$

Дальше А.С. использует 3 вида обозначений:

1. E_{ξ} 2. $E(\xi)$ 3. $E\xi$ — не боимся, это одно и то же.

Теорема (линейность мат ожидания)

$$E\lambda\xi = \lambda E_{\xi} \quad E_{(\xi+\eta)} = E_{\xi} + E_{\eta}$$

Доказательство:

$$E\lambda\xi = \sum_w p(w) \cdot \lambda\xi(w) = \lambda \sum_w p(w) \xi(w) = \lambda E_{\xi}$$

$$E(\xi + \eta) = \sum_w p(w)(\xi(w) + \eta(w)) = \sum_w p(w)\xi(w) + \sum_w p(w)\eta(w) = E(\xi) + E(\eta)$$

Q.E.D.

МАТ. ОЖИДАНИЕ ВСЕГДА ЛИНЕЙНО!!!

2.3 Незав. случайные величины

ξ, η - независимы, если $[\xi = a], [\eta = b]$ — независимы $\forall a, b$.

Эквивалентное утверждение — $[\xi \leq a], [\eta \leq b]$ — независимы $\forall a, b$.

Иначе говоря, две случайные величины называются *независимыми*, если по значению одной нельзя сделать выводы о значении другой.

Теорема (о мультипликативности мат. ожидания)

ξ, η — независимы $\Rightarrow E(\xi \cdot \eta) = E_\xi \cdot E_\eta$.

Доказательство:

$$\begin{aligned} E_{(\xi \cdot \eta)} &= \sum_a a P(\xi, \eta = a) = \sum_a a \sum_{\forall i, j: i \cdot j = a} P(\xi = i, \eta = j) = \\ &= \sum_a \sum_i \sum_j a P(\xi = i) P(\eta = j) = \sum_i i P(\xi = i) \cdot \sum_j j P(\eta = j) = E_\eta \cdot E_\xi \end{aligned}$$

Q.E.D.

2.4 Дисперсия случайной величины.

$D_\xi = Var(\xi)$ — дисперсия случайной величины.

$$D_\xi = E((\xi - E_\xi)^2) = E_{\xi^2} - (E_\xi)^2$$

Дисперсия случайной величины — это мера того, насколько сильно разбросаны значения этой случайной величины вокруг её математического ожидания (среднего значения). Другими словами, она показывает, насколько "широко" распределение вероятностей случайной величины.

Теорема (свойства дисперсии). Если ξ, η - независимы:

$$D_{c\xi} = c^2 D_\xi \quad D_{\xi+\eta} = D_\xi + D_\eta$$

Доказательство тривиально из линейности мат. ожидания.

3 Лекция 3.

3.1 Ковариация

$$Cov(\xi, \eta) = E_{\xi\eta} - E_{\xi}E_{\eta}$$

Ковариация или **корреляционный момент** показывает на сколько зависимы случайные величины это мера зависимости двух случайных величин.

Если ξ, η - независимые случайные величины

$$Cov(\xi, \eta) = 0$$

:

$$Cov(\xi, \xi) = D_{\xi} = Var_{\xi} - \text{вариация}$$

3.2 Корреляция

$$Corr(\xi, \eta) = \frac{E_{\xi\eta} - E_{\xi}E_{\eta}}{\sqrt{D_{\xi} \cdot D_{\eta}}} = \frac{Cov(\xi, \eta)}{\sqrt{D_{\xi} \cdot D_{\eta}}}$$

Корреляция - статистическая взаимосвязь двух случайных величин. Корреляция является **нормированной** версией ковариации, что позволяет сравнивать силу линейной зависимости между различными парами переменных, независимо от их масштаба.

Теорема (об ограниченности корреляции)

$$-1 \leq Cor(\xi, \eta) \leq 1$$

Доказательство:

Возьму $\alpha = \xi - \lambda\eta$:

$$D\alpha = D(\alpha) = E\xi^2 - 2\lambda E_{\xi\eta} + \lambda^2 E\eta^2 - (E\xi)^2 + 2\lambda E_{\xi}E_{\eta} - \lambda^2 (E\eta)^2 \geq 0$$

$$D\xi - 2\lambda Cov(\xi, \eta) + \lambda^2 D\eta \geq 0$$

Откуда, если рассматривать это, как уравнение относительно λ , то $D \leq 0$, то есть:

$$4Cov(\xi, \eta) - 4D_{\eta}D_{\xi} \leq 0$$

А если присмотреться, то это и есть то, что нам надо.

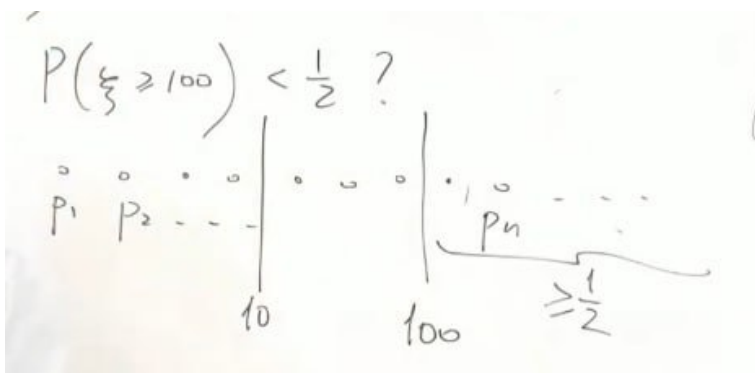
Q.E.D.

3.3 Хвостовые неравенства

Рассмотрим азартную игру. ~~не одобряем, не играем.~~

Проводится случайный эксперимент, смотрится значение ξ . Если оно получилось 100 или больше, то мы платим 100 рублей, а иначе наш друг платит нам 100 рублей. Мы знаем $E\xi = 10, \xi \geq 0$.

Хотим оценить $P(\xi \leq 100)$:



Давайте посмотрим, является ли наша вероятность меньше $\frac{1}{2}$. Тогда всё, что правее 100 имеет вероятность выпадения $\geq \frac{1}{2}$. Все левое оценивается нулем, откуда мат ожидание хотя бы 50. Такого быть не может. В общем случае:

Теорема (Неравенство Маркова)

$$\xi \neq 0, \xi \geq 0 : \forall a \geq 1 : P(\xi \geq a \cdot E\xi) \leq \frac{1}{a}$$

Доказательство:

$$E\xi = \sum_v v \cdot P(\xi = v) = \sum_{v < a \cdot E\xi} v P(\xi = v) + \sum_{v \geq a \cdot E\xi} v P(\xi = v) \geq \sum_{v \geq a \cdot E\xi} a E\xi P(\xi = v) = a E\xi \cdot P(\xi \geq a \cdot E\xi)$$

Q.E.D.

Теорема (Неравенство Чебышева)

Абсолютная версия и относительная версия ($\alpha = \lambda\sigma$):

$$P(|\xi - E\xi| \geq \alpha) \leq \frac{D\xi}{\alpha^2} \quad P(|\xi - E\xi| \geq \lambda\sigma) \leq \frac{1}{\lambda^2}$$

Доказательство:

Возьму вот такие величины:

$$D\xi = E(\xi - E\xi)^2 \quad \eta = (\xi - E\xi)^2$$

Заметим, что $E\eta = D\xi$. Используем неравенство Маркова для оценки дисперсии:

$$P(\eta \geq c \cdot E\eta) \leq \frac{1}{c}$$

Возьму $c = \frac{D\xi}{\alpha^2}$ и получу искомое.

Q.E.D.

Нечестная монета. Вот вам дали домашку, вместе с вопросом $p > \frac{1}{2}$ или $p < \frac{1}{2}$. Что вы можете делать? Только кидать ее, но при этом бесконечное количество раз вы не кинете, у вас дедлайн домашки через час.

Пусть мы бросили n раз. Выпало c единиц и $n - c$ нулей. Пусть $c \leq \frac{n}{2}$:

$$P(\xi = c) \leq P(\xi \leq c) \leq P(|\xi - pn| \geq pn - c) \leq P(|\xi - pn| \geq \frac{n}{2} - c) \leq \frac{n}{4} \cdot \frac{1}{\left(\frac{n}{2} - c\right)^2}$$

Что это концептуально значит? На самом деле, это дает нам оценку на распределение. Зачем? Чтобы **СДАТЬ** домашку.

Теорема (Граница Чернова)

$$P(\xi \geq (1 + \varepsilon)p) \leq e^{-\frac{\varepsilon^2}{2+\varepsilon}np} \quad \Leftrightarrow \quad P(\xi \geq (1 + \varepsilon)p) \leq e^{-\frac{\varepsilon^2}{2}np}$$

$$e^{-\frac{\varepsilon^2}{3}np} \leq \delta \quad \Leftrightarrow \quad -\frac{\varepsilon^2}{3}np \leq \ln \delta$$

$$n \geq \frac{3}{p\varepsilon^2} \ln \frac{1}{\delta}$$

Не знаю, что это концептуально, напишите пж

4 Лекция 4.

4.1 Введение в теорию информации.

информация = - неопределенность - сказал дяденька Шеннон

Для осознания нам поможет рисунок АС:



Есть что-то - неизвестное - облачко. Затем, вы с помощью глаза заглядываете туда, и ваша неопределенность уменьшается. Соответственно вы получили информацию. То есть сначала была неопределенность H_1 , потом H_2 . $I = H_1 - H_2$, откуда и получается наша формула. У него есть глубокий смысл, но создается вопрос: «И че? И что это за неопределенность?»

Ну наличие глаза мешает, непонятно, фу фу фу. Поэтому хотим ввести что-то более формальное и менее абстрактное.

Пусть у нас есть какой-то случайный эксперимент Ω , с вероятностями p_1, \dots, p_n . И вот мы получили информацию что выпало (например орел на монетке).

Случайный источник — черный ящик с красной кнопкой, который показывает номер эл. исхода, когда вы нажимаете на красную кнопку.

Возьмем монетку. Кинули, получили 0 или 1. Теперь возьмем кубик, получим число от 1 до 6. Когда мы кидаем кубик, мы получаем больше информации. И вот Шеннон решил систематизировать все это...

4.2 Энтропия

Пусть у нас есть случайный источник и вероятности p_1, p_2, \dots, p_n . Мы хотим померить численно сколько информации содержится в одном эксперименте:

$$H(p_1, \dots, p_n) : RS \rightarrow R^+$$

Энтропия Шеннона (H) - это мера неопределенности или случайности, связанная с случайной переменной. Она измеряет среднее количество информации, необходимое для описания результата случайной переменной. Иными словами, энтропия показывает, насколько непредсказуемым является источник информации.

Возьму пример $p_i = \frac{1}{n}$. Введем новое обозначение:

$$h(n) = H\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right)$$

Очевидно, что $h(n+1) > h(n)$.

Теперь рассмотрим вероятностное пространство и источник на нем:

$$\Omega = \{(1, 1), (1, 1), \dots, (1, m_1), (2, 1), \dots, (k, 1), \dots, (k, m_k)\}$$

И давайте теперь каждому причислим какую-то q_{ij} , так, что в сумме 1. $p_i = \sum_{j=1}^{m_i} q_{ij}$. Пусть наш случайный источник сломан и показывает только одно число. Если я возьму сломанный случайный источник от Ω , то мы получим столько же информации сколько и у случайного источника сделанного из p .

Теперь давайте делить это на 2 части. Что вот мы сначала видим первую часть информации, а потом хоба и видим вторую часть информации. И того мы получаем, что когда мы открываем вторую часть мы получим $p_i H(\frac{q_{i1}}{p_i}, \dots, \frac{q_{im_i}}{p_i})$ информации. Откуда благодаря таким рассуждение получаем свойство, которое называется аддитивностью энтропии:

$$H(p_1, \dots, p_k) + \sum_{i=1}^k p_i H(\frac{q_{i1}}{p_i}, \dots, \frac{q_{im_i}}{p_i}) = H(q_{11}, \dots, q_{mk})$$

Также для фиксированного n , H непр из $\mathbb{R}^n \rightarrow \mathbb{R}$.

Теорема.(Формула энтропии Шеннона)

$$H(p_1, \dots, p_n) = -\alpha \sum_{i=1}^n p_i \log_2 p_i$$

α отвечает за выбор единицы измерений.

Доказательство:

Лемма 1. $h(n \cdot m) = h(n) + h(m)$.

Доказательство:

Возьмем $k = n, m_i = m, p = \frac{1}{n}, q_{ij} = \frac{1}{nm}$. Из утверждения сверху это верно!

Q.E.D.

Фиксируем $h(2) = \alpha$. Тогда:

Лемма 2. $h(2^k) = k\alpha$. тривиально из Леммы 1.

Лемма 2,5. $h(n^r) = rh(n)$. тривиально из Леммы 1.

Лемма 3. $h(n) = \alpha \log_2 n$

Доказательство:

Найду i такое, что $2^i \leq n^r < 2^{i+1}$, где $r \in \mathbb{N}$.

Из монотонности h следует: $\alpha i \leq h(n^r) < \alpha(i+1)$. Поэтому:

$$\alpha i \leq rh(n) < \alpha \Leftrightarrow a \frac{i}{r} \leq h(n) \leq a \frac{i+1}{r}$$

Также мы знаем, что $i \leq r \log_2 n < i + 1$. Получим, что:

$$\alpha \frac{i}{r} \leq \alpha \log_2 n < \alpha \frac{i+1}{r}$$

То есть $\forall r : |h(n) - \alpha \log_2 n| \leq \frac{\alpha}{r}$. Откуда, получаем требуемое равенство.

Q.E.D.

Возвращаемся к доказательству теоремы. Пусть p_i рациональные. Приведем все p к общему знаменателю и пусть теперь $p_i = \frac{a_i}{b_i}$. Возьму $m_i = a_i$, $r_{ij} = \frac{1}{a_i}$, $q_{ij} = \frac{1}{n}$. Подставим во второе неравенство получим:

$$H\left(\frac{1}{b}, \frac{1}{b}, \dots, \frac{1}{b}\right) = H(p_1, p_2, \dots, p_k) + \sum_{i=1}^k p_i H\left(\frac{1}{a_i}, \dots, \frac{1}{a_i}\right)$$

Что тут происходит? Я разбиваю каждый исход изначальный, на a_i исходов по $\frac{1}{b_i}$. С одной стороны я получаю b исходов по $\frac{1}{b}$. С другой стороны я могу выбрать исход, а потом его разбить. Откуда по аддитивности и получается такая формула. А она в свою очередь уже удобная, так как в ней повторяются значения внутри H , так что можем заменить на h :

$$h(b) = H(p_1, \dots, p_k) + \sum_{i=1}^k p_i h(a_i)$$

Заметим, что $\sum_{i=1}^n p_i = 1$, так что левую часть на эту сумму:

$$\sum_{i=1}^n p_i h(b) = H(p_1, \dots, p_k) + \sum_{i=1}^k p_i h(a_i)$$

$$H(p_1, \dots, p_k) = \sum_{i=1}^n p_i (h(b) - h(a_i))$$

$$H(p_1, \dots, p_k) = \sum_{i=1}^n p_i (\alpha \log_2 b - \alpha \log_2 a_i) = -\alpha \sum_{i=1}^n p_i \log_2 p_i$$

Эта формула верна и не для рац. исходя непрерывности (любое не рац. можно зажать с двух сторон сходящимися последовательностями и мы победили)

Q.E.D.

α — бит, единица информации.

Обычно используется логарифм по основанию 2, тогда энтропия измеряется в битах (или "Шеннонах"). Если используется натуральный логарифм (основание e), то энтропия измеряется в натах. Использование логарифма по основанию 10 даёт единицы измерения в децитах (Hartleys). Выбор основания влияет только на масштаб энтропии, а не на её относительные значения.

Энтропия Шеннона имеет широкое применение в различных областях, включая:

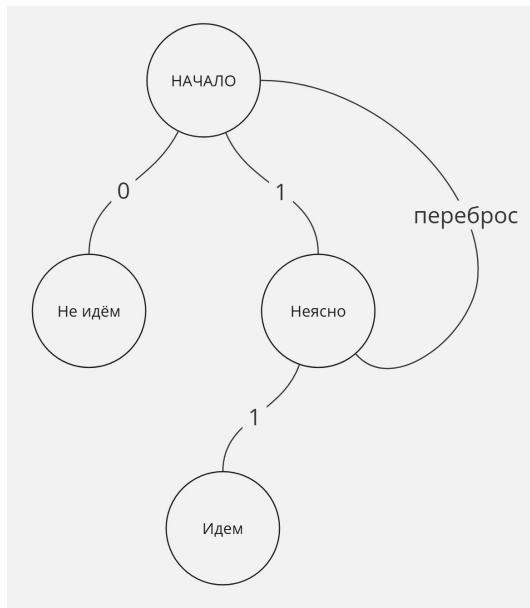
- Теория информации: Является фундаментальным понятием для измерения количества информации.
- Сжатие данных: Используется для оценки теоретического предела сжатия данных.
- Криптография: Оценка случайности ключей и стойкости шифров.
- Машинное обучение: В деревьях решений используется для выбора признаков, которые лучше всего разделяют данные.
- Обработка естественного языка (NLP): Оценка неопределенности в языковых моделях.
- Термодинамика: Аналогична термодинамической энтропии, отражает меру беспорядка в системе.

Также есть такие понятия, как взаимная энтропия и условная энтропия. Их определения появятся в конспекте после того, как пройдет неделя со сдачей домашки.

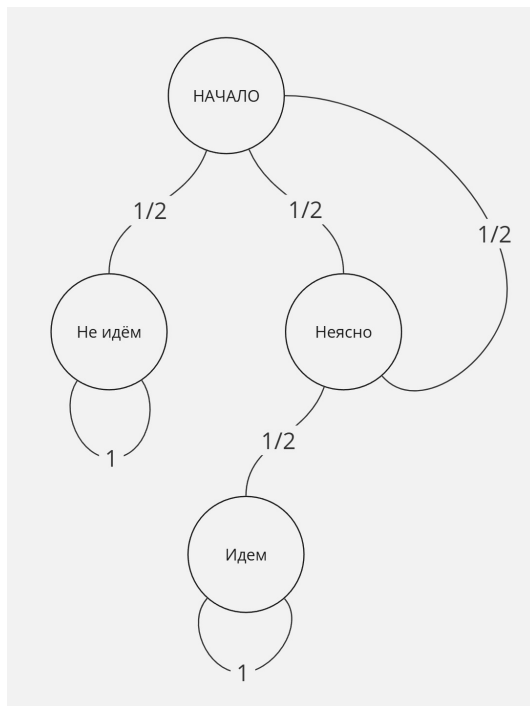
5 Лекция 5.

5.1 Марковские цепи

Вспоминаем задачку с Петей, монеткой и кинотеатром и визуализируем в виде ориентированного графа.



Заметим, что результаты нам не особо важны, важны только вероятности, перерисуем и дополнительно заведём ограничение на сумму значений вероятностей исходящих из вершины: сумма должна всегда должна быть равна единичке.



Давайте для такого графа введём определение.

Марковской цепью называется ориентированный граф с конечным числом вершин, у которых на исходящих рёбрах написаны вероятности, причём для каждой вершины верно, что сумма этих рёбер равна единице. Введём ещё парочку определений.

Вершина называется **поглощающей** если из неё исходит только петля.

Обозначим векторами $b_i^{(t)} = P(\xi_t = i)$, $b^t = (b_1^{(t)}, \dots, b_n^{(t)})$ - распределение ξ_t .

Давайте с такими данными построим матричку $P_{n \times n}$, где значение в ячейке – это вероятность перейти из i -той вершины в j -тую. Разберём пример:

$$P = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Хочется что-то с этой матрицей начать делать, начнём же:

Лемма 1. $b^{(t+1)} = b^{(t)} P$

Доказательство:

$$b_i^{(t+1)} = P(\xi_{t+1} = i) = \sum_{j=1}^n P(\xi_{t+1} = i \ \& \ \xi_t = j) P(\xi_t = j) = \sum_{j=1}^n b_j^{(t)} P_{ji} = (b^{(t)} P)_i$$

Q.E.D.

По коэффициенту очевидности $b^{(t)} = b^{(0)} P^t$, проверяйте сами

МЦ называется **поглощающей**, если $\forall u \exists$ путь такой, что $u \rightarrow$ поглощаемая вершина

Введём отношение эквивалентности такое, что $u \sim v$ если \exists пусть $u \rightarrow v$ и наоборот.

Можем заметить, что у нас образуется разбиение на классы эквивалентности, рассмотрим $[1; n] / \sim$, \tilde{u} – класс эквивалентности

\tilde{u} – **эргодический класс**. тем самым мы можем разделять задачи на две части: на наблюдение свойств в самом классе и между классами.

МЦ – **эргодическая**, если в ней ровно 1 эргодический класс

Эргодический класс называется **периодическим**, если $\exists d > 1$, что размер любого цикла делится на d – сама d называется **периодом**, очевидно, что период – нод длин всех циклов

Теорема о классификации Марковских Цепей.

В любой МЦ есть поглощающий эргодический класс, вероятность того, что МЦ окажется в одном таких классов стремится к единице

Вторая часть, я не могу разобрать свой почерк, на паре мы это не доказывали пока

Доказательство:

$$P = \left(\begin{array}{c|c} Q & R \\ \hline \mathbf{0} & E \end{array} \right), \text{ где } Q_{m \times m}, R_{m \times (n-m)}, \text{ где } m - \text{это непоглощающие эргодические классы}$$

тогда каждый вектор $b^{(t)}$ делится на две части: первые m координат обозовём $a^{(t)}$, вторые назовём $c^{(t)}$. $a_i^{(t)} = P(\xi_t = i)$, где i непоглощающая

Лемма 2. $a^{(t)} = a^{(0)}Q^t$, лемма очевидна просто взгляните, на то, как умножается вектор на матрицу

Лемма 3. $Q^k \xrightarrow{k \rightarrow +\infty} 0$

Доказательство:

Для того, чтобы найти предел матрицы нам нужно ввести норму, норму введём такую $|A| = \max_{ij} |a_{ij}|$, сами докажете, что это норма, не маленькие.

Тогда матрица стремится к нулю, если её норма стремится к нулю, наша задача показать, что $Q^n \xrightarrow{n \rightarrow +\infty} 0$

Пусть L - максимальная длина кратчайшего пути от i до поглощения, давайте посмотрим чему равно $X = Q^L$, рассмотрим $x_{ij} = \sum_{k_1, k_2, \dots, k_{l-1}} q_{ik_1} q_{k_1, k_2} q_{k_2, k_3} \dots q_{k_{l-1}, j}$. Чему равна эта сумма, неясно, каждое из этих произведений – это вероятность пройти по пути длиной ровно L от вершины i до j , давайте рассмотрим такую сумму

$$\sum_{j\text{-не погл}} x_{ij} = \sum_{k_1, k_2, \dots, k_{l-1}, j} q_{ik_1} q_{k_1, k_2} q_{k_2, k_3} \dots q_{k_{l-1}, j} = \delta < 1$$

Меньше единицы оно потому, что мы смотрим не на все пути, а лишь на часть. Посмотрим на матричку:

$$Q^n_{ij} = (Q^L Q^{n-L})_{ij} = \sum_k Q^L_{ik} Q^{n-L}_{kj} \leq (\sum_k Q^L_{ik}) |Q^{n-L}| \leq \delta |Q^{n-L}| \text{ т.к. } |Q^n| \leq \delta^{[n/L]}$$

то наше выражение стремится к нулю

Лемму доказали, стало быть, и теоремка доказана.

Передём к вычислительным моментам, мы хотим понять, где мы поглотимся. Найдём мат. ожидание времени до поглощения. Давайте введём T – случайная величина : число шагов до поглощения, тогда $T = \sum_{i=1}^m T_i$ – число посещений i -го состояния

$$T_i = \sum_{j=0}^{\infty} T_{ij}$$

$$T_{ij} = \begin{cases} 1, & \text{если на шаге } j \text{ марковская цепь находится в состоянии } i, \\ 0, & \text{иначе.} \end{cases}$$

$$P(\text{м.ц на шаге } j \text{ в состоянии } i) = (a^0 Q^j)_i$$

$$ET = \sum_{i=1}^n ET_i = \sum_{i=1}^m \sum_{j=0}^{\infty} ET_{ij} = \sum_{i=1}^m \sum_{j=0}^{\infty} (a^0 Q^j)_i$$

Воспользуемся линейной алгеброй, т.к. сумма матриц перестановочна относительно взятия i -го компонента

$$\sum_{i=1}^m \sum_{j=0}^{\infty} (a^0 Q^j)_i = \sum_{i=1}^m \left(\sum_{j=0}^{\infty} a^0 Q^j \right)_i$$

т.к. a^0 не зависит от j

$$\sum_{i=1}^m \left(\sum_{j=0}^{\infty} a^0 Q^j \right)_i = \sum_{i=1}^m a^0 \left(\sum_{j=0}^{\infty} Q^j \right)_i$$

Q.E.D.

Лемма $\sum_{j=0}^{\infty} Q^j = (I - Q)^{-1}$

Доказательство:

$$\begin{aligned} (E - Q)(E + Q + Q^2 + Q^3 + \dots) &= E - Q + Q - Q^2 + Q^2 + \dots - Q^{n+1} \\ &= E - Q^{n+1} = E \text{ т.к. } Q \text{ стремится к нулю} \end{aligned}$$

Q.E.D.

Фундаментальная матрица поглощения м.ц. – это $N = (I - Q)^{-1}$

$$\begin{aligned} P(\text{погл в } j) &= \sum_i^m P(\text{погл в } j \text{ из } i) P(j \text{ перед поглощением}) = \sum_t^{\infty} \sum_{i=1}^m P(\text{погл в } j \text{ из } i \text{ в } t) \\ &= P(\text{быть в } i \text{ в момент времени } t) = a^0 N R \end{aligned}$$

Откуда получаем 2 крутые и очень полезные в дальнейшем формулы:

6 Лекция 6.

Поглощающий эргодический класс – класс, из которого в конденсации нет рёбер

МЦ регулярная если $\forall i, j \ P_{i,j} > 0$

Теорема: Эргодическая теорема для марковских цепей

$$\exists b \ \forall b^{(0)} P^{(n)} \xrightarrow{n \rightarrow \infty} b \text{ и } bP = b$$

Доказательство:

Давайте возьмём наш вектор $b^{(0)}$ и умножим на матрицу A такую, что $A \in M_{n \times n}$ и $\forall j \ a_{ji} = \tilde{a}$

$$(b \cdot A)_i = \sum_{j=1}^n b_j a_{ji} = (\sum_{j=1}^n b^{(0)}_j) \tilde{a}_i = \tilde{a}_i. \text{ т.к. сумма в столбике } b^{(0)} \text{ всегда равна единичке}$$

Докажем, что $P^n \xrightarrow{n \rightarrow \infty} A$, удовлетворяющей условиям.

Введем $M_i^{(t)} = \max(P^t)_{ji}$, $m_i^{(t)} = \min(P^t)_{ji}$, посмотрим на разность максимума и минимума в столбце, правда ли он стремится к нулю?

Возьмём $\delta = \max P_{ij}$ и $\tilde{\delta} = \min P_{ij}$

$$\begin{aligned} (P^{(t+1)})_{ij} &= (P \cdot P^t)_{ij} = \sum_{k=1}^n P_{jk} (P^t)_{ki} \leq \sum_{k \neq \text{pos}(\min)} P_{jk} \cdot M_i^{(t)} + P_{j, \text{pos}(\min)} m_i^{(t)} = \\ &= \sum_{k=1}^n P_{jk} \cdot M_i^{(t)} + P_{j, \text{pos}(\min)} (m_i^{(t)} - M_i^{(t)}) \leq M_i^{(t)} - \delta A_i^{(t)} \end{aligned}$$

с другой стороны

$$(P^{(t+1)})_{ij} = (P \cdot P^t)_{ij} = \sum_{k=1}^n P_{jk} (P^t)_{ki} \geq \sum_{k=1}^n P_{jk} m_i^{(t)} + P_{j, \text{pos}(\max)} (M_i^{(t)} - m_i^{(t)}) = m_i^{(t)} + \tilde{\delta} A_i^{(t)}$$

$$\begin{aligned} M_i^{(t+1)} &\leq M_i^{(t)} - \delta A_i^{(t)} \\ m_i^{(t+1)} &\geq m_i^{(t)} + \tilde{\delta} A_i^{(t)} \end{aligned}$$

минусуем

$$A_i^{(t+1)} \leq A_i^{(t)} - (\delta - \tilde{\delta} A_i^{(t)}) = (1 - \delta - \tilde{\delta}) A_i^{(t)} \leq \dots \leq (1 - \delta - \tilde{\delta})^{(t+1)} \rightarrow 0$$

Если разность стремится к 0, то $b^{(0)} P^{(n)} \xrightarrow{n \rightarrow \infty} b^0 A = b$

Q.E.D.

Давайте научимся искать b , решим следующее уравнение: $(I - P)b = \mathbf{0}$. Казалось бы, это СЛОУ и она может иметь либо 0 решений, либо бесконечно много, тогда почему же у нас есть конкретное решение? На самом деле, $(I - P)$ – вырожденная, и она имеет ранг $n - 1$ и у нас одномерное пространство решений, но нам подойдёт только тот вектор, у которого сумма координат единичка. Тогда как в итоге найти b ?

У нас есть два способа, мы либо возведём много раз матрицу P в квадрат, с какого-то момента это всё будет стремиться к A , мы найдём ответ и, тогда сможем вычислить b , все операции займут у нас $O(n^3c)$, понятно, что если использовать какие-то продвинутые алгоритмы, оценку можно улучшить.

Либо же просто решим систему уравнений, но это тоже имеет свои тонкости, как минимум нам понадобится работать с вещественными числами, а мы знаем, что вещественные числа в компьютере несовершенны и дорого с ними работать, хоть и асимптотика номинально $O(n^3)$, но она упирается в проблемы работы с даблами.

Наши теоремы мы доказали только для регулярных МЦ, но что происходит с нерегулярными?

Теорема

Если P – матрица нерегулярного эргодического класса, тогда $\exists t : P^t$ - регулярное Доказывать мы это не будем, просто поверим на слово

7 Лекция 7.

7.1 Формальные языки

Алфавит - Σ - конечное или пустое

Множество слов длины n : Σ^n

$$\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$$

Язык - подмножество слов в алфавите $L \subset \Sigma^*$

Замечание. Множество языков 2^{Σ^*} несчетно.

7.2 Описание языка.

Вот мы хотим описать язык. И описать его словами, например: все слова четной длины, они содержат четное число единиц. Но мы не можем дать эти слова компьютеру или человеку, говорящему на другом языке.

Поэтому существует несколько способов описания языков:

1. Перечисление: Явное перечисление всех слов в языке (возможно только для конечных языков). Например, $L = \{a, ab, aba\}$.
2. Порождающая грамматика (формальная грамматика): Набор правил, определяющих, какие слова принадлежат языку.
3. Распознающий автомат: Машина (например, конечный автомат), которая принимает или отклоняет слова в зависимости от того, принадлежат они языку или нет.
4. Регулярное выражение: Шаблон, описывающий структуру слов в языке.
5. Предикат: Математическое условие, которому должны удовлетворять слова, чтобы принадлежать языку.

Замечание. Всего описаний счетное множество

Фанфакт. Если вы захотите найти лексикографически минимальный язык без описания, то когда вы найдете его, у него появится описание. Мы не будем думать об этом и такого рода парадоксах.

7.3 Регулярные (автоматные языки)

У нас есть операция конкатенации, определена она таким образом:

$$\alpha, \beta \in \Sigma^* : \varphi = \alpha\beta, \text{ причем } \alpha \in \Sigma^k, \beta \in \Sigma^l, \alpha\beta \in \Sigma^{k+l}, \text{ а также } \varphi_i = \begin{cases} i \leq k \Rightarrow \alpha_i \\ i > k \Rightarrow \beta_{i-k} \end{cases}$$

Она ассоциативна, и есть нейтральный элемент ε .

Но теперь хотим ввести понятие конкатенации языков:

$$AB = \{x \mid x = yz, y \in A, z \in B\}$$

Пример:

$A = \{0, 01\}, B = \{0, 10\}$, тогда $AB = \{00, 010, 0110\}$

А теперь обо всех операциях

Пусть L_1 и L_2 - языки над алфавитом Σ . Тогда можно определить следующие базовые операции:

1. Объединение: $L_1 \cup L_2 = \{w \mid w \in L_1 \mid\mid w \in L_2\}$
2. Конкатенация: $L_1 L_2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$
3. Степень: $L_1^0 = \{\varepsilon\}, L_1^{n+1} = L_1 L_1^n$
4. Замыкание Клини: $A^* = \bigcup_{k=0}^{\infty} A^k$

$Reg_0 = \{\emptyset, \{\varepsilon\}, \{c\} \text{ for } c \text{ in } \Sigma\}$ - базовые регулярные языки

$Reg_{i+1} = Reg_i \cup \{A \cup B, AB, A^* \mid A, B \in Reg_i\}$

Семейство регулярных языков: $Reg = \bigcup_{k=0}^{\infty} Reg_k$

Лемма:

$A, B \in Reg \Rightarrow A \cup B \in Reg, AB \in Reg, A^* \in Reg.$

Доказательство:

$A \in Reg_i, B \in Reg_j$, откуда по определению $A \cup B, AB, A^* \in Reg_{\max(i,j)+1}$.

Q.E.D.

$X \in Good, X \subset 2^{\Sigma^*}, X = \text{set}\langle \text{language} \rangle, Good = \text{set}\langle \text{set}\langle \text{language} \rangle \rangle$

Good - множество множеств языков. Мы называем его хорошим если:

1. $Reg_0 \subset X$
2. X замкнут относительно базовых операций.

Теорема. $Reg = \bigcap_{u \in Good} u$

Доказательства не будет.

\emptyset - пустое множество, ε - пустая строка.

Пусть у языка A описание α . У языка B описание β . Тогда введем обозначение:

1. AB , описание будет $\alpha\beta$ — средний приоритет
2. $A \cup B$, описание будет $\alpha \mid \beta$ — минимальный приоритет
3. A^* , описание будет α^* — максимальный приоритет

В зависимости от приоритета будем заключать выражения в скобки. Такие записи называются академическими регулярными выражениями.

Также есть еще две обозначения:

1. $\alpha^+ = \alpha\alpha^*$.

2. $\alpha^k = \alpha \dots \alpha$ - k раз.

7.4 Детерминированный Конечный Автомат (ДКА)

В разделе марковских цепей у нас были графы с вероятностями на ребрах. А давайте теперь удалим вероятности.

$A = \langle \Sigma, Q, S, T, \delta : Q \times \Sigma \rightarrow Q \rangle$, где

1. Σ - алфавит
2. Q - конечное множество состояний
3. $S \in Q$ - начальное состояние.
4. $T \subset Q$ - допускающее (конечное) состояние.
5. δ - функция перехода, которая по текущему состоянию переходит

$Snap = Q \times \Sigma^*$ — **множество мгновенных описаний** автомата или состояний. \vdash - операция перехода от одного описания к другому.

$\vdash: \langle q, \alpha \rangle \vdash \langle r, \beta \rangle$

1. $\alpha = c\beta, c \in \Sigma$
2. $r = \delta(q, c)$

Это описание того, как мы бегаем по графу. То есть например пусть у нас есть автомат, который описывает четное или нечетное число единиц в строке, и ему подали 0101. Тогда для него мгновенные описания:

$$\langle even, 0101 \rangle \vdash \langle even, 101 \rangle \vdash \langle odd, 01 \rangle \vdash \langle odd, 1 \rangle \vdash \langle even, \varepsilon \rangle$$

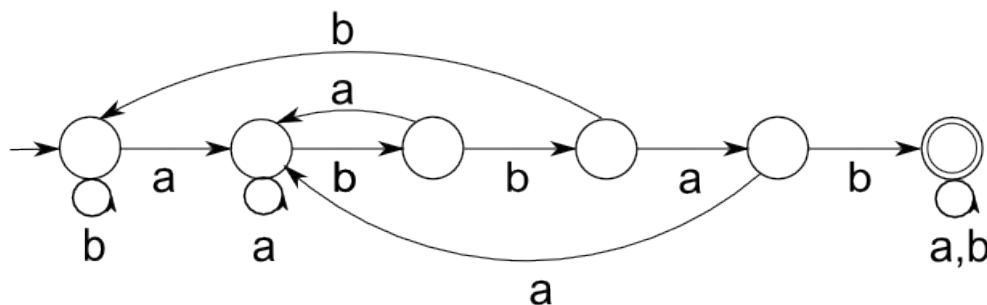
Язык автомата $L(A) = \{w \mid \langle s, w \rangle \vdash^* \langle t, \varepsilon \rangle, t \in T\}$

Теорема (Клини)

$Reg = Aut$, где Aut - множество языков, задаваемых ДКА: $Aut = \{X \mid \exists A : X = L(A)\}$, где A — ДКА.

Доказательства не будет

Пример ДКА:



Это автомат для поиска образца в тексте для строки "abbab".

8 Лекция 8.

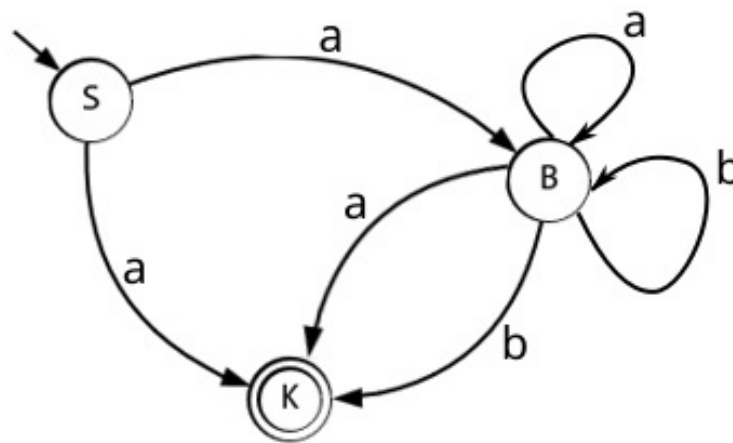
8.1 Недетерминированный конечный автомат (НКА).

Недетерминированность: Для каждого состояния и входного символа может быть несколько возможных переходов.

Ключевое отличие от ДКА: Функция перехода δ в НКА возвращает множество состояний, а не одно состояние. Это означает, что при чтении одного символа из текущего состояния, НКА может перейти в несколько состояний одновременно, либо не перейти ни в какое.

Мы можем считать, что S - множество стартовых позиций

Пример НКА:



Уйдем пока в другую сферу. Как хранить ДКА?

s будем хранить int-ом, t - $\text{vector}<\text{bool}>(n)$, δ $\text{vector}<\text{vector}<\text{int}> > (n,c)$.

И например операцию проверки в нем можно сделать вот так:

```

bool accept(x){
    cur = s
    for(i=0...len(x)-1)
        cur = delta[cur][x[i]]
    return t[cur]
}

```

Как хранить НКА?

s будем хранить int-ом(или set-ом), t - $\text{vector}<\text{bool}>(n)$, δ $\text{vector}<\text{vector}<\text{set}<\text{int}> > > (n,c)$.

И например операцию проверки в нем можно сделать вот так:

```

bool accept(x){
    can[0][s] = true
    for(i = 0... len(x)-1)
        for(q = 0... n-1)
            if can[i][q] \\можем идти дальше
                for r : delta[q][x[i]]
                    can[i+1][r]=true
}

```



```

    for(q = 0...n-1)
        if(can[len(x)][q] and t[q])
            return true
    return false
}

```

За сколько это работает это $\text{len}(x)n^2$.

НКА легче строить, но им сложнее оперировать и строиться. То есть в ДКА мы платим сложностью построения, чтобы легко потом проверять на содержание в автомате.

Теорема

Для языка L существует НКА $A_n \Leftrightarrow$ для L существует ДКА A_D

Доказательство:

Очевидно, что ДКА - частный случай НКА, поэтому нам надо доказывать только стрелку в правую сторону.

Алгоритм Томпсона.

Давайте в том алгоритме, который у нас был сверху сделаем функцию, которая будет брать i -ую строчку и будет возвращать $i + 1$:

```

vector<bool> next(vector<bool> a, char c){
    result = vector<bool> (n)
    for(q = 0... n-1)
        if a[q] \\можем идти дальше
            for r : delta[q][c]
                result[r] = true;
    return result;
}

```

И теперь заменим в исходном коде операции НКА на данную функцию:

```

bool accept(x){
    can[0][s] = true
    for(i = 0... len(x)-1)
        can[i+1] = next(can[i], x[i])
    for(q = 0...n-1)
        if(can[len(x)][q] and t[q])
            return true
    return false
}

```

Хмм, что-то похожее на то, что было в ДКА. ~~У вас кони на ста - 5 минусов, делай 3, до свидания~~

Пусть $Q_D = 2^{Q_N}$, Σ , $\delta_D = (A, c) = \{r | \exists q \in A, r \in \delta_N(a, c)\}$. $T_D = \{A | A \cap T_n \neq \emptyset\}$.

Получили автомат и детерминированный.

Q.E.D.

8.2 ε - НКА

ε -НКА — это расширение обычного НКА, позволяющее автомату переходить между состояниями, не потребляя никаких входных символов (так называемых ε переходы). Это добавляет еще больше гибкости в процесс распознавания языка.

ε - замыкание:

1. Сделаем транзитивное замыкание по графу ε - переходов, добавим новые ребра как ε - переходы.

То есть если x нужно было идти 2 раза подряд по ε ребру, то теперь всего 1 ε переход

2. Если есть вершины p, q , q - терминальная, то сделаем p терминальной.

Получается, что любое слово, задающееся этим автоматом, такое, что последний переход в терминальное не ε

3. Рассмотрим вершины p, q, r . Пусть из p в q есть ε переход, а из q в r переход по какому-то символу c . Добавим ребро из p в r по c .

Получается, что любое слово можно допустить без ε - переходов.

4. удалим ε -переходы и получили обычное НКА.

Мы показали, что любой язык задается через ДКА, НКА, ε -НКА, причем из любого автомата я могу получить любой другой автомат с помощью ε -замыкания или алгоритма Томпсона.

Теперь докажем теорему Клини, ту самую с прошлой лекции.

Теорема (Клини)

$$Reg = Aut$$

Доказательство:

1. $Reg \subset Auto$

Хочу доказать, что $\forall i : Reg_i \subset Aut$. Будем считать, что у нас ε -НКА автоматы с одним входом и выходом

База: $i = 0$ Очевидно.

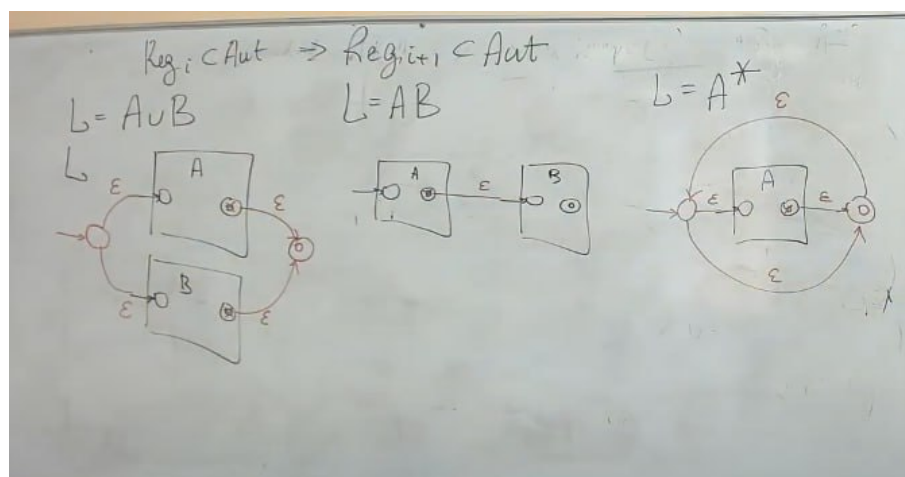
Индукционный переход:

Пусть $Reg_i \subset Aut \Rightarrow Reg_{i+1} \subset$.

Рассмотрим случаи как получился тот или иной язык:

- (a) $L = A \cup B$: Для A, B мы уже умеем строить автоматы. Давайте сделаем новую стартовую вершину, которая будет вести в стартовые A, B , с помощью ε -переходов, а так же соединим их терминальные и сделаем еще одну новую вершину терминальной
- (b) $L = AB$: Для A, B мы уже умеем строить автоматы. Соединю ε -переходом конец A и старт B . Старт A сделаю стартом нового автомата, а терминальную вершину B соответственно сделаю терминальной вершиной нового автомата
- (c) $L = A^*$. Для A мы уже умеем строить автомат. Добавлю стартовое состояние, которое ведет(ε -переходом) в стартовое состояние A . Добавлю новую вершину, в которую я поведу конец A (ε -переходом), и соединю вершины туда обратно.

Все это крайне подробно изображено на рисунке:



Индукционный переход доказан и первая часть доказательства тоже!

9 Лекция 9.

9.1 Теорема Клини.

Продолжаем доказывать, но уже в правую сторону

2. $Aut \subset Reg$

$L \in Aut : A$ - ДКА для L .

$Q = \{1, \dots, n\}$

ξ_{ijk} - слова, такие, что i, j и переход по вершинам номера которых меньше k .

Построим n^3 таких регулярных выражений таким образом:

$$\xi_{ij(k+1)} = \xi_{ijk} | \xi_{i(k+1)k} (\xi_{(k+1)(k+1)k})^* \xi_{(k+1)jk}$$

(Случай $k = 0$ тривиален, используем его как за базу)

Откуда наше регулярное: $\varphi = \xi_{st_1n} | \xi_{st_2n} | \dots$

Q.E.D.

9.2 Нерегулярность

Язык правильных скобочных последовательностей не является регулярным языком.

Доказательство:

Пусть ПСП $\in Reg$, A - ДКА для ПСП. Пусть n число сост. A .

Скормим ему $(, ((, (((, ((((, \dots$ до их размера $n+1$. По принципу Дирихле у нас будут 2 конечных состояния у этих строк, которые совпадут. Пусть $q_i = q_j = u$. Тогда посмотрим на строки $x = (i)^i$ и $y = (j)^i$. Заметим, что они придут в конечное одно состояние. Причем x —ПСП, y —нет. Противоречие.

Q.E.D.

Нерегулярный язык - это формальный язык, который не может быть описан ни одним из вышеупомянутых способов (регулярным выражением, конечным автоматом или регулярной грамматикой).

Примеры:

$$L = \{a^n b^n | n \geq 0\}$$

$$L = \{w w^R | w \in \{a, b\}^*\} - \text{язык палиндромов}$$

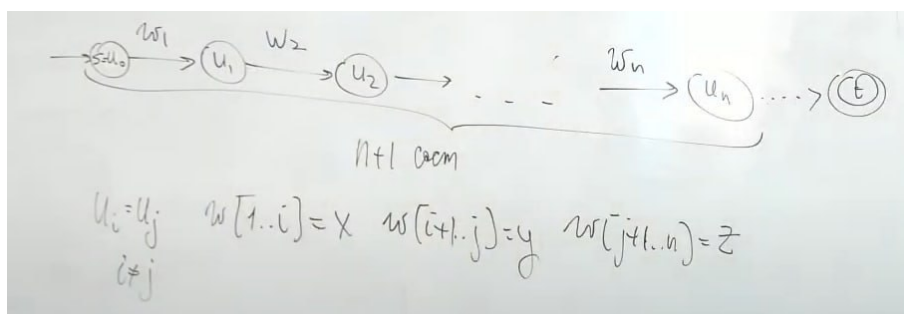
Теорема (Лемма о разрастании/накачке)

L - регулярный язык. Тогда:

$$\exists n > 0 : \forall w : w \in L, |w| \geq n, \exists x, y, z : w = xyz, \text{ такие, что } y \neq \varepsilon, |xy| \leq n, \forall k \geq 0, xy^k z \in L$$

Доказательство:

Пусть L - регулярный язык, тогда существует ДКА для L . Пусть n - число состояний автомата A . $\forall w \in L, |w| \geq n$.



Возьму первые $n + 1$ состояний. По принципу Дирихле будут совпадающие (мы нашли цикл). Возьмем за y - данный цикл, x - всего до него z - все после. Откуда верность утверждения тривиальна.

Следствие. Не выполнено данное утверждение \Rightarrow не регулярный.

У любого языка есть минимальный автомат (линейная алгебра передает привет).

Алгоритм минимизации автомата.

Д.К.А. u и v назовем **различимыми** строкой s : если из u можно перейти в какой-то x , а из v можно перейти в y с помощью слова s , причем либо x , либо y терминальная.

Назовем $a \sim b$, если a и b не различимы никакой строкой. Очевидно \sim это отношение эквивалентности, откуда можно разбить на классы эквивалентности.

Лемма: $u \sim v \Rightarrow \delta(u, c) \sim \delta(v, c)$. Очевидно.

Алгоритм: Обозначим $D_k = \{(u, v) | u, w \text{ различимы } s, |s| \leq k\}$

$$D_0 = \{(u, v) | u \in T \oplus v \in T\}$$

$$D_k = \{(u, v) | (u, v) \in D_{k-1} | \exists c \in \Sigma : (\delta(u, c), \delta(v, c)) \in D_{k-1}\}$$

Тогда у нас вот примерно такой алгоритм поиска эквивалентных:

```

\\Положим D0 в Q, D0 в D, Q - очередь
while not Q.empty():
    (u,v) = Q.pop()
    for c in Sigma:
        for a in In[u][c]
            for b in In[v][c]:
                if(a,b) not in D:
                    q.push(a,b)
                    D.add(a,b)

```

То есть мы умеем искать пары эквивалентных.

Теорема. A - ДКА для L не содержит эквивалентных и любое состояние достижимо из s тогда:

1. A - минимальный.
2. A' - ДКА для L . $|Q'| = |Q|$, тогда A' изоморфно A .

Доказательство:

Пусть есть автомат с меньшим количеством вершин. Назовем его B . Посмотрим на автомат Объединения. Есть два стартовых состояния. Забьем на них пока что. Давайте на получившемся автомате запустим автомат из теоремы. Что мы знаем? Стартовые состояния, очевидно, оказались эквивалентными. Возьмем какую-нибудь вершину из A (назовем ее u) и придем в нее через стартовую, повторяя эти действия во втором автомате, то по лемме они придут в эквивалентные состояния. Так как у нас достижимо из стартовой каждое состояние в A , то у меня будет n пар эквивалентностей A и B , откуда, тк справа вершин $< n$ или по-другому $\leq n - 1$, то по принципу Дирихле у какие-то 2 состояния из A эквивалентны одному из B , а так как это эквивалентность, то они сами эквивалентны друг другу, что противоречит условию. Откуда, B не может быть меньшего размера. Если B такого же размера, как и A , то мы показали, что у нас образуются пары эквивалентных.

Q.E.D.

10 Лекция 10.

План на лекцию:

1. Система линейных уравнений в Регулярных выражениях.
2. Быстрый алгоритм минимизации.
3. Анализ свойств языков.

10.1 Система линейных уравнений в регулярных выражениях.

Пусть α и β - регулярные выражения

$$L = \alpha L + \beta$$

Лемма: $\alpha^*\beta$ - решение. Очевидно

Теорема.

$\varepsilon \notin \alpha \Rightarrow \alpha^*\beta$ — единственное решение.

$\varepsilon \in \alpha \Rightarrow L = \alpha^*\beta \cup N$ - решение, для $\forall N \subset \Sigma^*$.

Доказательство:

1. $\varepsilon \notin \alpha$:

База индукции: $|x| = 0, x \in L \Rightarrow x \in \alpha L + \beta \Rightarrow x \in \beta \Rightarrow x \in \alpha^*\beta$.

ИП. Пусть для всех $|y| < |x| \Rightarrow y \in \alpha^*\beta$. $x \in L \Rightarrow x \in \alpha L + \beta$:

(a) $x \in \beta \Rightarrow x \in \alpha^*\beta$

(b) $x \in \alpha L \Rightarrow x = yz, y \in \alpha, z \in L, |z| < |x|$, откуда по индукции $x \in \alpha^*\beta$

2. $\varepsilon \in \alpha, \beta \subset L \Rightarrow \alpha\beta \subset L \Rightarrow \dots \Rightarrow \alpha^*\beta \subset L$.

$$\forall N : N \subset \alpha N$$

Q.E.D.

Теперь решим систему уравнений. Пусть X_1, X_2, \dots, X_n - неизвестные языки:

$$\begin{cases} X_1 = \alpha_{11}X_1 + \alpha_{12}X_2 + \dots + \alpha_{1n}X_n + \beta_1 \\ X_2 = \alpha_{21}X_1 + \alpha_{22}X_2 + \dots + \alpha_{2n}X_n + \beta_2 \\ \vdots \\ X_n = \alpha_{n1}X_1 + \alpha_{n2}X_2 + \dots + \alpha_{nn}X_n + \beta_n \end{cases}$$

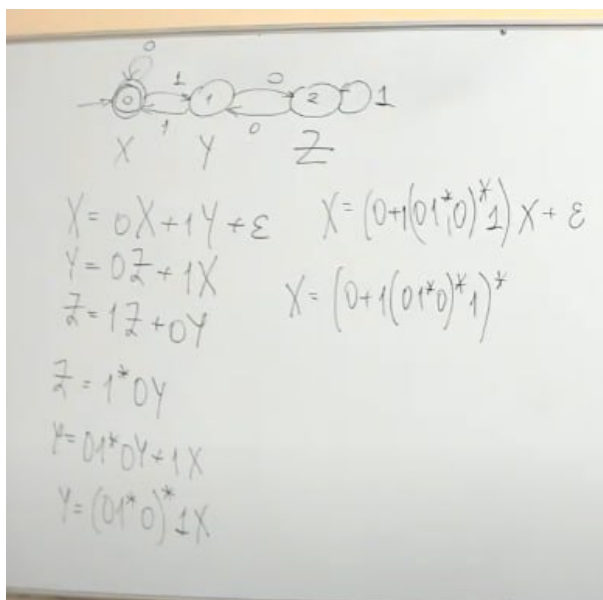
Мы тут не можем перенести в левую часть, потому что у нас нет вычитания языков. $\varepsilon \notin \alpha_{ij}$.

Берем первое уравнение. Возьмем теорему, у нас единственное решение

$$X_1 = \alpha_{11}^*(\alpha_{12}X_2 + \alpha_{13}X_3 + \dots + \alpha_{1n}X_n + \beta_1)$$

Подставим во второе. Повторим аналогично, как с A_1 с X_1 и так далее.

Благодаря прошлому мы можем вот таким образом искать регулярное по автомату.



10.2 Алгоритм анализа регулярных языков.

Теорема:

Пусть A, B - регулярные языки. $\varphi : \Sigma^* \rightarrow P^*$ - линейная.

$A \cup B, \varphi(A), \varphi^{-1}(A) = \{x | \varphi(x) \in A\}$ - регулярные языки.

Доказательство:

Докажем $A \cup B$:

Введем произведение автоматов $A \times B$:

$Q_A \times Q_B : \delta(\langle u_A, u_B \rangle) = \langle \delta_A(u_A, c), \delta_B(u_B, c) \rangle$. $s = \langle s_A, s_B \rangle$, $T = \{ \langle t_A, t_B \rangle | t_A \in T_A, t_B \in T_B \}$.

Заметим, что такое произведение задает объединение автоматов.

Докажем $\varphi(A)$:

Возьмем автомат соответствующий A , заменим все ребра на $\varphi(c)$ (тк по условию нам могут выдавать слова, то возможно на ребре будет записано слово, если так, то построим посимвольную цепочку). Откуда есть автомат \Rightarrow регулярный.

Докажем $\varphi^{-1}(A)$

Возьму автомат для A . Посмотрю пути из u, v , которые по $\varphi(c)$ и на новом автомате между u, v проведу ребро с c . Откуда есть соотв. автомат \Rightarrow регулярный.

10.3 Алгоритм минимизации.

Алгоритм Хопкрофта.

Алгоритм основан на итеративном разбиении множества состояний на группы эквивалентности.

Начальное разбиение: Все состояния разбиваются на две группы: принимающие состояния (те, которые являются конечными) и непринимаящие состояния.

На каждой итерации алгоритм пытается разбить существующие группы эквивалентности на более мелкие, если обнаруживает, что состояния внутри группы ведут себя по-разному для некоторого входного символа.

Назовем A, B - согласованны, если из B все ребра ведут либо в A , либо не в A .

Цель: Добиться, что все пары согласованны.

Грамматика называется линейной, если любое правило имеет вид $A \rightarrow \xi B \delta$, или $A \rightarrow \xi$. (не более одного B в любом правиле). А грамматика называется праволинейной, если любое правило имеет вид $A \rightarrow \alpha B$

Теорема:

A - регулярный равносильно тому, что A задается праволинейной грамматикой.

Доказательство:

Тривиально.

Следствие: $Reg \subset CF$, $Reg \neq CF$

Иерархия Хомского.

Формальная грамматика 0-ого класса - грамматика, но правила задаются из N^+ , а не из N .

12 Лекция 12.

12.1 Приведение грамматики к НФХ.

Грамматика в нормальной форме Хомского, если любое правило имеет вид:

$$A \rightarrow BC \quad A \rightarrow c$$

А также, возможно правило из $S \rightarrow \varepsilon$, но тогда S не встречается в правых частях правил.

Давайте научимся приводить грамматику в нормальный вид Хомского:

1. Для всех терминалов заведем их личные нетерминалы и добавим правило $N_c \rightarrow c$, а также заменим во всех правилах c на N_c . У нас остались такие правила с нетерминалами:

- (a) $A \rightarrow \varepsilon$, 0 нетерминалов. ε -правила
- (b) $A \rightarrow B$, 1 нетерминал. Цепные правила.
- (c) $A \rightarrow BC$, 2 нетерминала.
- (d) $A \rightarrow N_1 \dots N_k, k \geq 3$. Длинные правила.

Осталось победить 3 вида правил.

2. Победим правила. Начнем с длинных. Возьмем правило $A \rightarrow N_1 \dots N_k, k \geq 3$.

Заменим его на правила: $A \rightarrow N_1 Y_1, Y_1 \rightarrow N_2 Y_2, Y_{k-2} \rightarrow N_{k-1} N_k$. Победили!

3. Удалим ε правила. Найдем все эpsilon порождающие терминалы (из которых можно вывести ε). Как такие искать?

Перебираем все правила, если из правила мы получаем в правой стороне все эpsilon порождающие нетерминалы, то отметим текущее правило эpsilon порождающим. Будем продолжать так, пока изменения не прекратятся.

Выкинем все эpsilon. S' - новый стартовый символ. Если S - эpsilon порождающий, то $S' \rightarrow S, S' \rightarrow \varepsilon$. И выкинем эpsilon правила.

Победили второго врага!

4. Остались цепные правила. Сделаем транзитивное замыкание по ним. Оставим только нецепные правила. Все работает.

В чем прикол НФХ? Дерево разбора бинарное!!!

12.2 Алгоритм Коко-Янгера-Касами (КЯК или СУК)

Алгоритм тунной динамики

Алгоритм Кока-Янгера-Касами — алгоритм, позволяющий по слову узнать, выводимо ли оно в заданной КС-грамматике в нормальной форме Хомского. Любую КС-грамматику можно привести к НФХ, как мы показали ранее, поэтому алгоритм является универсальным для любой КС-грамматики.

Пусть у меня есть слово x длины n . Сделаем динамику:

$d_A[l][r]$ - можно ли из A вывести подстроку $x[l \dots r - 1]$

Тогда для $r - l > 1$, будет выполнено:

$$d_A[l][r] = \bigvee_{A \rightarrow BC} \bigvee_{k=l+1}^{r-l} (d_B[l][k] \wedge d_C[k][r])$$

где \bigvee это логическая дизъюнкция.

Теперь порядок вычисления. Вычисляем по возрастанию длины. Ответ лежит в $d_S[0][n]$. Первый слой задаем руками.

12.3 МП автомат

Автомат с магазинной памятью - это конечный автомат, который использует стек для хранения состояний.

Есть автомат и стек. z_0 - маркер дна.

13 Информация о курсе

Поток — у2024.

Группы М3138-М3142.

Преподаватель — Станкевич Андрей Сергеевич.

В данном семестре фокусируются 2 темы: Дискретная теория вероятности и представление слов (токенов) в компьютере.

