

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Макаренко Елена Николаевна

Должность: Ректор

Дата подписания: 29.07.2022 18:15:38

Уникальный программный ключ:

c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

ЛЕКЦИЯ 1

НЕЙРОННЫЕ СЕТИ. ОСНОВНЫЕ ПОЛОЖЕНИЯ

КРАТКАЯ ИСТОРИЯ РАЗВИТИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

- 1943 г. – формальный персептрон МакКаллока и Питтса
- 1949 г. – правило Хебба
- 1957 г. – Персептрон Розенблатта
- 1975 – Когнитрон Фукусимы
- 1982 г. – сеть Кохонена
- 1986 г. – алгоритм обратного распространения ошибки
- 1989 г. – Сверточная нейронная сеть

Фрэнк РОЗЕНБЛАТТ



фото - Википедия

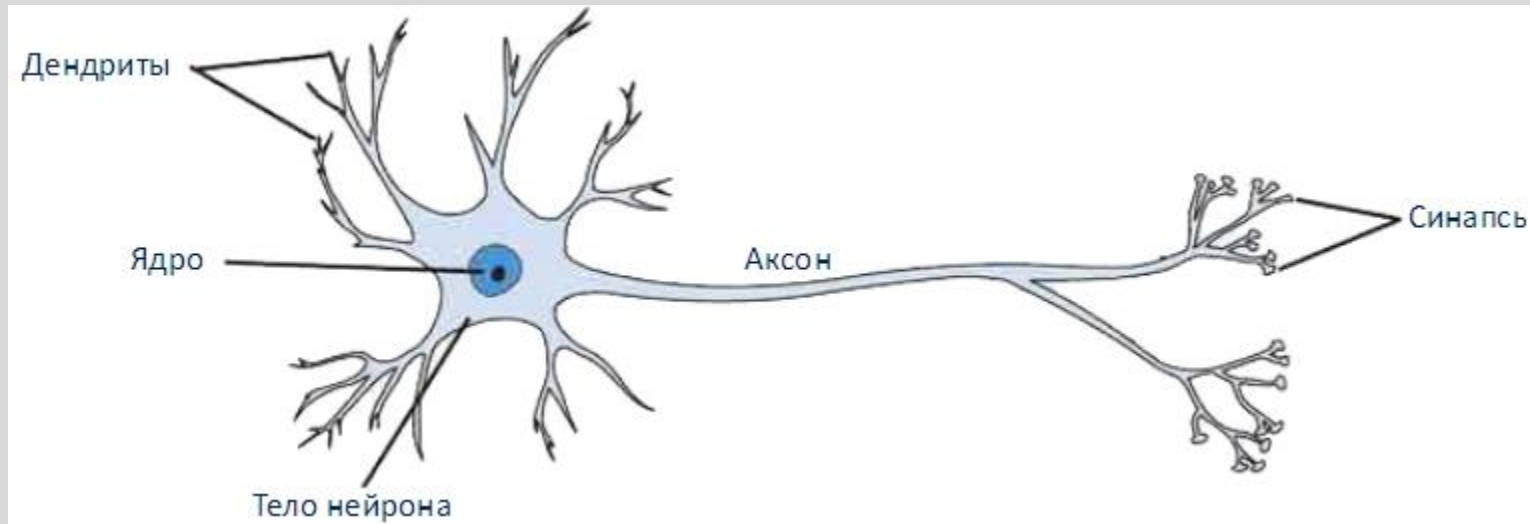
ОСНОВНЫЕ СВОЙСТВА НЕЙРОННЫХ СЕТЕЙ

- Нелинейность
- Адаптивность
- Отказоустойчивость
- Масштабируемость

Нейронные сети – это:

- гибкая модель для аппроксимации многомерных функций.
- средство прогнозирования во времени процессов, зависящих от большого количества переменных.
- средство распознавания образов
- инструмент для поиска по ассоциациям
- модель для поиска закономерностей в массивах данных

БИОЛОГИЧЕСКИЙ НЕЙРОН ЧЕЛОВЕЧЕСКОГО МОЗГА



Дендриты получают входные сигналы от других нейронов.

Тело нейрона суммирует входные сигналы, полученные от других нейронов, и в случае, если сумма превышает определенный порог, генерирует выходной сигнал.

Аксон служит для передачи выходного сигнала другим нейронам.

Синапсы соединяют аксон с дендритами других нейронов. Величина сигнала зависит от силы (синаптического веса) связи.

РАБОТА МОЗГА ЧЕЛОВЕКА

Структура мозга человека



Хорошая литература по функционированию мозга человека:
Хьюбел Д., Стивенс Ч., Кэндел Э. и др. Мозг. — М.: Мир, 1982. — 280 с.

Мозг — центральный отдел нервной системы животных, обычно расположенный в головном (переднем) отделе тела и представляющий собой компактное скопление нейронов и дендритов.

Отделы мозга:

-Продолговатый мозг: регулирует дыхание и кровообращение

-Мозжечок: отвечает за координацию движений, равновесие и мышечный тонус

-Варолиев мост: передает информацию из спинного мозга в отделы головного. Отвечает за глазные рефлексy, моторику кишечника и др.

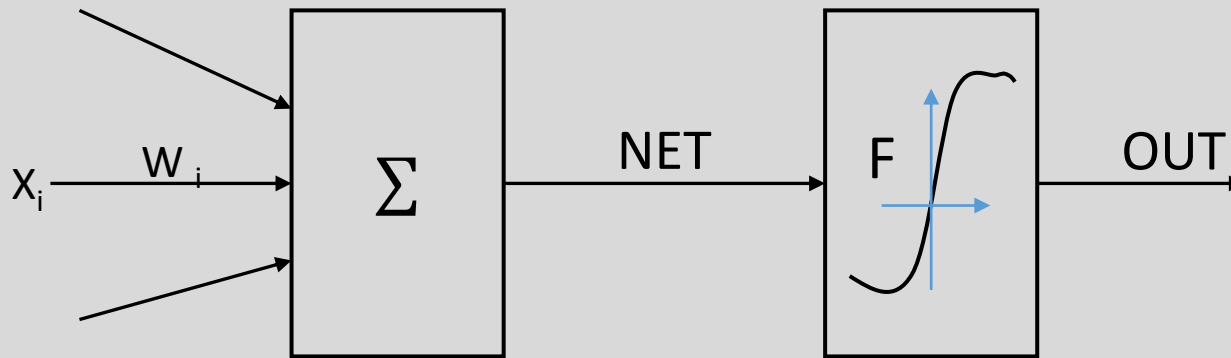
-Средний мозг: отвечает за зрение, слух, контроль движений, регуляция сна, концентрация внимания, температуру тела и многое др.

-Таламус: передача сенсорной и двигательной информации, процессы сна и бодрствования, концентрация внимания и др.

-Гипоталамус: регулирует выделение гормонов, регулирует ощущение голода и жажды и др.

-И т.д. (базальные ганглии, шишковидное тело, гипофиз и др.)

ФОРМАЛЬНЫЙ ПЕРСЕПТРОН МАККАЛОКА И ПИТТСА



$$NET = \sum W_i * X_i$$

$$OUT = F(NET - \Theta)$$

- 1- нейроны, выходные сигналы которых поступают на вход (x_i), w_i — веса входных сигналов
 - 2- сумматор входных сигналов, умноженных на их весовые коэффициенты;
 - 3- вычислитель передаточной функции (**функции активации**);
 - 4- нейроны, на входы которых подается выходной сигнал данного нейрона.
- Нейрон имеет один выход, часто называемый аксоном по аналогии с биологическим прототипом.
- С единственного выхода нейрона сигнал может поступать на произвольное число входов других нейронов.

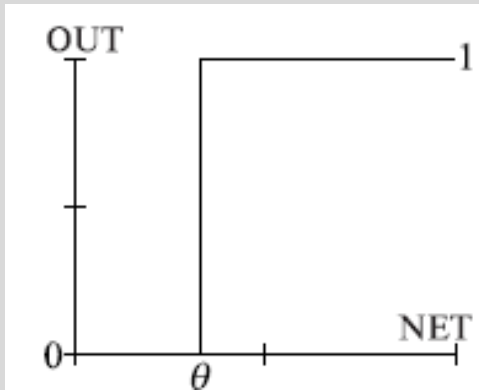
ОГРАНИЧЕНИЯ ФОРМАЛЬНОГО ПЕРСЕПТРОНА

Считается, что:

1. Вычисления нейрона происходят мгновенно, не внося задержку.
2. Нет четких алгоритмов для выбора функции активации.
3. Нет механизмов, регулирующих работу сети в целом, наподобие гормональной регуляции активности в нервных клетках.

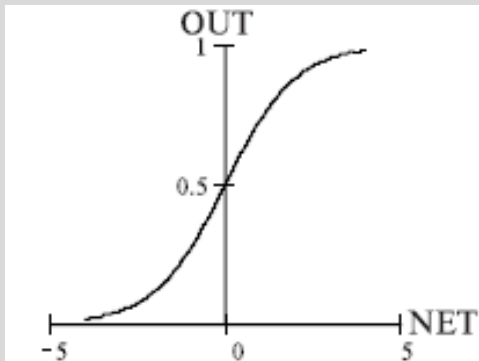
Модель формального нейрона не является биоподобной
и это скорее математическая абстракция.

ФУНКЦИИ АКТИВАЦИИ



Жесткая ступенька

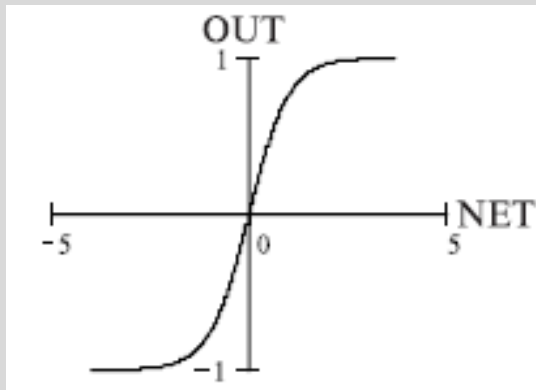
$$OUT = \begin{cases} 0, & NET < \theta \\ 1, & NET \geq \theta \end{cases}$$



Сигмоида (функция Ферми)

$$OUT = \frac{1}{1 + e^{-NET}}$$

ФУНКЦИИ АКТИВАЦИИ



Тангенсоида (гиперболический тангенс)

$$OUT = \text{th}(NET) = \frac{e^{NET} - e^{-NET}}{e^{NET} + e^{-NET}}$$

SOFTMAX функция

$$OUT = \frac{e^{NET}}{\sum_i e^{NET_i}}$$

Обобщение логистической функции для многомерного случая

ЗАДАЧА ОБУЧЕНИЯ С УЧИТЕЛЕМ

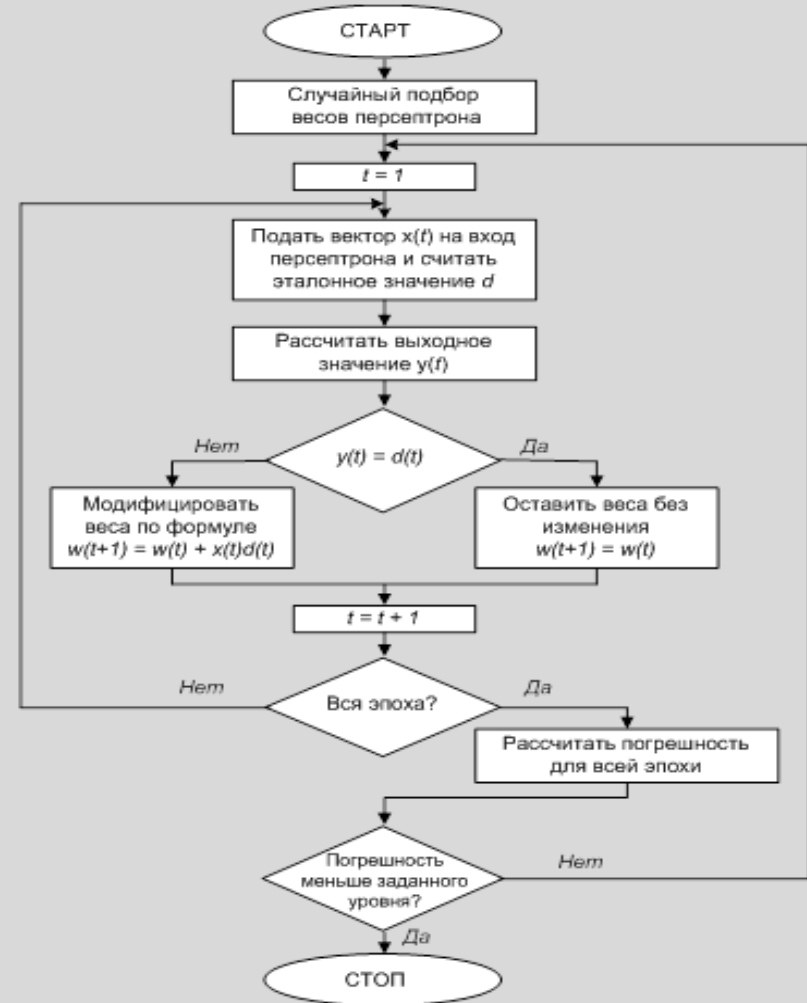
Известны входы X и выходы Y , задача натренировать сеть, чтобы на X_i выдавался Y_i

Ошибка на выходе сети

Ошибка на выходе

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2$$

Градиентный спуск $\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}$



ЗАДАЧА ОБУЧЕНИЯ БЕЗ УЧИТЕЛЯ

Существует широкий пласт задач, в которых ответы заранее не известны, например, в задачах многомерной классификации. Обучение без учителя – «самостоятельно», здесь можно провести аналогию с обучением ребенка – что-то он познает с помощью учителя, чему – то учится на собственном опыте. По сути существует лишь независимая от задачи мера качества работы сети.

алгоритм Хебба берет свое начало в книге «The Organization of Behavior» (Хебб, 1946 год), в которой приводится четкое определение физиологического правила обучения для синаптической модификации «эффективность переменного синапса между двумя нейронами повышается при многократной активации этих нейронов через данный синапс».

(unsupervised learning, неконтролируемое обучение)

Сигнальный метод обучения Хебба заключается в изменении весов по следующему правилу

$$w_{ij}(t) = w_{ij}(t-1) + \alpha * y_i^{n-1} * y_j^n \quad (\text{X.1})$$

y_i^{n-1} - Выходное значение нейрона i слоя $(n-1)$

y_j^n - Выходное значение нейрона j слоя n

Дифференциальный метод Хебба:

$$w_{ij}(t) = w_{ij}(t-1) + \alpha \cdot [y_i^{(n-1)}(t) - y_i^{(n-1)}(t-1)] \cdot [y_j^{(n)}(t) - y_j^{(n)}(t-1)] \quad (\text{X.2})$$

АЛГОРИТМ ОБУЧЕНИЯ ПО ХЕББУ

1. При инициализации сети всем весовым коэффициентам присваиваются небольшие случайные значения
2. На входы подается выбранный случайным образом образ и сигналы возбуждения распространяются по всем слоям согласно принципам прямопоточных сетей
3. На основании полученных выходных значений нейронов по формулам (X.1) или (X.2) происходит коррекция весовых коэффициентов.
4. Цикл с шага 2, пока выходные значения сети не застабилизуются с заданной точностью

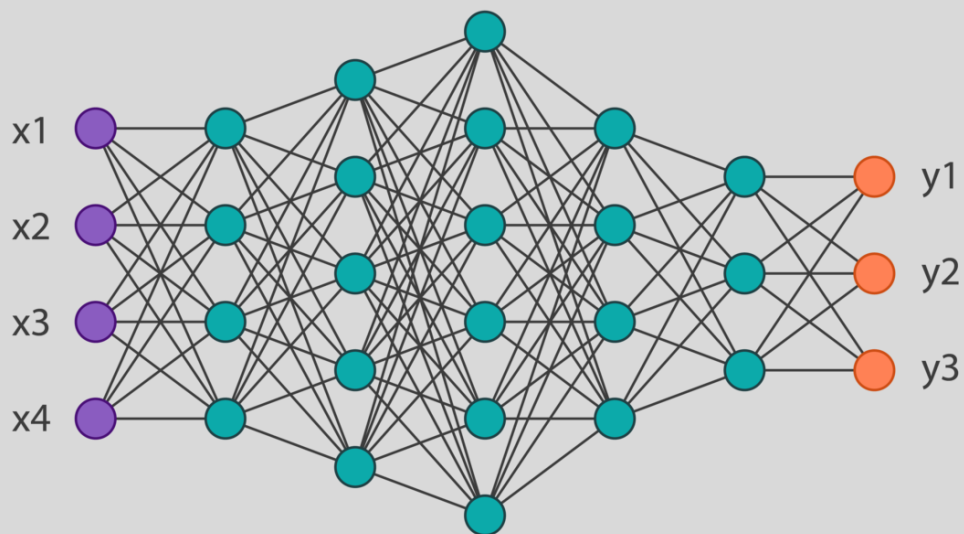
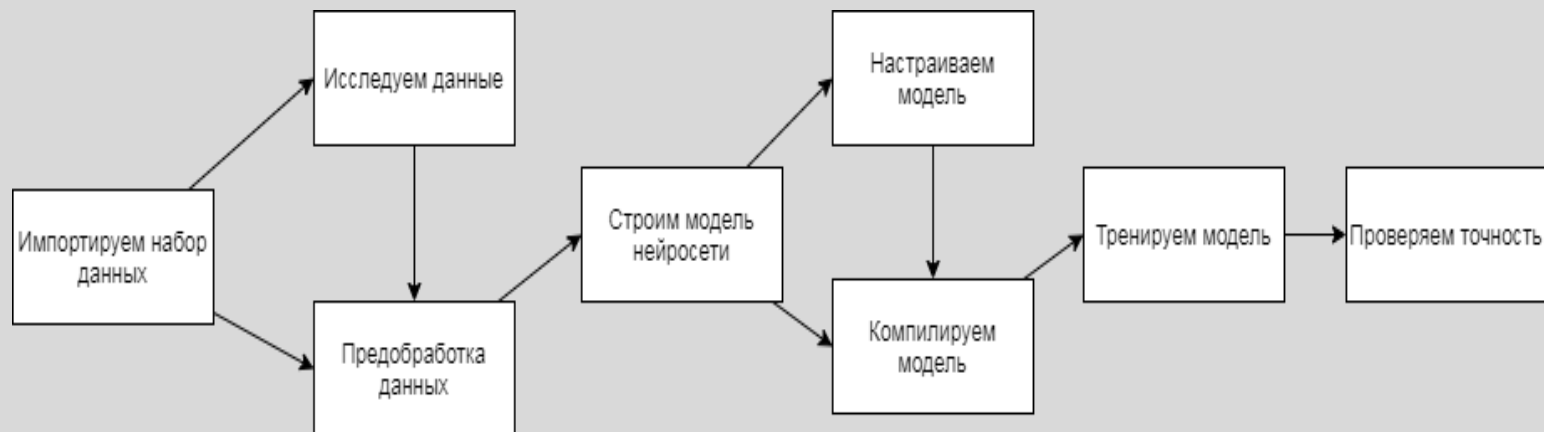
ОБУЧЕНИЕ ПО ПРАВИЛУ КОХОНЕНА

$$w_{ij}(t) = w_{ij}(t-1) + \alpha \cdot [y_i^{(n-1)} - w_{ij}(t-1)]$$

Из формулы видно, что обучение сводится к минимизации разницы между входными сигналами нейрона, поступающими с выходов нейронов предыдущего слоя и весовыми коэффициентами его синапсов

Общий алгоритм такой же, как у Хебба

ПОСТРОЕНИЕ АРХИТЕКТУРЫ НЕЙРОННОЙ СЕТИ



Достоинства	Недостатки
Способность к обучению	Отсутствие объяснительной компоненты
Адаптивность	Большое время обучения
Возможность работы с нелинейностью	Трудность формирования топологии сети
Возможность работы с неточными данными	Проблемы забывания
	Необходимость достаточно представительной обучающей выборки
	Эвристичность параметров алгоритмов обучения различных типов НС



Рисунок 1 – Топология многослойного персептрона {34–69–1} для прогнозирования латеральной миграции радионуклидов в пределах элементарного ландшафта

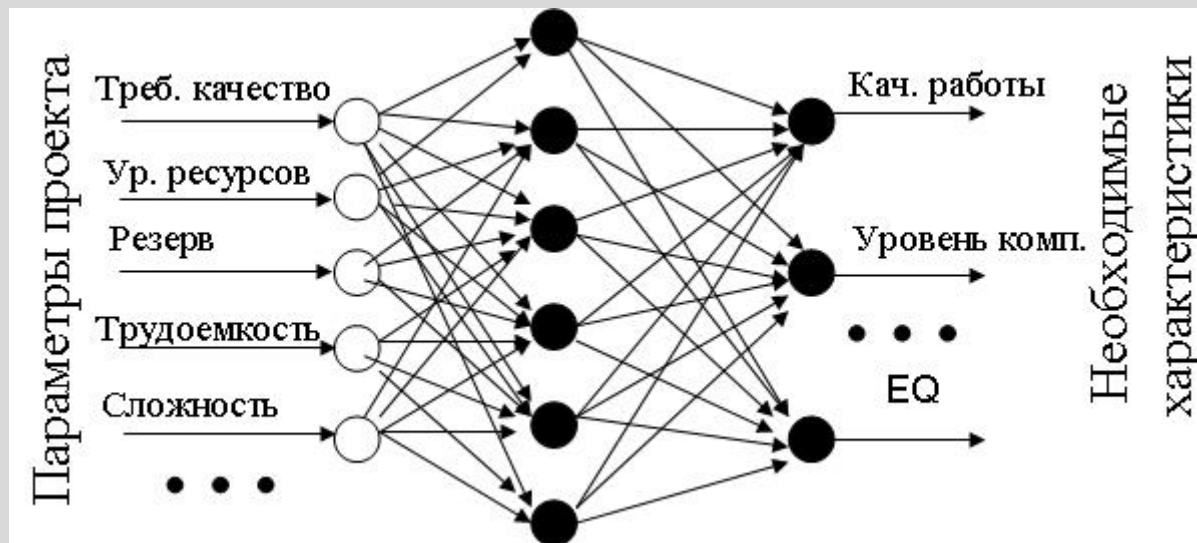
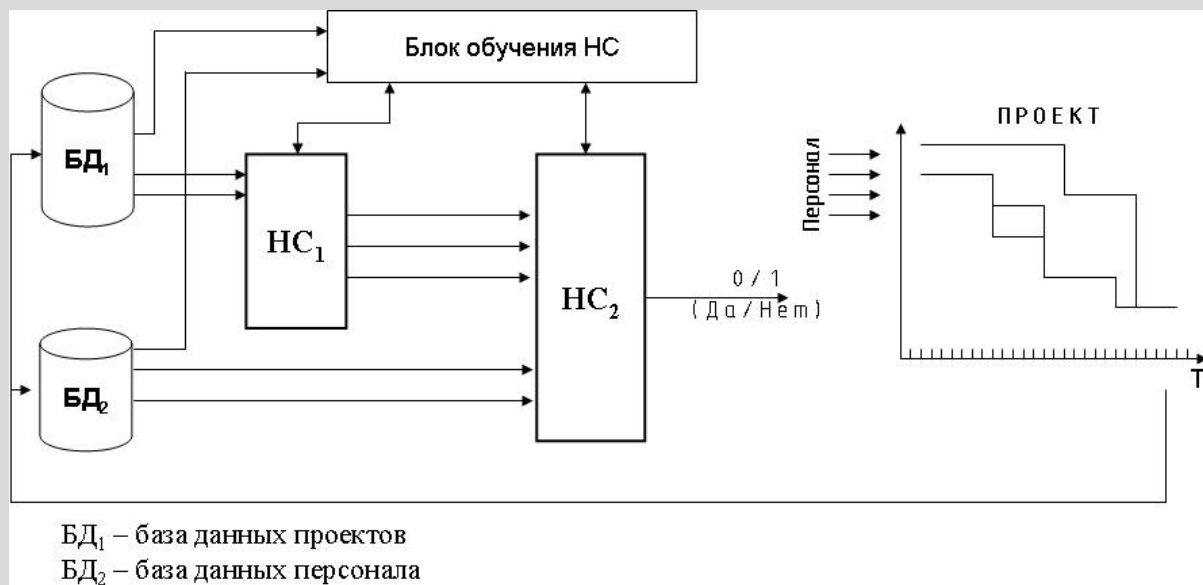
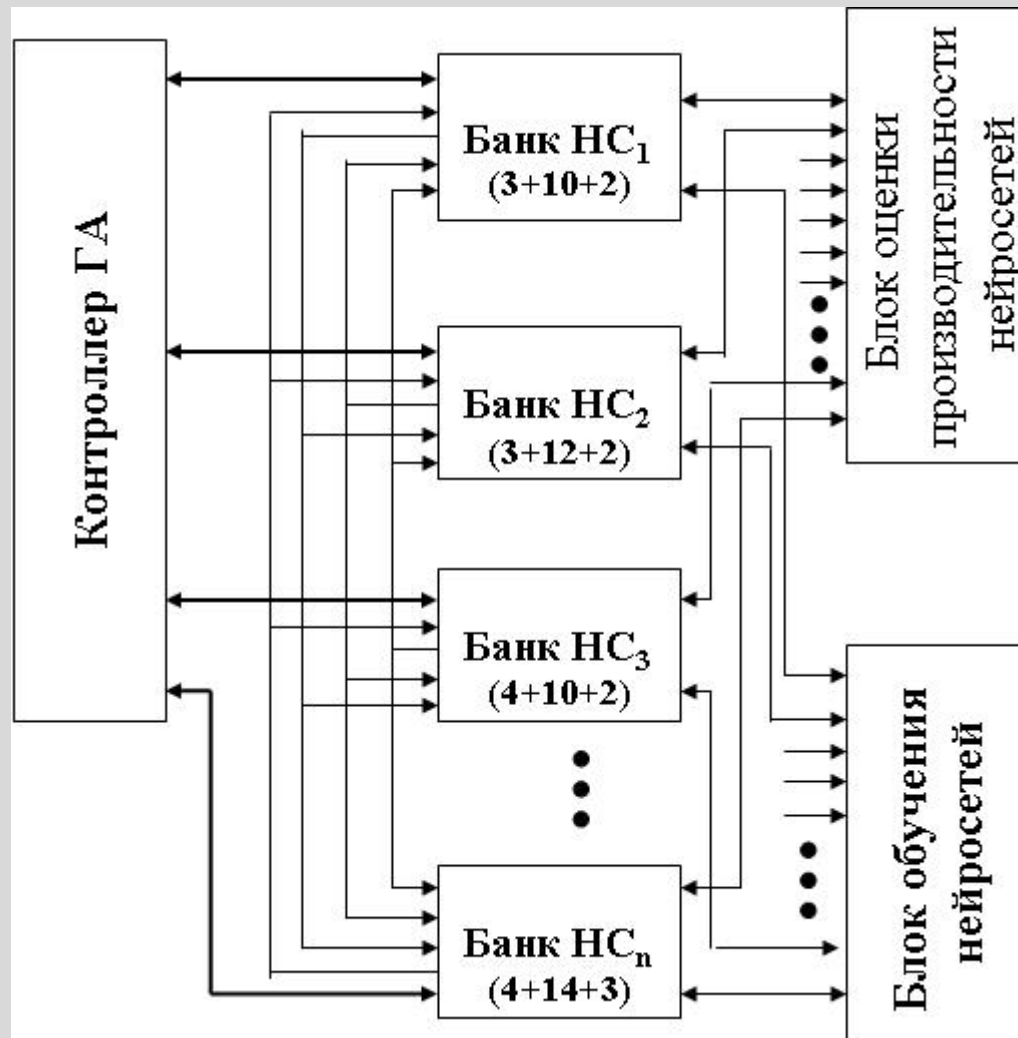


Схема нейронной сети для поиска характеристик специалистов



Шумков Е.А. Применение нейронных сетей для подбора состава группы проекта // Электронный журнал КубГАУ, 2019, №153



Архитектура системы поиска нейросетевых моделей с помощью генетического алгоритма

Шумков Е.А. Поиск нейросетевых моделей с помощью генетических алгоритмов // Электронный сетевой политематический журнал «Научные труды КубГТУ». 2021, №4

МЕТОДЫ ПОВЫШЕНИЯ КАЧЕСТВА УПРАВЛЕНИЯ СУДНОМ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ

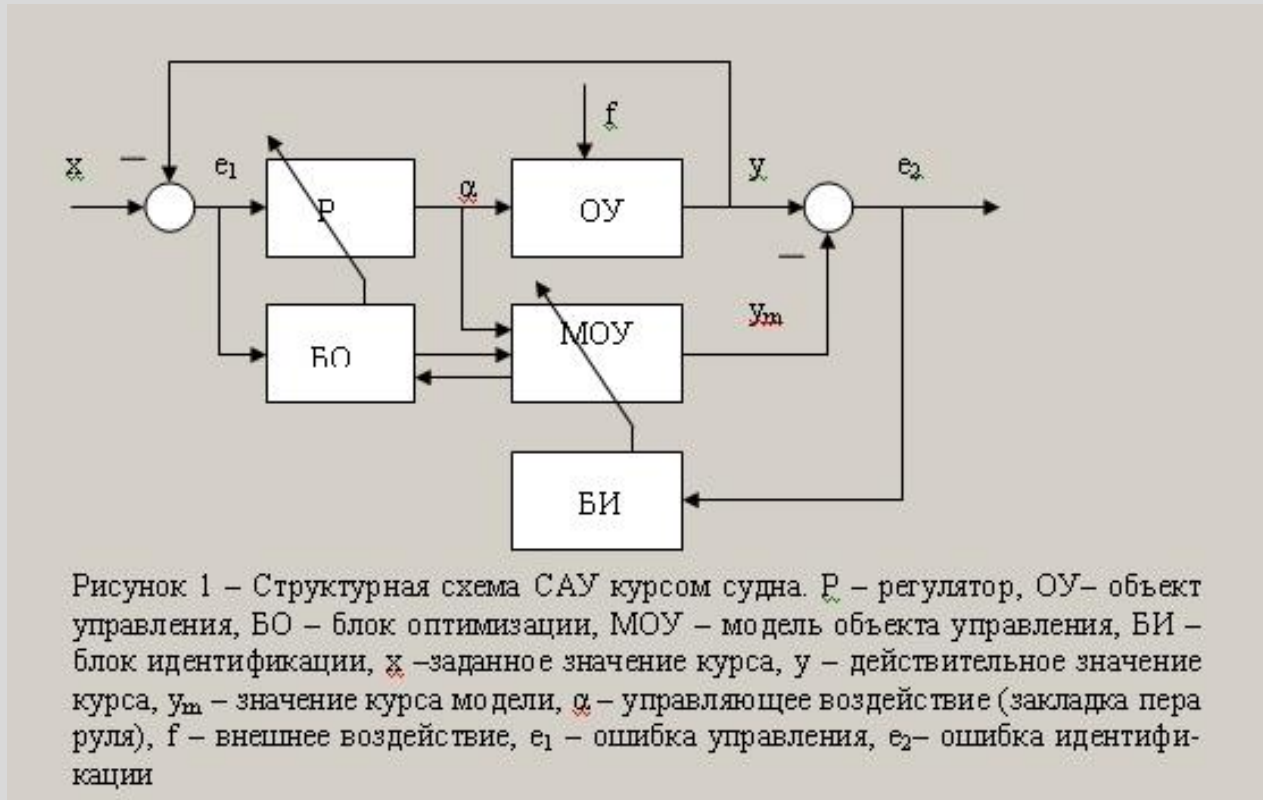


Рисунок 1 – Структурная схема САУ курсом судна. Р – регулятор, ОУ– объект управления, БО – блок оптимизации, МОУ – модель объекта управления, БИ – блок идентификации, x – заданное значение курса, y – действительное значение курса, y_m – значение курса модели, u – управляющее воздействие (закладка пера руля), f – внешнее воздействие, e_1 – ошибка управления, e_2 – ошибка идентификации

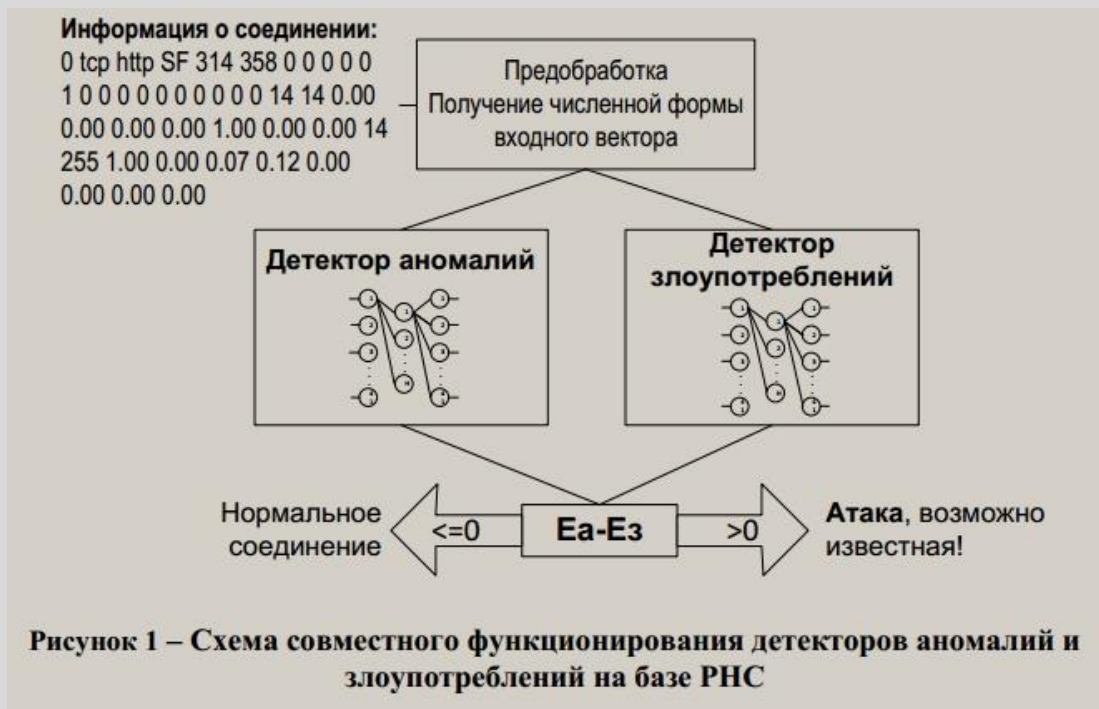
Глушков С.В. д.т.н.
Владивосток 2008
Морской ГУ им. адм. Невельского

Нейросетевое моделирование Предприятия



Долгова Е.В.
Пермь

ПерГТУ



Нейродетектор сетевых атак

Кочурко П.А.,
Беларусь

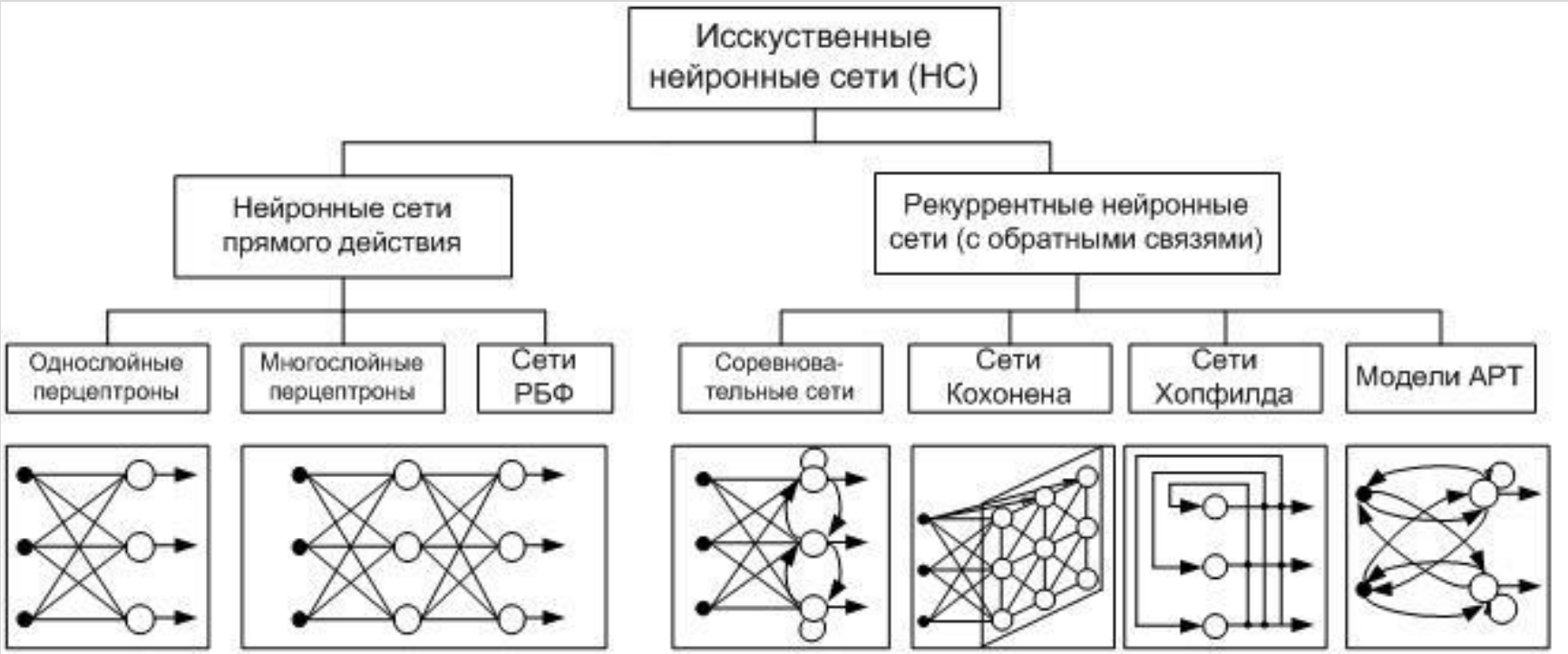
ПРИМЕРЫ ПРОГРАММНОЙ РЕАЛИЗАЦИИ НЕЙРОННЫХ СЕТЕЙ

Можно найти на сайтах:

- <http://www.github.com>
- <http://www.codeproject.com>
- <http://www.habr.com>
- <http://www.kaggle.com>
- <http://apsheronk.bozo.ru>

и многих других

ОДИН ИЗ ВАРИАНТОВ КЛАССИФИКАЦИИ НЕЙРОННЫХ СЕТЕЙ



Литература:

1. Хайкин С. Нейронные сети: полный курс. 2-е изд.: Пер. с. англ. – М.: Издательский дом «Вильямс», 2006 – 1104 с.
2. Осовский С. Нейронные сети для обработки информации. М,,: Финансы и статистика. 2002. 344 с.
3. Мкртчян С. О. Нейроны и нейронные сети. – М: Энергия, 1971. -232 с.
4. Суровцев И. С., Ключкин В. И., Пивоварова Р. П., Нейронные сети. – Воронеж: ВГУ, 1994. – 224 с.
5. Заенцев И. В. Нейронные сети: основные модели. Учебное пособие . Воронеж: ВГУ. 1998.- 76с.
6. Нейронные сети: история развития теории. Кн. 5. Учеб. пособие для вузов. / Под общей ред. А.И. Галушкина и Я.З. Цыпкина. – М.: ИПРЖР. 2001. 840 с.
7. Розенблатт Ф. Принципы нейродинамики. Перцептроны и теория механизмов мозга. М., 1965.

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 3

**Топологии нейронных сетей и алгоритмы их обучения:
карты Кохонена, рекуррентные нейронные сети**

АЛГОРИТМ RPROP
Resilient Propagation – «упругое распространение»
Авторы: M.Riedmiller & H.Braun

В данном алгоритме устранен главный недостаток BackProp – скорость обучения

$$\Delta_{\tilde{y}}^{(t)} = \left\{ \begin{array}{l} \eta^+ \Delta_{\tilde{y}}^{(t)}, \quad \frac{\partial E^{(t)}}{\partial w_{\tilde{y}}} \frac{\partial E^{(t-1)}}{\partial w_{\tilde{y}}} > 0 \\ \eta^- \Delta_{\tilde{y}}^{(t)}, \quad \frac{\partial E^{(t)}}{\partial w_{\tilde{y}}} \frac{\partial E^{(t-1)}}{\partial w_{\tilde{y}}} < 0 \end{array} \right. \quad (1)$$
$$0 < \eta^- < 1 < \eta^+$$

shumkoff.ru

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}$$

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}$$

<https://basegroup.ru/community/articles/rprop>

АЛГОРИТМ RPROP

Если на текущем шаге частная производная по соответствующему весу изменила знак, то из этого следует, что последнее изменение было большим и алгоритм проскочил локальный минимум. Следовательно, величину изменения необходимо уменьшить на η^- и вернуть предыдущее значение весового коэффициента, то есть сделать «откат».

$$\Delta w_{ij}(t) = \Delta w_{ij}(t) - \Delta_{ij}^{(t-1)}$$

Если знак частной производной не изменился, то нужно увеличить величину коррекции на η^+

Рекомендуется $\eta^+ = 1.2$ $\eta^- = 0.5$

Для вычисления значения коррекции весов используется следующее правило

$$\Delta w_{ij}(t) = \begin{cases} -\Delta_{ij}^{(t)}, & \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)}, & \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0, & \frac{\partial E^{(t)}}{\partial w_{ij}} = 0 \end{cases}$$

Шаги алгоритма RPROP:

- 1) Проинициализировать величину коррекции Δ_{ij}
- 2) Предъявить все примеры из выборки и вычислить частные производные
- 3) Подсчитать новое значение дельты по ф. 1 и 3 Δ_{ij}
- 4) Скорректировать веса по формуле 4
- 5) Если условие останова не выполнено, то перейти к 2

Скорость примерно в 4-5 раз выше, чем у алгоритма обратного распространения ошибки

ОБЩИЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧ С ПОМОЩЬЮ МСПРС

1. Рассмотреть поставленную задачу и подумать, а не проще ли ее решить с помощью других методов
2. Определиться с начальным пулом значащих входов. Пул входов обычно перебирается в течение времени и выбираются те, на которых дается наименьшая ошибка
3. Выбрать, что будет на выходе нейронной сети, то есть определить выходные переменные
4. Определиться с начальным количеством слоев и нейронов в них

СЕТЬ КОХОНЕНА

Нейронные сети Кохонена – класс нейронных сетей, основным элементом которых является слой Кохонена.

Предложены в 1982 году финским ученым **Тейву Кохоненом**.

Слой Кохонена состоит из линейных формальных нейронов, которые обрабатывают информацию по принципу **«Победитель забирает всё»**.

Т.е. на каждой итерации обучения корректируются веса синаптических связей только того нейрона, у которого наибольший сигнал на выходе.

ПРИМЕНЕНИЕ СЕТИ КОХОНЕНА

Сеть Кохонена обычно используется в задачах кластеризации.

Кластеризация - это разбиение набора примеров на несколько компактных областей (кластеров), причем число кластеров заранее неизвестно.

Кластеризация позволяет представить неоднородные данные в более наглядном виде и использовать далее для исследования каждого кластера различные методы.

ПОСТАНОВКА ЗАДАЧИ ДЛЯ СЕТИ КОХОНЕНА

Есть множество классов $C^1, \dots, C^M = \{C^m\}$

Есть объекты $x^p = (x_1^p, \dots, x_N^p)$

Задача – разбить объекты по классам

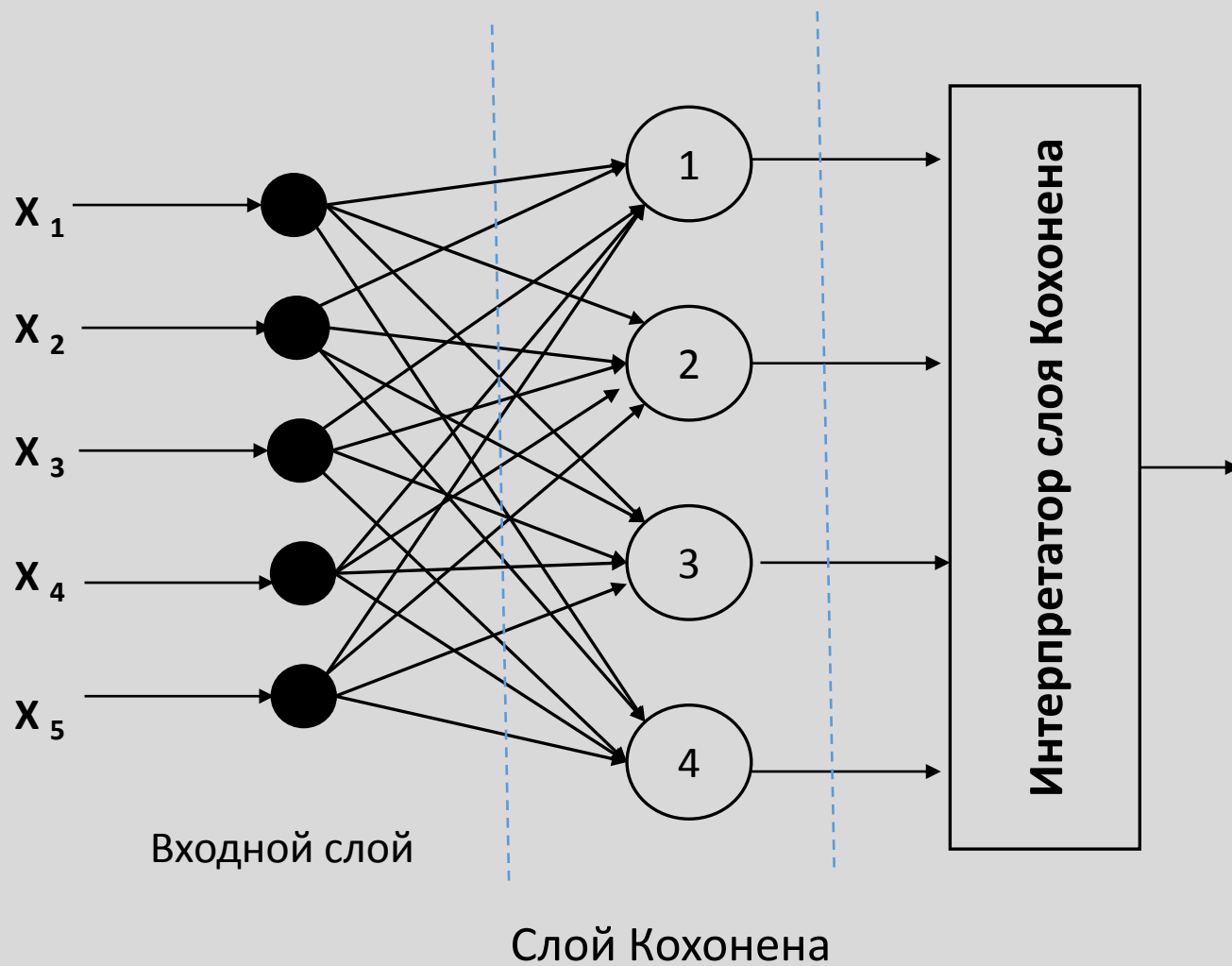
Мера близости

$$\sum_p d(x^p, c^{m(p)}) \rightarrow \min$$

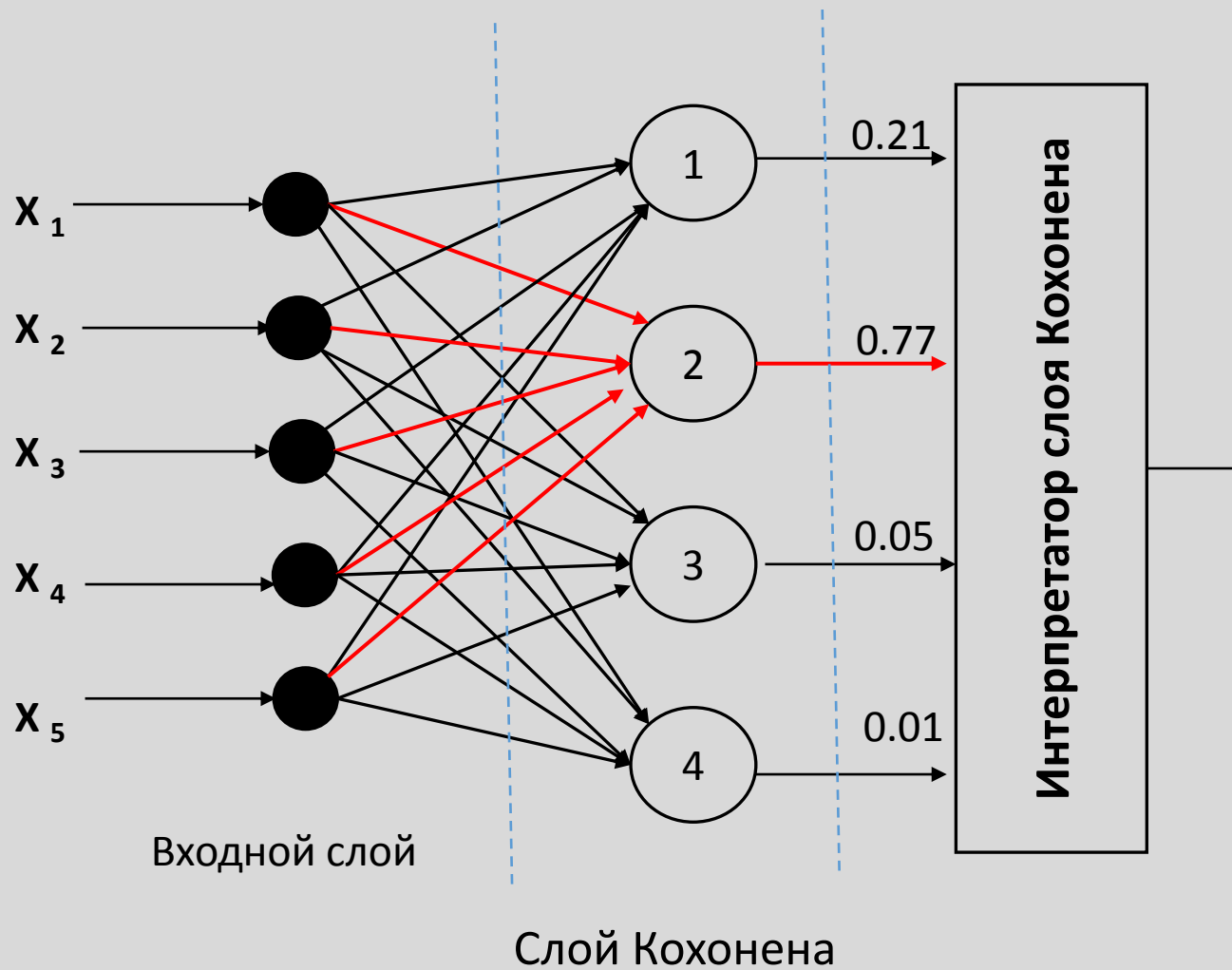
Обычно d – евклидова мера

$$d(x, y) = \sum_i (x_i - y_i)^2$$

СТРУКТУРА СЕТИ КОХОНЕНА



Принцип «ПОБЕДИТЕЛЬ ЗАБИРАЕТ ВСЕ»



→ На данной итерации корректируются только веса 2-го нейрона слоя Кохонена

АЛГОРИТМ ОБУЧЕНИЯ СЕТИ КОХОНЕНА

1. Подаем на входы один из векторов X^P
2. Рассчитываем выход слоя Кохонена и определяем номер выигравшего нейрона m , выход которого максимален.
3. Корректируем веса только выигравшего нейрона m по формуле

$$w_{m_0} = w_{m_0} + \alpha(x^P - w_{m_0})$$

где α – скорость обучения. Обычно используется монотонно убывающая функция $a(t)$

Обучение происходит пока не застabilизируются веса связей. На шаге 1 входной пример выбирается случайным образом.

НЕКОТОРЫЕ ЗАМЕЧАНИЯ ПО ФУНКЦИОНИРОВАНИЮ СЕТИ КОХОНЕНА

1. Если использовать функцию SOFTMAX, то выходы можно интерпретировать как вероятность.
2. Инициализация весов связей производится случайным образом небольшими значениями (обычно в интервале $(0;1]$).
3. Желательно, чтобы значения весов были равномерно распределены.
4. Входной слой не производит никаких вычислений, только передает сигналы на слой Кохонена.



ТЕЙВУ КОХОНЕН (фото - Википедия)

• Высшая техническая школа Университета Аалто

САМОРГАНИЗУЮЩАЯСЯ КАРТА КОХОНЕНА

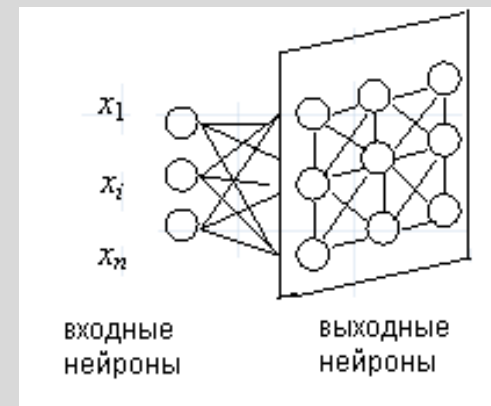
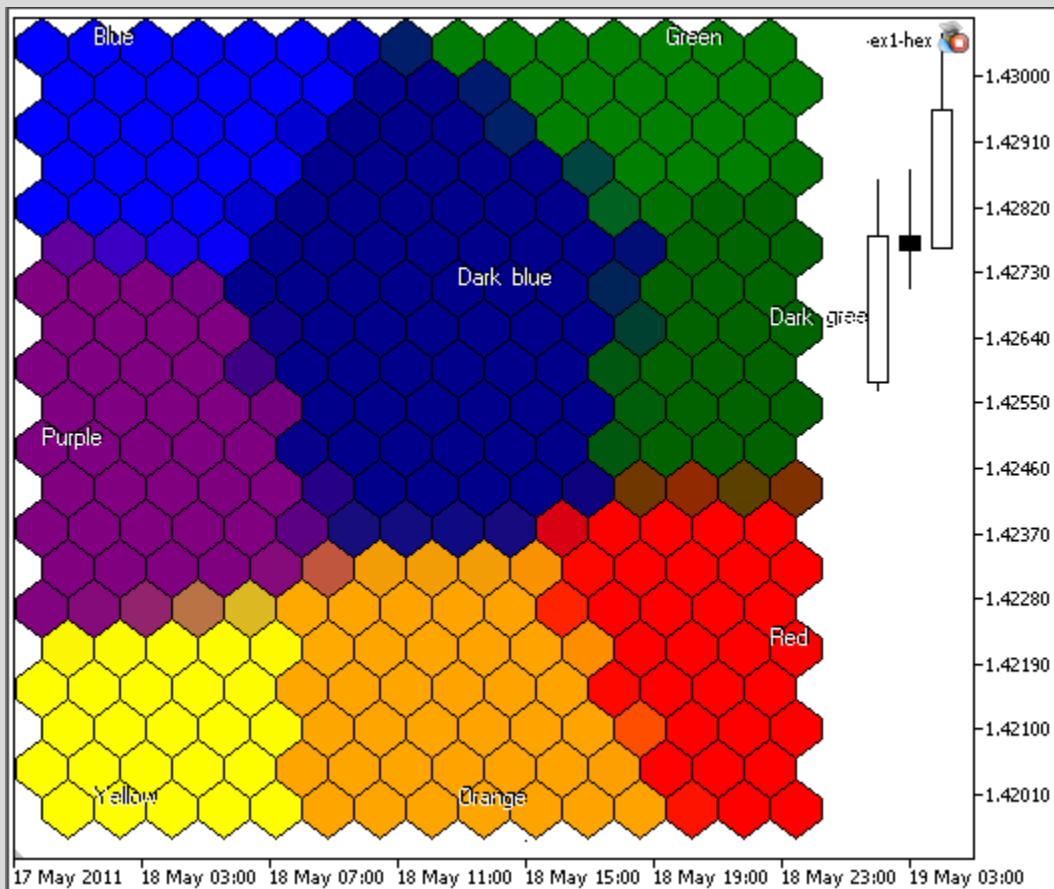


Схема - Интуит

Рисунок – mql5.com

ЗВЕЗДЫ ГРОССБЕРГА

Входная и выходная звезды Гроссберга

Входная звезда: Нейрон в форме входной звезды имеет N входов $X_1..X_N$, которым соответствуют веса $W_1..X_N$, и один выход Y , являющийся взвешенной суммой входов. Входная звезда обучается выдавать сигнал на выходе всякий раз, когда на входы поступает определенный вектор. Таким образом, входная звезда является детектором совокупного состояния своих входов. Процесс обучения представляется в следующей итерационной форме:

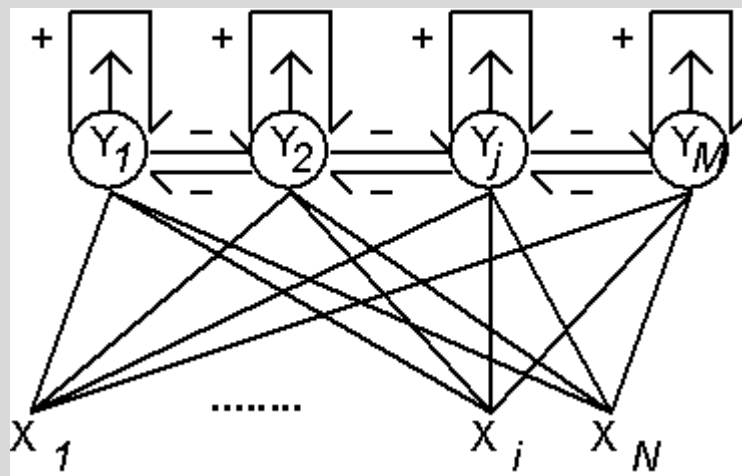
$$W_i(t + 1) = W_i(t) + \alpha(X_i - W_i(t))$$

Выходная звезда Гроссберга выполняет противоположную функцию – функцию командного нейрона, выдавая на выходах определенный вектор при поступлении сигнала на вход. Нейрон этого типа имеет один вход и M выходов с весами $W_1..M$, которые обучаются по формуле:

$$W_i(t + 1) = W_i(t) + \beta(Y_i - W_i(t))$$

Принцип Winner Take All (WTA) - Победитель Забирает Все – модель Липпмана-Хемминга.

1987 г.



Торможение нейронов

$$y^m(t+1) = f\left(y^m(t) - (1 / (M + 1)) \sum_{n \neq m} y^n(t)\right)$$

СЕТЬ ВСТРЕЧНОГО РАСПРОСТРАНЕНИЯ

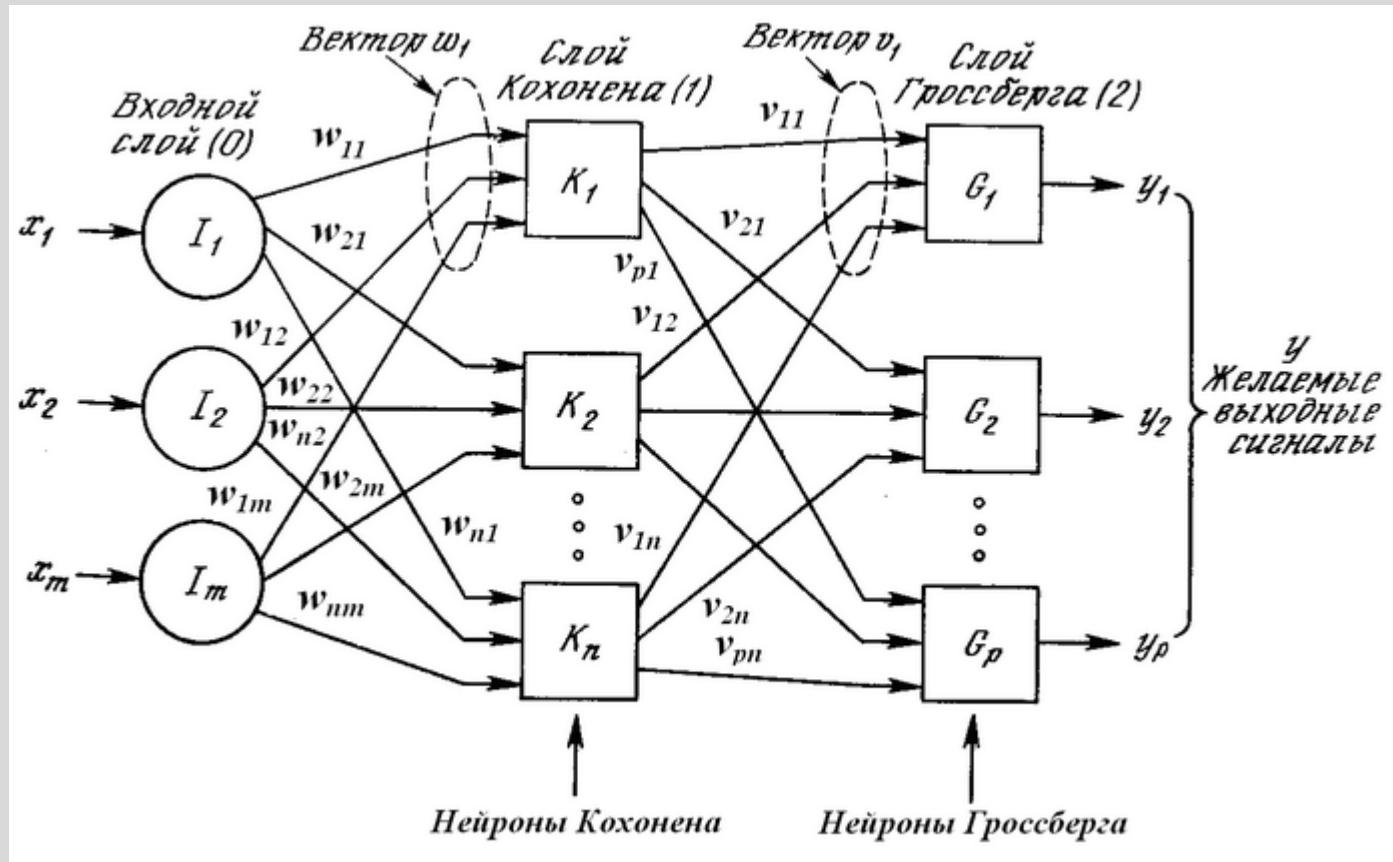
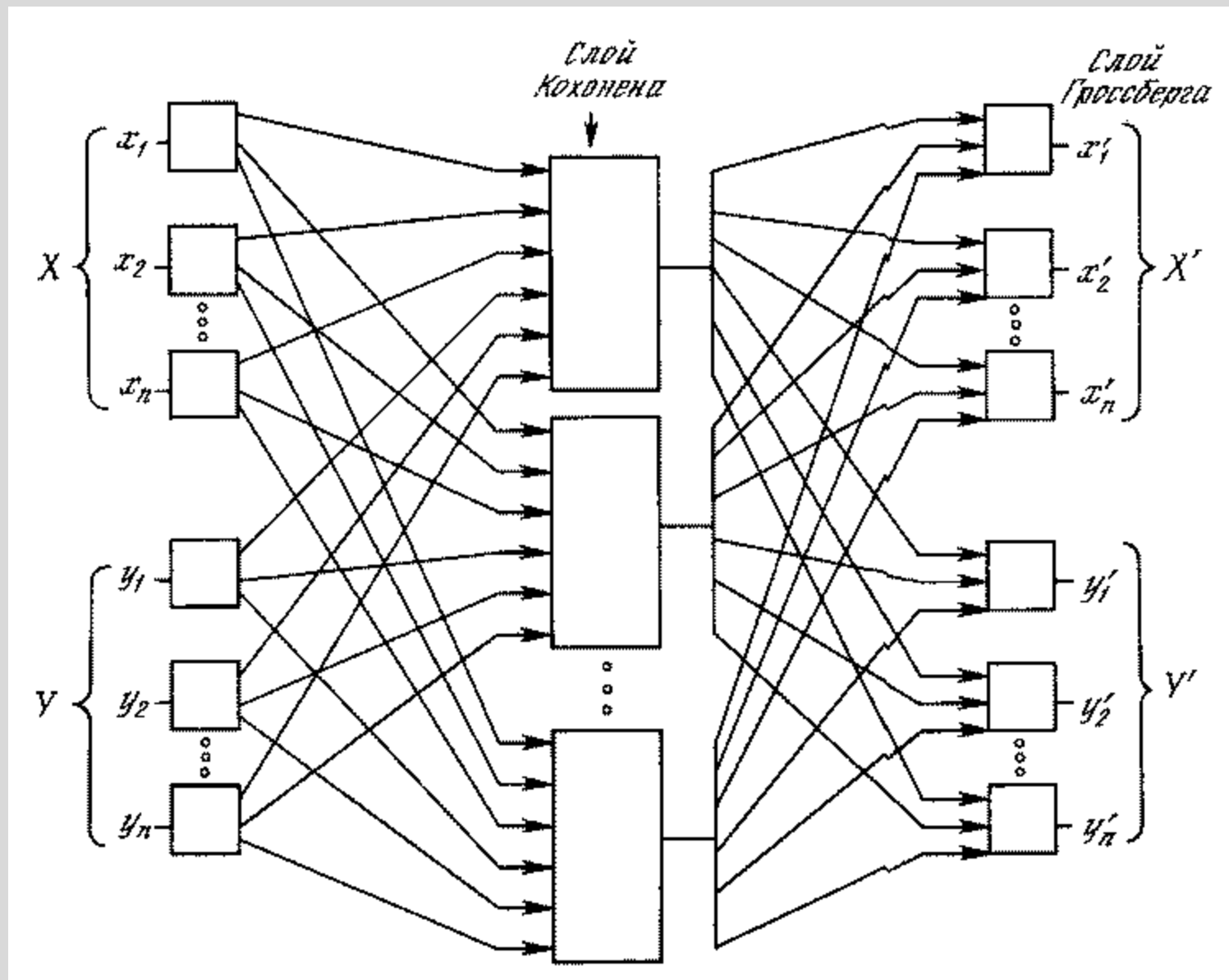
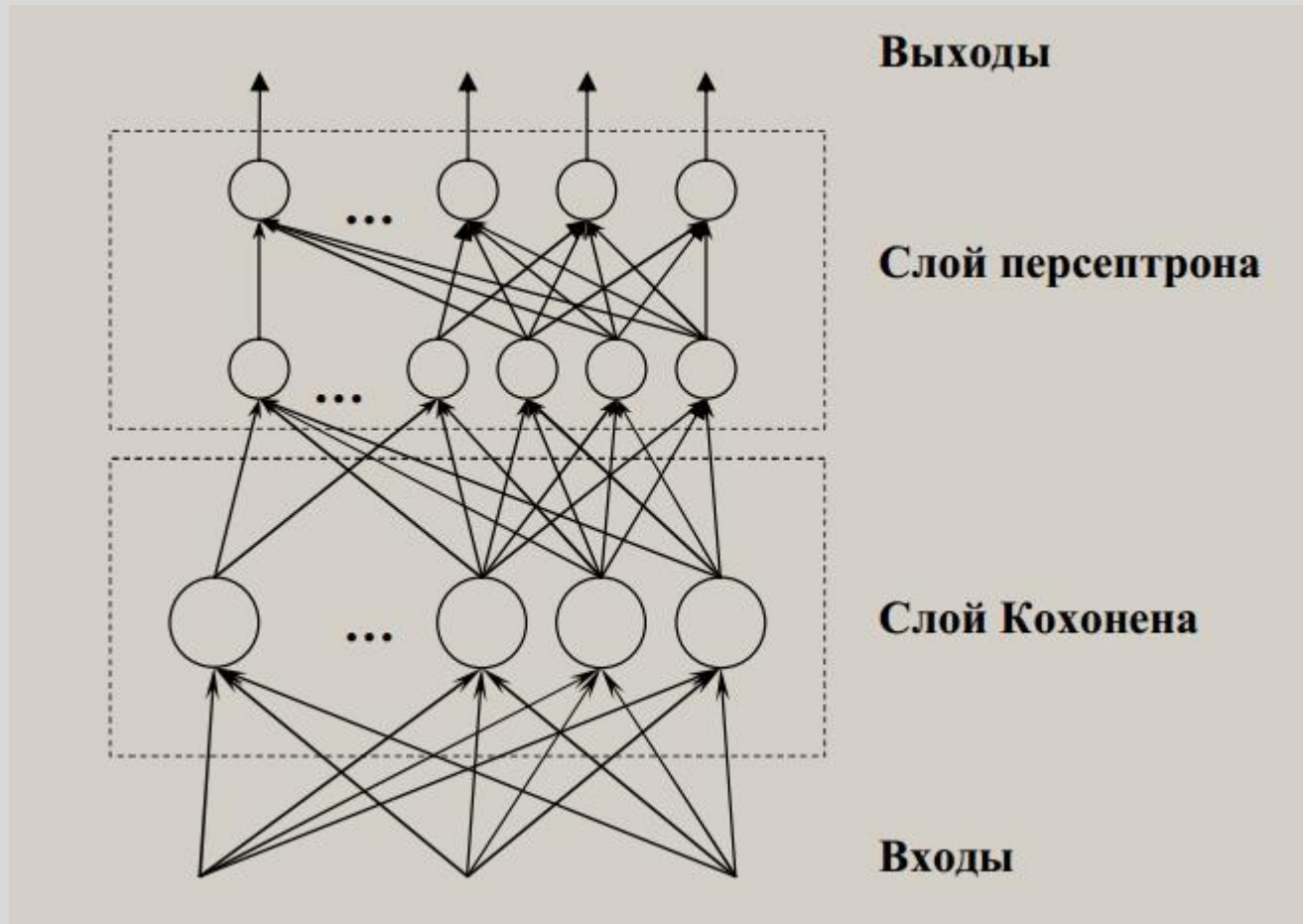


Схема – studfile.net

Режим интерполяции сети встречного распространения



ГИБРИДНАЯ НЕЙРОННАЯ СЕТЬ



РЕКУРРЕНТНАЯ НЕЙРОННАЯ СЕТЬ

Recurrent neural network; RNN

Вид нейронных сетей, где связи между элементами образуют направленную последовательность. Благодаря этому появляется возможность обрабатывать серии событий во времени или последовательные пространственные цепочки. В отличие от многослойных перцептронов, рекуррентные сети могут использовать свою внутреннюю память для обработки последовательностей произвольной длины.

В связи с особенностью архитектуры, РНС могут применяться в задачах, где целое разбито на части: распознавание речи или рукописного текста.

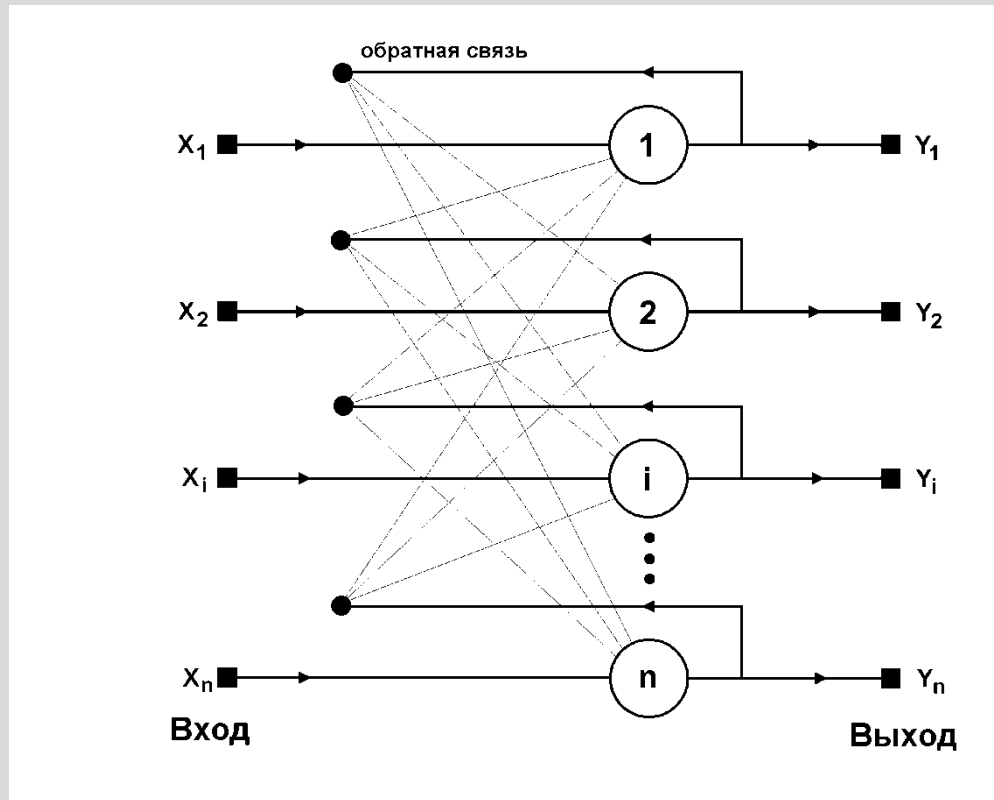
Существует большое количество рекуррентных нейросетей, но наибольшее распространение получили сеть с долговременной и кратковременной памятью (LSTM) и управляемый рекуррентный блок (GRU)

А началось все с нейросети Хопфилда в 1982 г.

РЕКУРРЕНТНЫЕ НЕЙРОННЫЕ СЕТИ

СЕТЬ ХОПФИЛДА

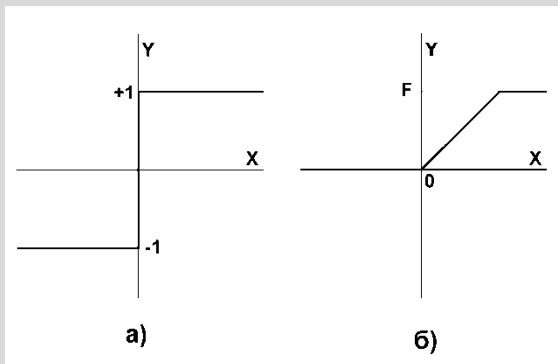
1982 г.



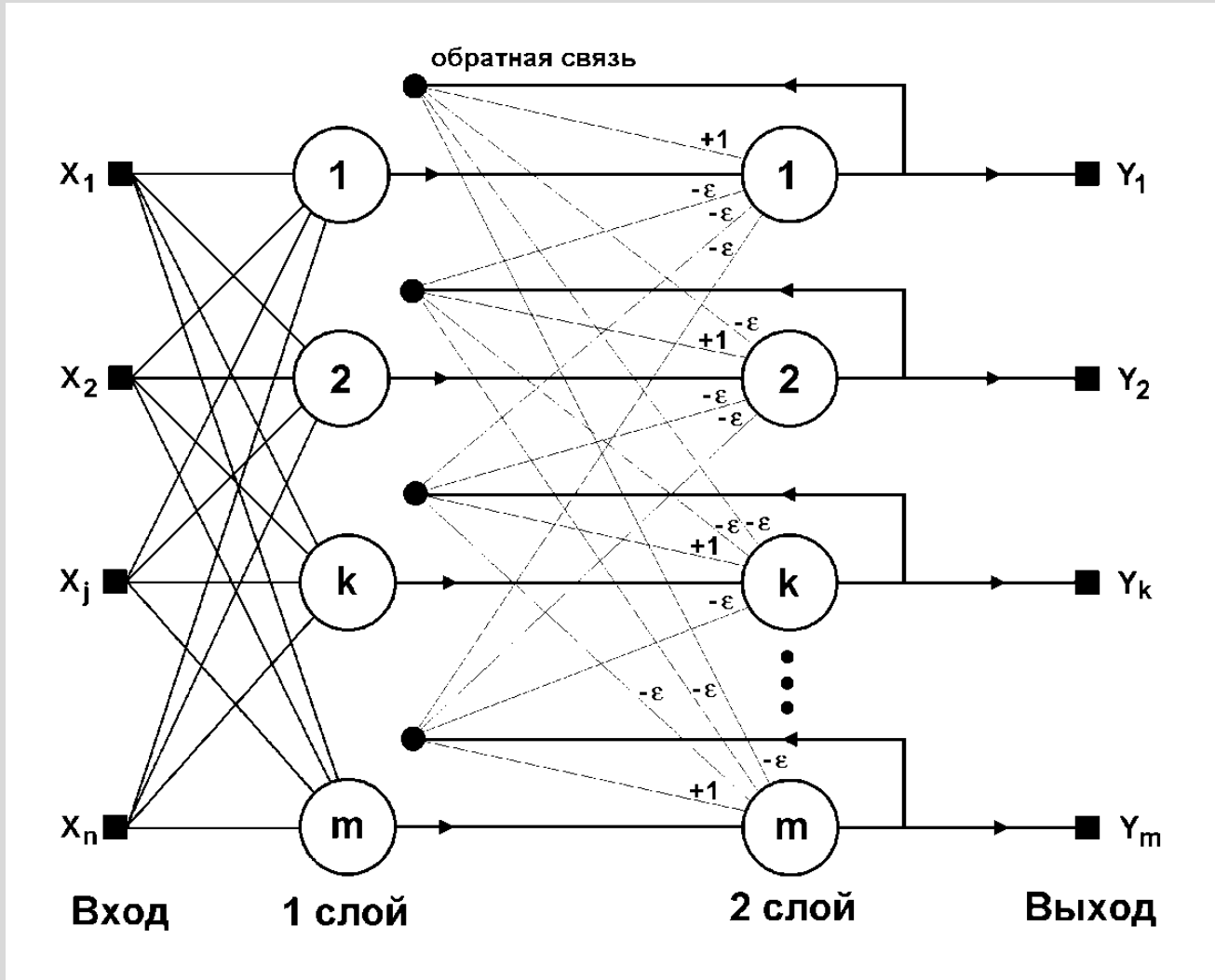
$$W_{ij} = \begin{cases} \sum_{k=0}^{m-1} x_i^k x_j^k, & i \neq j \\ 0, & i = j \end{cases}$$

$$s_j(p+1) = \sum_{i=0}^{n-1} w_{ij} y_i(p)$$

$$y_j(p+1) = f[s_j(p+1)]$$



СЕТЬ ХЕММИНГА



$$W_{ik} = \frac{x_i^k}{2}$$

$$0 < \varepsilon < 1/m$$

Схема – С.Короткий

$$y_j^{(1)} = s_j^{(1)} = \sum_{i=0}^{n-1} w_{ij} x_i + T_j$$

Long short-term memory LSTM

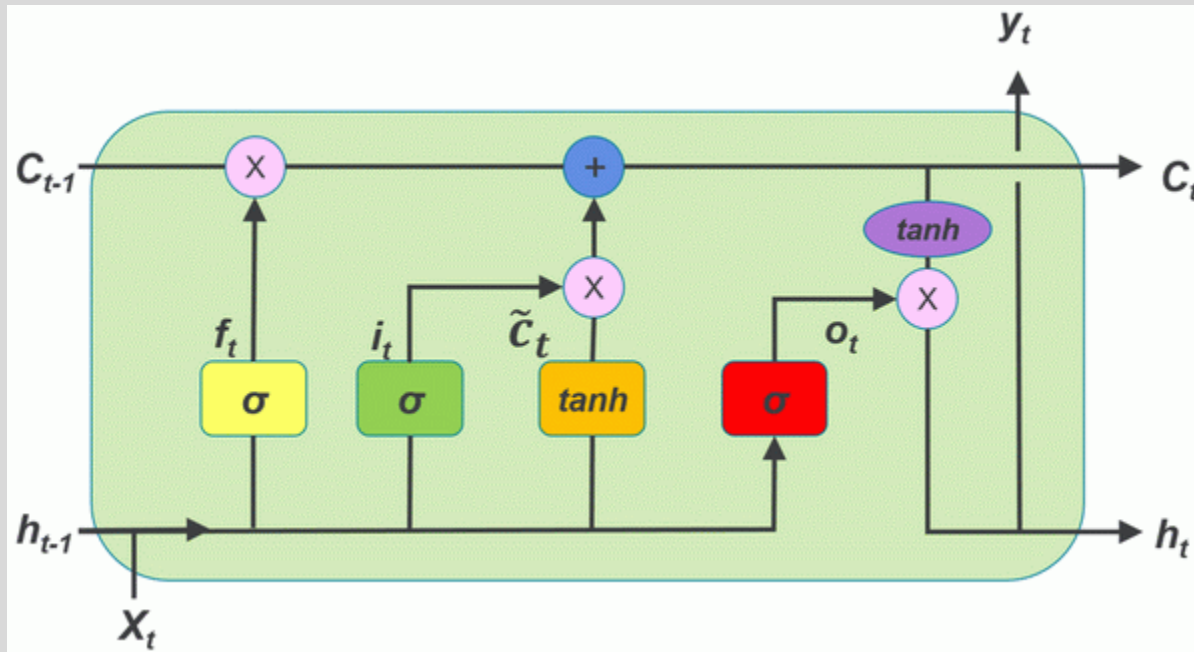


Рис. www.researchgate.net

LSTM – модуль (блок)

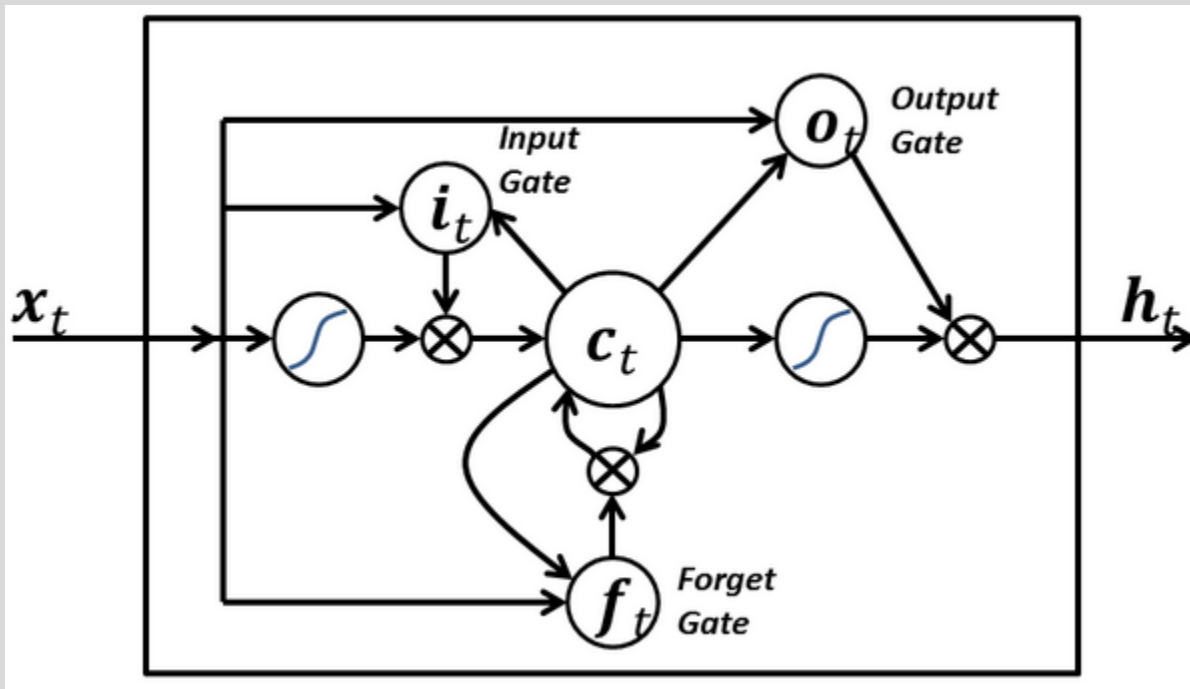
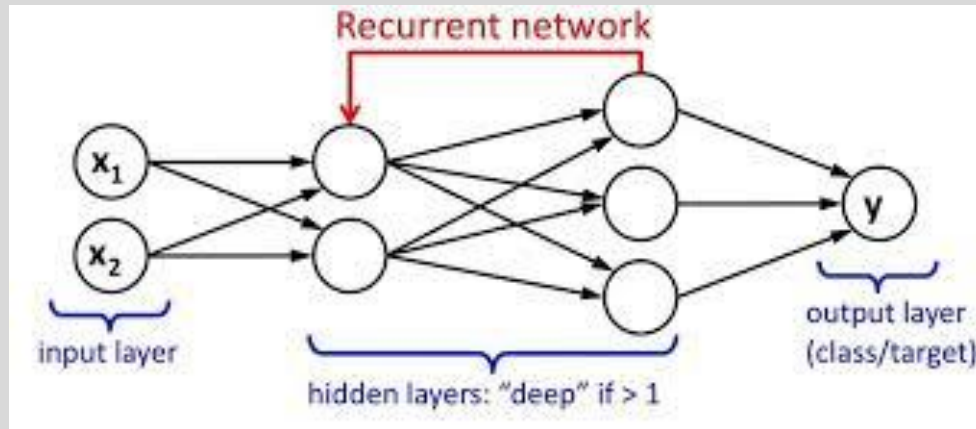


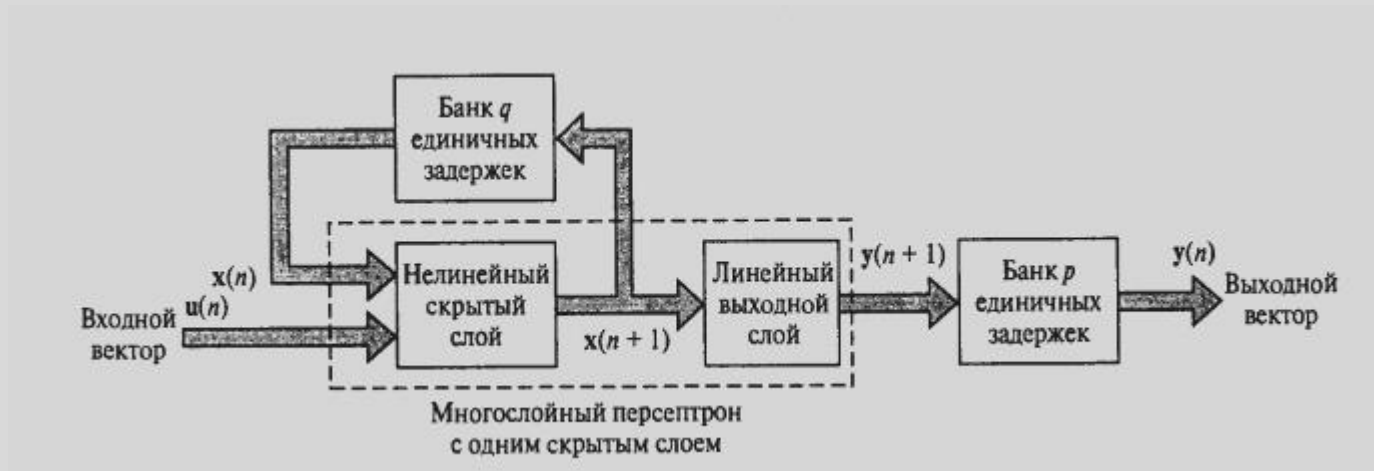
Схема - Википедия

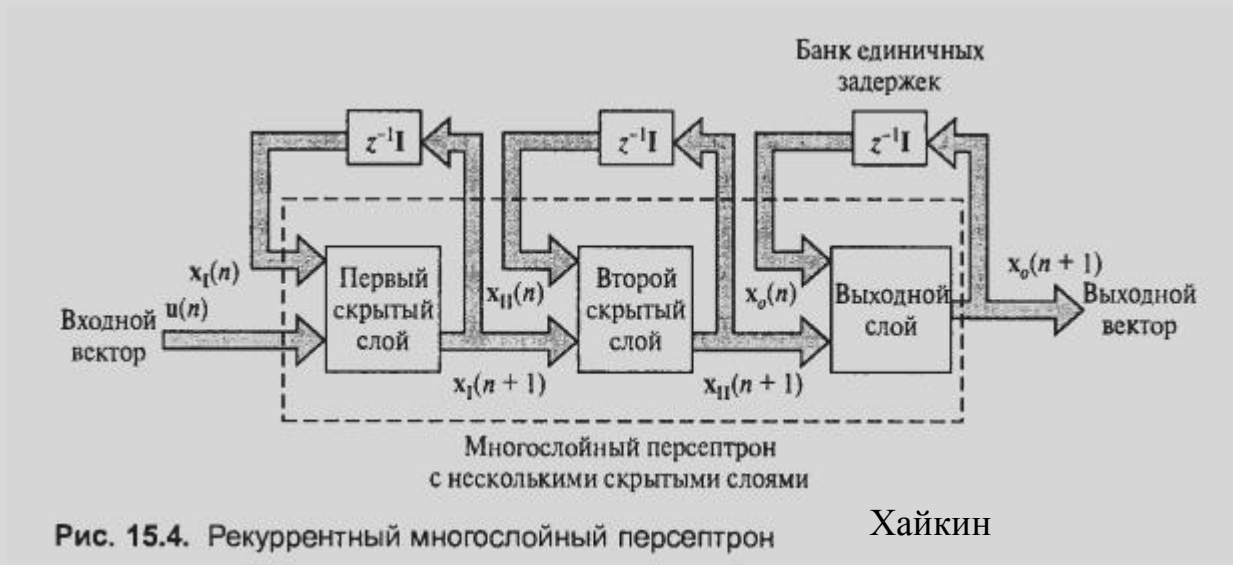
F – вентиль забывания

I – входной вентиль



deepai.org

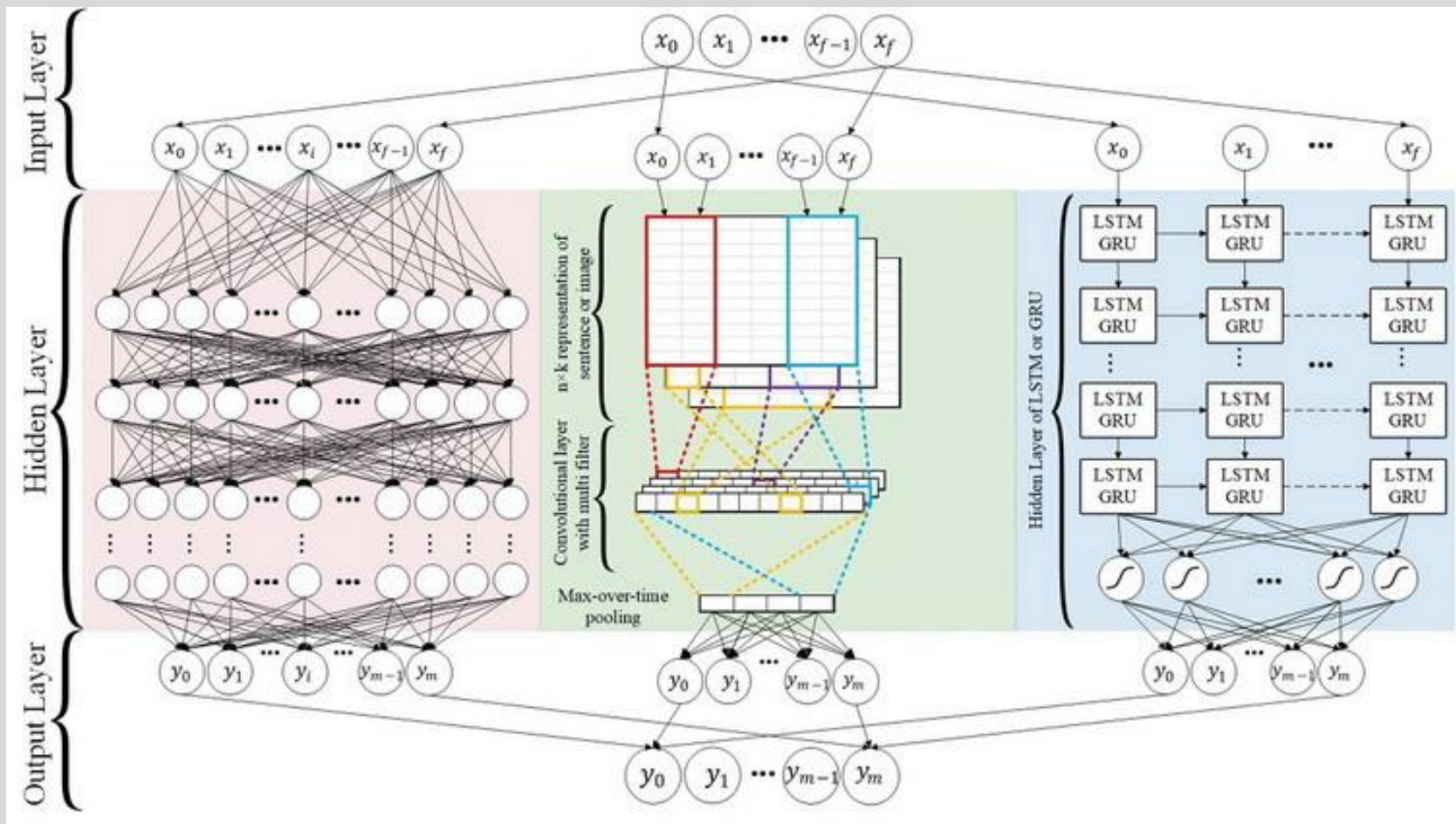




$$u_j(k+1) = \sum_{j=1}^n w_{ji}^{[1]} x_i(k) + \sum_{i=1}^{n1} w_{ji}^c o_i(k) + \theta_j^{[1]}$$

$$o_j(k+1) = \phi(u_j(k+1)) = \tanh(u_j(k+1)), \quad j = 1, 2, 3 \dots n1$$

$$y_j(k+1) = \sum_{i=1}^{n1} w_{ji}^{[2]} o_i(k+1) + \theta_j^{[2]}$$



Вариант сложной топологии с РНС

Сеть ЭЛМАНА

Простая рекуррентная сеть (Simple Recurrent Network SRN)

Частично-рекуррентная сеть

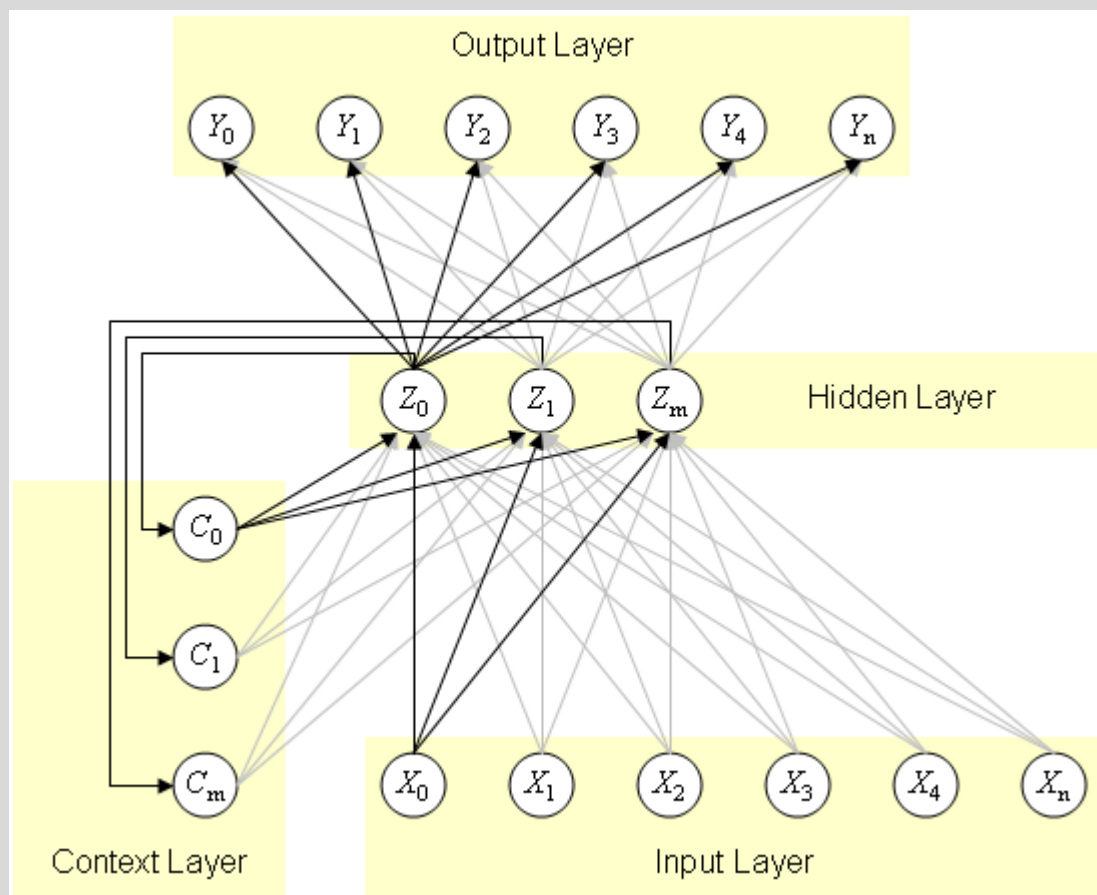
Выходной слой может быть нелинейным и банк нелинейных задержек на выходе отсутствует.

Сеть Элмана содержит рекуррентные связи скрытых нейронов со слоем контекстных элементов (context unit), состоящим из единичных задержек. Эти контекстные элементы сохраняют выходы скрытых нейронов на один шаг времени, после чего передают их на входной слой.

Элементы сети Элмана – стандартные нейроны, адаптивные линейные ассоциаторы и элементы задержки

Тип функции активации сети Элмана – гиперболический тангенс

ТОПОЛОГИЯ СЕТИ ЭЛМАНА

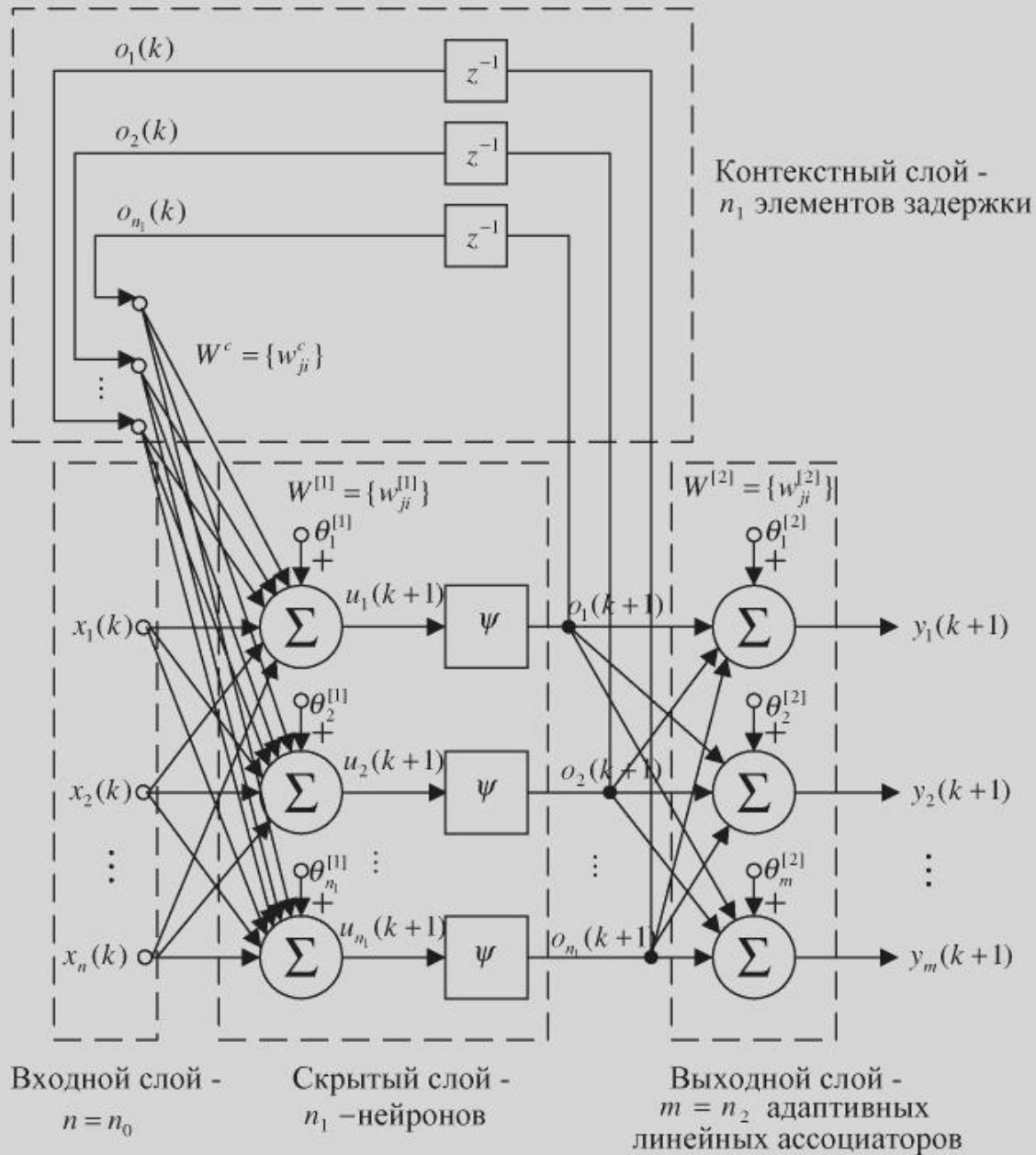


АЛГОРИТМ РАБОТЫ СЕТИ ЭЛМАНА

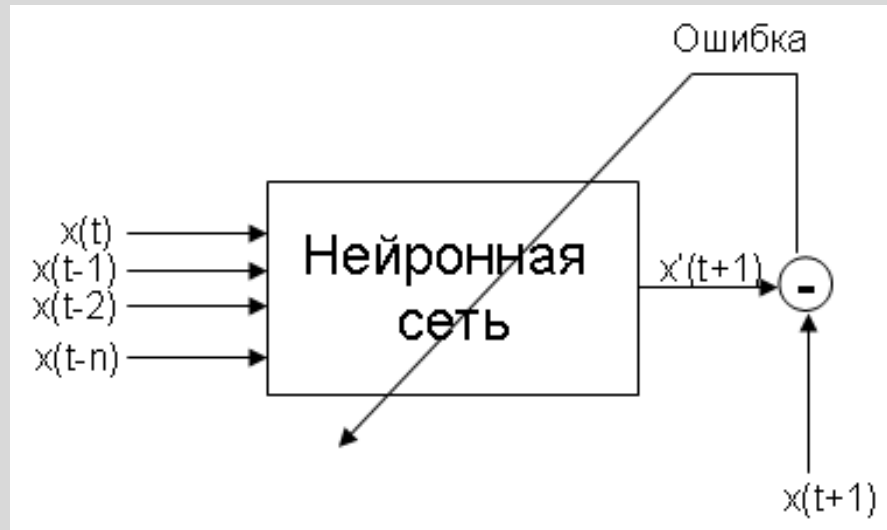
$$u_j(k+1) = \sum_{j=1}^n w_{ji}^{[1]} x_i(k) + \sum_{i=1}^{n1} w_{ji}^c o_i(k) + \theta_j^{[1]}$$

$$o_j(k+1) = \phi(u_j(k+1)) = \tanh(u_j(k+1)), \quad j = 1, 2, 3 \dots n1$$

$$y_j(k+1) = \sum_{i=1}^{n1} w_{ji}^{[2]} o_i(k+1) + \theta_j^{[2]}$$



ЗАДАЧА ПРОГНОЗИРОВАНИЯ С ПОМОЩЬЮ НЕЙРОСЕТИ



Выход у прогнозирующей НС обычно один

Скользящее окно данных

Второй вариант «окна» данных

LastY	Hist1	Hist2	Hist3	Hist4	Hist0
L	D-1	D-2	D-3	D-4	D
L-1	D-2	D-3	D-4	D-5	D-1
L-2	D-3	D-4	D-5	D-6	D-2
...

<http://apsheronk.bozo.ru/Neural/Lec9.htm>

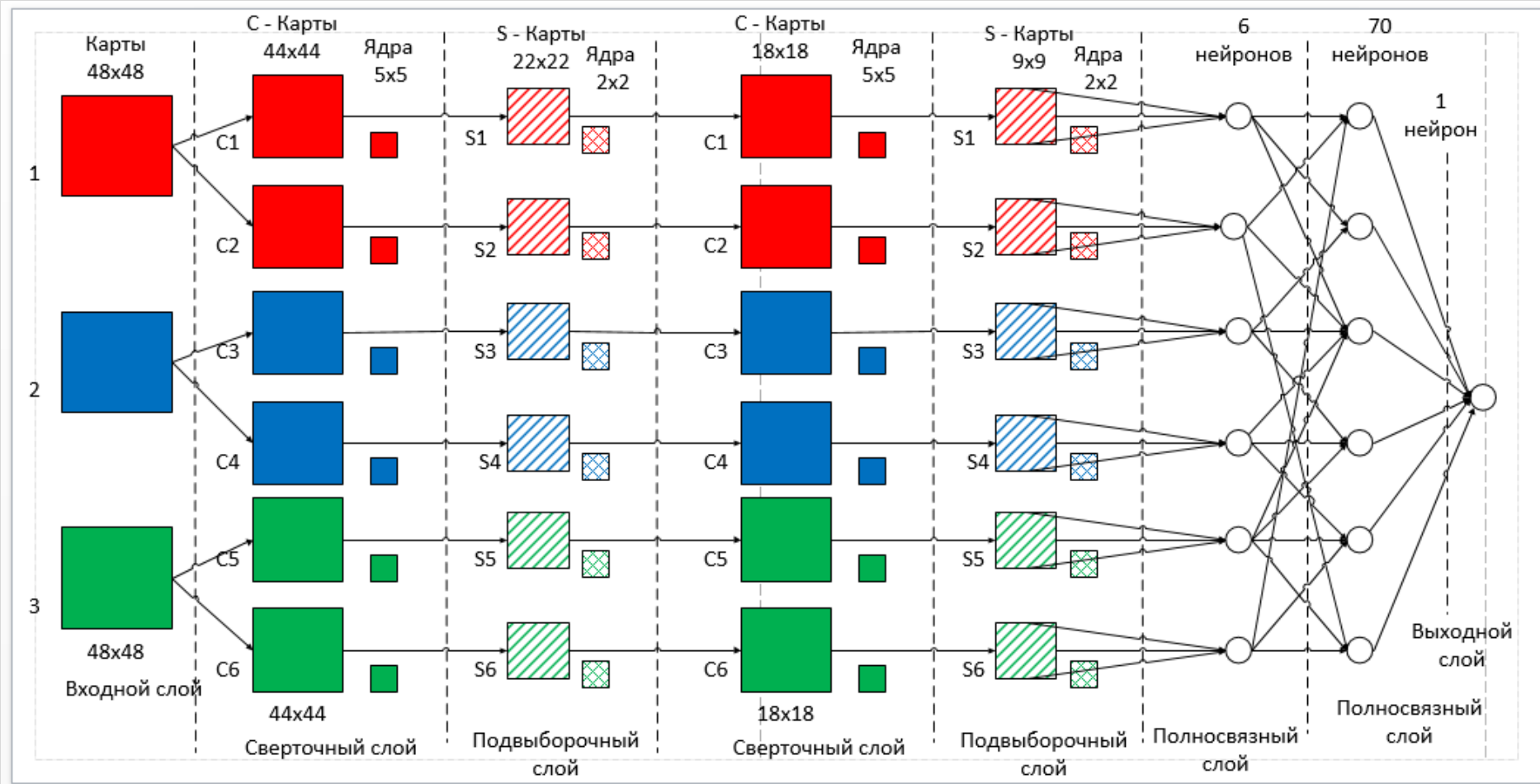
БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 4

СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ

ОБЩИЙ ВИД СВЕРТОЧНОЙ НЕЙРОНОЙ СЕТИ (схема –www.habr.com)



Сверточная нейронная сеть – тип глубокой нейронной сети, показывающий наилучшие результаты в области распознавания изображений.

Работу сверточных нейронных сетей можно описать как переход от конкретных особенностей изображения (признаков) к более абстрактным деталям, и так далее вплоть до абстракций высокого уровня.

Сеть сама настраивает необходимую иерархию признаков, выделяя самые существенные.

Преимущества:

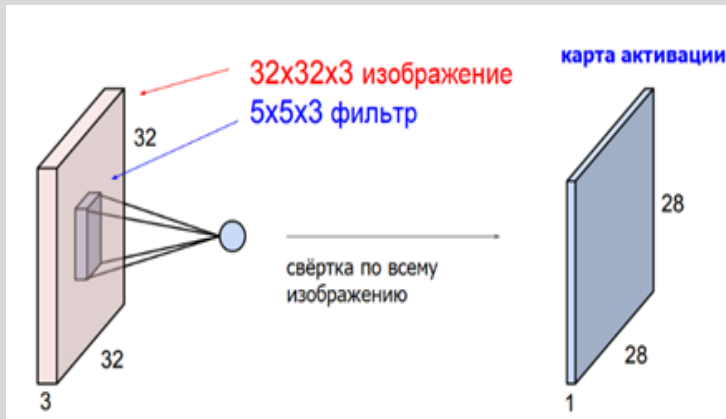
1. Один из лучших алгоритмов для распознавания и классификации изображений.
2. Устойчивость к сдвигу и повороту изображения.
3. Возможность реализации алгоритмов работы и обучения на графических процессорах.

Недостатки:

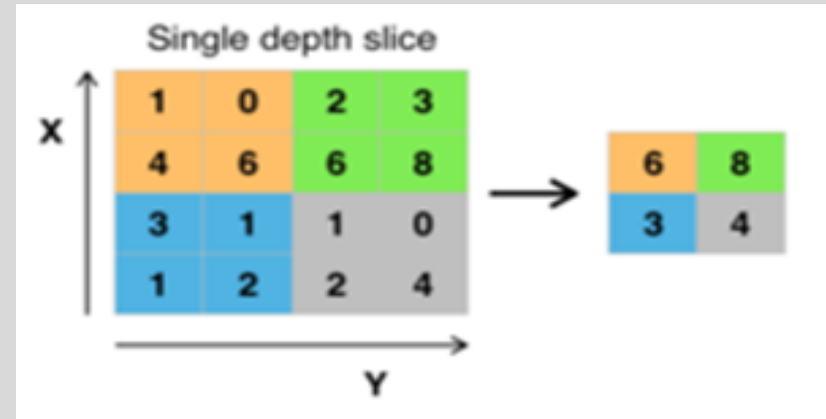
1. Большое количество настраиваемых параметров выбираемые пользователем эмпирически.

СЛОИ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ

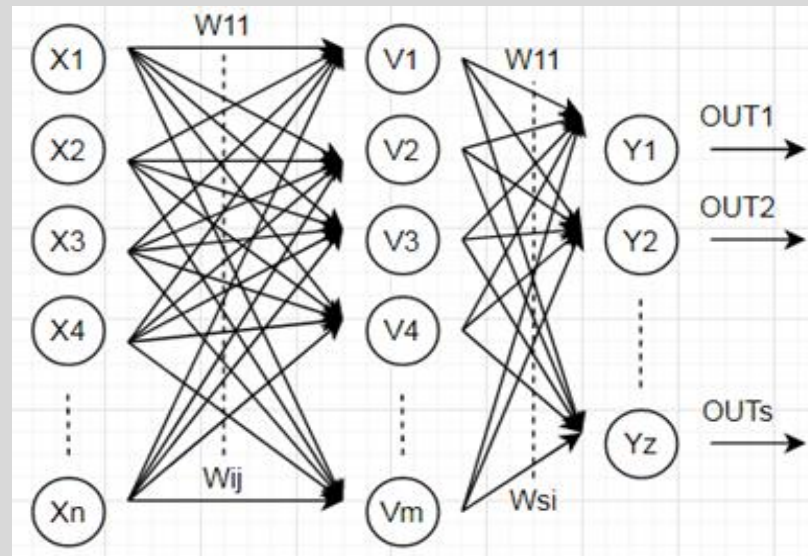
1. Сверточные слои



2. Слои подвыборки



3. Полносвязный слой



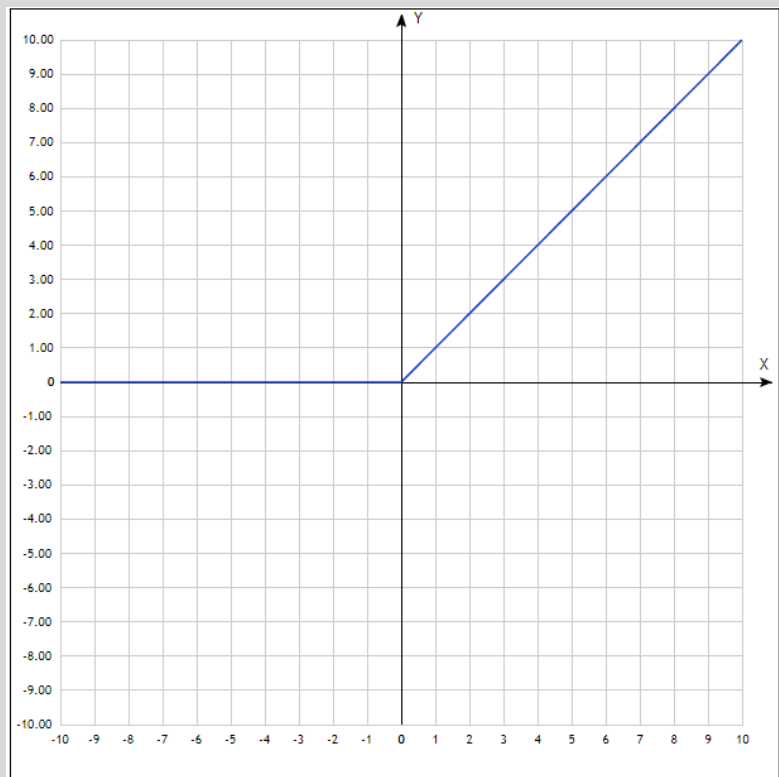
Примеры задач, решаемых с помощью сверточных нейросетей

Задачи, решаемые сверточными нейронными сетями:

- * Определение границ объектов изображения.
- * Определение вектора к нормали реконструирует трёхмерное изображение из двухмерного.
- * Определение объектов внимания - то, на что человек, в первую очередь обратит внимание.
- * Семантическая сегментация. Разделяет объекты на классы по их структуре, ничего не зная об этих объектах.
- * Семантическое выделение границ – это выделение границ, разбитых на классы.
- * Выделение частей тела человека.
- * Распознавание самих объектов на изображении.

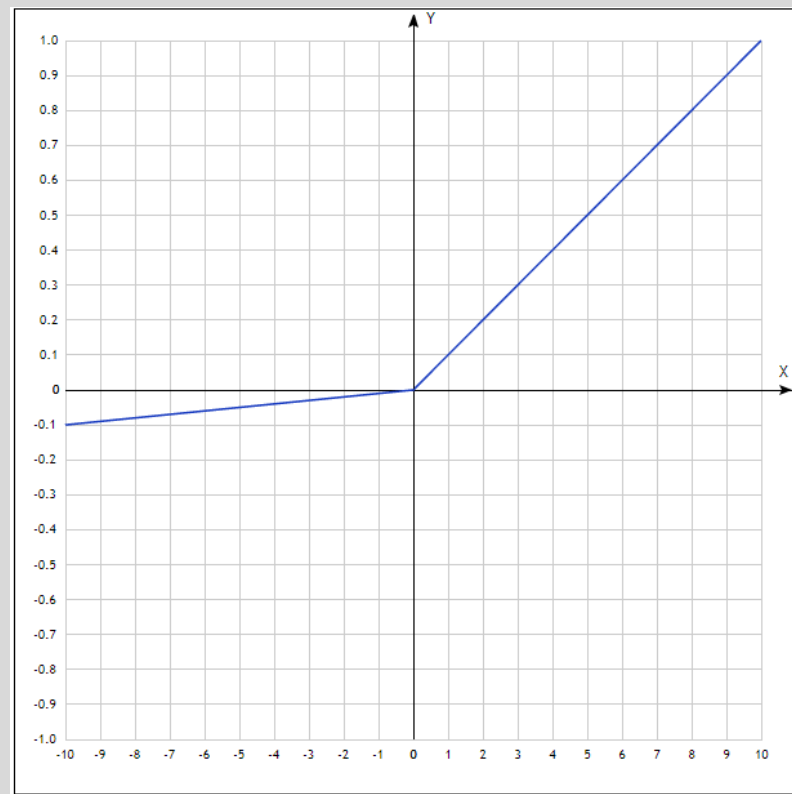
И многие другие...

ФУНКЦИЯ АКТИВАЦИИ ReLU



$$f(x) = \max(0, x)$$

ФУНКЦИЯ АКТИВАЦИИ Leaky ReLU



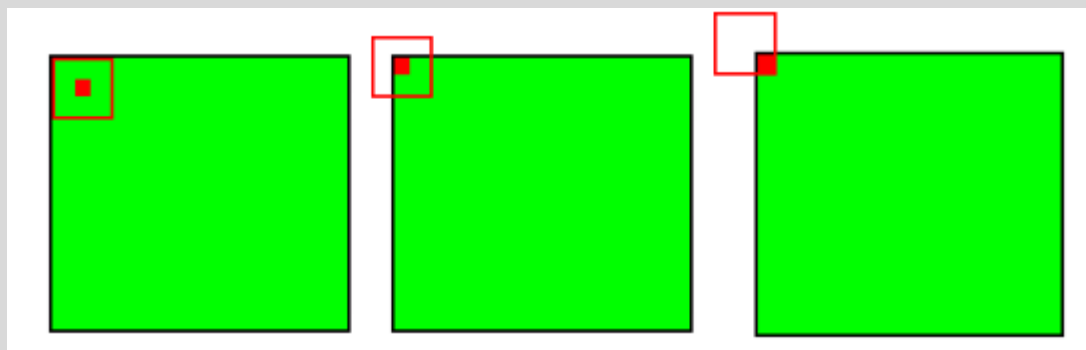
$$f(x) = \max(0.01x, x)$$

Пример свертки изображения до применения функции активации

3_0	3_1	2_2	1	0
0_2	0_2	1_0	3	1
3_0	1_1	2_2	2	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

Обработка краев



$$(f * g)[m, n] = \sum_{k, l} f[m - k, n - l] * g[k, l],$$

где f – исходная матрица изображения;

g – ядро свертки.

w_{22}	w_{21}
w_{12}	w_{11}

rot_180(w)

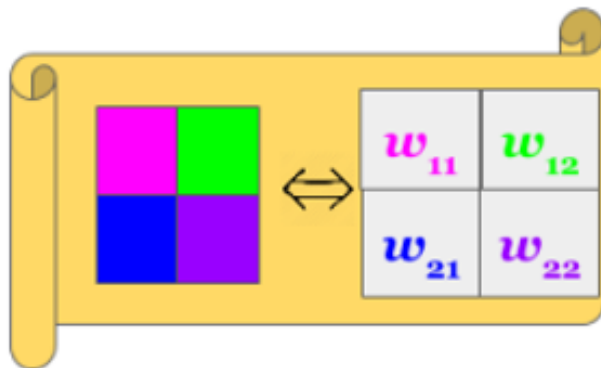
*

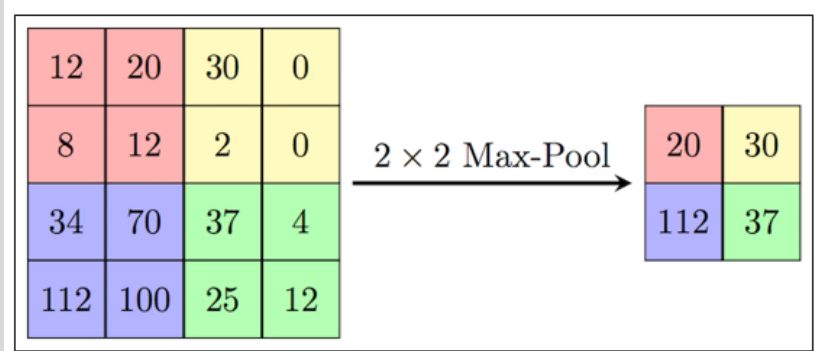
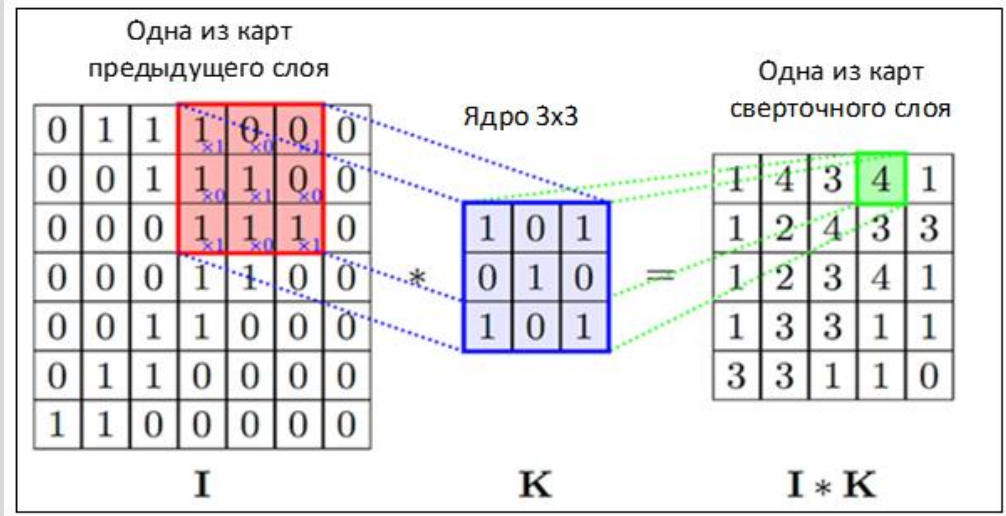
δ_{11}	δ_{12}
δ_{21}	δ_{22}

grads from orange layer

=

$\delta_{11} w_{22}$	$\delta_{11} w_{21} + \delta_{12} w_{22}$	$\delta_{12} w_{21}$
$\delta_{11} w_{12} + \delta_{21} w_{22}$	$\delta_{11} w_{11} + \delta_{12} w_{12} + \delta_{21} w_{21} + \delta_{22} w_{22}$	$\delta_{12} w_{11} + \delta_{22} w_{21}$
$\delta_{21} w_{12}$	$\delta_{21} w_{11} + \delta_{22} w_{12}$	$\delta_{22} w_{11}$





$$x^l = f(a^l * \text{subsample}(x^{l-1}) + b^l),$$

где x^l – выход слоя l ;
 $f()$ – функция активации;
 a^l, b^l – коэффициенты сдвига слоя l ;
 $\text{subsample}()$ – операция выборки локальных максимальных значений.

НЕМНОГО ПРО ТЕНЗОРЫ

p
y
t
h
o
n

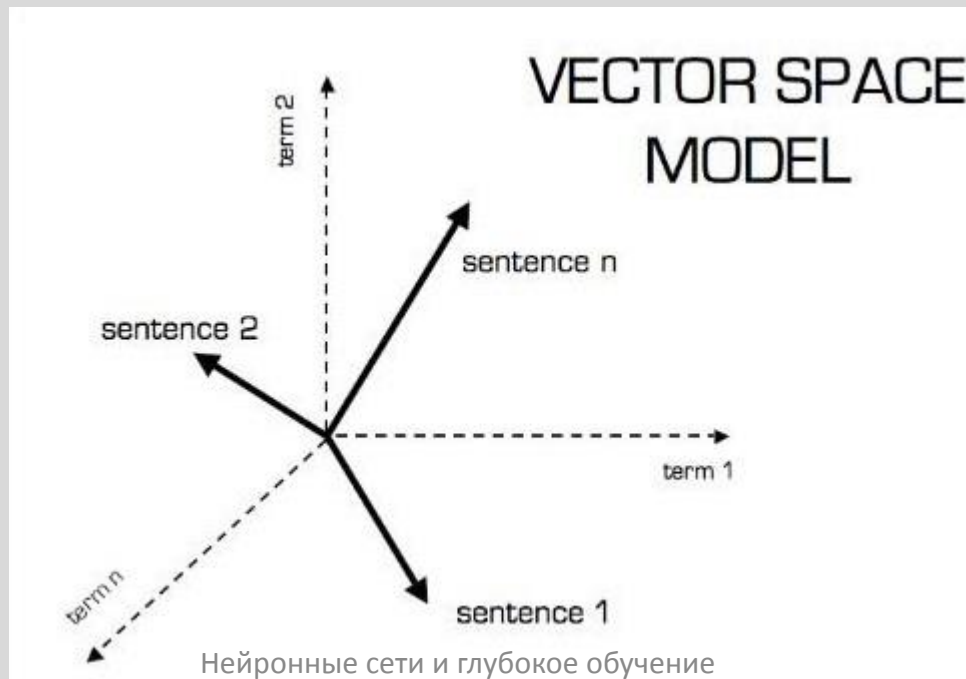
Tensor of one dimension

p	t	a	1
y	e	b	2
t	n	c	3
h	s	d	4
o	o	e	5
n	r	f	6

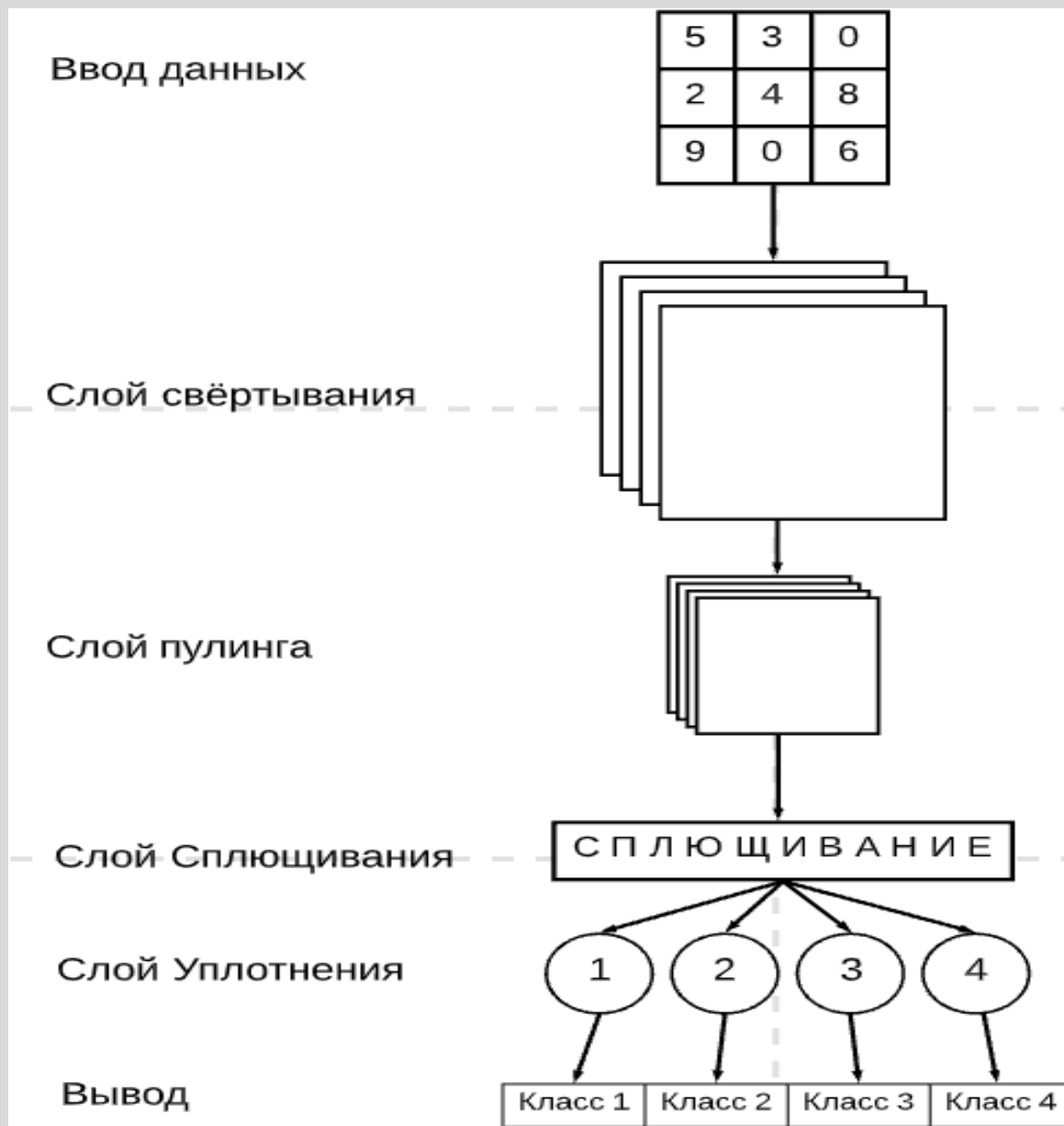
Tensor of two dimensions

	p	t	a	1
p	y	e	b	2
y	t	n	c	3
t	h	s	d	4
h	o	o	e	5
o	n	r	f	6
n	r	f	6	

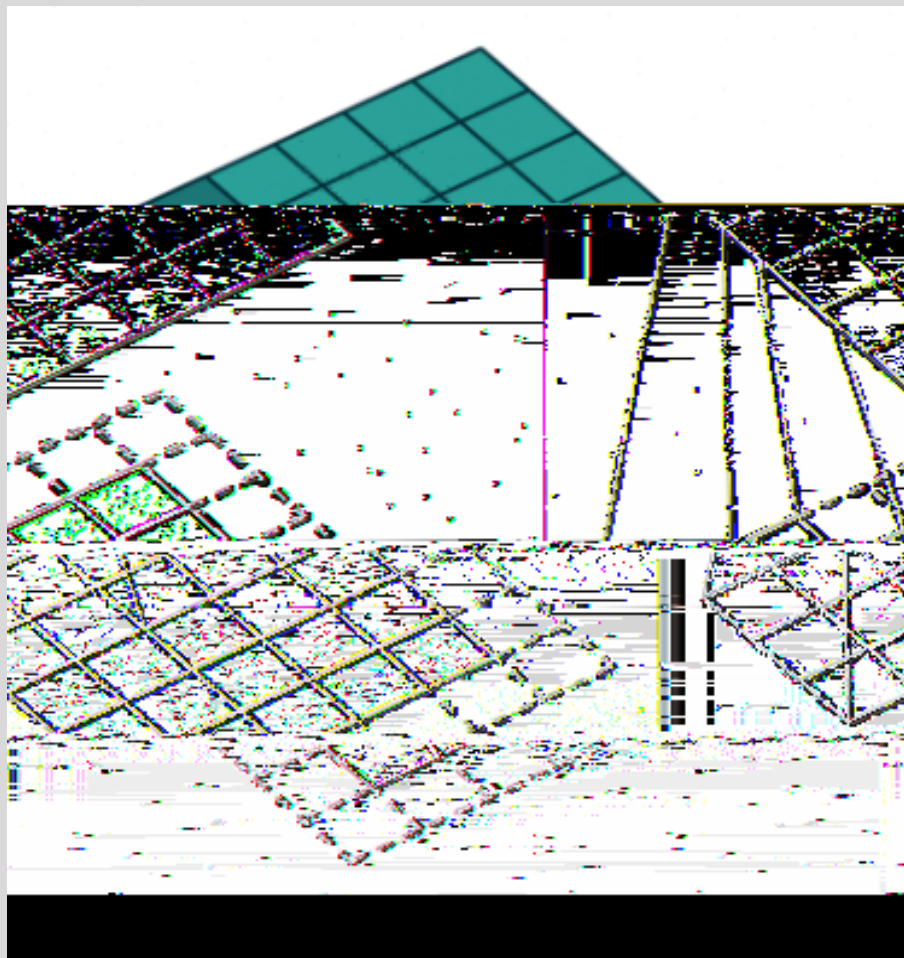
Tensor of three dimensions



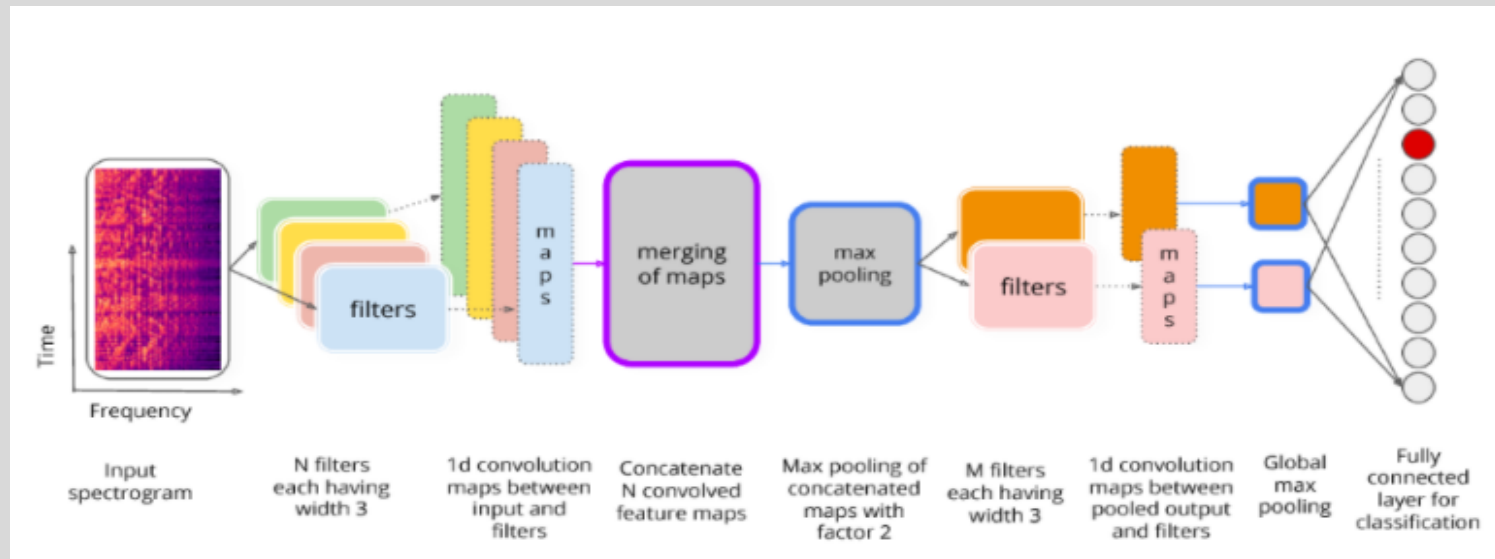
ЕСТЬ И ТАКОЙ ВАРИАНТ ... (из чьей-то курсовой)



Пример использования техники padding



ПРИМЕР ПРЕДСТАВЛЕНИЯ СНС



ОБУЧЕНИЕ СНС

$$E_p = \frac{1}{2} \sum_j (t_{pj} - y_{pj})^2, \quad (2.8)$$

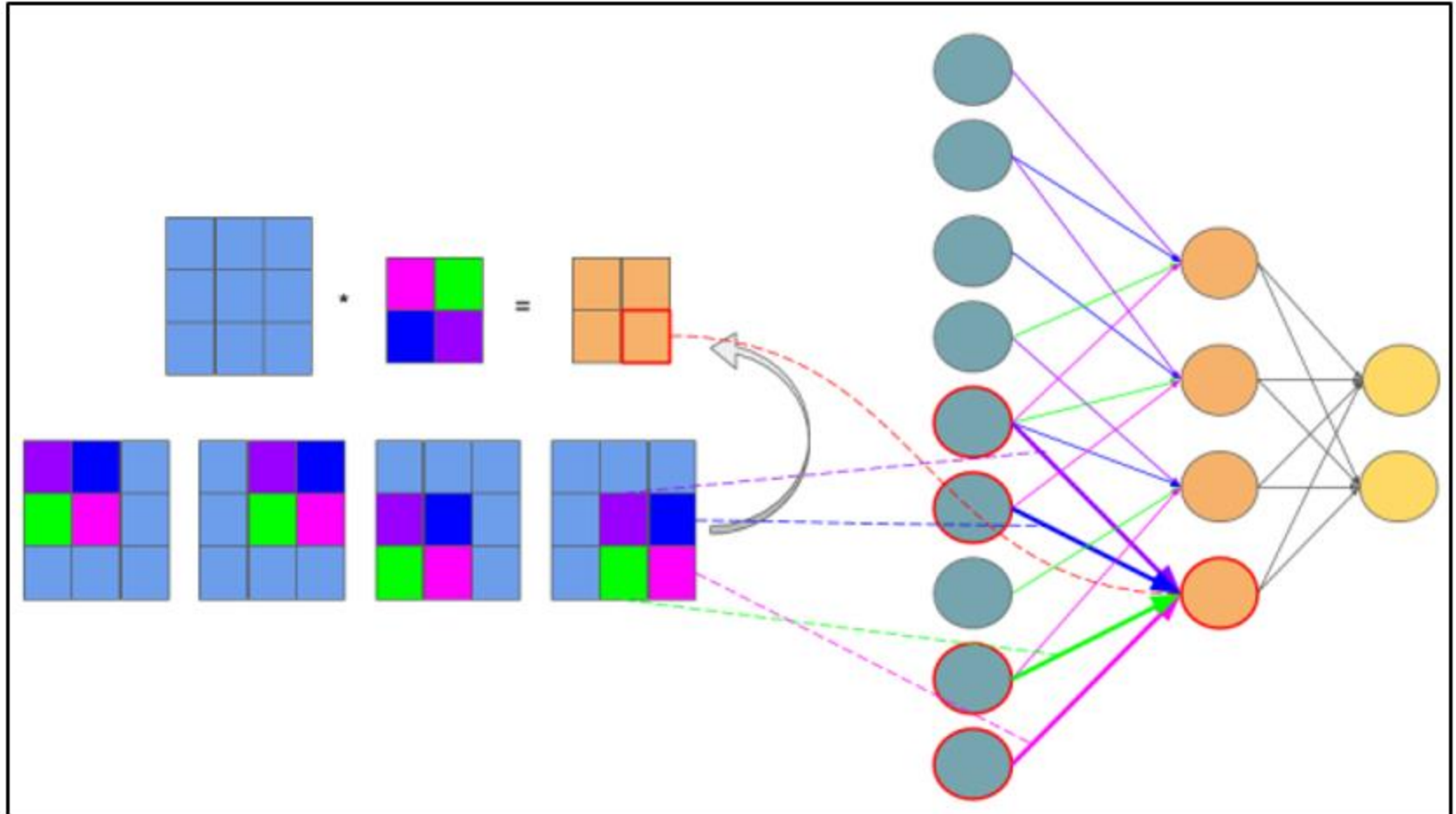
где E_p — величина функции ошибки для образа p ;

t_{pj} — желаемый выход нейрона j для образа p ;

y_{pj} — активированный выход нейрона j для образа p .

А далее... алгоритм обратного распространения ошибки...

ЕЩЕ ОДИН ВАРИАНТ ПРЕДСТАВЛЕНИЯ РАБОТЫ СВЕРТОЧНОЙ НС (автор - ?)



ФУНКЦИЯ ПОТЕРЬ ПРИ РАБОТЕ СНС

$$L = \frac{1}{N} \sum_i L_i(f(x_i, W), y_i)$$

где L_i – это функция потерь;

x_i – входные данные;

W – весовая матрица;

y_i – прогноз для класса;

L – общие потери всех обучающих данных.

Расширенная функция потерь

$$L_i = \sum_{j \neq y_i} \begin{cases} 0, \text{if } s_{y_i} \geq s_j + 1 \\ s_j - s_{y_i} + 1, \text{otherwise} \end{cases} = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

где s_j – ложное предсказание;

s_{y_i} – истинное предсказание

График точности работы СНС

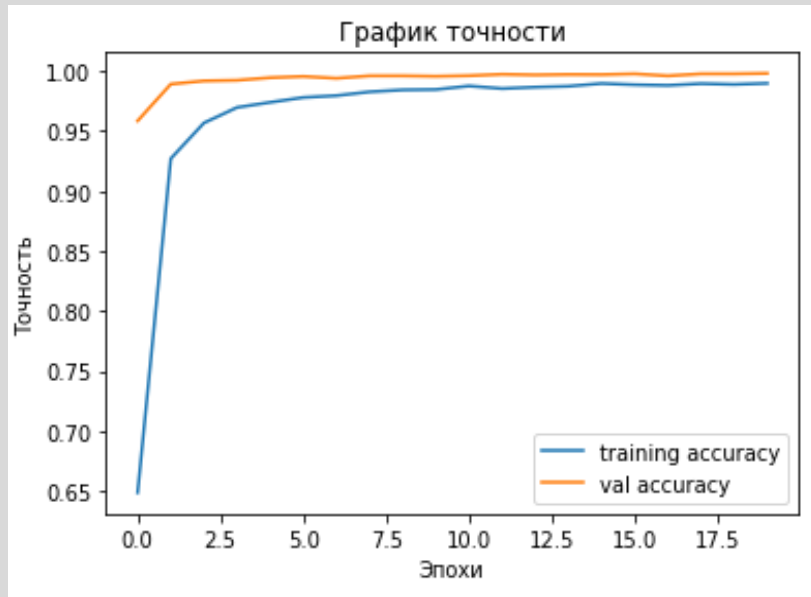
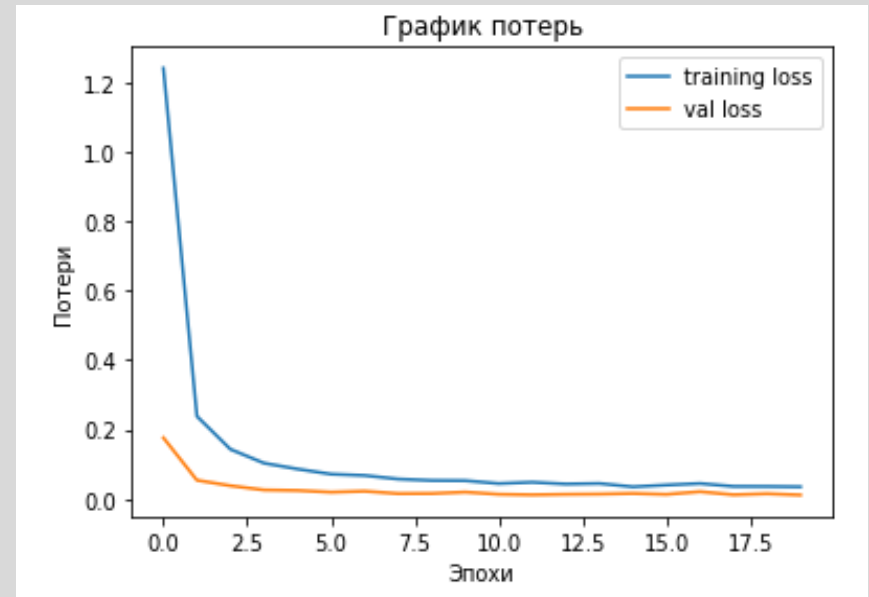


График потерь при обучении СНС

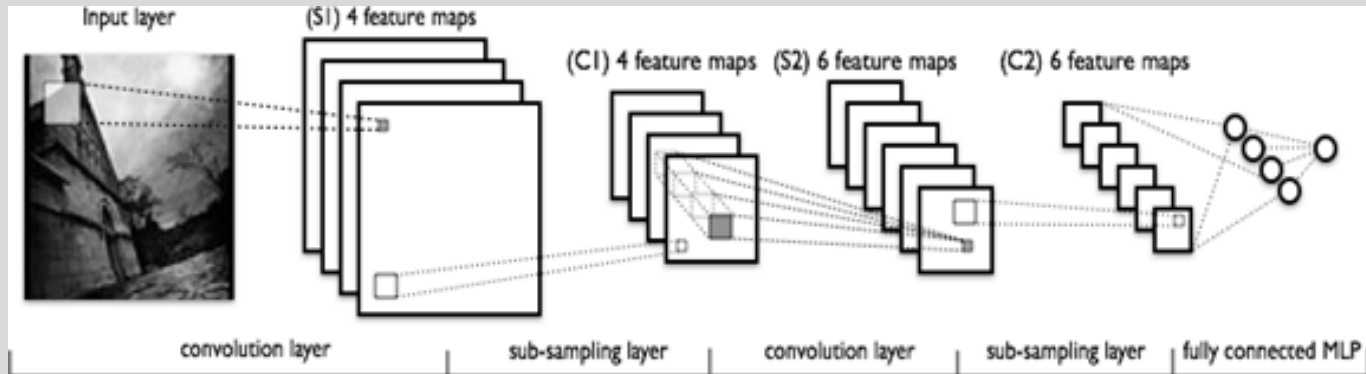


ПОДГОТОВКА ДАННЫХ Процесс Аугментации



Аугментация на примере изображения кошек
(взято <http://robocraft.ru>)

Аугментация данных – метод создания дополнительных данных из уже имеющихся. Есть множество вариантов аугментации, но самыми популярными являются отражение по горизонтали и вертикали, случайное кадрирование, то есть обрезание фотографии со случайным центром и изменение цвета. Так же можно применять различные комбинации этих способов, например, поворот и затем сразу масштабирование и изменять насыщенность пикселей как целого, так и отдельных частей изображения.



$$f(p, min, max) = \frac{p - min}{max - min},$$

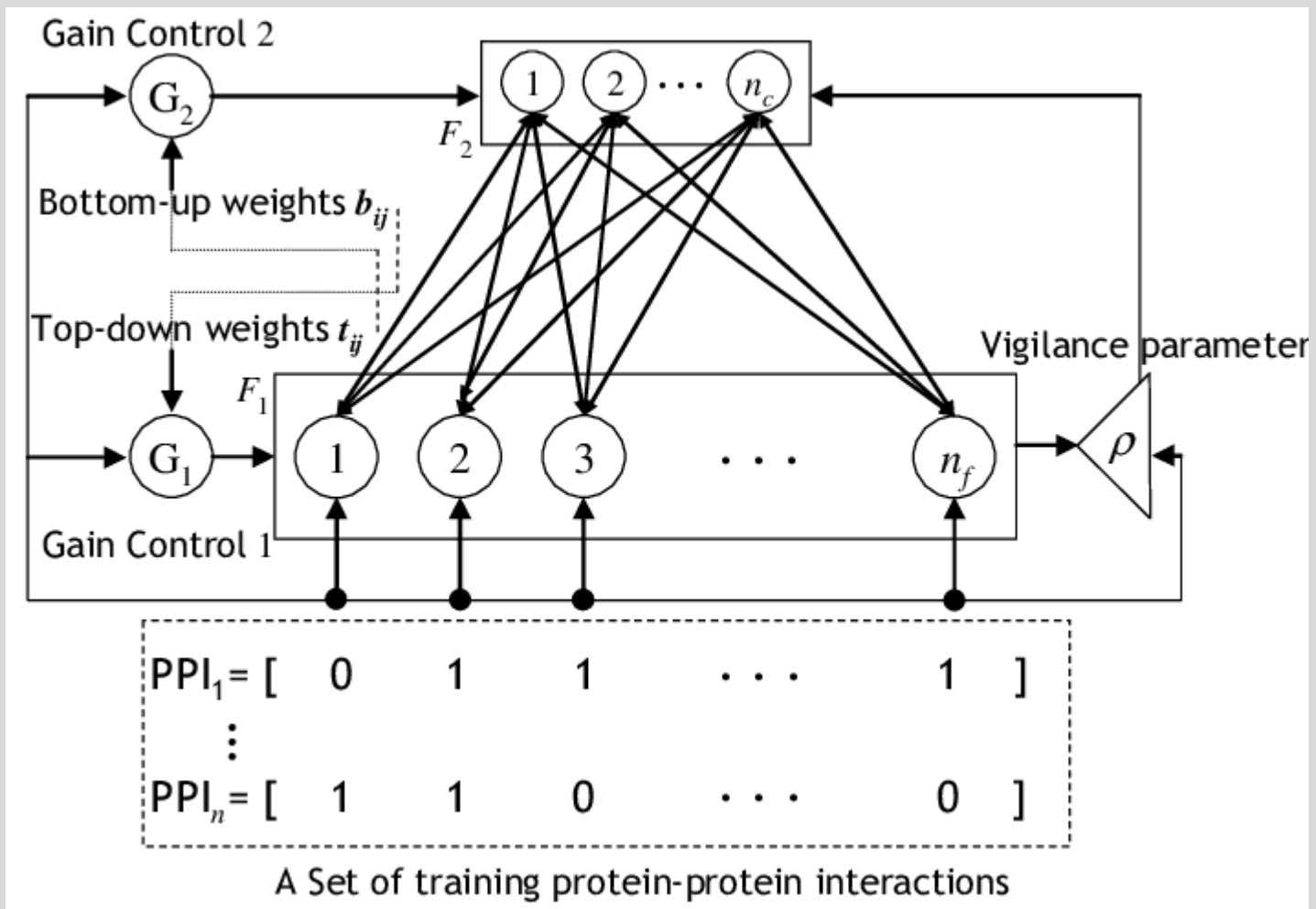
где f – функция нормализации;

p – значение конкретного цвета пикселя от 0 до 255;

min – минимальное значение пикселя – 0;

max – максимальное значение пикселя – 255.

Авторы – Карпентер и Гроссберг



НЕМНОГО О ПРОГРАММИРОВАНИИ НЕЙРОННЫХ СЕТЕЙ

Современный подход

Можно так (библиотека Keras)

Установка:

```
from google.colab import files
files.upload()
!pip install -q kaggle
!mkdir -p ~/.kaggle
!cp kaggle.json ~/.kaggle/
!chmod 600 ~/.kaggle/kaggle.json
```

Загрузка датасета:

```
!kaggle datasets download -d meowmeowmeowmeowmeow/gtsrb-german-traffic-sign
```

```
Downloading gtsrb-german-traffic-sign.zip to /content
 98% 602M/612M [00:06<00:00, 123MB/s]
100% 612M/612M [00:06<00:00, 94.2MB/s]
```

Далее загрузка библиотек (Python)

```
from future import absolute_import, division,
print_function, unicode_literals
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow as tf
import cv2
from PIL import Image
import os
```

Пропустим процесс преобразования файлов в нужный формат...

Перемешиваем данные:

```
s=np.arange(Cells.shape[0])
np.random.seed(43)
np.random.shuffle(s)
Cells=Cells[s]
labels=labels[s]
```

Разделение данных на две подвыборки:

```
(X_train,X_val)=Cells[(int)(0.2*len(labels)):],
Cells[::(int)(0.2*len(labels))]
X_train = X_train.astype('float32')/255
X_val = X_val.astype('float32')/255
(y_train,y_val)=labels[(int)(0.2*len(labels)):],
labels[::(int)(0.2*len(labels))]
```

Преобразование меток изображений в категории:

```
from keras.utils import to_categorical
y_train = to_categorical(y_train, 43)
y_val = to_categorical(y_val, 43)
```

Далее строим архитектуру сети:

```
from keras.models import Sequential
from keras.layers import Conv2D, MaxPool2D, Dense, Flatten, Dropout
model = Sequential()
model.add(Conv2D(filters=32, kernel_size=(5,5), activation='relu', input_shape=X_train.shape[1:]))
model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu'))
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(rate=0.25))
model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu'))
model.add(MaxPool2D(pool_size=(2, 2)))
model.add(Dropout(rate=0.25))
model.add(Flatten())
model.add(Dense(256, activation='relu'))
model.add(Dropout(rate=0.5))
model.add(Dense(43, activation='softmax'))
```

Компилируем модель:

```
model.compile(
    loss='categorical_crossentropy',
    optimizer='adam',
    metrics=['accuracy'])
```

Обучаем нейросеть:

```
epochs = 20
history = model.fit(X_train, y_train, batch_size=32,
    epochs=epochs, validation_data=(X_val, y_val))
```

Все! Ну и вывод графика с таблицей

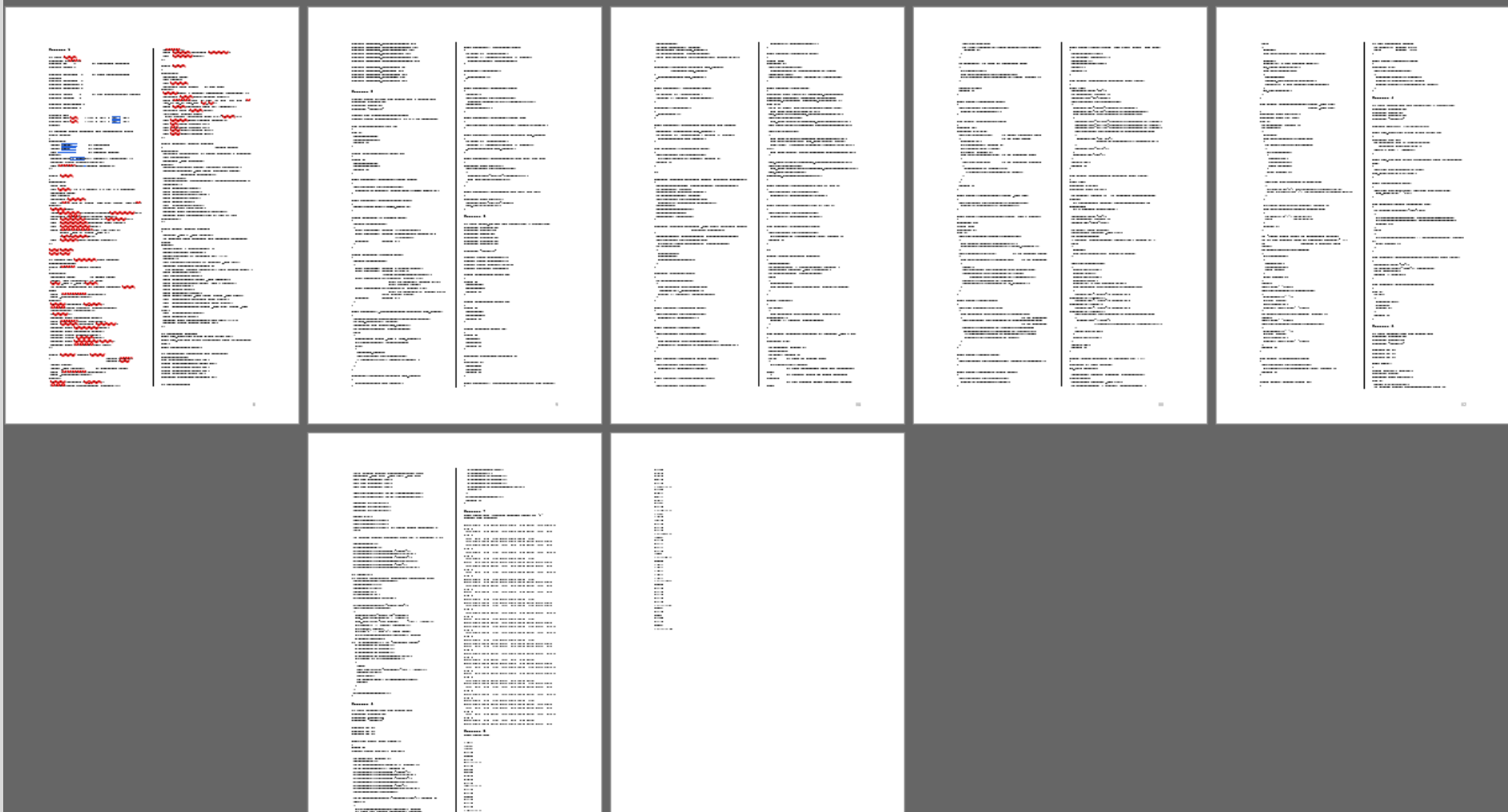
... и получаем результат:

```
Model: "sequential_1"
Layer (type)                Output Shape                Param #
-----
conv2d_1 (Conv2D)           (None, 26, 26, 32)         2432
conv2d_2 (Conv2D)           (None, 24, 24, 64)         18496
max_pooling2d_1 (MaxPooling2 (None, 12, 12, 64)         0
dropout_1 (Dropout)         (None, 12, 12, 64)         0
conv2d_3 (Conv2D)           (None, 10, 10, 64)         36928
max_pooling2d_2 (MaxPooling2 (None, 5, 5, 64)         0
dropout_2 (Dropout)         (None, 5, 5, 64)          0
flatten_1 (Flatten)         (None, 1600)                0
dense_1 (Dense)             (None, 256)                 409856
dropout_3 (Dropout)         (None, 256)                 0
dense_2 (Dense)             (None, 43)                  11051
-----
Total params: 478,763
Trainable params: 478,763
Non-trainable params: 0
```

Использован диплом
Харция М. (КубГТУ, 2020 г.)

А теперь многослойный персептрон на C++

Старый подход



Да-да, если не использовать существующие библиотеки, то очень-очень много строк кода... (на скриншоте код С. Короткого, конец 90-х годов)

НЕМНОГО О ПРОМЫШЛЕННОМ ПРИМЕНЕНИИ СНС

* в 2012 году, Алекс Крижевски благодаря им выиграл конкурс ImageNet (грубо говоря, это ежегодная олимпиада по машинному зрению), снизив рекорд ошибок классификации с 26% до 15%, что тогда стало прорывом.

- **Facebook** использует нейронные сети для алгоритмов автоматического проставления тегов
- **Google** — для поиска среди фотографий пользователя
- * **Amazon** — для генерации рекомендаций товаров
- * **Pinterest** — для персонализации домашней страницы пользователя
- * **Instagram** — для поисковой инфраструктуры

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 5

ЗАДАЧА РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Общие сведения о графических файлах (изображениях)

«Цифровое изображение – это последовательность кадров (или один кадр) в виде дискретного массива точек (т.н. ‘пикселей’), расположенных в памяти либо устройства ввода, либо непосредственно системы технического зрения (СТЗ)»

Выделяют 4 типа изображений: монохромные, полутоновые, в естественных цветах и палитровые

1) Монохромные (другое название - двухградационные) до сих пор часто используются при решении простых прикладных задач, в частности, если требуется определить только наличие какого – либо объекта в поле зрения. Это самое компактное изображение, один пиксель в нем кодируется одним битом! Лучше сказать – кодируется яркость пикселя. Но нередко используют байт.

2) Полутоновые изображения. Здесь яркость пикселя кодируется одним байтом, т.е. можно закодировать 256 оттенков ($2^8 = 256$ или от 0 до 255). В ряде источников [Воротников С.А.] отмечается, что это самый распространенный тип изображений в промышленных и бытовых задачах. Как пример – система видеонаблюдения, работающая ночью.

3) Цветное изображение в естественных тонах. Здесь цвет каждого пикселя сохраняется в так называемом формате RGB – тройки, т.е. требуется 3 байта, с помощью которых можно закодировать примерно 16,8 миллионов различных цветов! Это называется True Color. Этот формат, по сути, наиболее близок к тому, что воспринимают глаза человека – он естественен. Данный тип часто применяется в таких областях, как металлургия, медицина, безопасность и т.д. Но для данного формата уже требуются значительные массивы хранения.

4) Палитровое представление изображения. Можно рассматривать, как упрощенный True Color – он более компактен. Здесь палитра передается с помощью 16 или 256 RGB – троек и цвета здесь определяются косвенно. Цвет пикселя кодируется либо 4, либо 8 битами, при этом используется ссылка на цветовую палитру, а не прямое представление цвета. Это позволяет уменьшить размер изображения, но возникает проблема с подменой цветов – часто встречаются цвета, которых не было в исходном изображении. Как с этим бороться? Поступают просто – преобразовывают в полутоновый формат.

Векторное изображение – это совокупность независимых математических объектов (контуров), при этом каждый из них можно перемещать и масштабировать (DXF, SGO, POV)

Растровое изображение есть набор отдельных пикселей, записанных в единицу объема памяти в виде матрицы (т.н. «битовая карта»). При этом физический размер пикселя напрямую связан с разрешением по полю устройства получения изображения [2], т.е. число пикселей на дюйм (dot per inch – dpi). || *растровые – они реалистичны...*

Отметим существование комбинированного формата изображения, в котором закодирована и векторная и растровая информация – это так называемые метафайлы, среди которых наибольшее распространение получили форматы **CGM** (фирма ISO) и **WMF** (фирма Microsoft)

Структура графического файла

Графический файл – это заголовок и данные.

Структура заголовка:

- Магическое число – стоит вначале файла и указывает формат, допустим, GIF.
- содержание заголовка: формат, тип (монохромный, полутоновое, в естественных цветах);
- использовано ли сжатие;
- позиция видеоданных (если есть) и др.

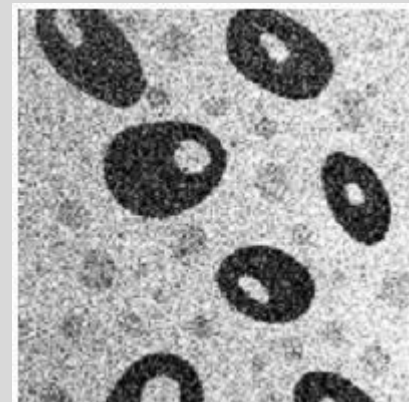
Вопрос обработки изображения при зашумлении

Зашумления условно можно разделить на естественные и искусственные. К естественным можно отнести такие критерии как:

- освещённость;
- затуманенность;
- природные осадки, дождь, снег, туман, морось;
- блики, тени.

К искусственным обычно относят такие критерии как:

- характеристики оптических систем (расфокусировка, замутнённость зеркал и линз, дисторсия);
 - характеристики устройств оцифровки;
 - характеристики электронной регистрирующей аппаратуры;
- характеристики каналов передачи данных.



ВОПРОС ФИЛЬТРАЦИИ ИЗОБРАЖЕНИЯ

Существует достаточно большое множество различных фильтров. В зависимости от типа шума применяется какой-либо тип фильтра. Можно выделить четыре самых распространённых из них:

- сглаживающие фильтры;
- фильтры Винера;
- медианные фильтры;
- ранжирующие фильтры.

Также существуют линейные и нелинейные фильтры.

Так как одним из самых распространённых шумов является белый шум Гаусса, соответственно для шумов такого вида часто применяются линейные фильтры. Среди линейных фильтров можно выделить следующие:

- линейная оконная фильтрация;
- скользящее среднее в окне;
- фильтрация Гаусса.

$$g(x) = ae^{-\frac{(x-b)^2}{2c^2}}$$

БИНАРИЗАЦИЯ ИЗОБРАЖЕНИЯ



Общая схема

«Бинаризация – это один из простых методов препарирования изображения. В общем понимании, после необходимой обработки изображение бесцветным чёрно-белым, у которого убраны лишние детали. Самый важный параметр данного преобразования – это порог. Порогом является критерий проверки интенсивности точки изображения»

МЕТОДЫ БИНАРИЗАЦИИ

Самыми распространёнными методами бинаризации изображения являются:

- глобальный фиксированный порог;
- локальный адаптивный порог;
- метод **Оцу**.

«В отличие от методов глобального и адаптивного порогов, значение порога которых задаётся вручную, в методе Оцу значение порога вычисляется автоматически».

$$\eta(k) = \max_{k=1} \left(\frac{\sigma_{\text{кл}}^2(k)}{\sigma_{\text{общ}}^2} \right)$$

где $\sigma_{\text{кл}}^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2$ – межклассовая дисперсия, а $\sigma_{\text{общ}}^2$ – это общая дисперсия для всего изображения целиком

Некоторые подходы к обработке изображений

Выделим следующие подходы:

- Bayesian – belief propagation (2003 год) // первое применение можно описать так – человек может узнать другого (знакомого) человека, наблюдая только половину лица и данный алгоритм восстанавливает изображение по некоторому фрагменту;

- классификация с помощью стандартного байесовского алгоритма с использованием байесовской нейронной сети, в которой значение синаптической связи интерпретируется, как вероятность.

С помощью данного алгоритма можно достаточно точно рассчитать вероятность, используя как ранее известную информацию, так и данные новых наблюдений (т.е. пополнение датасета). Но данный подход требует большого количества вычислений;

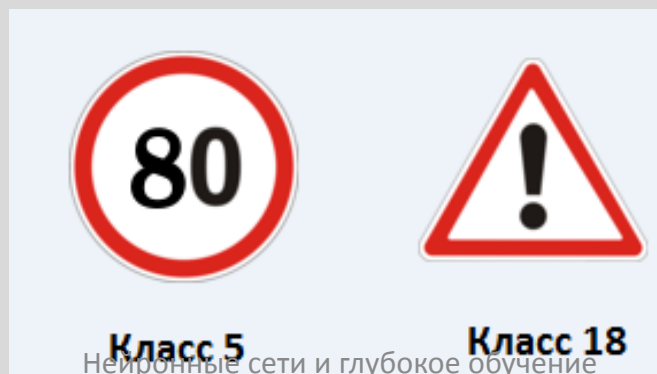
- алгоритм Виолы – Джонса. Датирован 2001 годом, считается «прорывным» в области распознавания (детектирования лиц). Используется технология скользящего окна. Использует примитивы Хоара.

ПРИМЕР ТЕСТОВОЙ ВЫБОРКИ



Дорожные знаки

Классы, которые хуже всего распознаются СНС



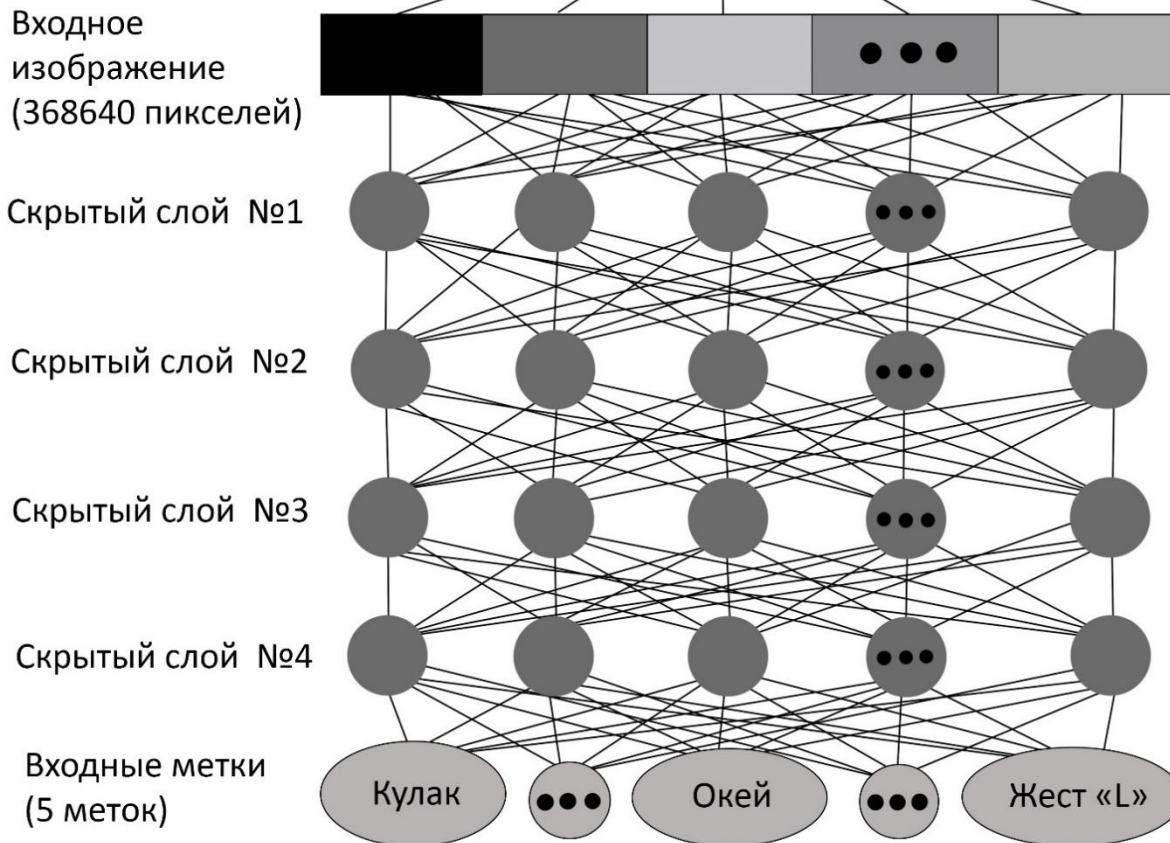
ЗАДАЧА РАСПОЗНАВАНИЯ ЖЕСТОВ РУК

(Пустовалов Вадим, КубГТУ. 2021)



Метка	Класс
0	Кулак
1	L
2	Окей
3	Ладонь
4	Мир

«Метод распознавания жестов для автоматизированной системы управления зданием на основе искусственных нейронных сетей»
(магистерская диссертация)



Пустовалов В. 2021 г.

ЗАДАЧА РАСПОЗНАВАНИЯ ЖЕСТОВ РУК

(Пустовалов Вадим, КубГТУ. 2021)

Метка	Класс	Вероятность
0	Кулак	0.01
1	Жест «L»	0.03
2	Окей	0.93
3	Ладонь	0.02
4	Мир	0.01



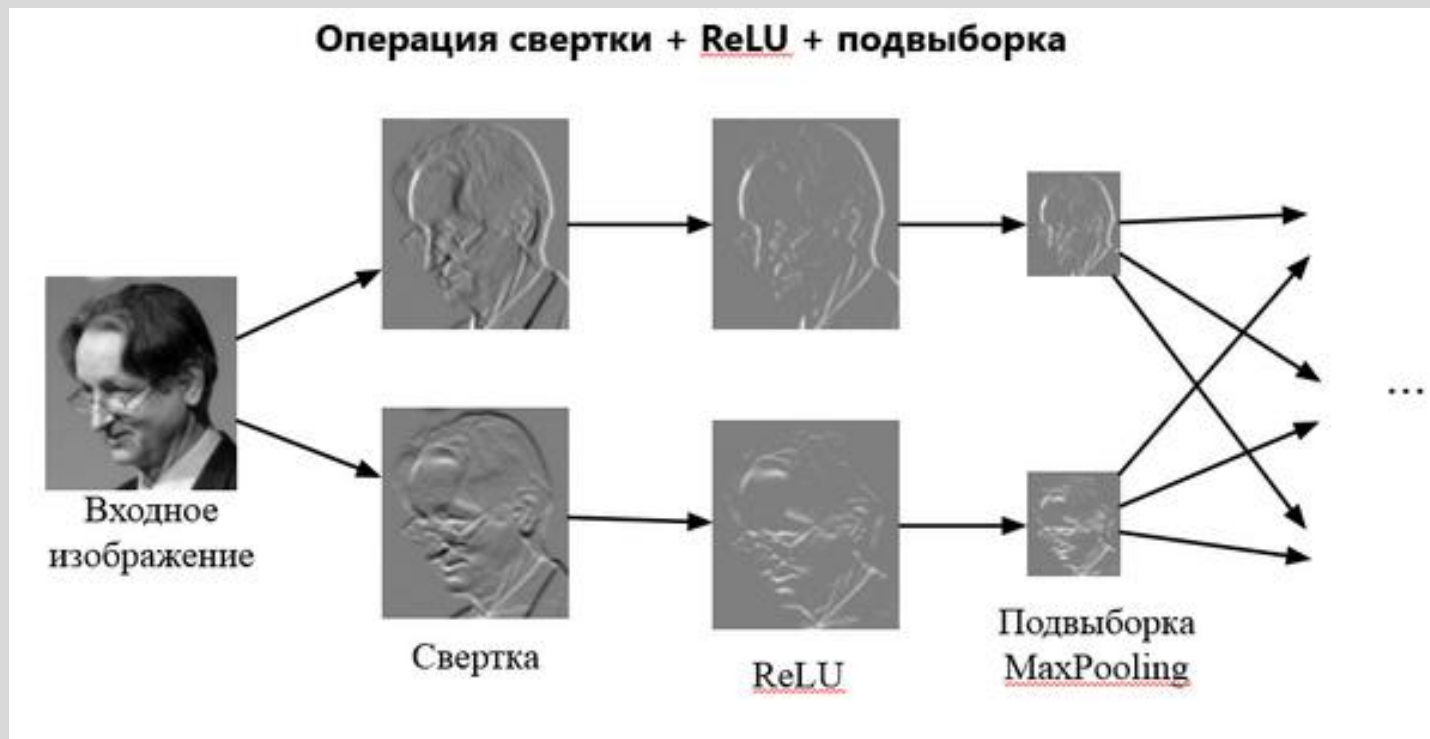
ОБРАБОТКА ДОРОЖНОЙ СИТУАЦИИ



А КАК БЫТЬ НОЧЬЮ??....

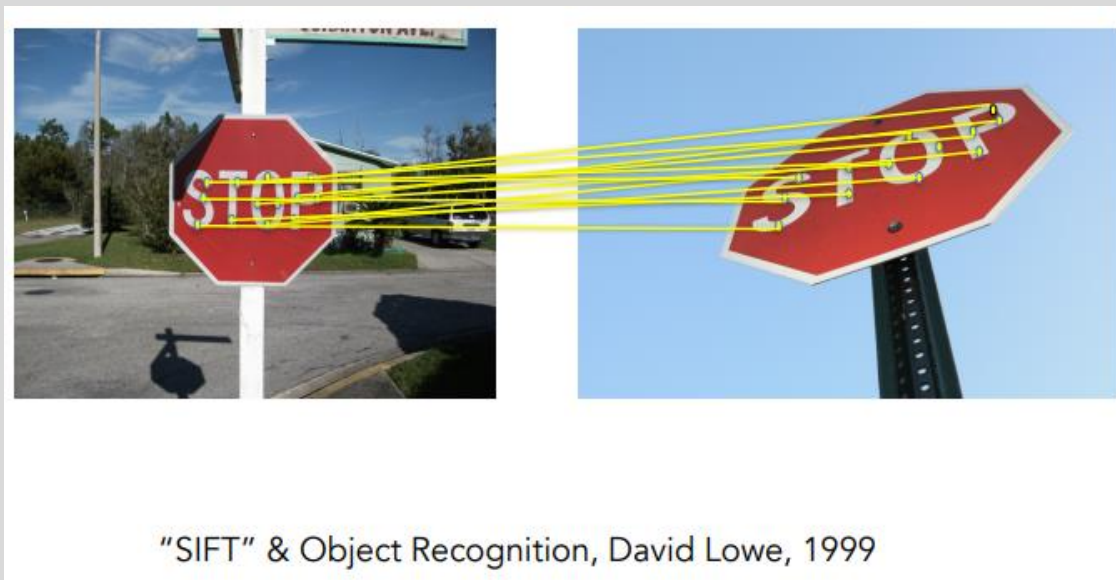


ПРИМЕНЕНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

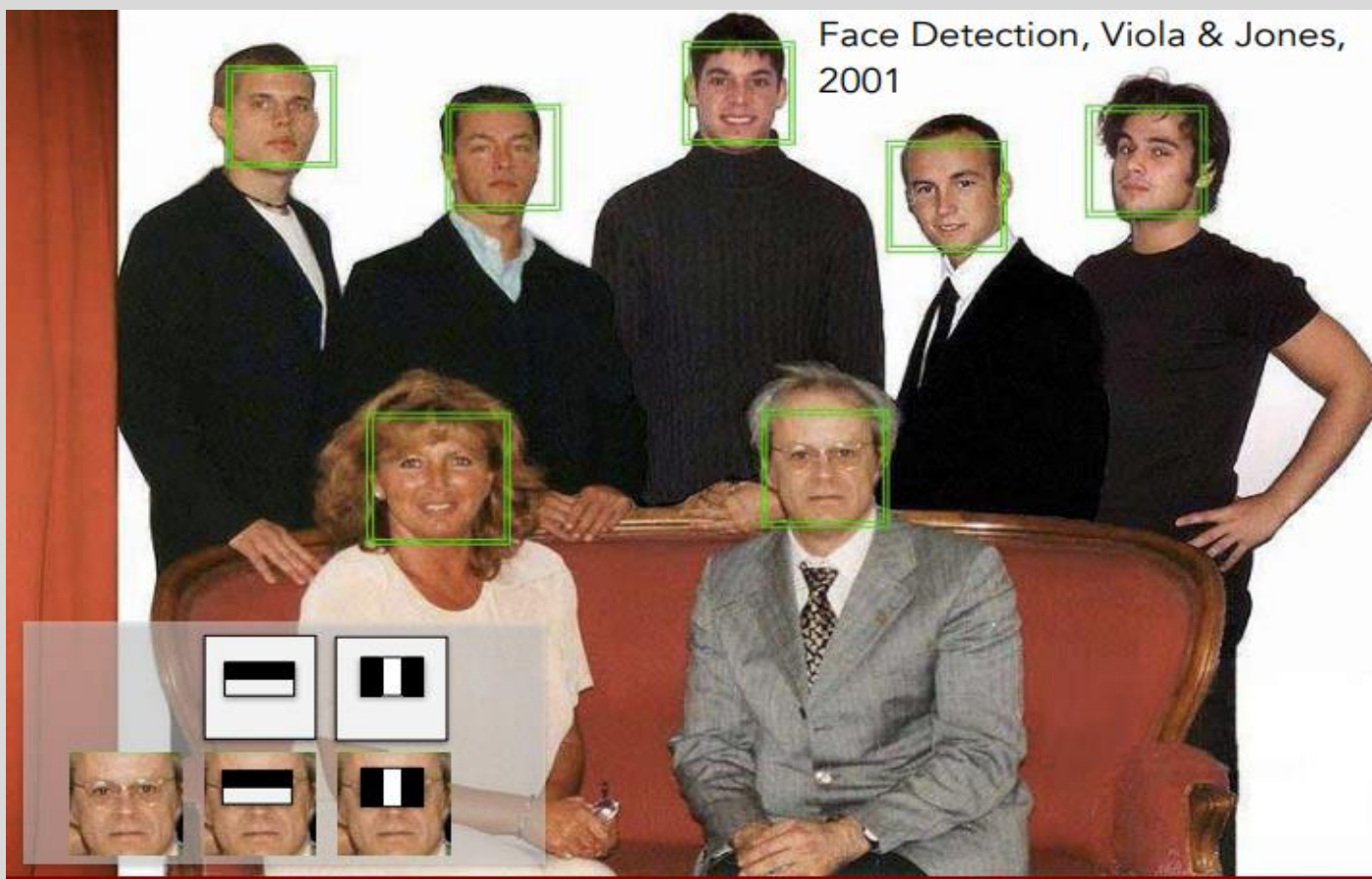


<https://habr.com/ru/post/348000/>

алгоритм **SIFT** Д.Лоу (*масштабно инвариативная трансформация признаков*). Данный алгоритм позволяет находить локальные признаки изображения, которые не зависят от различных изменений (угла съемки, масштаба, освещения) и сопоставлять их с похожими объектами

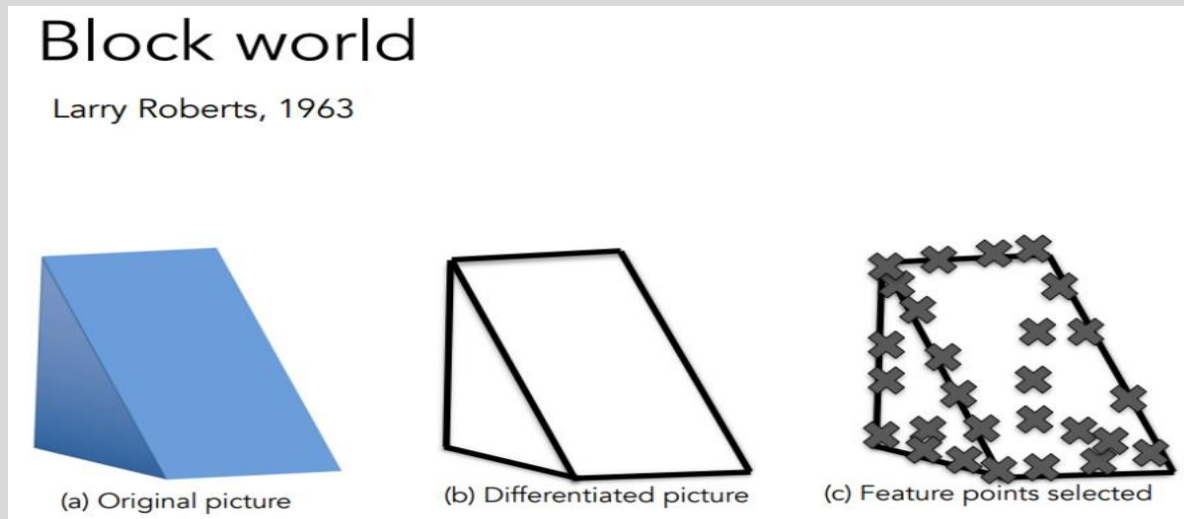


Алгоритм распознавания лиц Виолы-Джонса



ИЗ ИСТОРИИ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Хотя распознаванием изображений, а точнее символов, занимался еще Френк Розенблатт в конце 50-х годов, считается, что история компьютерного зрения началась в начале 60-х годов, когда ученый Ларри Робертс опубликовал набор работ «Block World» (рисунок ниже) в которых говорилось о восстановлении трехмерной геометрии, состоящей из простых сцен.



ЗАДАЧА РАСПОЗНАВАНИЯ ДОКУМЕНТОВ



Водительское удостоверение 93%



Паспорт 86%



ПТС 75%



Страховой полис 96%



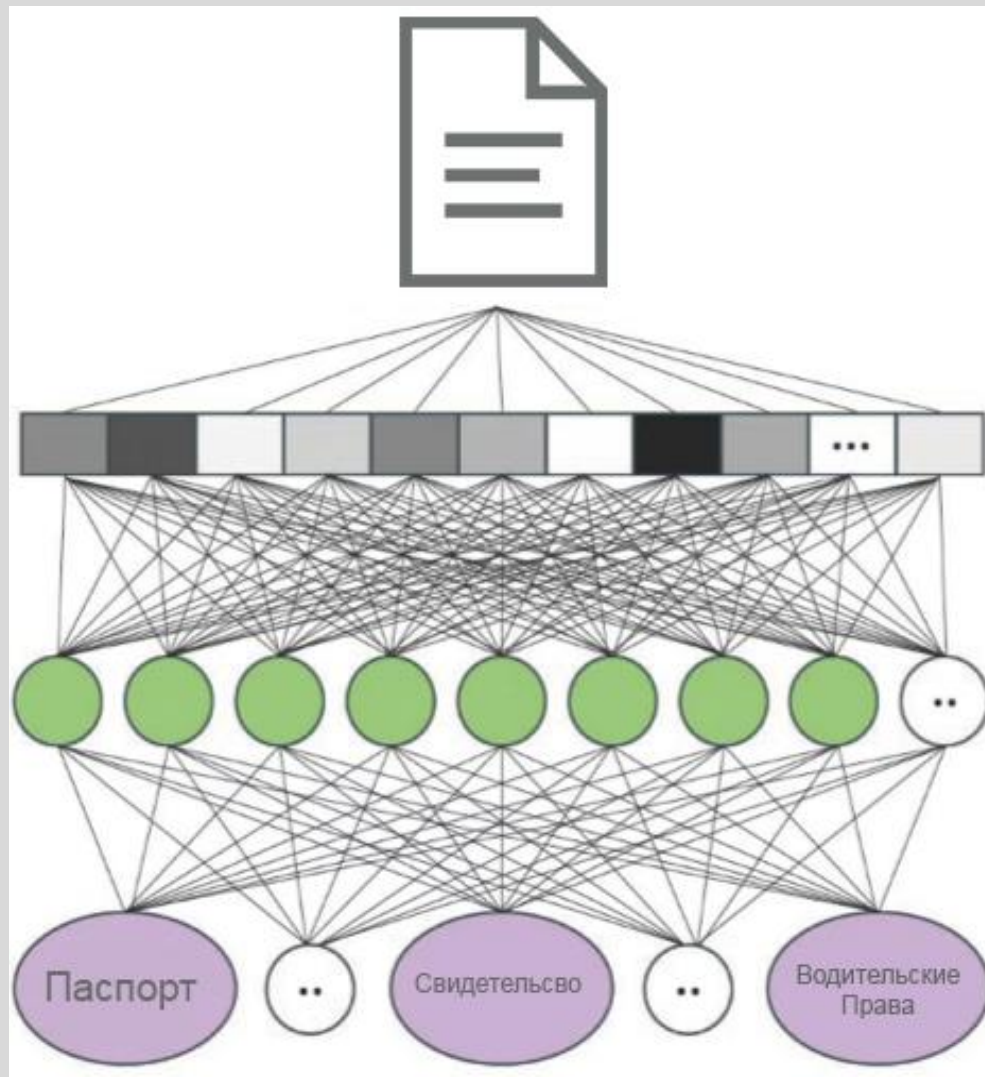
Тех. паспорт 83%



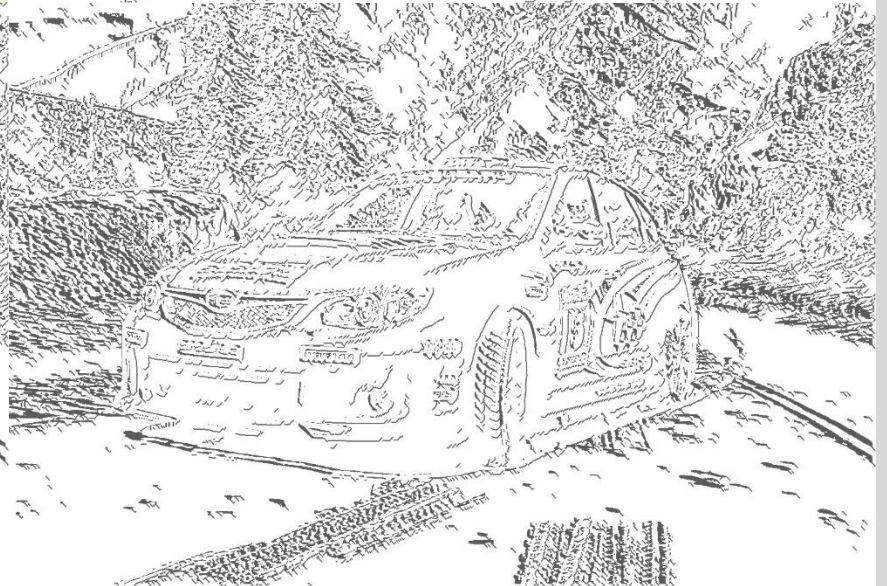
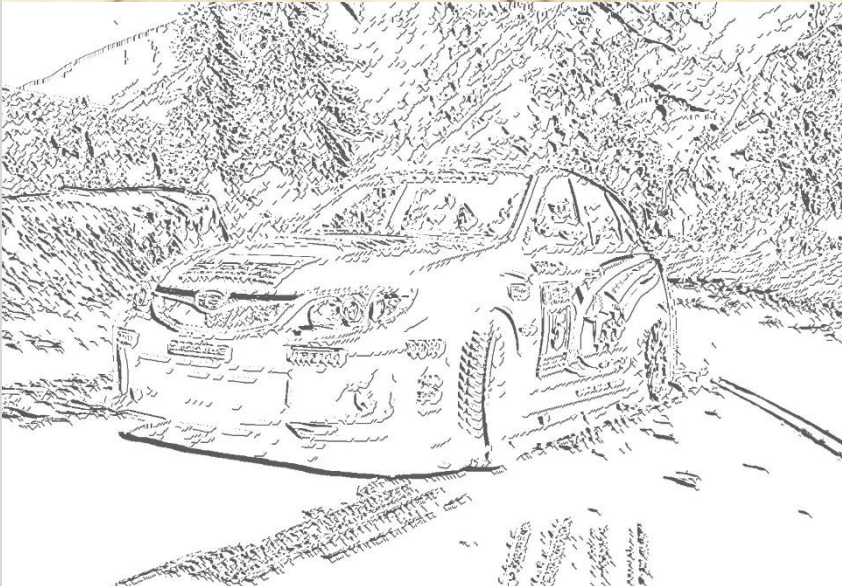
Водительское удостоверение 94%

Метка	Класс
0	Паспорт
1	Водительское удостоверение
2	ПТС
3	Тех. паспорт
4	Страховой полис

ВАРИАНТ НЕЙРОСЕТЕВОЙ ТОПОЛОГИИ ДЛЯ РАСПОЗНАВАНИЯ ДОКУМЕНТОВ



ЗАДАЧА ОТРИСОВКИ КОНТУРА С ПОМОЩЬЮ НЕЙРОСЕТИ



ЛИТЕРАТУРА:

1. Исрафилов Х.С. Исследование методов бинаризации изображений // Вестник науки и образования. – 2017. – №6. – С.44-51.
2. Воротников С.А. Информационные устройства робототехнических систем
3. Прохоренок Н. А. OpenCV и Java. Обработка изображений и компьютерное зрение. — СПб.: БХВ-Петербург, 2018. — 320 с.

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 6

**Задача автоматической обработки текстов:
задача обработки естественных языков,
составление семантических словарей**

СЕМАНТИЧЕСКИЕ ОТНОШЕНИЯ

Синонимия. Синонимами считаются лексические единицы с полностью или частично совпадающими значениями. В первом случае речь идет о явлении точной или полной синонимии, во втором – о квазисинонимии. Примерами полных синонимов в русском языке могут считаться пары "заснуть" – "уснуть", "езде" – "всюду".

Родовидовые отношения (гипер/гипонимия), также известные как отношения типа is-a, связывают лексические единицы, обозначающие родовое понятие (гипероним), с отдельными видами (гипонимами). Такими отношениями связаны пары "собака" – "пудель", "автомобиль" – "внедорожник", "спорт" – "хоккей".

Антонимами называют такие лексические единицы, значения которых противоположны друг другу. Выделяется три основных вида антонимии. 1. Комплементарная, когда один из антонимов исключает присутствие второго ("живой" - "мертвый", "спать" - "бодрствовать"), при этом вместе они полностью охватывают некий смысловой домен (т.е. пары типа "сидеть" – "стоять", "деревянный" - "железный" не подходят, хотя и отвечают первому требованию). 2. Векторная, когда антонимы относятся к разнонаправленным действиям или процессам. Е.g: "приехать" - "уехать", "вспыхнуть" – "потухнуть", "взлететь" – "приземлиться" и т.п. 3. Контрарная – антонимы обозначают две крайние точки одной шкалы: "жара" - "холод", "высокий" – "низкий".

Согласно Википедии

А также - Конверсивность

СИСТЕМЫ АВТОМАТИЧЕСКОЙ ОБРАБОТКИ ТЕКСТОВ (АОТ)

Последовательность действий систем АОТ

Графематический анализ. Выделение из массива данных предложений и слов (токенов).

Морфологический анализ. Выделение грамматической основы слова, определение частей речи, приведение слова к словарной форме.

Синтаксический анализ. Выявление синтаксических связей между словами в предложении, построение синтаксической структуры предложения.

Семантический анализ. Выявление семантических связей между словами и синтаксическими группами, извлечение семантических отношений.

главная

[о нас](#)

[онлайн демо](#)

[скачать](#)

[технологии](#)

[контакт](#)

11 декабря 2021 Добавил 118 тысяч фамилий в морфологический словарь из данных [Дискложес](#)

10 декабря 2021 Выложил на рурі [библиотеку на питоне](#), которая использует морфологию с сайта aot.ru

2 марта 2021 года Из-за отсутствия времени перестали поддерживать бинарные дистрибутивы программ, включая MorphWizard.

11 января 2021 года Исправление ошибок (буква ё и ударения) в русском морф. словаре по [тикету](#)

8 декабря 2020 года С этого дня работает снова [поиск по массиву](#). Я по-прежнему использую сервер СПбГУ

15 июля 2020 года Исходники переехали с sf.net на [github](#)

15 февраля 2020 года К сожалению, мы вынуждены уехать с сервера, который 15 лет предоставлял нам СПбГУ (Спасибо ему, особенно Виктору Захарову). Проект оказался слишком старый даже для университета. Часть демок будет отключена. Этот сайт - уже почти музей.

25 января 2019 года Все управление проектами исходников теперь осуществляется через [смаке](#).

ПОСЛЕДОВАТЕЛЬНОСТЬ ОБРАБОТКИ ТЕКСТА

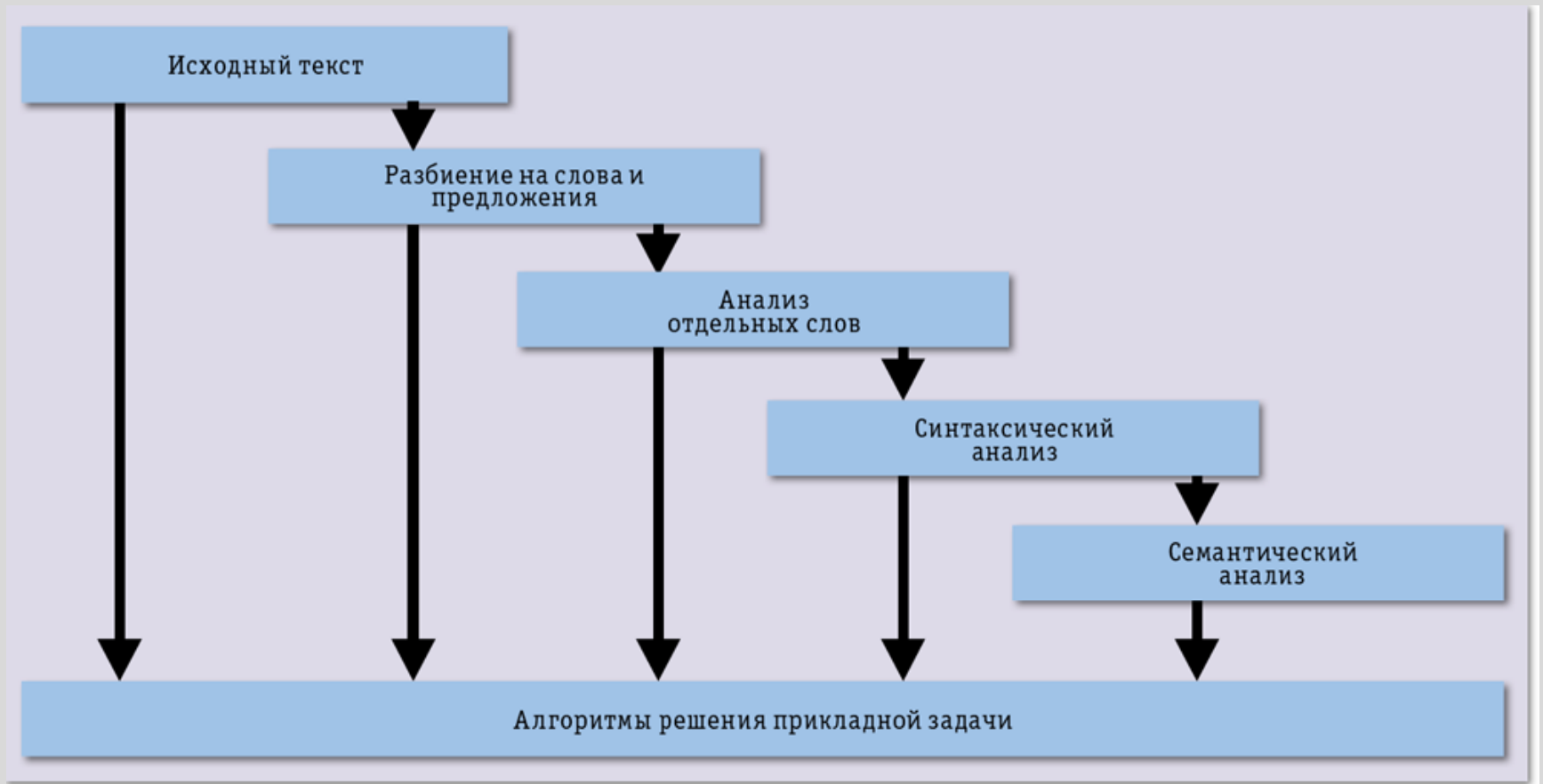
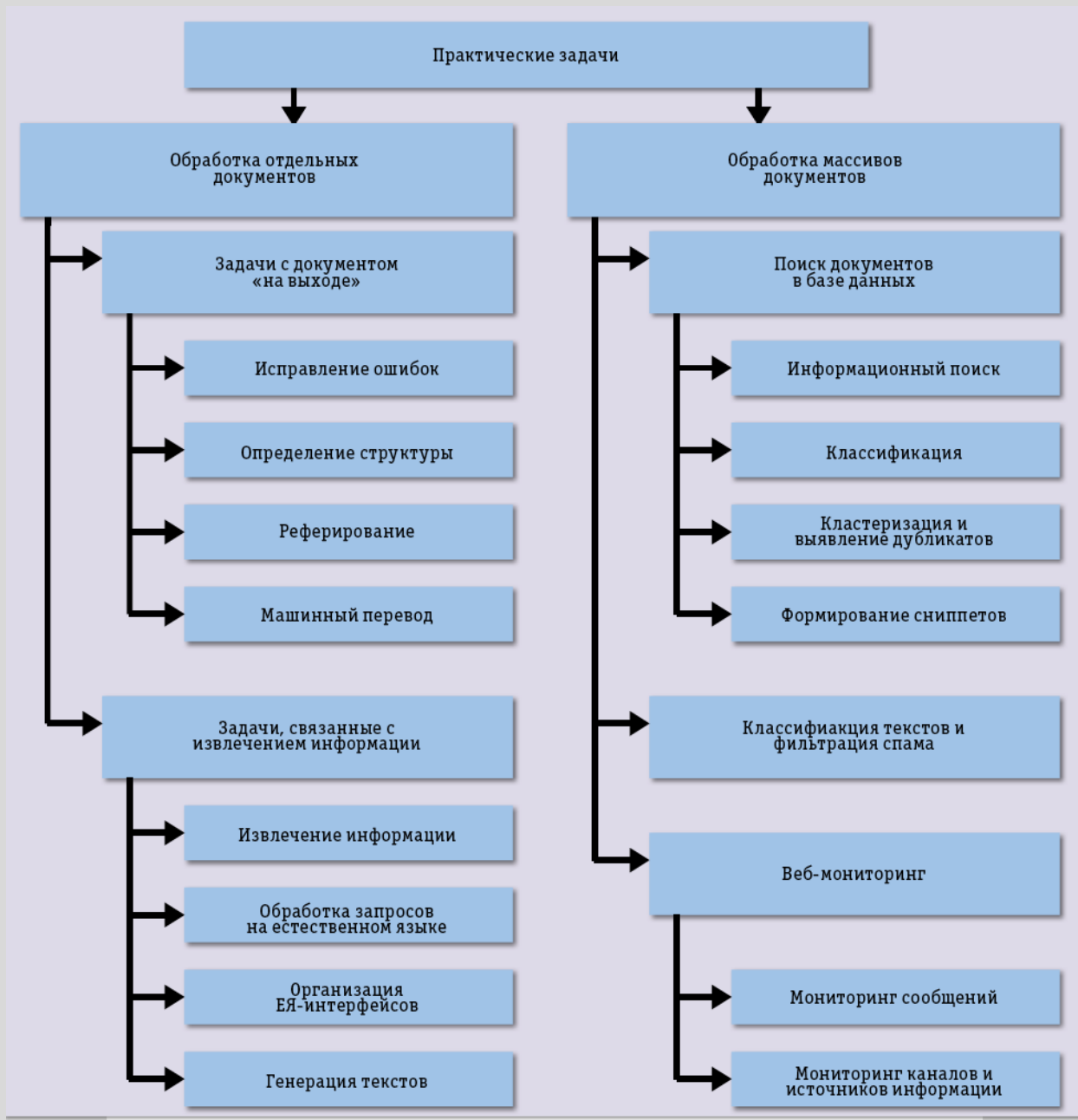
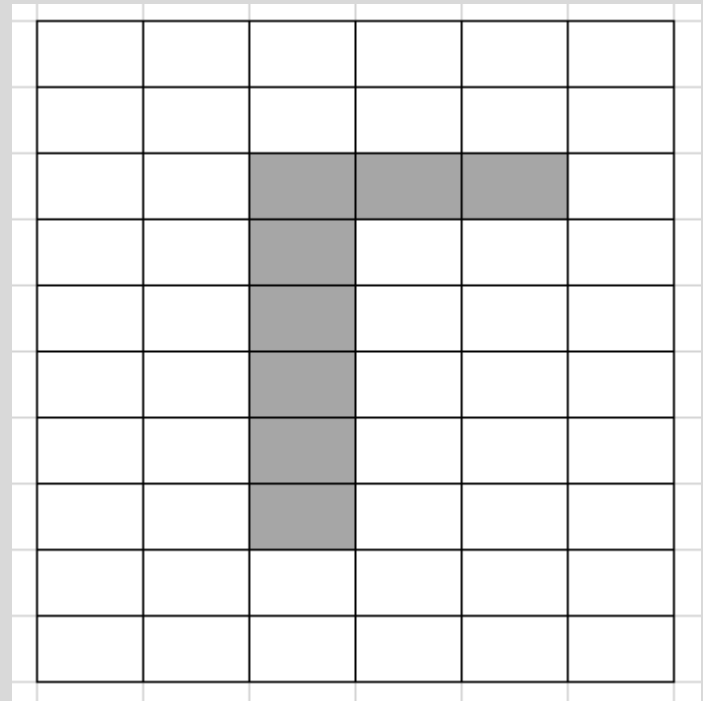
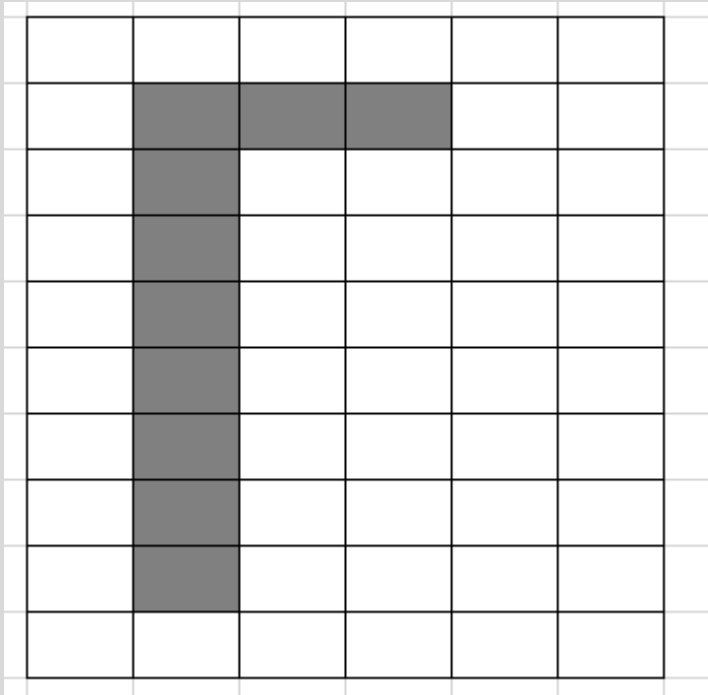


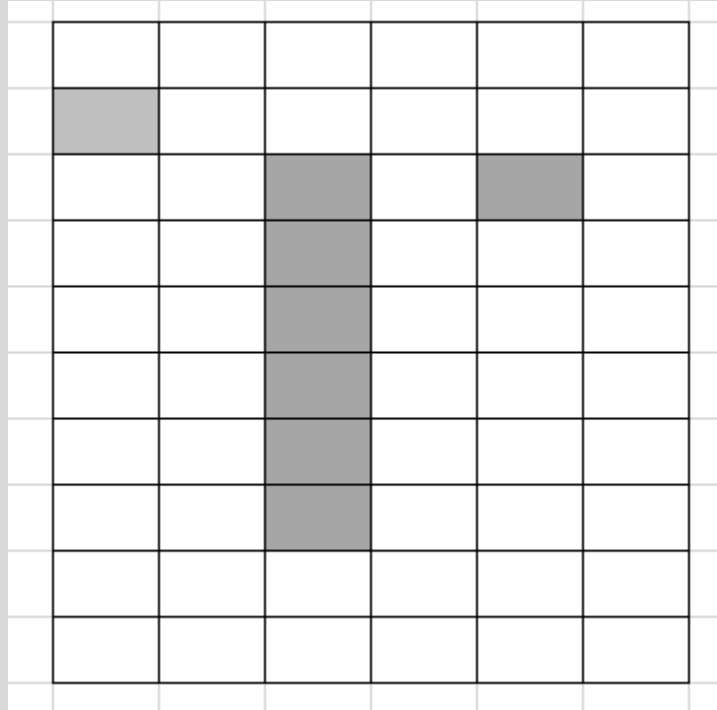
Схема – журнал «Открытые системы»



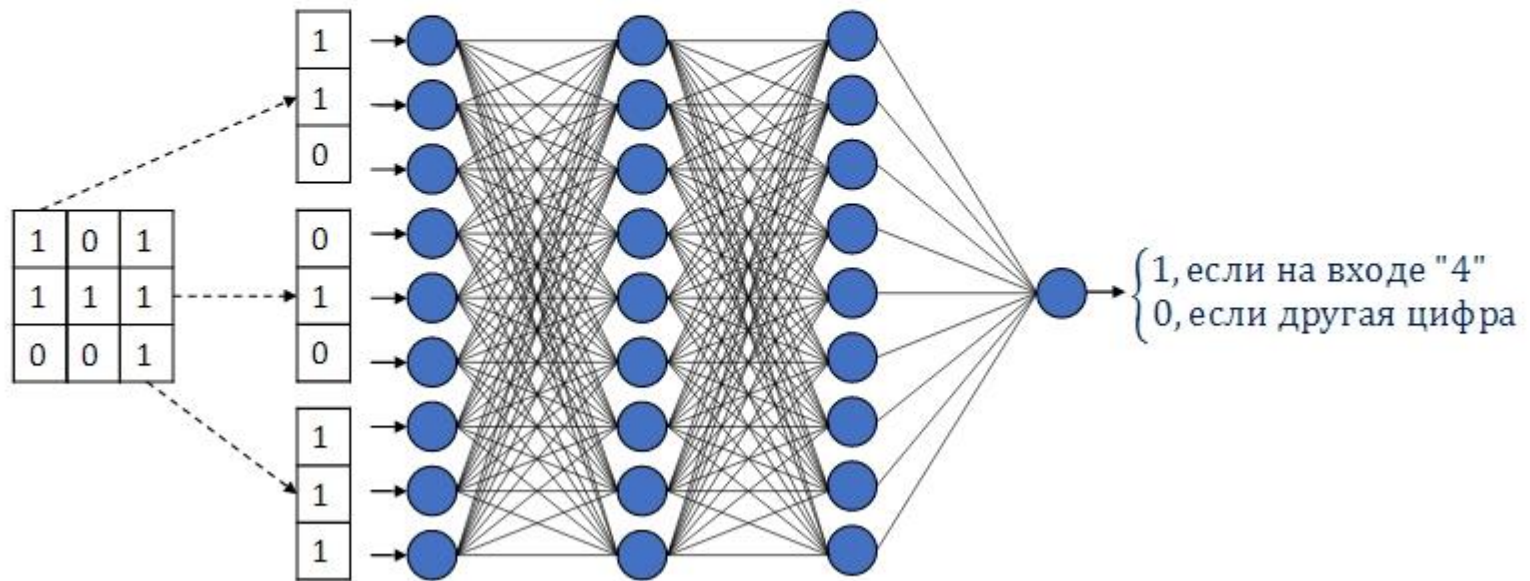
ПРИМЕР БУКВЫ НА ФОТОМАТРИЦЕ (... Ф.Розенблат)



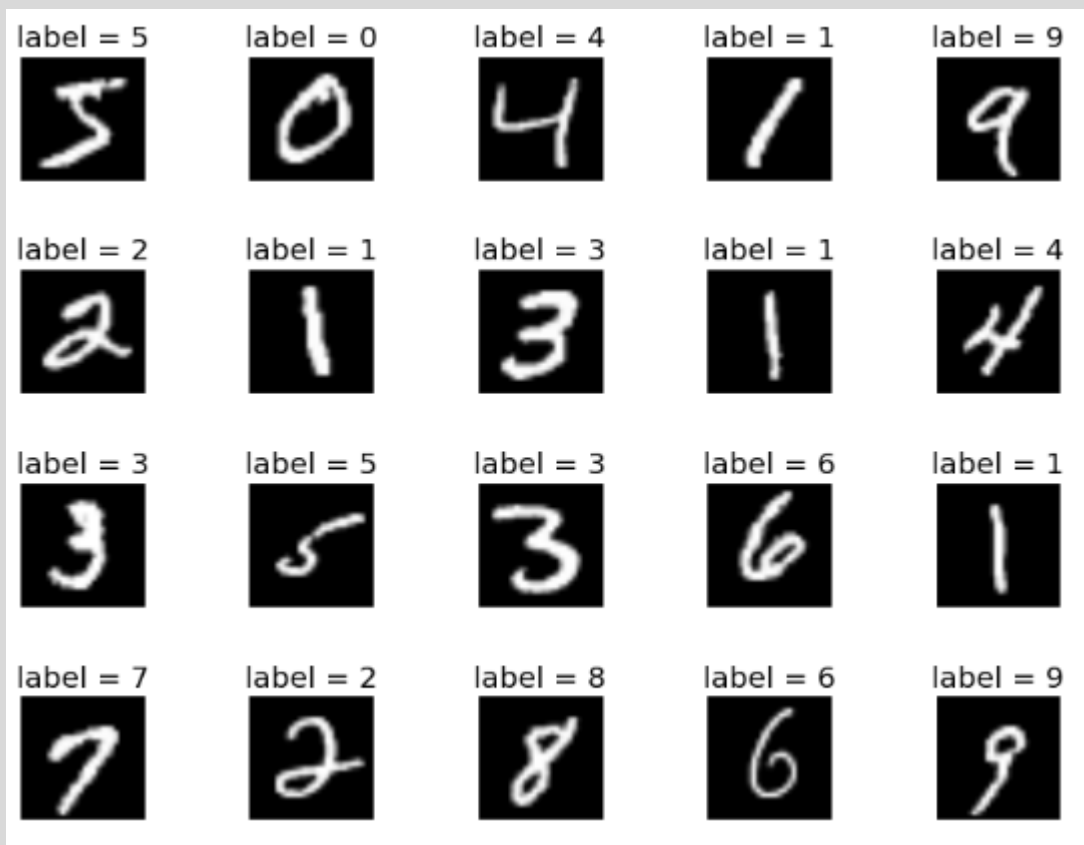
А ЕСЛИ ТАКОЙ НАБОР НА ФОТОМАТРИЦЕ?



Распознавание цифры «4»

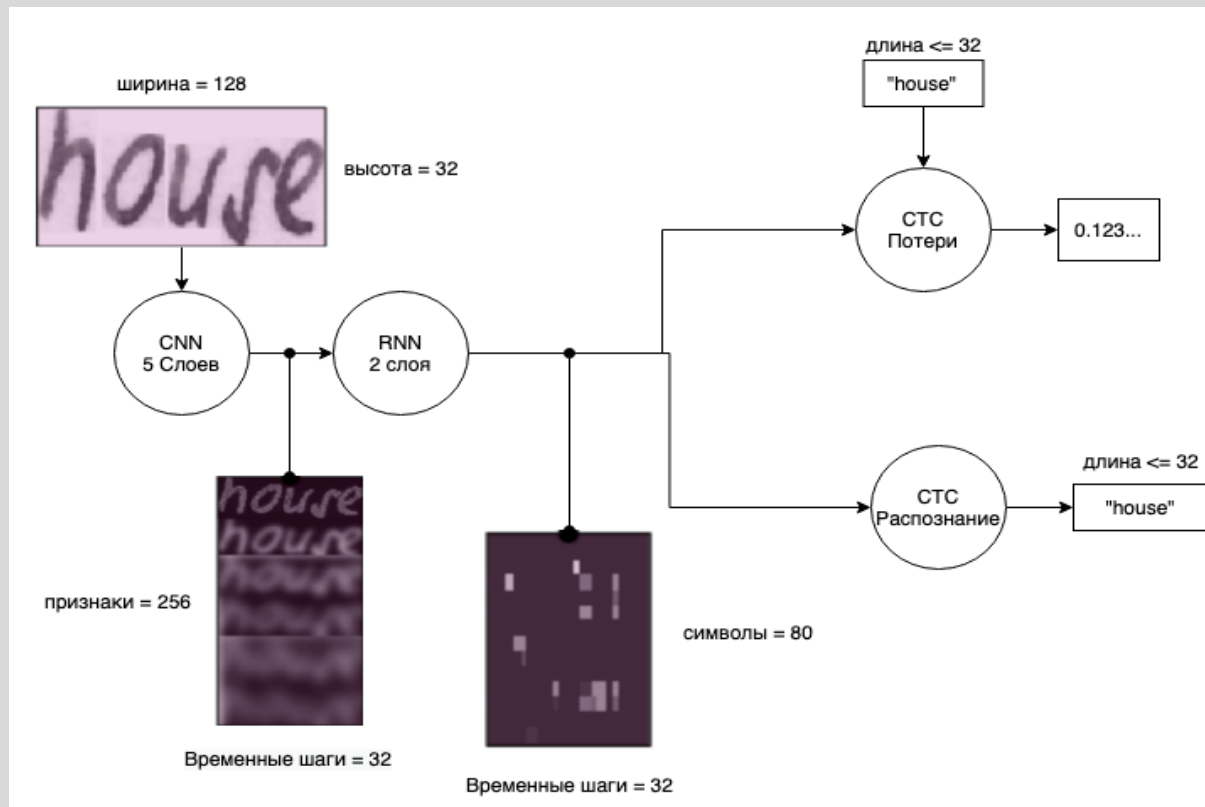


автор - ?



Изображения цифр в библиотеке MNIST

Распознавание рукописного текста сложнее, чем распознавание машинного. Трудности возникают: а) из-за почерка автора, б) из-за дефектов на бумаге, в) ... При этом различают рукописное написание онлайнное и оффлановое. Первое – имеется в виду, что текст написан на экране (здесь лучше сказать – на устройстве ввода) электронного устройства – стилусом или пальцем. В данном случае системе сразу понятно, где пробел, где символ. Второй – имеется в виду, что написано от руки на бумаге.



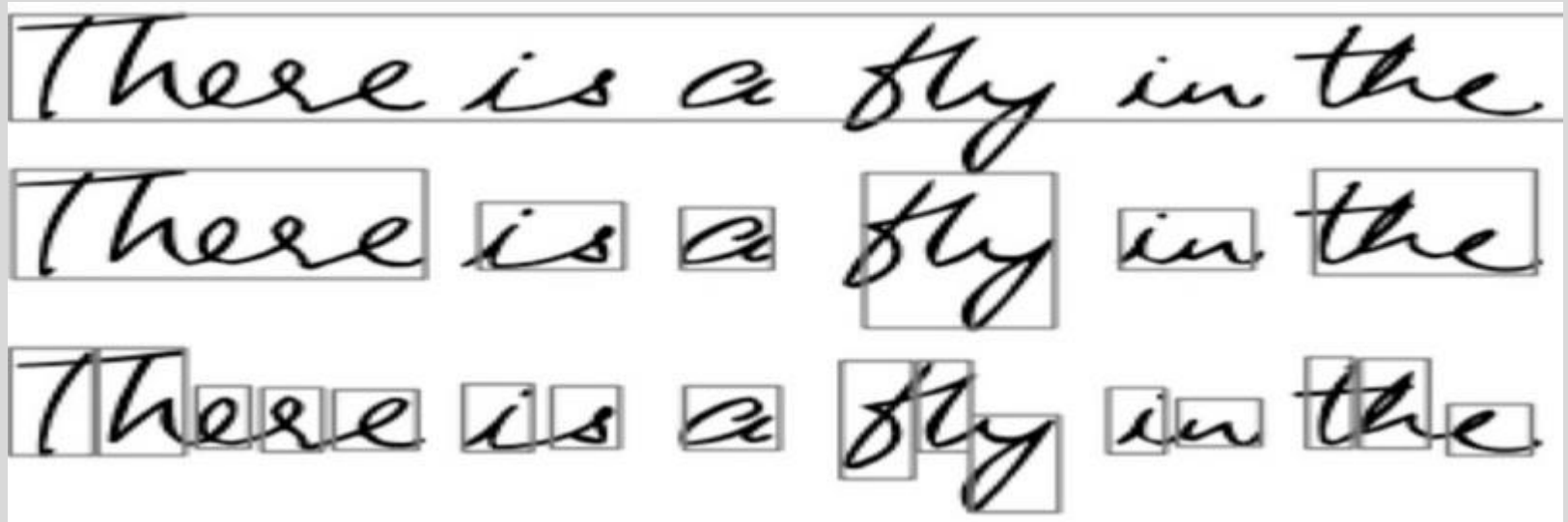
ЗАДАЧА РАСПОЗНАВАНИЯ РУКОПИСНОГО ТЕКСТА



Apple Newton – источник <https://vc.ru/>

Нейронные сети и глубокое обучение

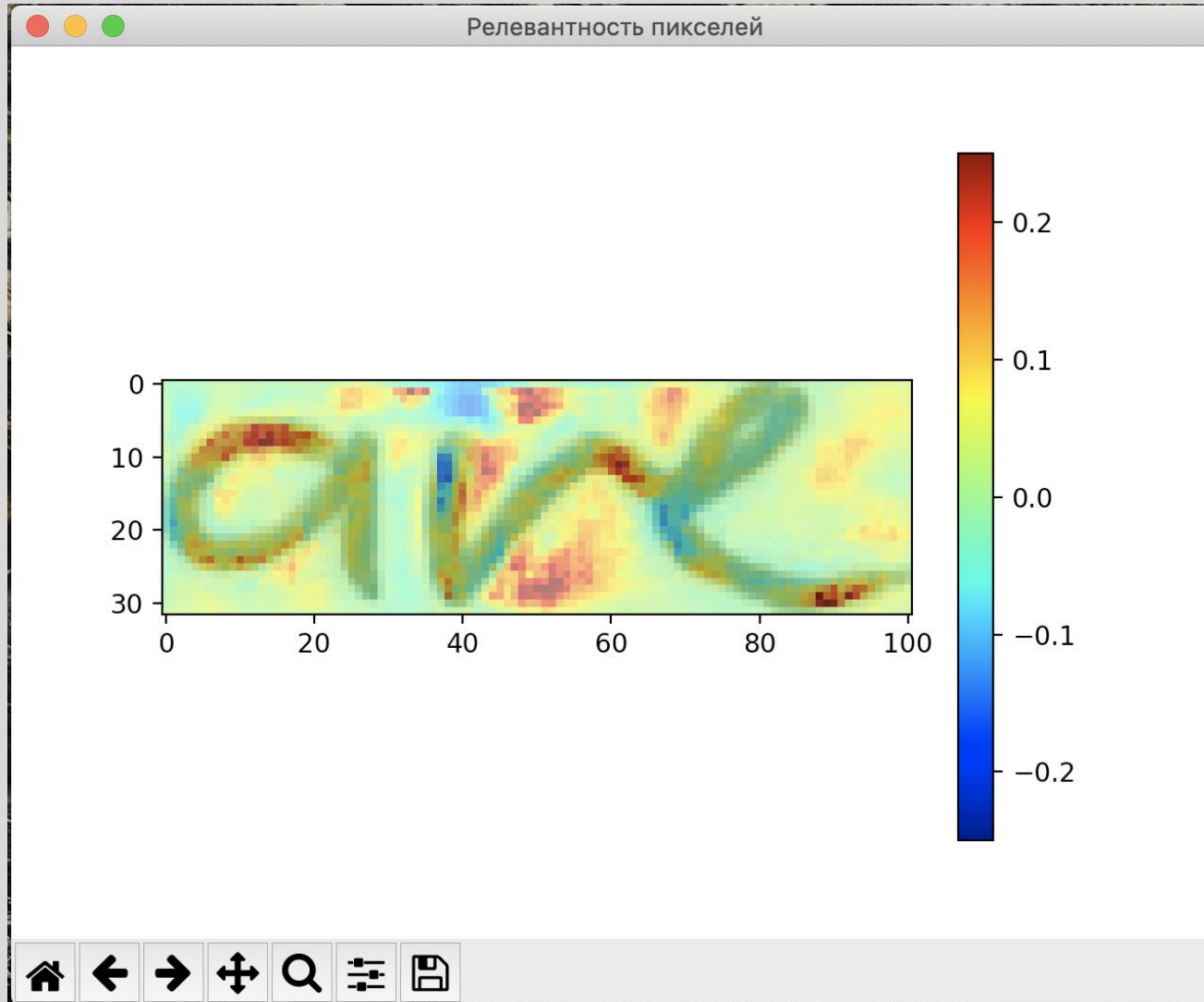
ПРОЦЕСС СЕГМЕНТАЦИИ



источник <https://vc.ru/>



Релевантность пикселей слова “are”



ЕЩЕ ОДИН ПРИМЕР ПРИМЕНЕНИЯ – ОБРАБОТКА АУДИОИНФОРМАЦИИ

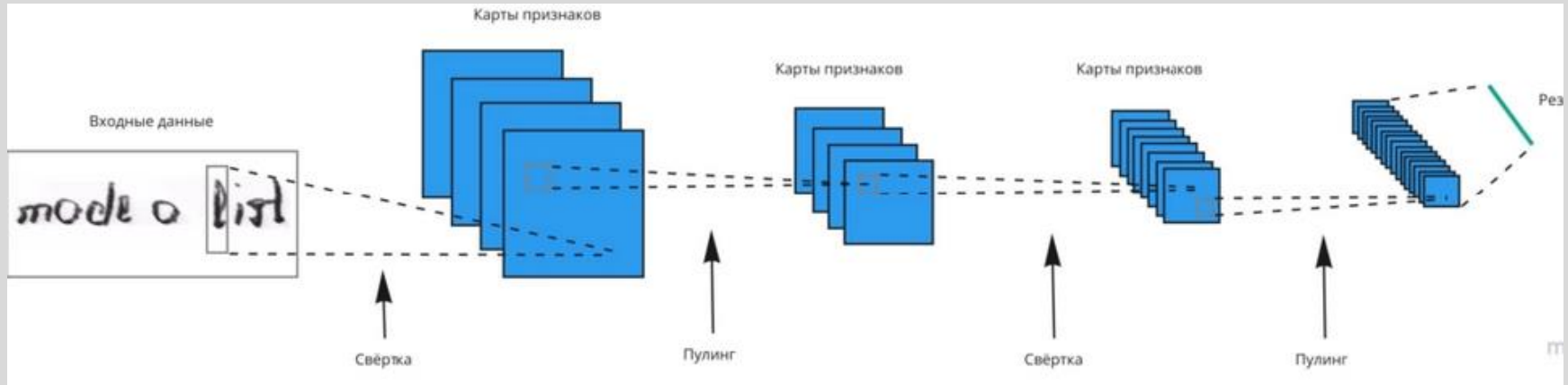
Таблица – Виды цифровых аудио форматов и их сравнение

Название формата	Разрядность (бит)	Частота дискретизации (Гц)	Степень и тип сжатия
WAV	16	до 44,1	Без сжатия
MP3	16-24	до 48	11:1 с потерями
WMA	до 24	до 96	2:1 с потерями
ACC	16-24	до 96	3:1 с потерями

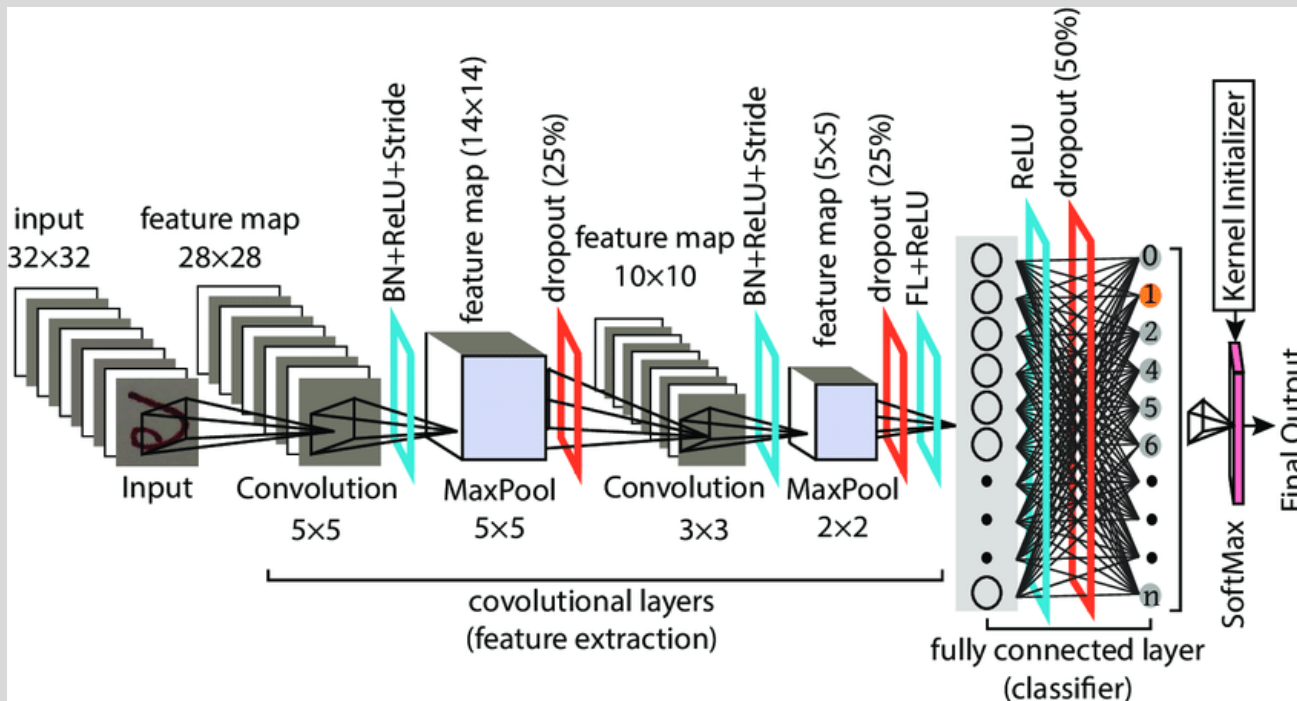
Некоторые «современные» задачи для глубокого обучения:

- индексирование музыкальных коллекций согласно их аудио признакам (Youtube Music);
- рекомендация музыки для радио миксов (Spotify).
- поиск сходства аудиоданных в звуке (Shazam, TrackId).
- обработка и синтез речи – генерирование искусственного голоса для диалоговых агентов (Siri, Алиса)

СВЕРТОЧНАЯ НЕЙРОННАЯ СЕТЬ ДЛЯ РАСПОЗНАВАНИЯ РУКОПИСНОГО ТЕКСТА



источник <https://vc.ru/>



Проверка уникальности

Уникальность: **100.00%**

- [Получить ссылку на проверку](#)
- [Зафиксировать уникальность](#)
- [Получить кнопку уникальности](#)

Подробнее

Проверка орфографии

В тексте найдено **7** ошибок:

- Grohe
- Allure
- излива

Подробнее

SEO-анализ текста

Всего символов: **947** Заспамленность: **39%**
Без пробелов: **823** Вода: **8%**
Количество слов: **124**

Подробнее

Подсвечено: Неуникальные фрагменты

Смеситель для раковины Grohe Allure 19386000 из новой коллекции Allure, стоимостью всего 5800 рублей. Скрытый монтаж обеспечивает повышенное удобство эксплуатации и, конечно, установки. Система GROHE SilkMove позволяет обеспечить исключительно легкое движение рычага. Специальное покрытие, произведенное по технологии StarLight создает долговечность и сохраняет хороший вид изделия на протяжении долгих лет. Вертикальный монтаж с двумя монтажным отверстиями весьма удобен и не должен вызвать трудностей. Величина выноса излива здесь равна 220 мм. Большой размер выноса приводит к тому, что использовать изделие становится намного проще. Все изделие в общем имеет вес равный 1,955 кг. Минимальное давление для данной модели равняется 1 бар. В подключении к электричеству нет необходимости. Бесплатная доставка и надежное, проверенное годами, качество широко известного всем немецкого бренда - главные причины купить смеситель Grohe Allure 19386000

Текст сохранен

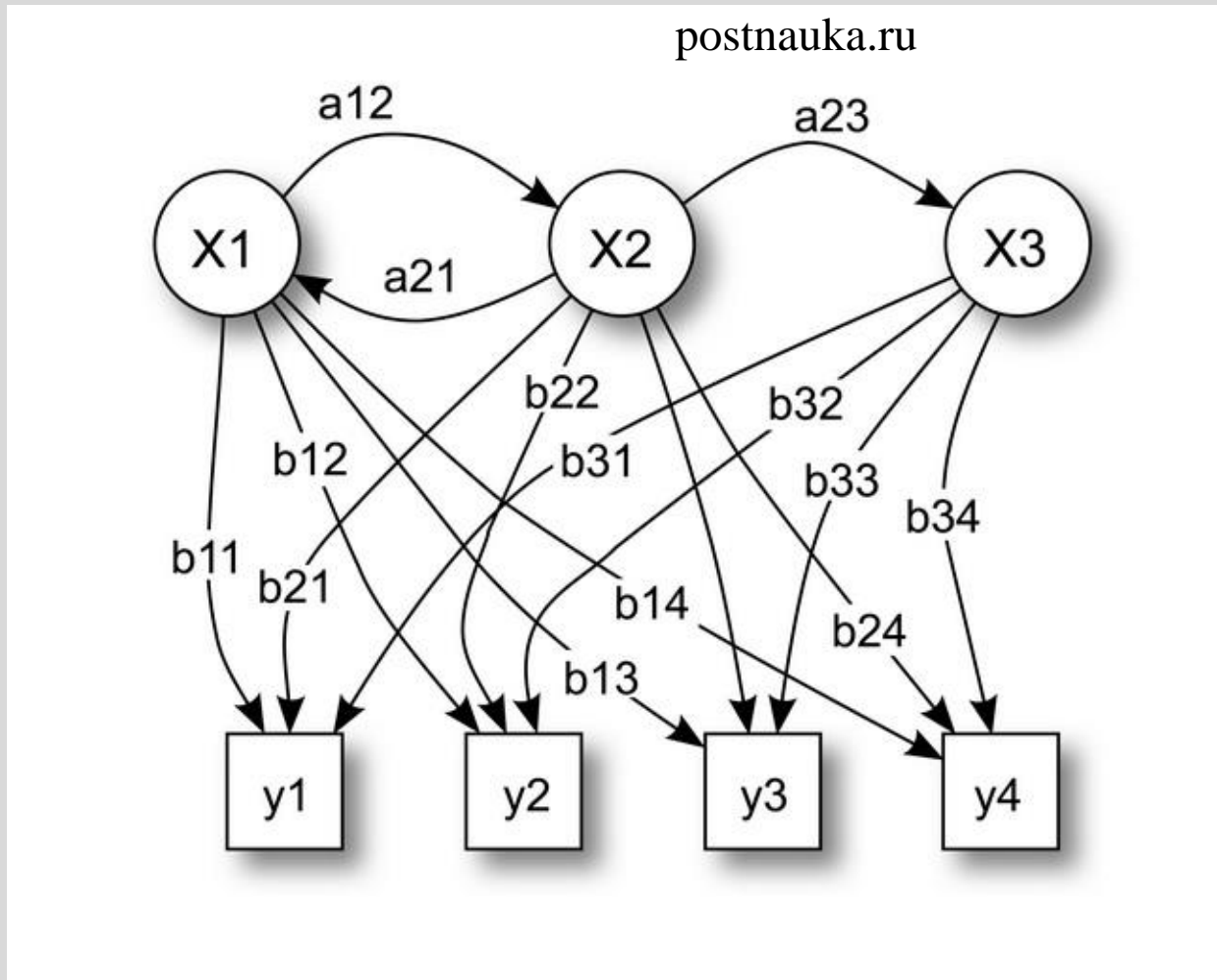
Проверить уникальность

Версии текста:

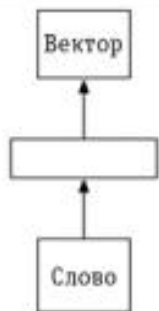
🔒 Несколько секунд назад (UTC +03:00)			
Уникальность	100%	Орфография	7
Всего символов	947	Заспамленность	39%
Без пробелов	823	Вода	8%
Количество слов	124		

SEO анализ текста
Рисунок – habr.com

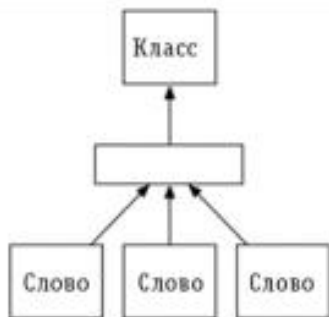
ОБРАБОТКА ТЕКСТА



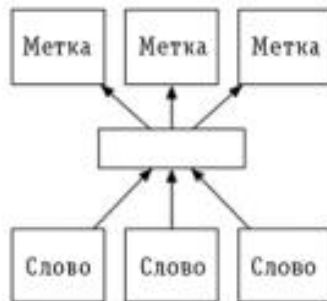
Переходы в скрытой марковской модели



1. Нахождение векторов слов



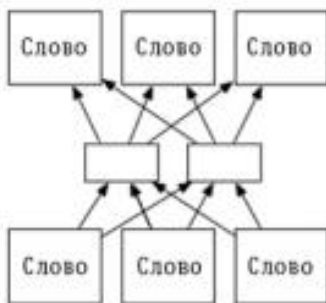
2. Классификация предложений или текстов



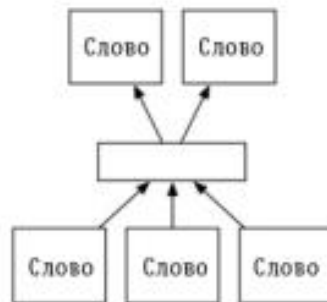
3. Классификация последовательностей



4. Генерация описания по входному объекту

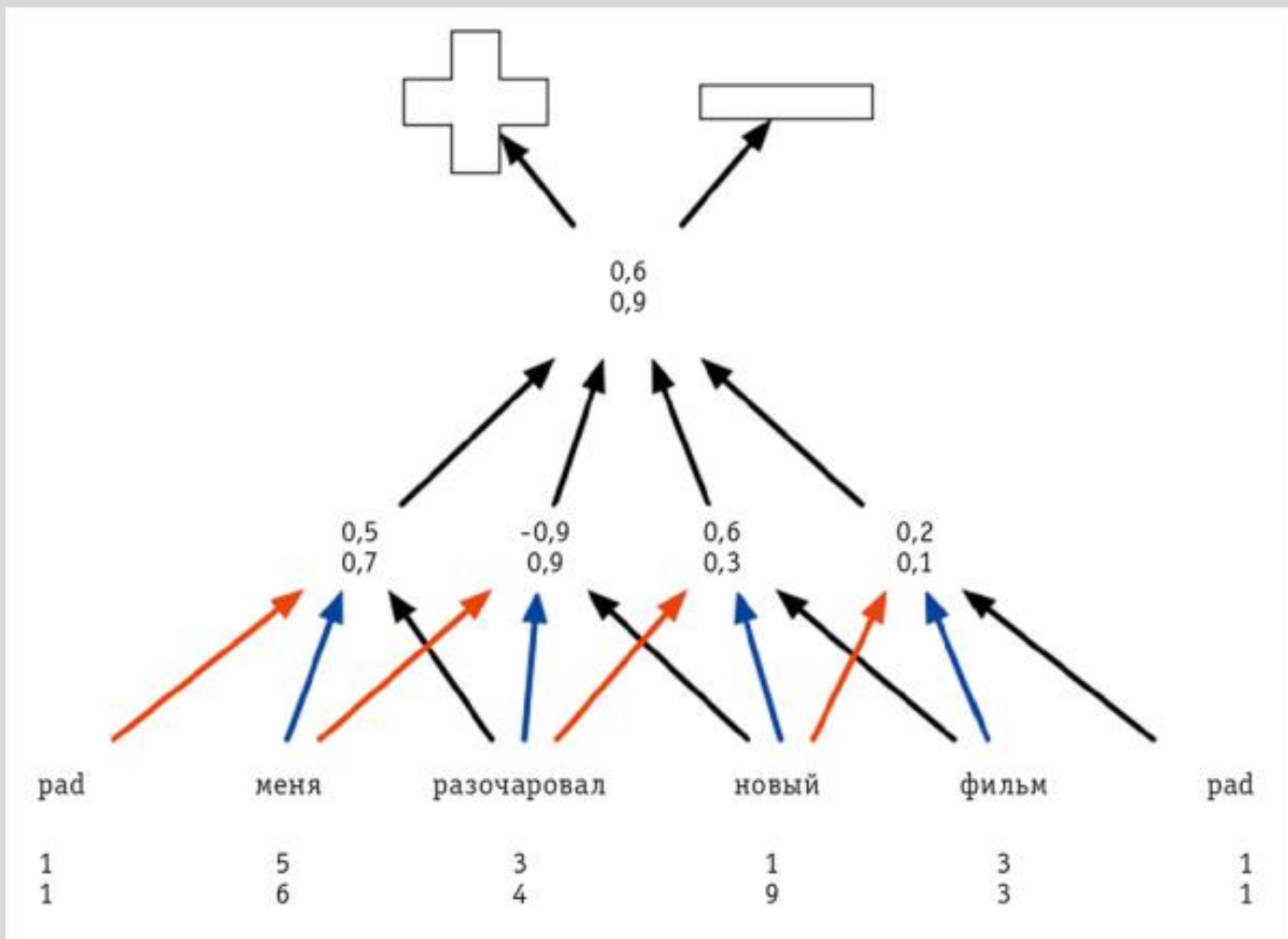


5. Автокодирование текста для снижения размерности



6. Машинный перевод
7. Построение вопросно-ответных систем и чат-ботов

Источник – «Открытые системы»



СНС для классификации предложения по тональности (журнал «Открытые Системы»)

СЕМАНТИЧЕСКИЕ СЛОВАРИ

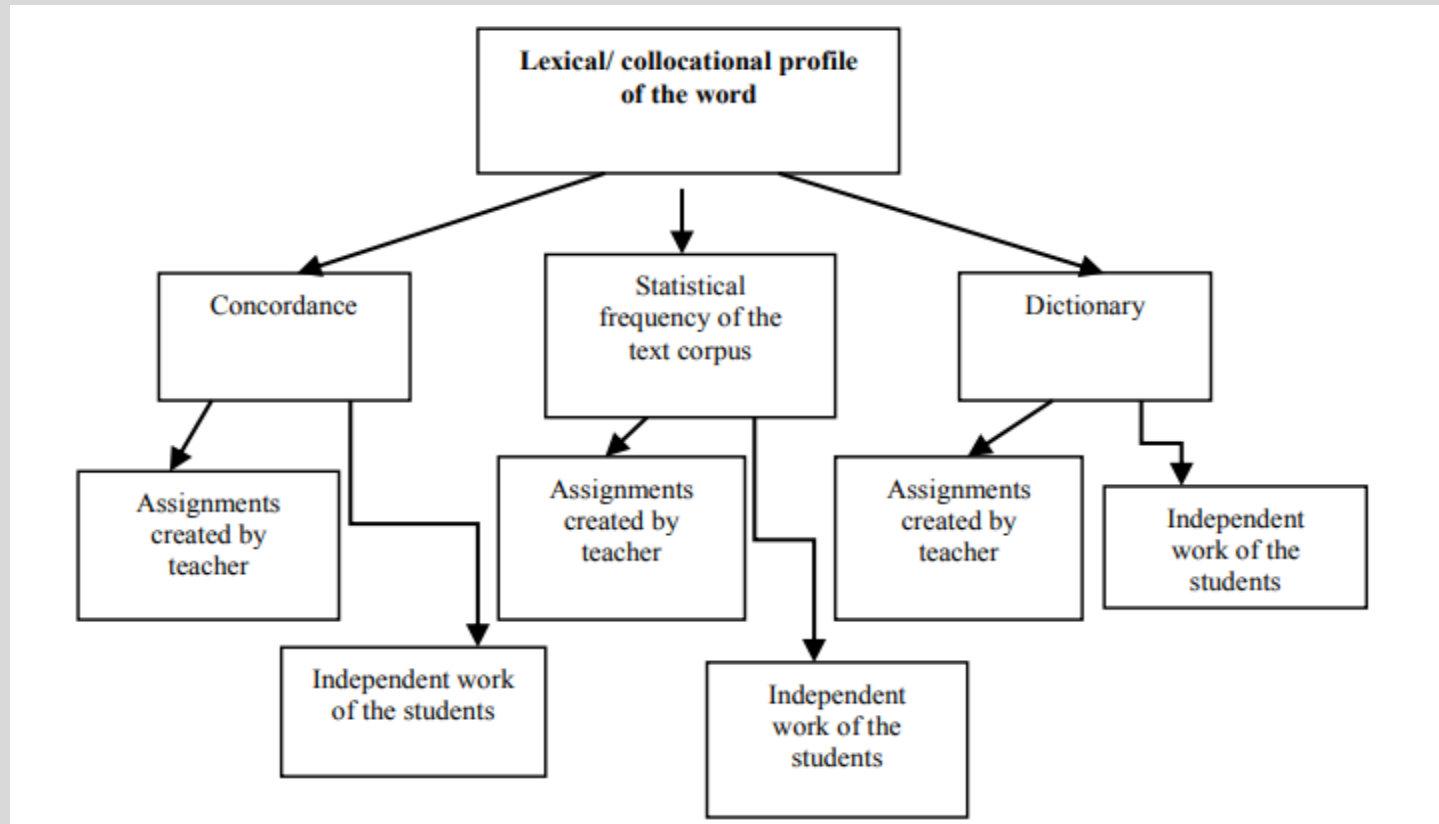
Семантический частотный словарь - словарь, в котором вся семантика входных единиц. Такие словари могут быть язычными (например, словари языка писателя, где толкуются значения и указываются частоты слов), двуязычными и язычными (например, учебные словари минимумы). Семантические частотные словари используются в системах машинного перевода, перевода с помощью машины, в обучающих лингвистических автоматах и в качестве учебных пособий: учебных словарей-минимумов, лексических справочников и пр., сохраняя при этом все достоинства обычного частотного словаря, но существенно расширяя сферу своего применения и круг пользователей

Для составления семантических словарей используется частотный анализ текста и составление корпуса слов языка

**НАЦИОНАЛЬНЫЙ КОРПУС РУССКОГО ЯЗЫКА
(поддерживается Яндексом)**

СЕМАНТИЧЕСКИЕ СЛОВАРИ

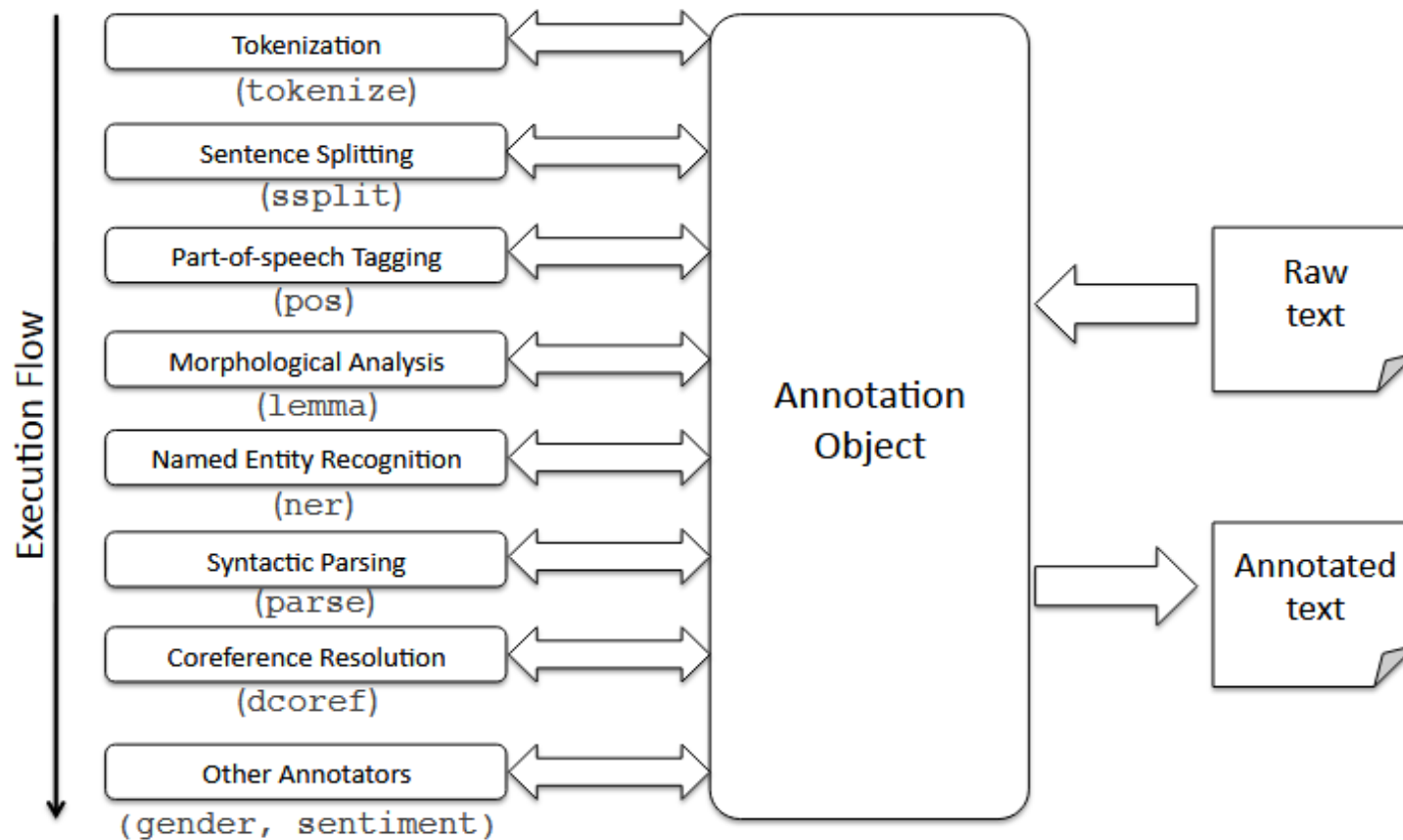
Семантический словарь должен содержать толкование единиц языка, обладать динамичностью и гибкостью, ограничивать сверху разномножество формализаций единиц языка, уметь выбирать адекватный ситуации смысл единицы языка.



Создание корпуса слов и другие проблемы анализа текста.

<https://www.redalyc.org>

Архитектура CoreNLP



ЛИТЕРАТУРА:

1. Тузов В.А. Компьютерная семантика русского языка. – СПб: Изд-во СПбГУ, 2004
2. Баймурзина Д.Р. Нейросетевые модели и диалоговая система для ведения разговора на общие темы. Дисс. канд. техн. наук. МФТИ. 2021. 136 с.
3. Бонгард М.М. Проблема узнавания. — М.: Наука, 1967. — 320 с.
4. Куторба А.Ю. Обработка англоязычных текстов на основе семантического словаря // Вестник Санкт – Петербургского университета. 2005.
5. Добров Б.В., Лукашевич Н.В., Автоматическая рубрикация полнотекстовых документов по классификаторам сложной структуры // Восьмая национальная конференция по искусственному интеллекту — Коломна, 2002.
6. Аношкина Ж.Г. Морфологический процессор русского языка. // Бюллетень машинного фонда русского языка / отв. редактор В.М. Андрющенко — М., 1996. — Вып.3, с.53-57.

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 7

**Задача автоматической обработки текстов:
машинное обучение в задаче обработки текстов,
системы автоматической обработки текстов**

NLP (*англ. Natural Language Processing*)

NLP лежит на стыке нескольких направлений науки – лингвистика, математика и искусственный интеллект

Выделим основные области лингвистики:

- 1) Фонология;
- 2) Морфология;
- 3) Синтаксис
- 4) Семантика и прагматика
- 5) Лексикография

NLP изучает проблемы компьютерного анализа и синтеза естественных языков. Применительно к искусственному интеллекту анализ означает понимание языка, а синтез — генерацию грамотного текста.

Есть также раздел в искусственном интеллекте – **компьютерная лингвистика**, в рамках которого также занимаются обработкой текстовых документов.

Распространенные задачи обработки естественного языка:

- распознавание речи, перевод текста в текстовом документе;
- синтез речи;
- информационный поиск: извлечение информации из текстов, путем выявления сущности некоторых типов и установления отношения между ними;
- классификация текстовых данных;
- кластеризация текстовых данных;
- анализ социальных сетей: извлечение мнений пользователей социальных сетей о заданных объектах;
- создание вопросно-ответных систем, диалоговых систем и задач машинного перевода и др..

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Конвейер (pipeline) — это метод объединения последовательности преобразований. Смысл этого подхода в том, чтобы разбить проблему на части и решать их отдельно.

Корпус (corpus/corpora) — это коллекция родственных документов или высказываний на естественном языке. Выделяют даже направление компьютерной лингвистики – **корпусная лингвистика**.

Мешок слов (или Bag of Words) — это модель текстов на натуральном языке, в которой каждый документ или текст выглядит как неупорядоченный набор слов без сведений о связях между ними. Его можно представить в виде матрицы, каждая строка в которой соответствует отдельному документу или тексту, а каждый столбец — определенному слову. Ячейка на пересечении строки и столбца содержит количество вхождений слова в соответствующий документ.

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Стемминг (stemming) — это процесс нахождения основы слова (неизменяемой части слова, которая выражает его лексическое значение) для заданного исходного слова.

Стеммер — программная реализация алгоритм *стемминга*.

Стоп-слова (stopwords) — слова в тексте, которые не несут смысловой нагрузки.

Токенизация (tokenization) — это разбиение текста на более мелкие части (токены).

Частотность (frequency) — термин, предназначенный для определения наиболее употребительных слов. Это отношение числа вхождений слова к общему числу слов документа.

Центроид — точка, которая является центром кластера. Каждый центроид является вектором, элементы которого представляют собой средние значения соответствующих признаков, вычисленные по всем записям кластера.

Классически лингвистический модуль обработки текста действует по цепочке:

- определение языка (если это многоязычная лингвистическая система);
- отфильтровывание нечитаемых символов;
- нормализация текста (разделение введенного текста на слова и остальные последовательности символов);
- исправление орфографических и синтаксических ошибок (т.н. спелчекеры);
- удаление предлогов;
- лексический анализ (анализ слов);
- синтаксический анализ (анализ порядка слов в предложении с учётом правил грамматики);
- семантический анализ (анализ значения предложения самого по себе и в его связи с другими предложениями).

Небольшой пример семантического анализа для русского языка (на базе продукционных правил):

Правило 1: ЕСЛИ определение стоит на первом месте и за ним идет существительное, ТО существительное является подлежащим.

Правило 2: ЕСЛИ за подлежащим идет глагол, ТО этот глагол является сказуемым и поясняет, что делает подлежащее.

Правило 3: ЕСЛИ за подлежащим идет сказуемое, а за ним следует существительное, ТО это существительное является дополнением.

Правило 4: ЕСЛИ предложение имеет следующий порядок слов: подлежащее, глагол, дополнение, ТО вся фраза говорит о том, что подлежащее делает (действие, выраженное сказуемым) по отношению к дополнению.

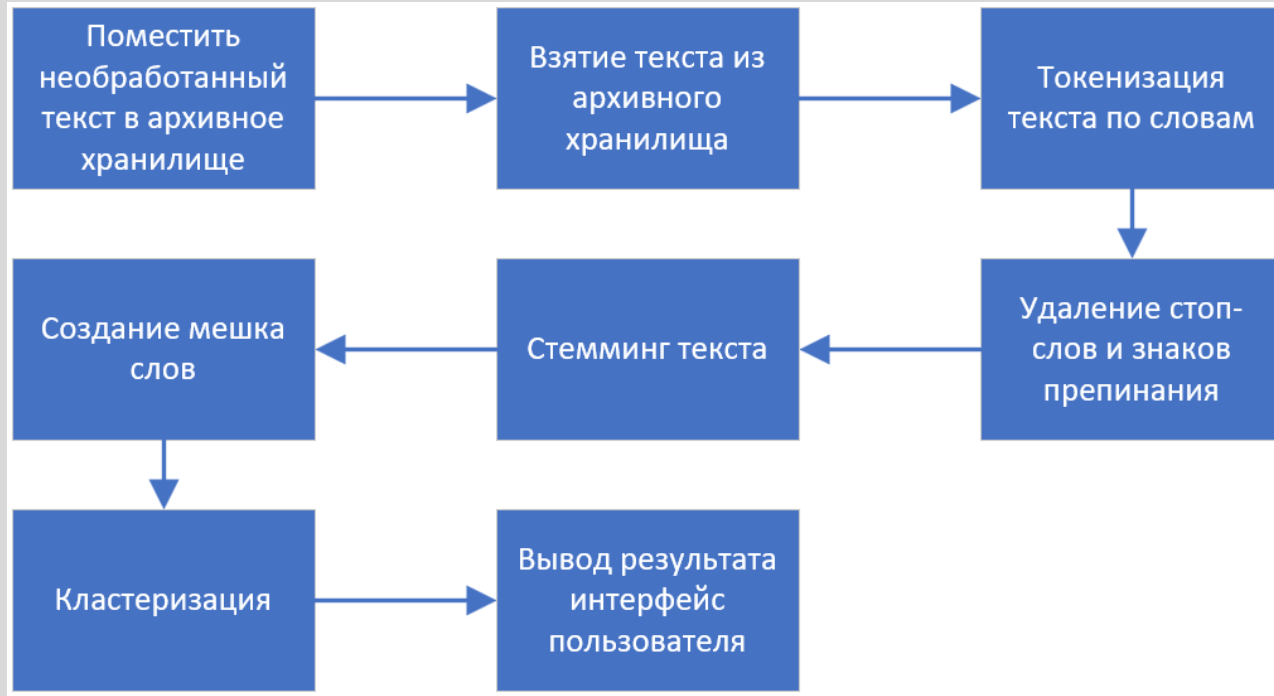
Современные примеры применения NLP

- идентификация различных категорий пользователей или клиентов для прогнозирования их оттока, прибыли от клиентов, продуктовых предпочтений и т.д.;
- детектирование отзывов пользователей о том или ином товаре, услуги или событии (при этом могут быть «вложенные задачи» - например, упоминание в отзывах размера одежды или ее цвета);
- классификация текста в соответствии с его смыслом по таким разделам, как запрос элементарной помощи или подсказки, решение определенной проблемы.

ОДИН ИЗ ВАРИАНТОВ ОБЩИХ СХЕМ ЛИНГВИСТИЧЕСКОЙ СИСТЕМЫ



Структура конвейера

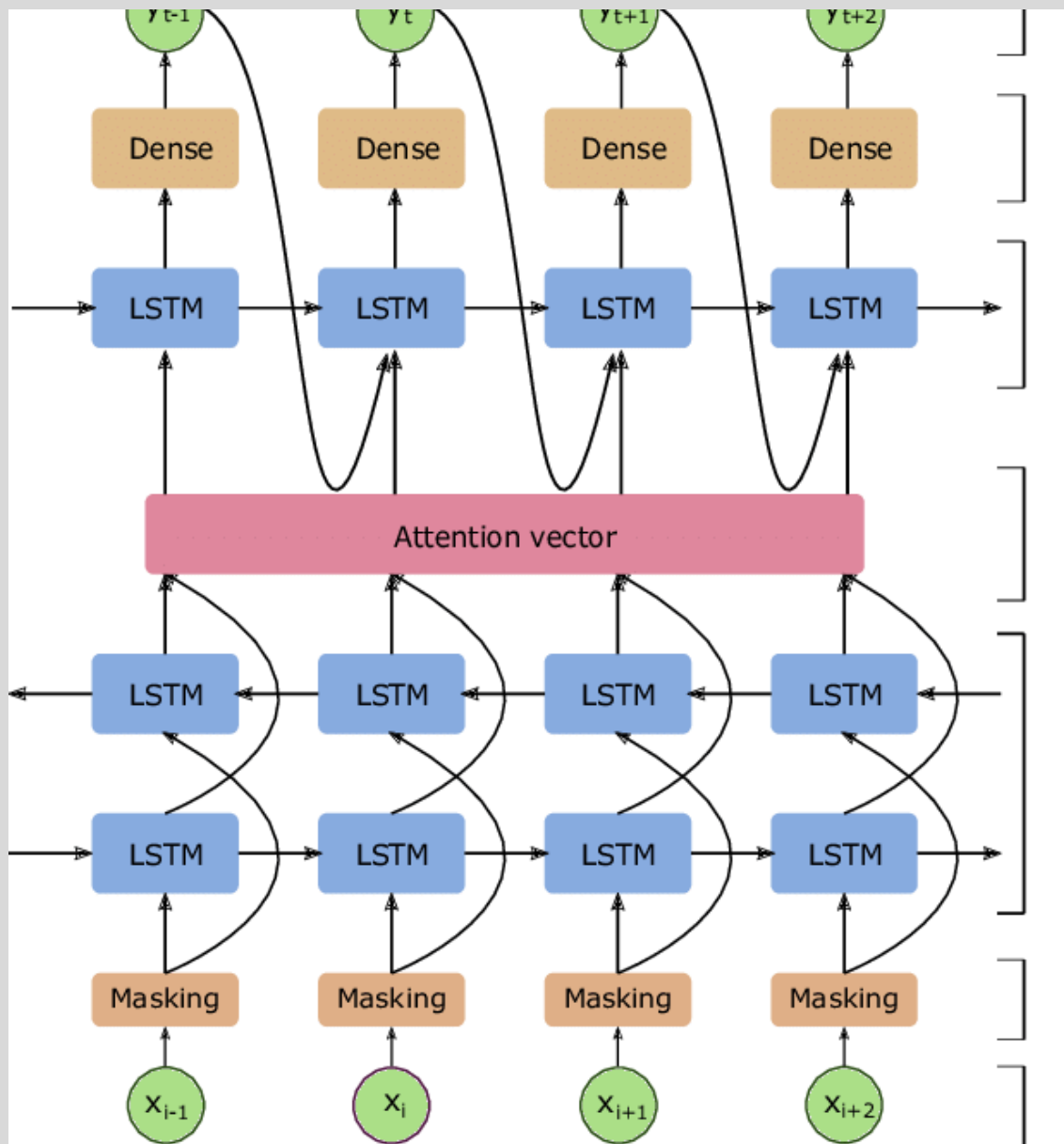


ПРОБЛЕМА ОЧИСТКИ ДАННЫХ

При очистке данных обычно производят следующие шаги:

- удаление нерелевантных символов (но до этого необходимо составить словарь нерелевантных символов);
- токенизация текста;
- удаление нерелевантных слов;
- приведение всех символов в нижний регистр;
- совмещение слов, написанных с ошибками или имеющих альтернативное описание (например, «Апшеронск / Апширонск» или «хорошо / круто»);
- проведение лемматизации, если это необходимо

ПРИМЕР ПРИМЕНЕНИЯ LSTM сети

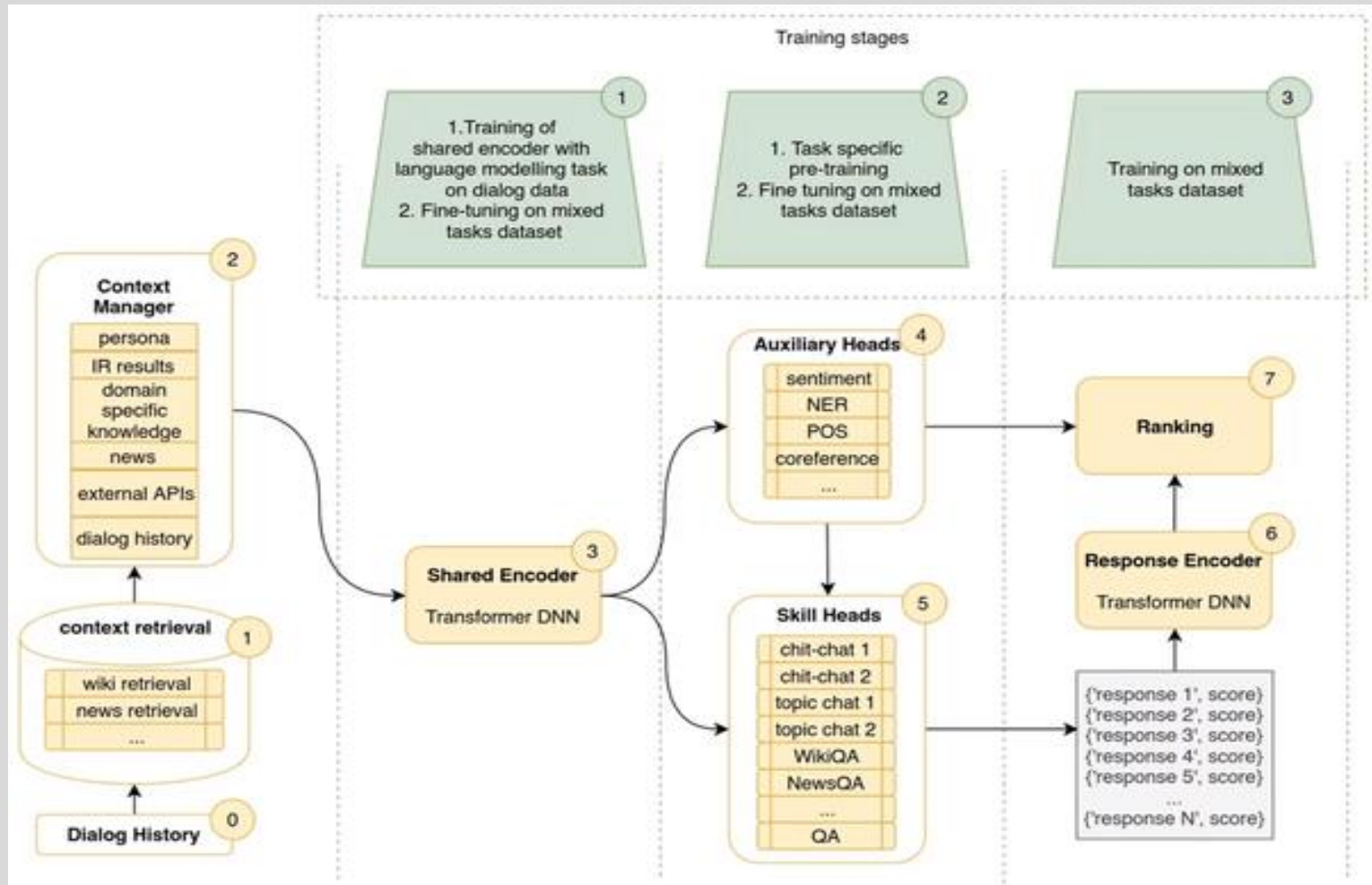


ДИАЛОГОВЫЕ СИСТЕМЫ (ЧАТ-БОТЫ)

Определение: «*ДИАЛОГОВАЯ СИСТЕМА, или интерактивная система, или система вопрос-ответ — автоматизированная человекомашинная система, работающая в режиме диалога, при котором она отвечает на каждую команду пользователя и по мере надобности обращается к нему за информацией*»

```
=====
EEEEEEEE L      IIIIII  ZZZZZZZ  AAA
E         L      I      Z      A   A
E         L      I      Z      A   A
EEEEEE   L      I      Z      A   A
E         L      I      Z      AAAAAA
E         L      I      Z      A   A
EEEEEEEE LLLLLLLL IIIIII  ZZZZZZ  A   A
=====
ELIZA > (PLEASE TYPE IN ALL CAPS) WHAT'S YOUR NAME DEAR ?
AMIT
ELIZA > HI, I'M ELIZA. WHAT DO YOU WANT TO TALK ABOUT ?
AMIT > CAN I TALK ABOUT MY PROBLEM ?
ELIZA > SURE... !
AMIT > DUE TO ON GOING EXAMS IN COLLEGE, I HAVE A LOT OF STRESS.
ELIZA > PLEASE RELAX A LITTLE. SLEEP WELL.
AMIT > THANKS FOR YOUR ADVICE.
ELIZA > NO MENTION.
AMIT > BYE
ELIZA > BYE AND KEEP IN TOUCH...
=====
```

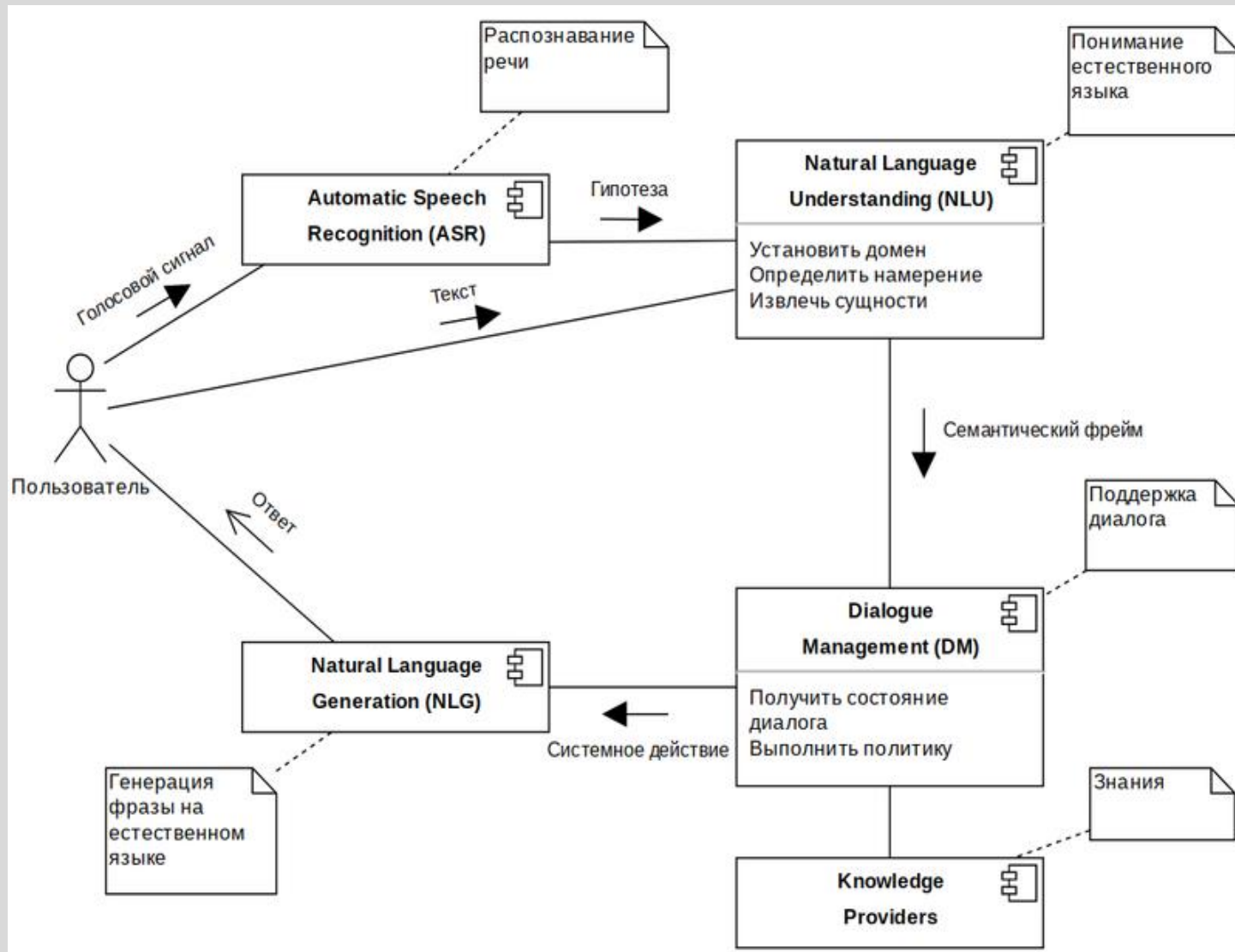
СХЕМА АБСТРАКТНОЙ ДИАЛОГОВОЙ СИСТЕМЫ



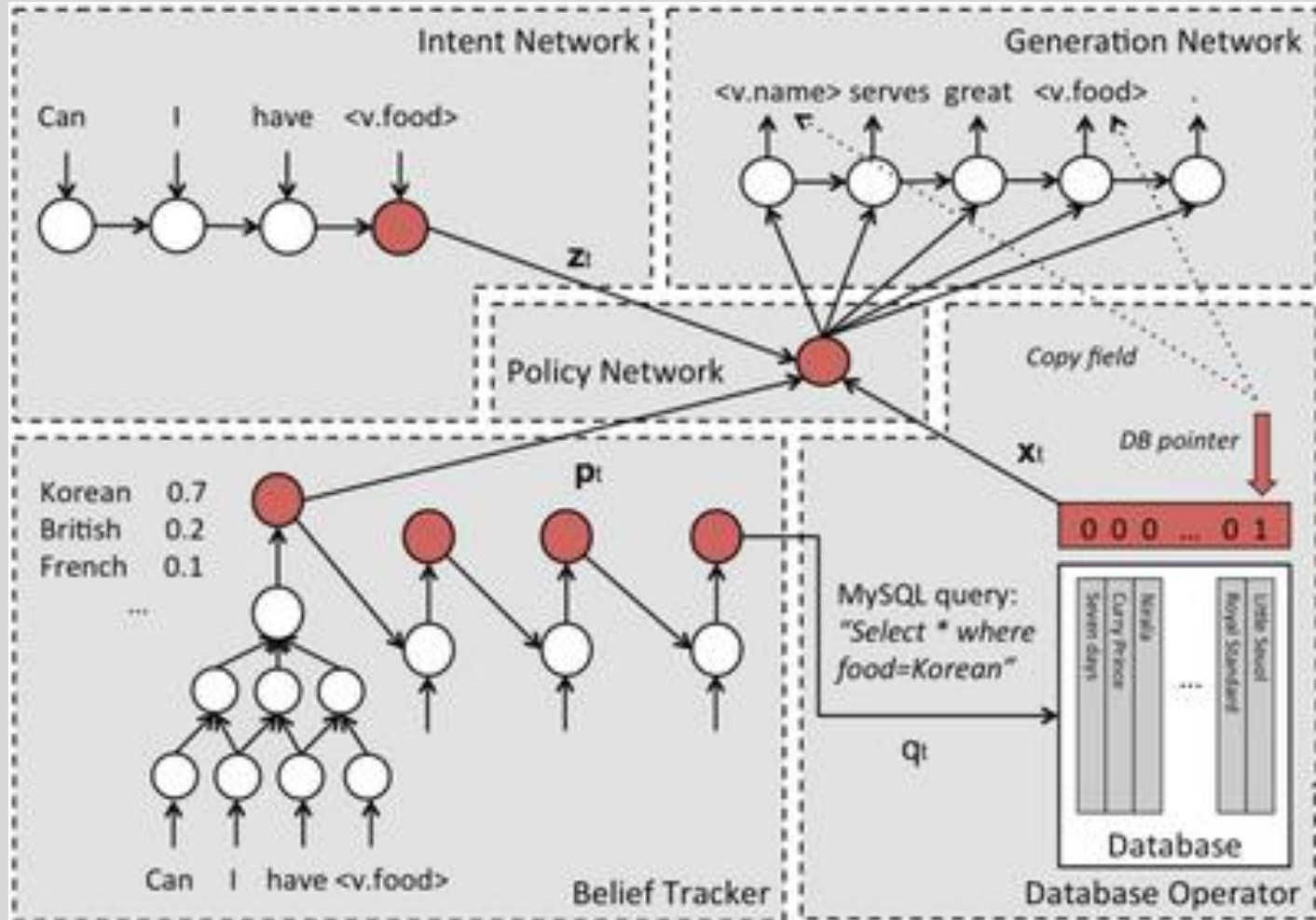
Обычно работа диалоговой системы содержит следующие шаги:

- Пользователь вводит информацию любым из возможных способов, и она преобразуется в текст с помощью декодера или распознавателя, который может автоматически распознавать человеческую речь (ASR), жесты или обычный текст;
- Системой анализируется преобразованный текст, с которым работает блок распознавания естественного языка (NLU), выполняющим идентификацию своего имени, част речную разметку, семантический или семантический анализ;
- Семантическая информация анализируется менеджером диалога, хранящим историю диалога и его состояние и управляющим общим потоком;
- Менеджер диалога может связываться с менеджерами задач, обладающими знаниями о конкретной области задач;
- Менеджер диалога производит выходные данные с помощью генератора выходных данных, в состав которого могут входить генераторы естественного языка, жестов и менеджер компоновки;
- В конечном счете производится вывод информации с помощью средства визуализации вывода, который может включать преобразование текста в речь (TTS) и взаимодействие с пользователем посредством виртуального аватара.

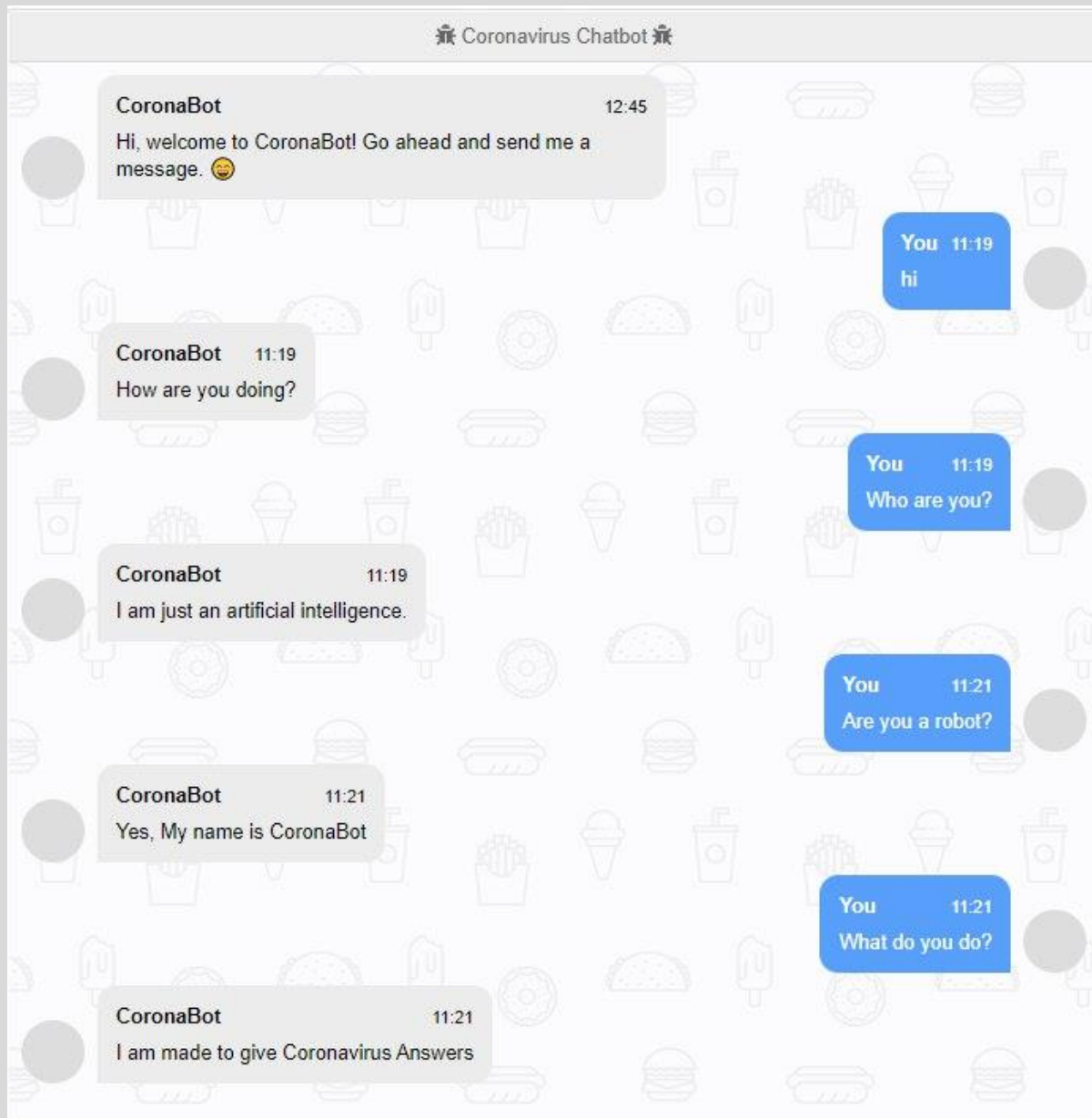
ДИАГРАММА КЛАССИЧЕСКОЙ АРХИТЕКТУРЫ ЧАТ-БОТА



НЕЙРОСЕТЕВАЯ АРХИТЕКТУРА В ДИАЛОГОВОЙ СИСТЕМЕ 2



ПРИМЕР ЧАТ-БОТА (CoronaBot, Овезов М. КубГТУ 2021)



АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ ТЕКСТА

Система автоматической генерации метеорологических сводок FoG
(начало 90-х)

0) Исходные данные			1) Понятия ПО		2) Элементы текста
время	Напр. ветра	Скорость ветра	Скорость ветра	Напр. ветра	
7 a.m.	235	17	15-20	southwest	wind 15-20 southwest
9 a.m.	231	21	15-20	southwest	diminish to
...	
9 p.m.	280	12	light	(west)	wind light southwest
10p.m.	307	11	light	(northwest)	
11p.m.	182	8	light	(south)	
12p.m.	246	10	light	(southwest)	

Варианты подходов генерации текста

- Canned-based methods – на базе неизменяющихся шаблонов;
- Template-based methods – с изменяющимся шаблоном;
- Phrase-based methods – на базе контекстной вставки;
- Feature-based methods – синтез на базе грамматических признаков

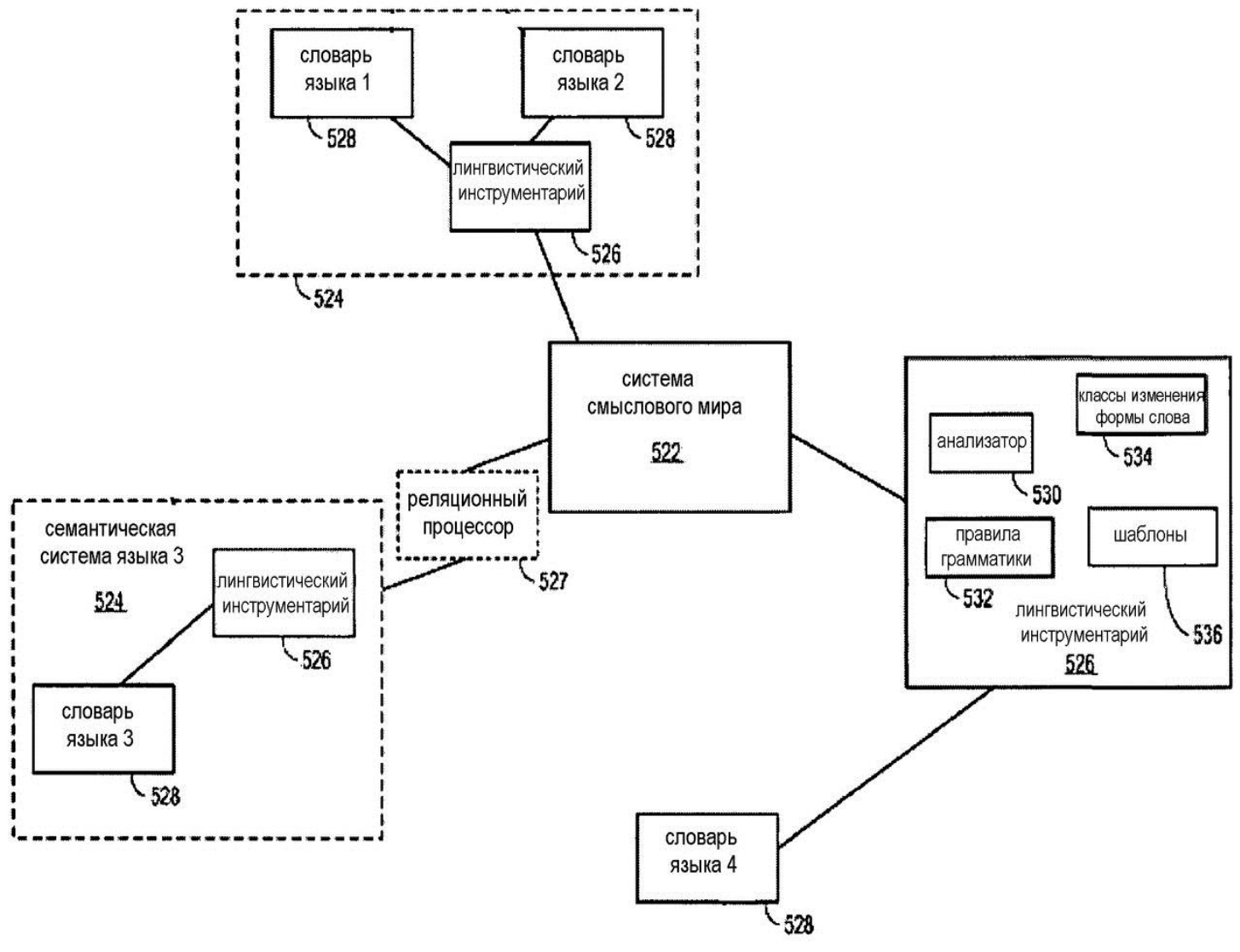
МАШИННЫЙ ПЕРЕВОД

Обычно выделяют следующие направления машинного перевода:

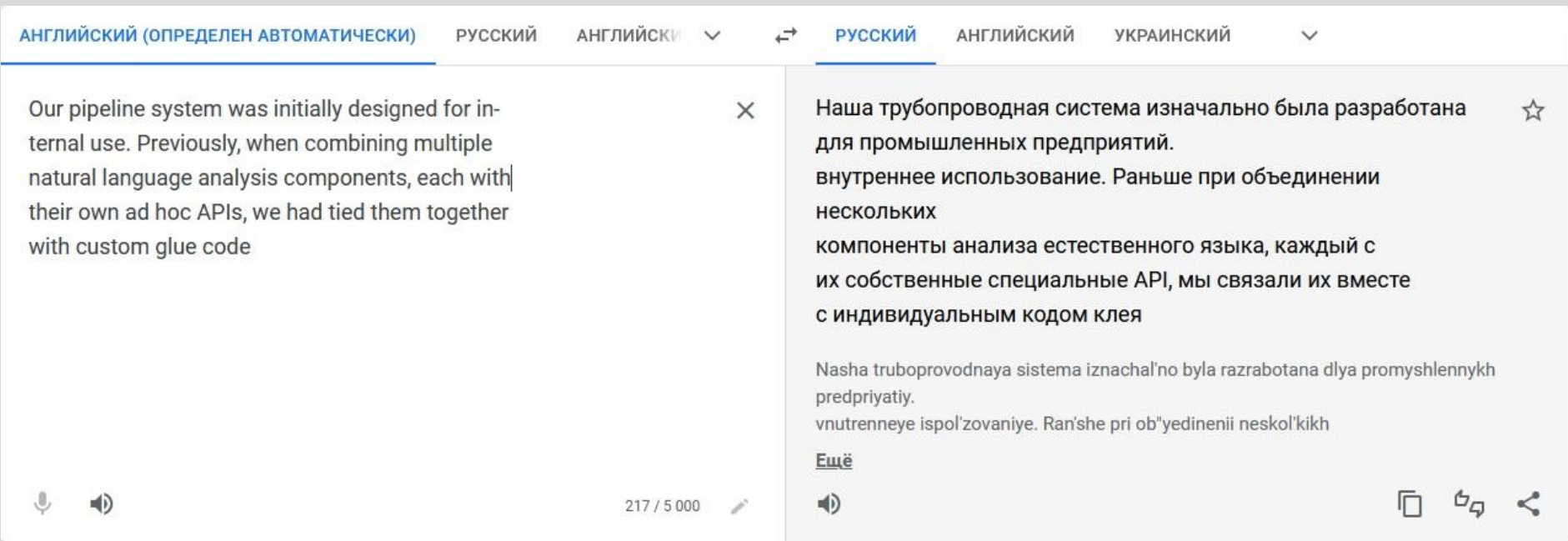
- Rule-based machine translation (RBMT);
- статистическая трансляция (Statistical machine translation, SMT) – базируется на статистику переводных пар слов и словосочетаний;
- Нейронный машинный перевод (Neural machine translation, NMT) – считается подразделом глубокого обучения.

Нейронный машинный перевод кардинально отличается от предыдущих подходов к машинному переводу. С одной стороны, NMT использует непрерывные представления вместо дискретных символьных представлений в SMT. С другой стороны, NMT использует одну большую нейронную сеть для моделирования всего процесса перевода. Обучение NMT является сквозным, в отличие от отдельно настраиваемых компонентов в статистическом переводе (SMT).

Схема обработки в NLTK



МАШИННЫЙ ПЕРЕВОД



The screenshot shows the Google Translate web interface. At the top, there are language selection tabs: 'АНГЛИЙСКИЙ (ОПРЕДЕЛЕН АВТОМАТИЧЕСКИ)', 'РУССКИЙ', 'АНГЛИЙСКИЙ', and 'УКРАИНСКИЙ'. The 'АНГЛИЙСКИЙ' tab is selected. Below the tabs, the source text in English is: 'Our pipeline system was initially designed for internal use. Previously, when combining multiple natural language analysis components, each with their own ad hoc APIs, we had tied them together with custom glue code'. The translated text in Russian is: 'Наша трубопроводная система изначально была разработана для промышленных предприятий. внутреннее использование. Раньше при объединении нескольких компоненты анализа естественного языка, каждый с их собственные специальные API, мы связали их вместе с индивидуальным кодом клея'. The interface also includes a microphone icon, a speaker icon, a character count '217 / 5 000', and a star icon for bookmarks.

Современное состояние на GoogleTranslate

ЕЩЕ ОДИН ПРИМЕР С GOOGLE TRANSLATE

Google Переводчик

Текст

Документы

АНГЛИЙСКИЙ (ОПРЕДЕЛЕН АВТОМАТИЧЕСКИ)

РУССКИЙ

АНГЛИЙСКИЙ



РУССКИЙ

АНГЛИЙСКИЙ

УКРАИНСКИЙ



SOM or Kohonen neural network is one of the basic types of artificial neural networks. Its architecture represents a two-dimensional grid of connected neurons, which are multi-dimensional vectors. (The dimension of vectors is equal to the number of descriptors.) The learning of SOM is the projection from multi-dimensional space onto two-dimensional grid (array) of neurons. The projection or learning of network runs in two-steps, the first step is the selection of the winning neuron and the second step is the self-organization of the map [



SOM или нейронная сеть Кохонена - одна из основных типы искусственных нейронных сетей. Его архитектура представляет двумерная сетка связанных нейронов, которые многомерные векторы. (Размерность векторов равно количеству дескрипторов.) Изучение SOM - это проекция из многомерного пространства на двумерное размерная сетка (массив) нейронов. Проекция или обучение сети выполняется в два этапа, первый шаг - это выбор победившего нейрона, и второй шаг - это самоорганизация карты [



ЕЩЕ ПРИЛОЖЕНИЯ КОМПЬЮТЕРНОЙ ЛИНГВИСТИКИ

- информационный поиск (Informational Retrieval), с которым связаны задачи реферирования, классификации и категоризации, а также рубрицирования документов и текстов. В настоящее время этим занимается т.н. “Text Mining”, который в свою очередь относится к “Data – mining”.

Здесь интересным видится направление реферирования текста – «выжимки» главного из входного текста, его сути. Можно сказать, что краткое содержание документа. Основной подход к решению проблемы – статистический анализ. Здесь есть еще одна схожая проблема – аннотирование текста, т.е. автоматическое составление аннотации к тексту.

- Задача формирования ответов на вопросы (т.н. FAQ);
- Задача выделение мнений (популярная современна задача. Пример – выявление мнений пользователей определенной соцсети на или иное событие или оценка качества товаров);

Задача анализа тональности текста – оценка общей тональности высказываний по тому или иному вопросу. Примеры тональных оценок: позитивная, негативная, нейтральная.

МАШИННОЕ ОБУЧЕНИЕ В ЗАДАЧАХ ОБРАБОТКИ ТЕКСТОВ

Методы машинного обучения, применяемые в обработке текстов:

- метод Байеса;
- метод k-ближайших соседей;
- линейная регрессия;
- нейронные сети;
- классификатор Роше;
- деревья решений;
- машина опорных векторов....

Можно сказать, что используется практически весь спектр методов машинного обучения.

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 8
«Нейронные сети и глубокое обучение»
Глубокое обучение с подкреплением:
обучение с подкреплением, Q – обучение

ОСНОВНЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

* *Подкрепление* – это некоторая оценка действий агента (обычно безразмерная).

* *Функция поощрения* определяет цель в задаче обучения с подкреплением, т.е. ставит в соответствие каждому состоянию среды или паре «состояние – действие» единственное число, вознаграждение, показывающее степень желательности данного состояния. По сути, определяет, насколько хороши или плохи те или иные события для данного агента.

* *Функция ценности* определяет ценность некоторого состояния – общую сумму вознаграждений, которую агент планирует получить в будущем, если данное состояние будет для него начальным

ОСНОВНЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ (*продолжение*)

TD – временная разность (от англ. Time – Difference). Ошибка временной разности.

Оценка стратегии – вычисление функции ценности состояния для произвольной стратегии (иногда данную задачу называют задачей прогнозирования)

Приведем определение подкрепления (*англ. reinforcer*) из области психологии – «*Подкрепление – любое событие, стимул, действие, реакция или информация, которые, если следуют за реакцией, служат увеличению относительной частоты или вероятности возникновения этой реакции*»

ПОДХОДЫ К ОБУЧЕНИЮ С ПОДКРЕПЛЕНИЕМ

1) Классический подход. В данном подходе процесс обучения происходит за счет поощрения и наказания («кнута и пряника») для достижения высококлассного поведения. Данный подход уходит корнями в давние работы по психологии, в частности Павлова – по условному рефлексу и Торндайка– обучение животных. Также к классическому подходу относят использование генетических алгоритмов для задач с подкреплением.

2) Современный подход. Данный подход основан на методе динамического программирования, используемом для формирования последовательности Действий с учетом будущих состояний без фактического их осуществления. Также используются различные статистические техники.

Но фундаментально можно выделить три класса методов обучения с подкреплением:

- С использованием принципов динамического программирования
- На базе методов Монте – Карло
- На базе метода временных различий

(понятно, что существуют комбинированные методы...)

Но есть и другой подход к классификации методов обучения с подкреплением:

- 1) с использованием модели (*model - based*)
- 2) без использования модели (*model - free*).

Второй тип оценивает оптимальную политику без использования динамики окружающей среды, первый наоборот ее использует для оценки оптимальной политики

РАСЧЕТ ПОДКРЕПЛЕНИЯ

$$R_t = r_{t+1} + r_{t+2} + \dots + r_T$$

R – суммарное подкрепление

r – подкрепление на временной итерации

$$R_t = r_{t+1} + \gamma \cdot r_{t+2} + \gamma^2 \cdot r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1}$$

РАСЧЕТ ОШИБКИ ВРЕМЕННОЙ РАЗНОСТИ

Прогноз функции ценности в момент времени t

$$V_{St} \leftarrow V(S_t) + \alpha \cdot [r_{t+1} + \gamma \cdot V(S_{t+q}) - V(S_t)]$$

Корректировка прогноза

$$\Delta V_t = \alpha \cdot \left[\sum_{k=t}^{\infty} \gamma \cdot r_k - V_t \right]$$

или в терминах временной разности между следующими друг за другом прогнозами:

$$\begin{aligned} \Delta V_t &= \alpha \cdot [(r_t + \gamma \cdot V_{t+1} - V_t) + \gamma(r_{t+1} + \gamma \cdot V_{t+2} - V_{t+1}) + \dots] = \\ &= \alpha \cdot \sum_{k=t}^{\infty} (r_t + \gamma \cdot V_{k+1} - V_k) \cdot \gamma^{(k-t)} \end{aligned}$$

ЦЕННОСТЬ ДЕЙСТВИЯ

Ценность действия a в момент времени t (r_1, r_2, \dots - вознаграждения полученные за все время при выборе действия a). В общем случае $k \neq t$

$$Q_t = \frac{r_1 + r_2 + \dots + r_{ka}}{k_a}$$

Другой вариант формулы $Q_{t+1} = Q_t + \frac{1}{t+1} [r_{t+1} - Q_t]$

ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ (схематично)

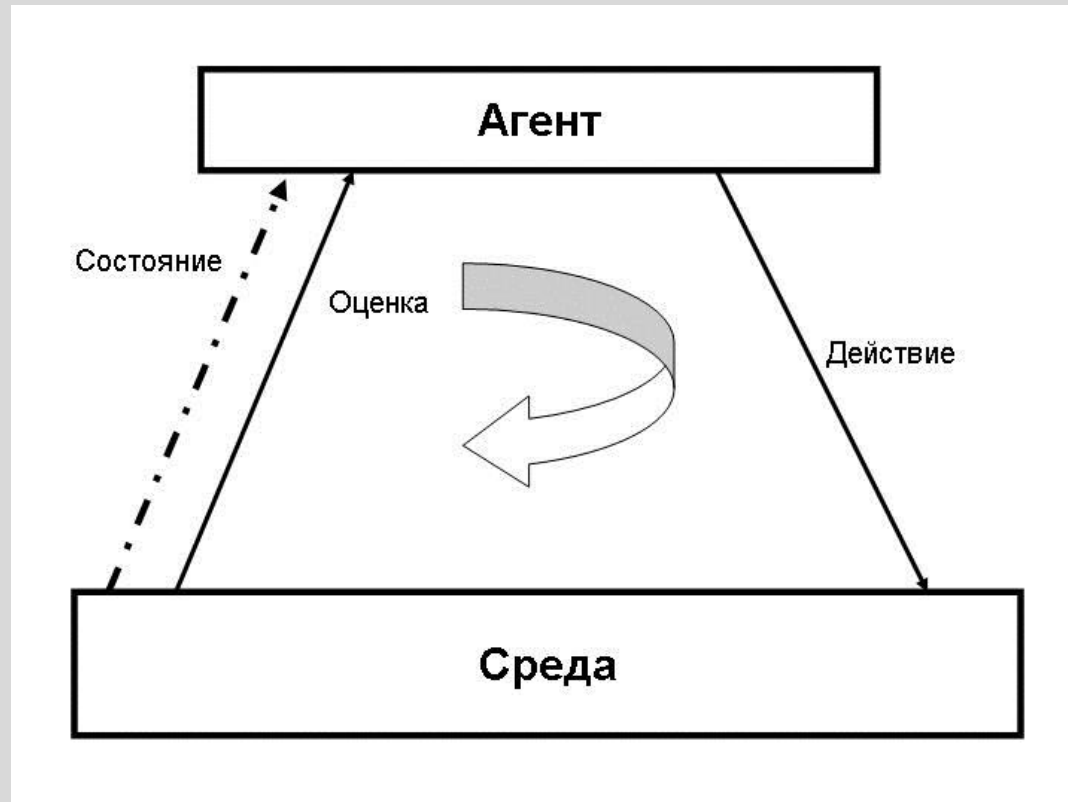
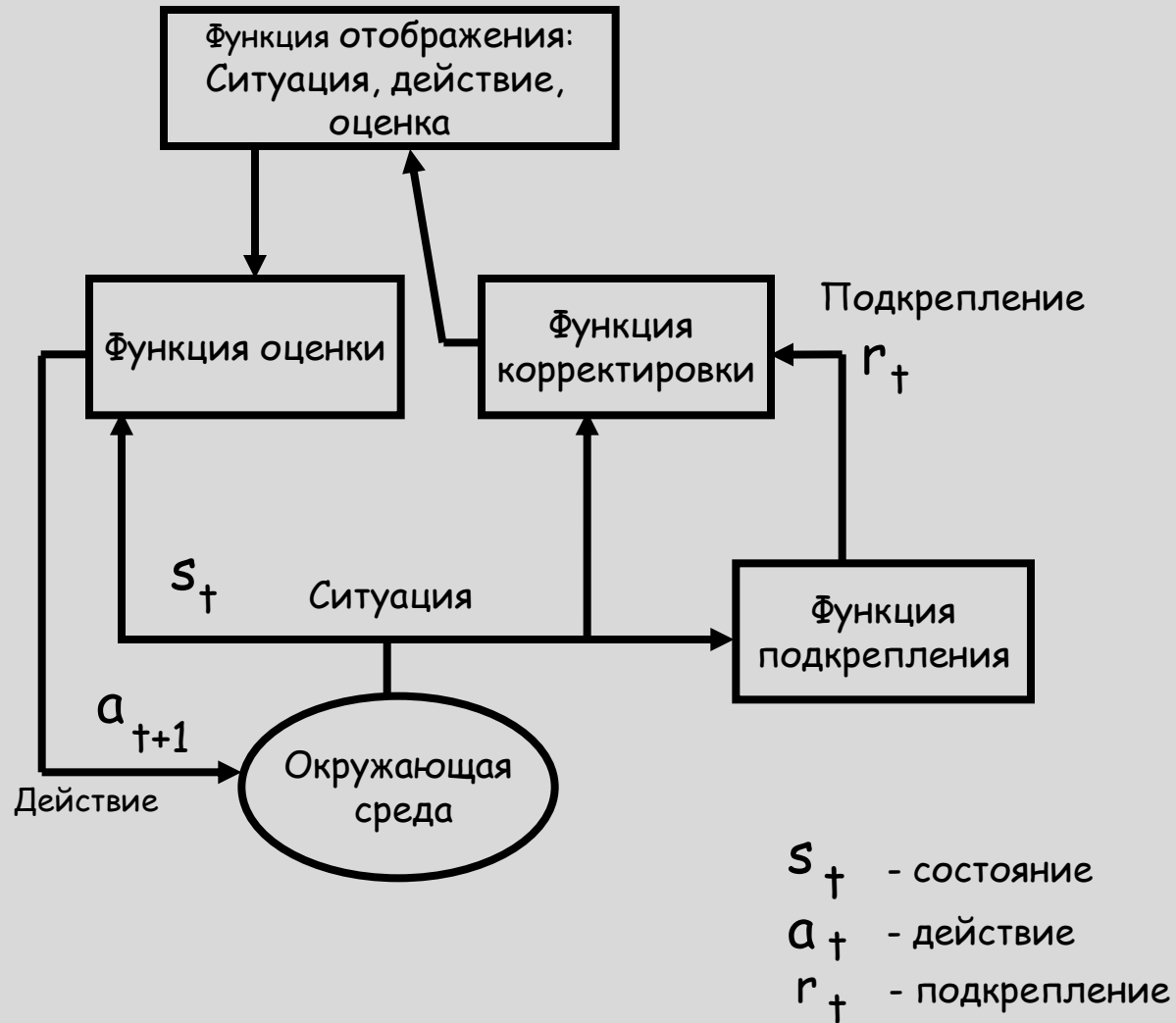
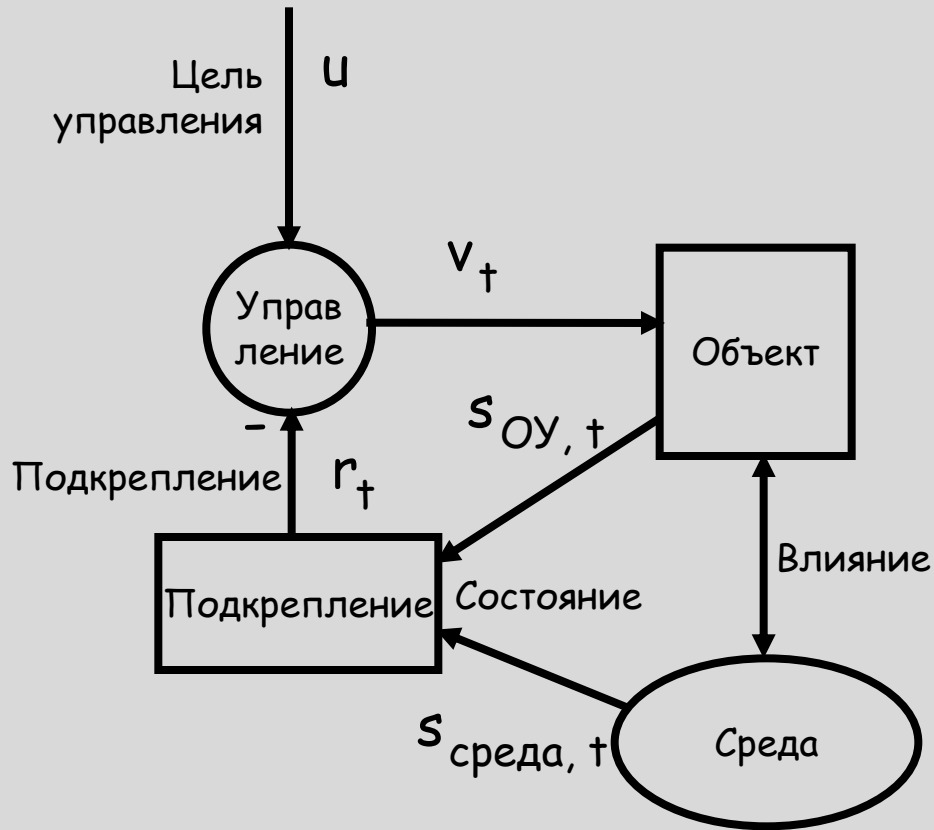


СХЕМА ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ



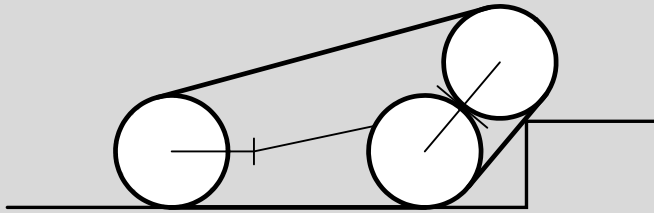
ДРУГОЙ ВАРИАНТ СХЕМЫ



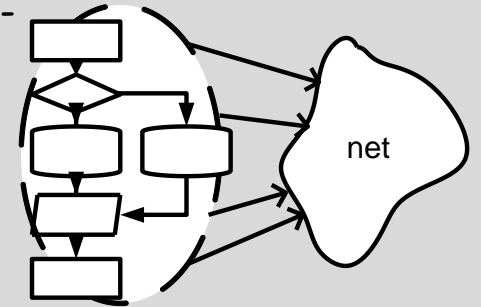
Использованы плакаты Стасевича В.П. (МГТУ - КубГТУ)

Примеры систем

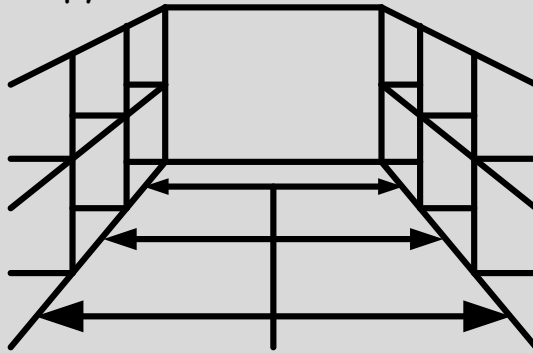
Мобильные роботы



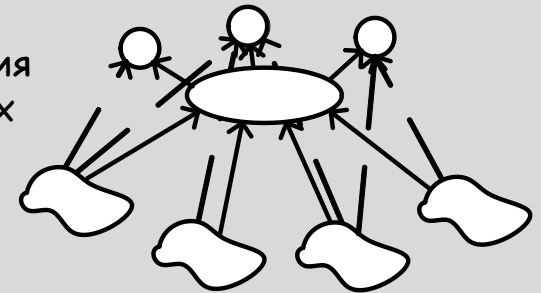
Программные роботы-агенты, собирающие необходимую информацию в глобальной сети



Мобильные роботы-грузчики

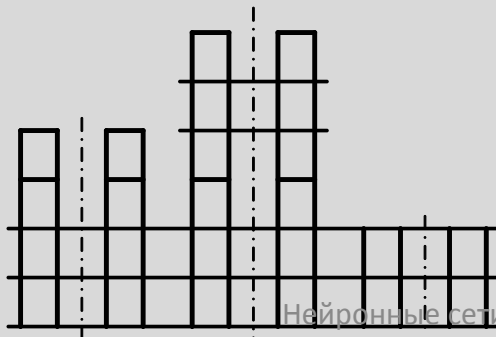


Система распределения ограниченных ресурсов

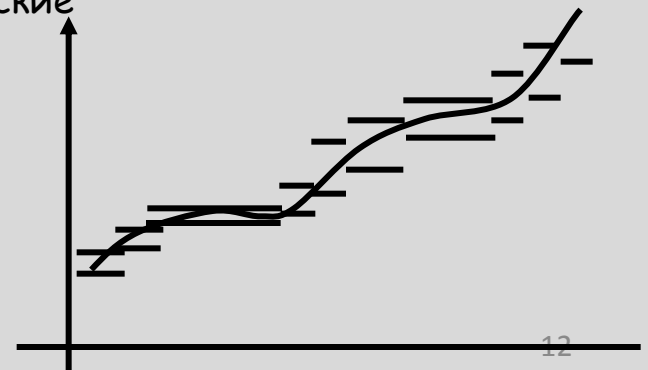


Распределение бюджета

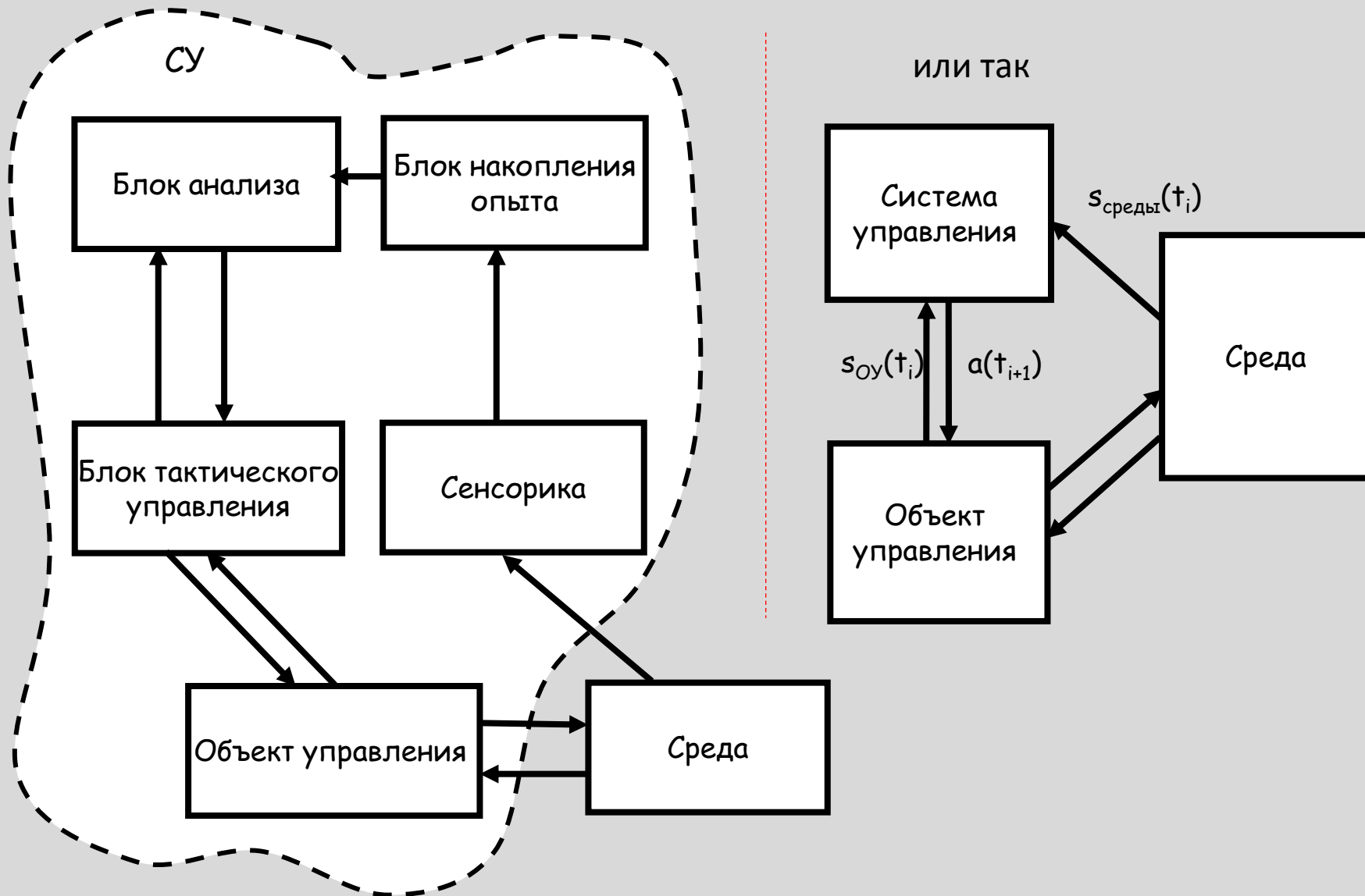
Комплекс лифтов в многоэтажном здании



Автоматические трейдеры

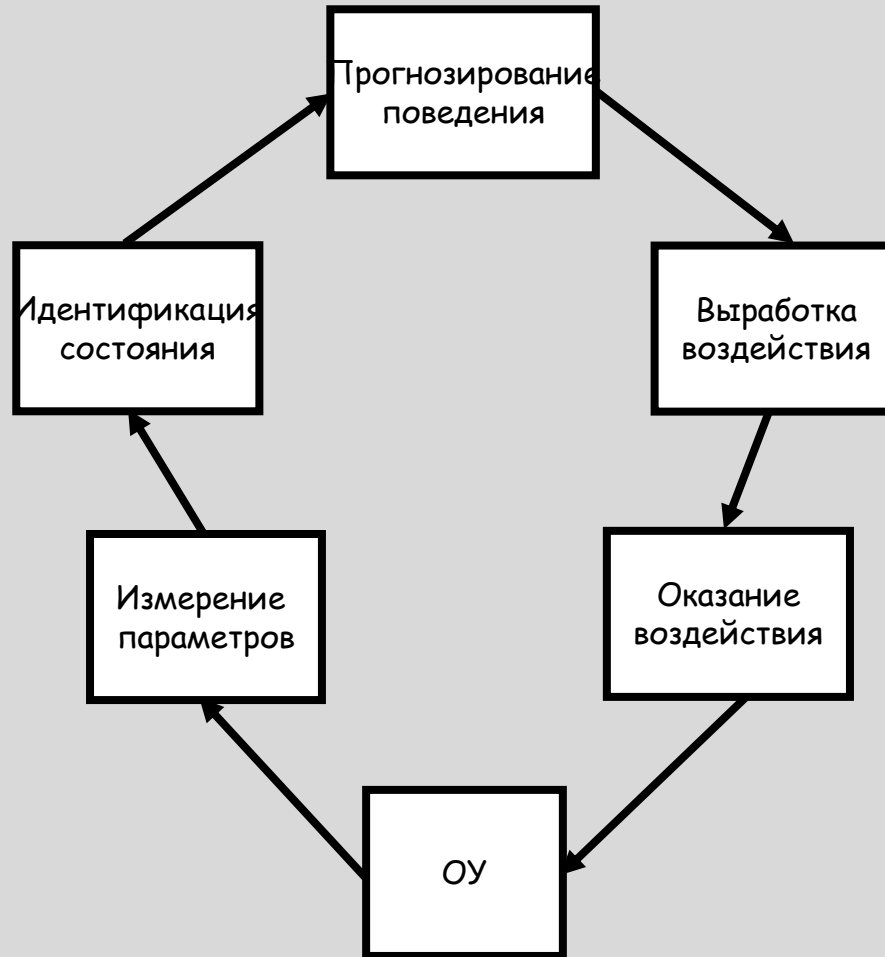


Система Управления и среда



ОБЩИЙ ПРИНЦИП РАБОТЫ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

(но взято из ТАУ)



Стандартное представление систем с критиком

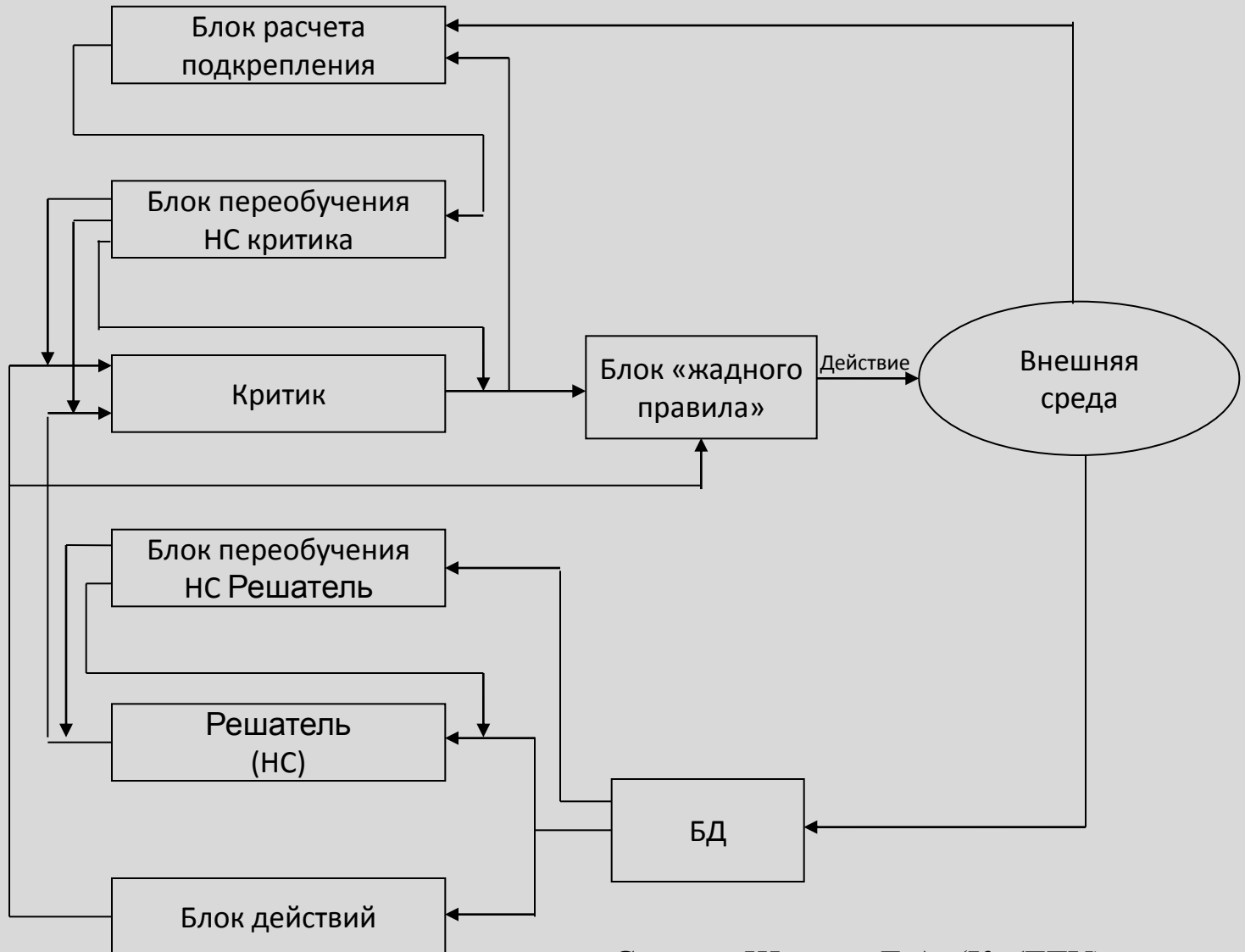


Схема – Шумков Е.А. (КубГТУ)

ДРУГОЕ ПРЕДСТАВЛЕНИЕ СХЕМЫ АДАПТИВНОГО КРИТИКА



Работа модели:

$$X_p = F\left(\sum_{i_2} w_{i_2 j_2 z} \dots \sum_{i_2} w_{i_2 j_2 2} * F\left(\sum_{i_1} w_{i_1 i_1 1} I_{i_1 j_1 1} - \theta_{i_1 j_1 1} \dots - \theta_{i_1 j_1 2}\right)\right)$$

Нейронные сети и глубокое обучение

S_t – состояние среды и объекта

a_t – действие агента

X_t – прогноз ситуации

r_t - подкрепление

ОШИБКА ВРЕМЕННОЙ РАЗНОСТИ

Прогноз функции ценности в момент времени t

$$V_{S_t} \leftarrow V(S_t) + \alpha \cdot [r_{t+1} + \gamma \cdot V(S_{t+q}) - V(S_t)]$$

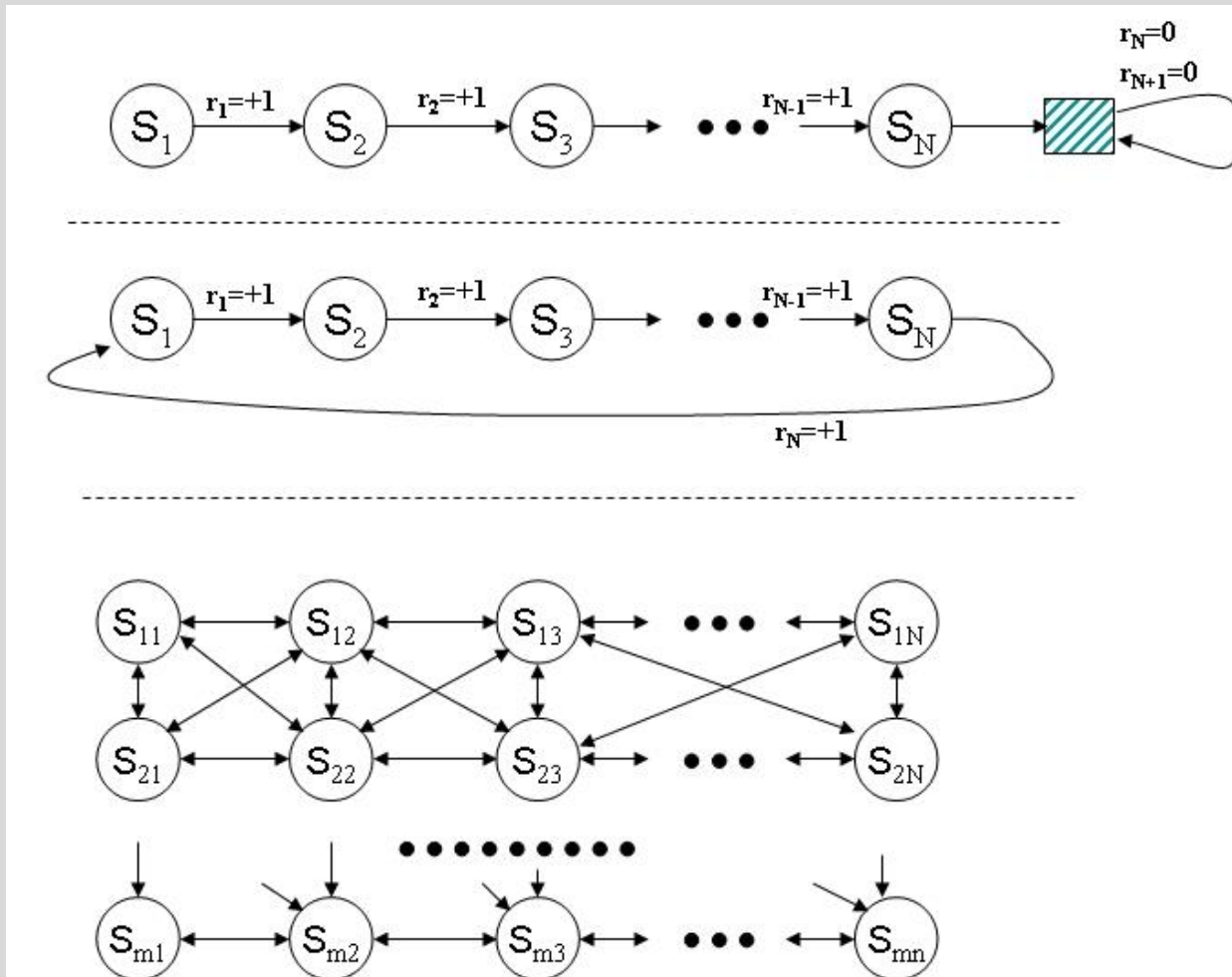
Корректировка прогноза

$$\Delta V_t = \alpha \cdot \left[\sum_{k=t}^{\infty} \gamma \cdot r_k - V_t \right]$$

или в терминах временной разности между следующими друг за другом прогнозами

$$\begin{aligned} \Delta V_t &= \alpha \cdot [(r_t + \gamma \cdot V_{t+1} - V_t) + \gamma(r_{t+1} + \gamma \cdot V_{t+2} - V_{t+1}) + \dots] = \\ &= \alpha \cdot \sum_{k=t}^{\infty} (r_t + \gamma \cdot V_{k+1} - V_k) \cdot \gamma^{(k-t)} \end{aligned}$$

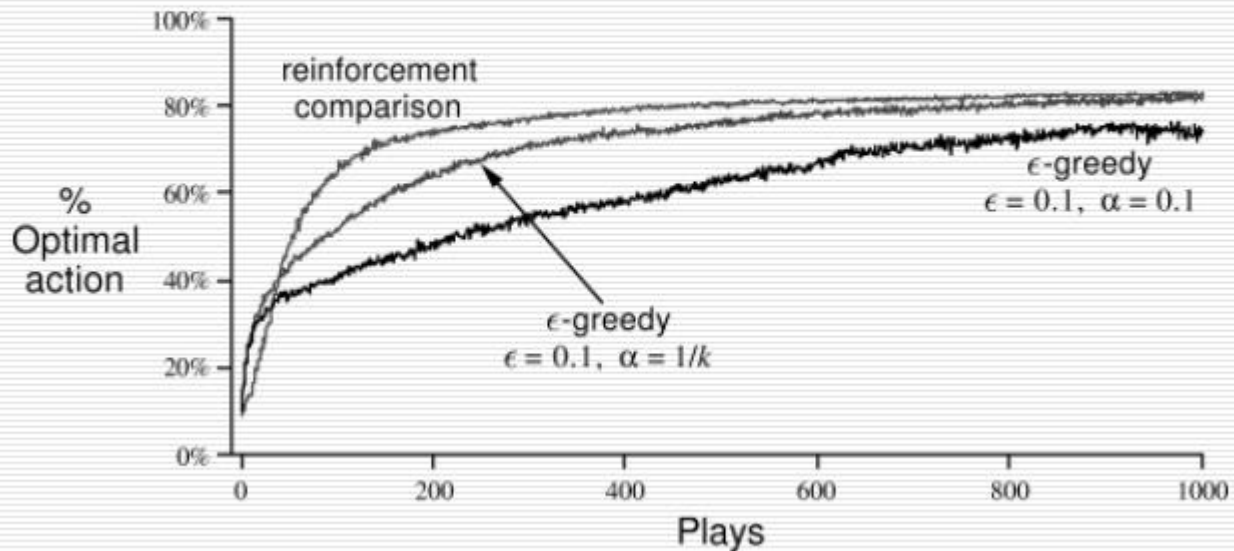
ЭПИЗОДЫ (переходы между состояниями)



«ЖАДНОЕ» - правило (ϵ – жадный алгоритм)

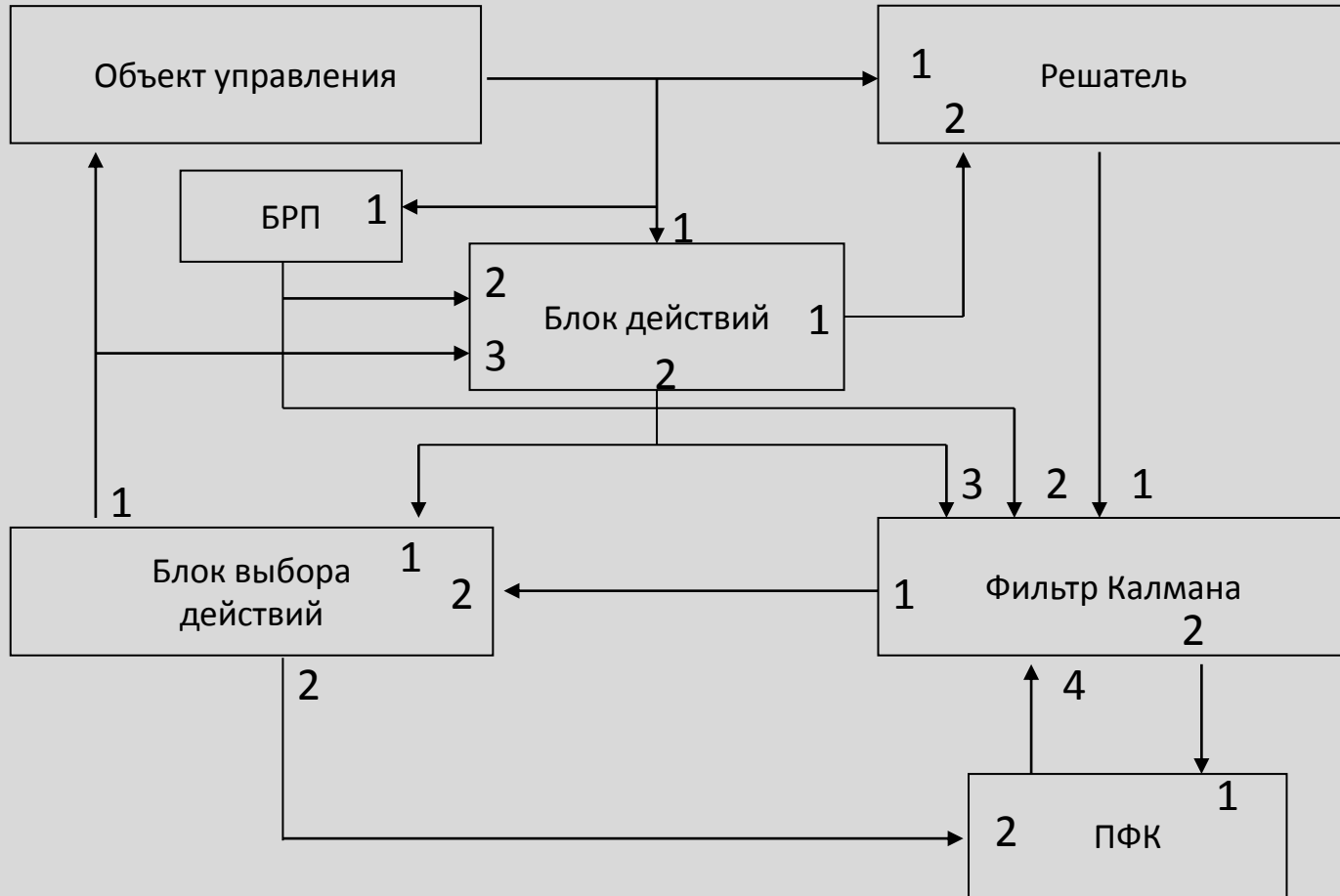
- 1) С вероятностью $(1-\epsilon)$ выбираем жадное действие
- 2) С вероятностью ϵ выбираем не жадное действие

Сравнение подкреплений



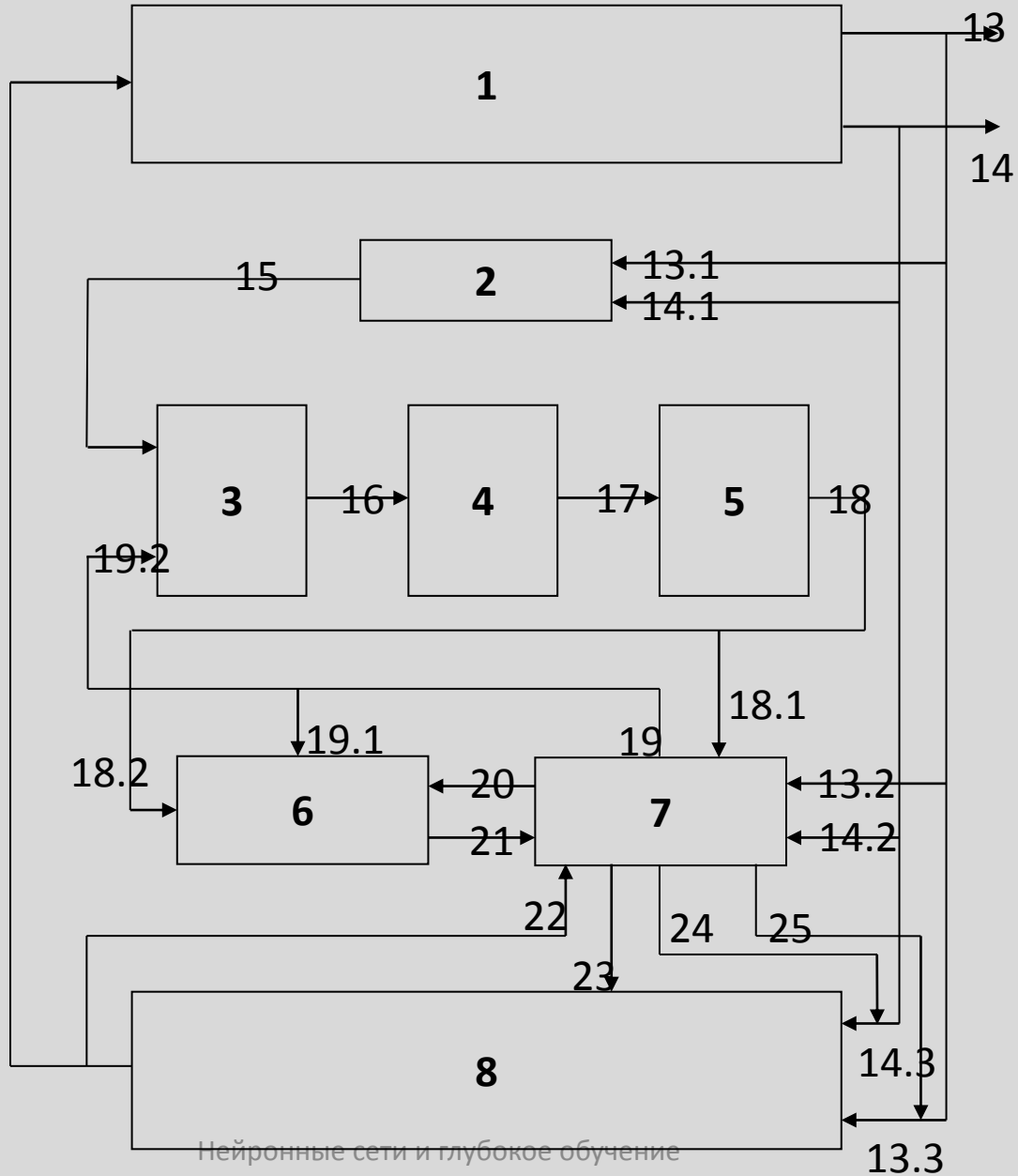
Жадный алгоритм — это алгоритм, заключающийся в принятии локально оптимальных решений на каждом этапе, допуская, что конечное решение также окажется оптимальным

ПРИМЕРЫ ПАТЕНТОВ НА БАЗЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ



С фильтром Калмана. Патент

Модифицированный интеллектуальный контроллер
с нечеткими правилами



патент № 2504002

Q - обучение

$$Q(s_t, a_t) \leftarrow r_t + \gamma \cdot \max_a Q(s_{t+1}, a)$$

Алгоритм Q - обучения

1. Для каждой пары состояние – действие инициализировать ячейку таблицы $Q(s_i, a_j)$

в ноль (или небольшими случайными величинами).

2. Обследовать текущее состояние среды и объекта управления.

3. Цикл пока не будет достигнуто терминальное состояние (или когда задача агента будет выполнена):

3.1. Выбрать действие агента a

и выполнить его.

3.2. Получить подкрепление r

3.3. Обследовать новое состояние среды S_{t+1} и объекта управления.

3.4. Обновить ячейку таблицы для пары состояние – действие по формуле $Q(s, a) \leftarrow r + \gamma \cdot \max_{a' \in A} Q(s', a')$

3.5 Обновить $s \leftarrow s'$

Алгоритмы Q - обучения

Без использования нейронных сетей:

- Q – обучение с использованием Хеммингова расстояния;
- Q – обучение с статической кластеризацией.

С использованием нейронных сетей:

- Q – обучение на основе послойно - полносвязных нейронных сетей прямого распространения;
- Q – обучение на основе сети Кохонена;
- Dyna – Q;
- Competitive MLP (на основе конкуренции).

Представление Q – таблицы (ситуация - действие)

	s_1	s_2	s_3	...	s_n
a_1	Q_{11}	Q_{12}	Q_{13}	...	Q_{1n}
a_2	Q_{21}	Q_{22}	Q_{23}	...	Q_{2n}
a_3	Q_{31}	Q_{32}	Q_{33}	...	Q_{3n}
				...	
a_m	Q_{m1}	Q_{m2}	Q_{m3}	...	Q_{mn}

ВЕРОЯТНОСТИ ПЕРЕХОДА СИСТЕМЫ ИЗ ОДНОГО СОСТОЯНИЯ В ДРУГОЕ

	s_1	s_2	s_3	...	s_n
s_1	V_{11}	V_{12}	V_{13}	...	V_{1n}
s_2	V_{21}
s_3	V_{31}
...
s_n	V_{n1}

хранения оценок Q-значений.

action \ state	s_1	s_2	...	s_6
a_1 (NORTH)	$Q(s_1, a_1)$	$Q(s_1, a_1)$...	$Q(s_1, a_1)$
a_2 (EAST)	$Q(s_1, a_2)$	$Q(s_1, a_2)$...	$Q(s_1, a_2)$
a_3 (WEST)	$Q(s_1, a_3)$	$Q(s_1, a_3)$...	$Q(s_1, a_3)$
a_4 (SOUTH)	$Q(s_1, a_4)$	$Q(s_1, a_4)$...	$Q(s_1, a_4)$

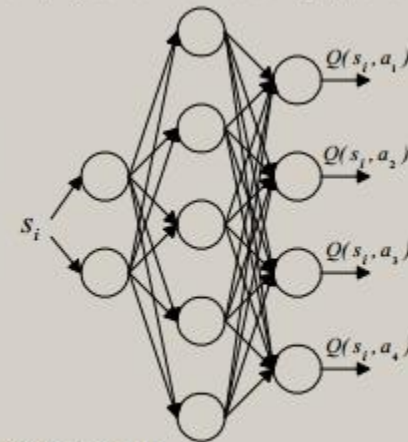


Рис.2.6. Замена средства представления Q-функции

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?

ЛЕКЦИЯ 6-3

Тема: «Адаптивные критики (продолжение) и глубокое обучение с подкреплением»

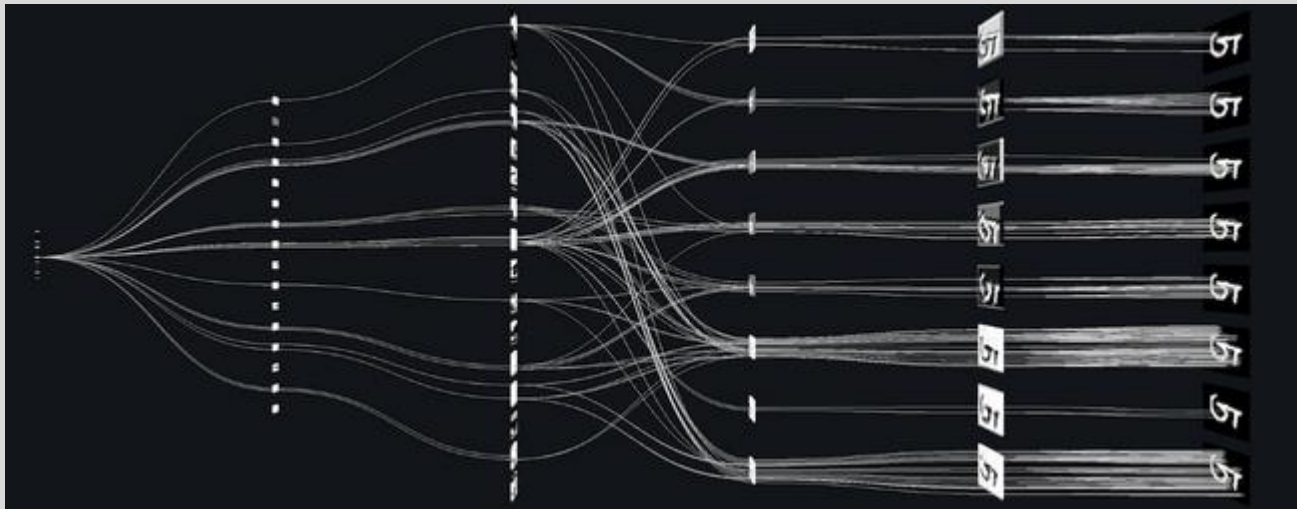
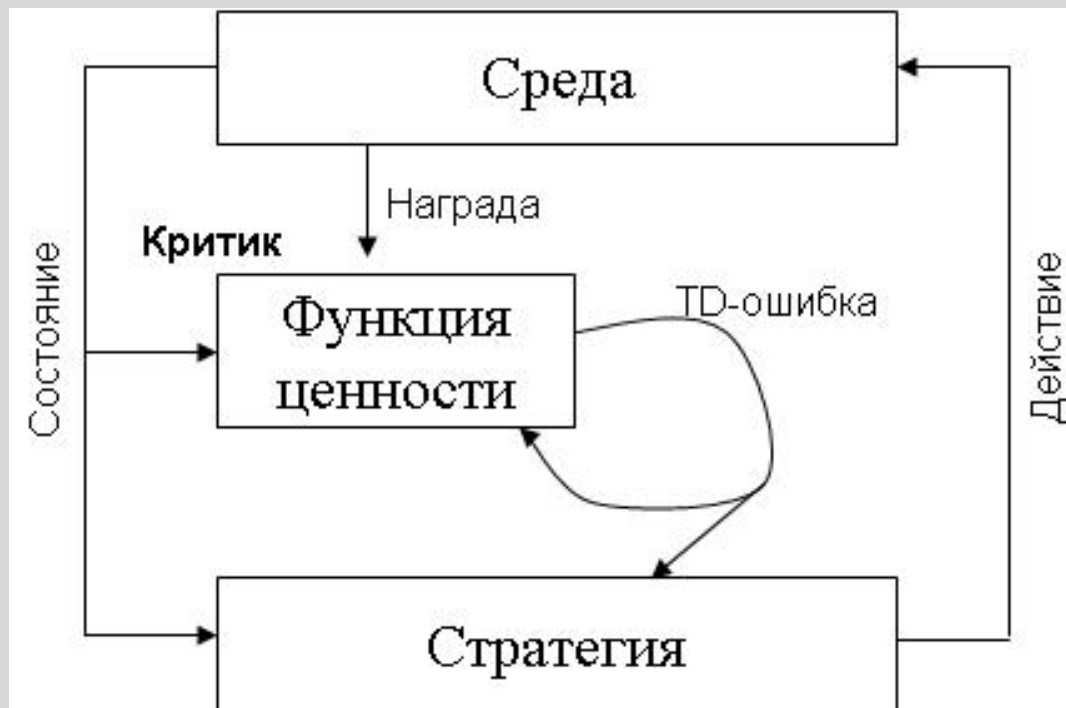


рис - хабрахабр

СЕТИ АДАПТИВНОЙ КРИТИКИ

Обобщенная схема функционирования адаптивных критиков



АЛГОРИТМ SARSA

$$S(t) \rightarrow a(t) \rightarrow r(t) \rightarrow S(t+1) \rightarrow a(t+1)$$

$$Q(S(t), a(t)) = E\{r(t) + \gamma \cdot Q(S(t+1), a(t+1)) + \gamma^2 \cdot r(t+1) + \dots\} \mid S = S(t), a = a(t)$$

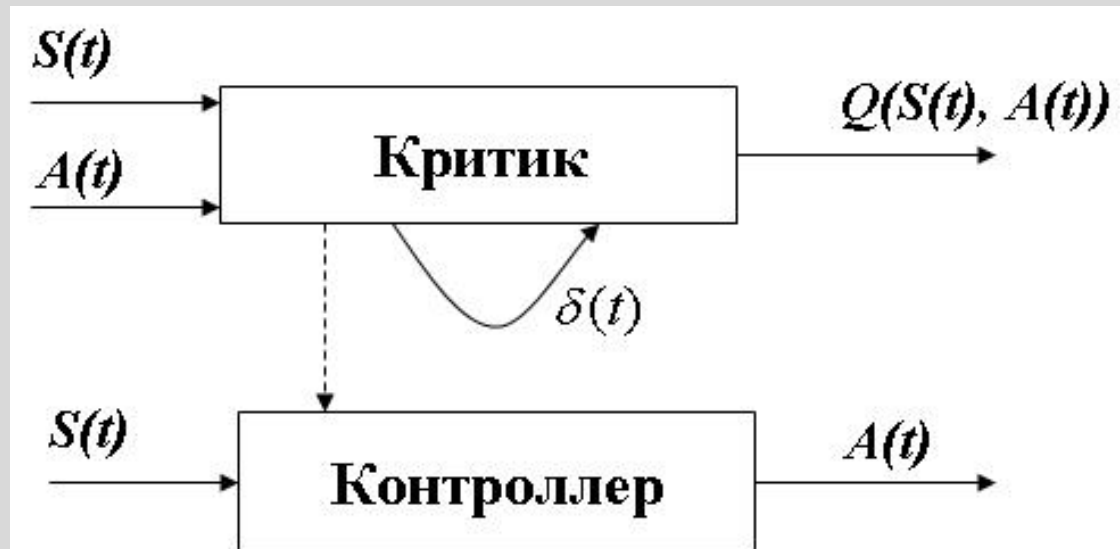
Ошибка временной разности

$$\delta(t) = r(t) + \gamma \cdot Q(S(t+1), a(t+1)) - Q(S(t), a(t))$$

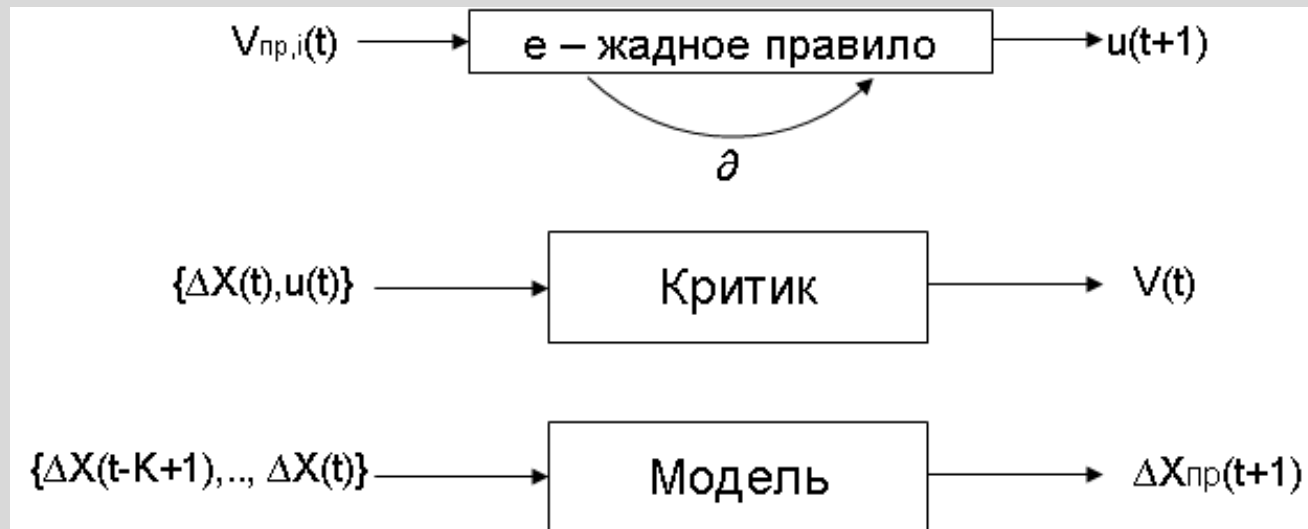
Переоценка

$$\Delta Q(S(t), a(t)) = \alpha \cdot \delta(t) = \alpha \cdot [r(t) + \gamma \cdot Q(S(t+1), a(t+1)) - Q(S(t), a(t))]$$

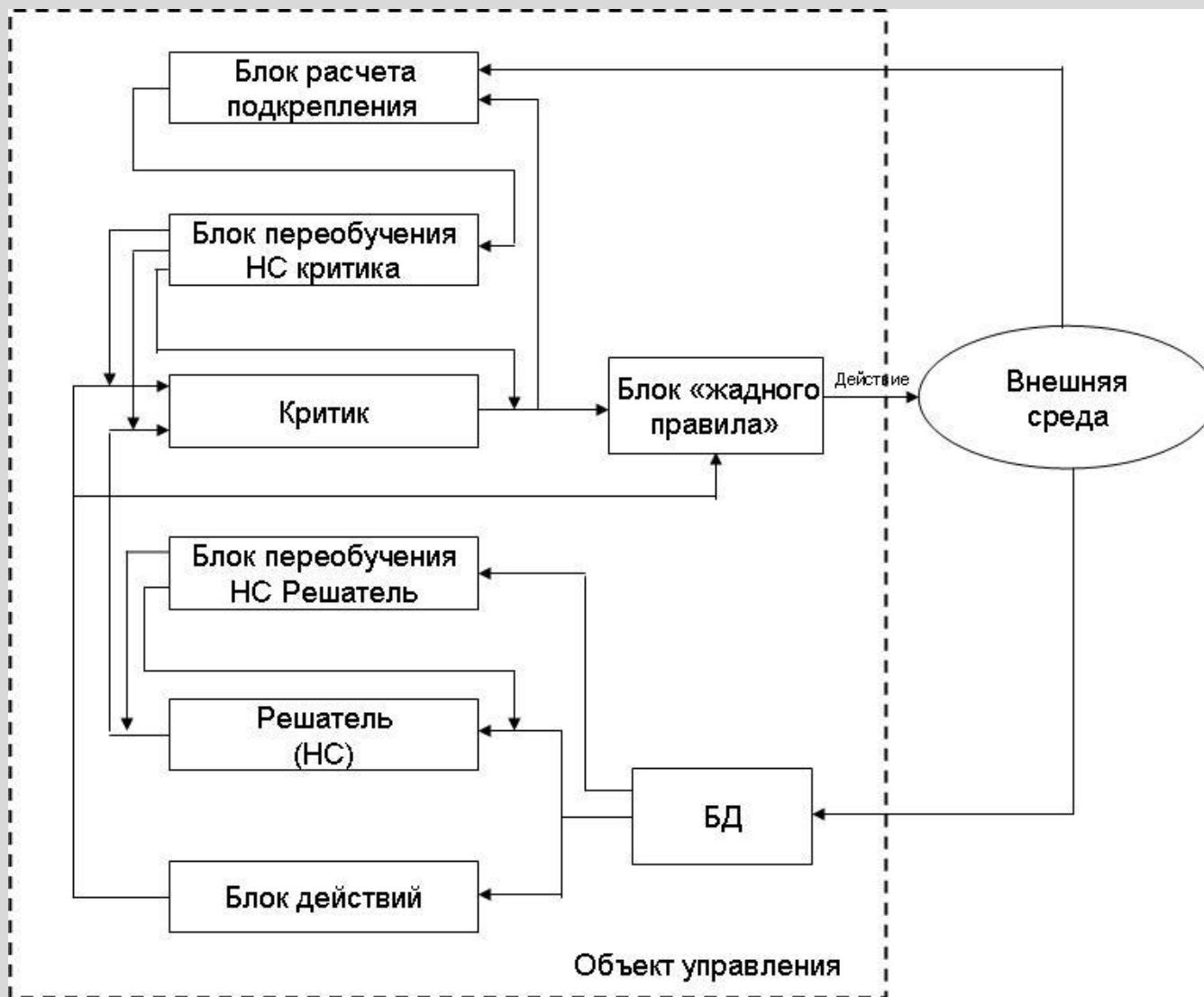
Q - критик



V - критик



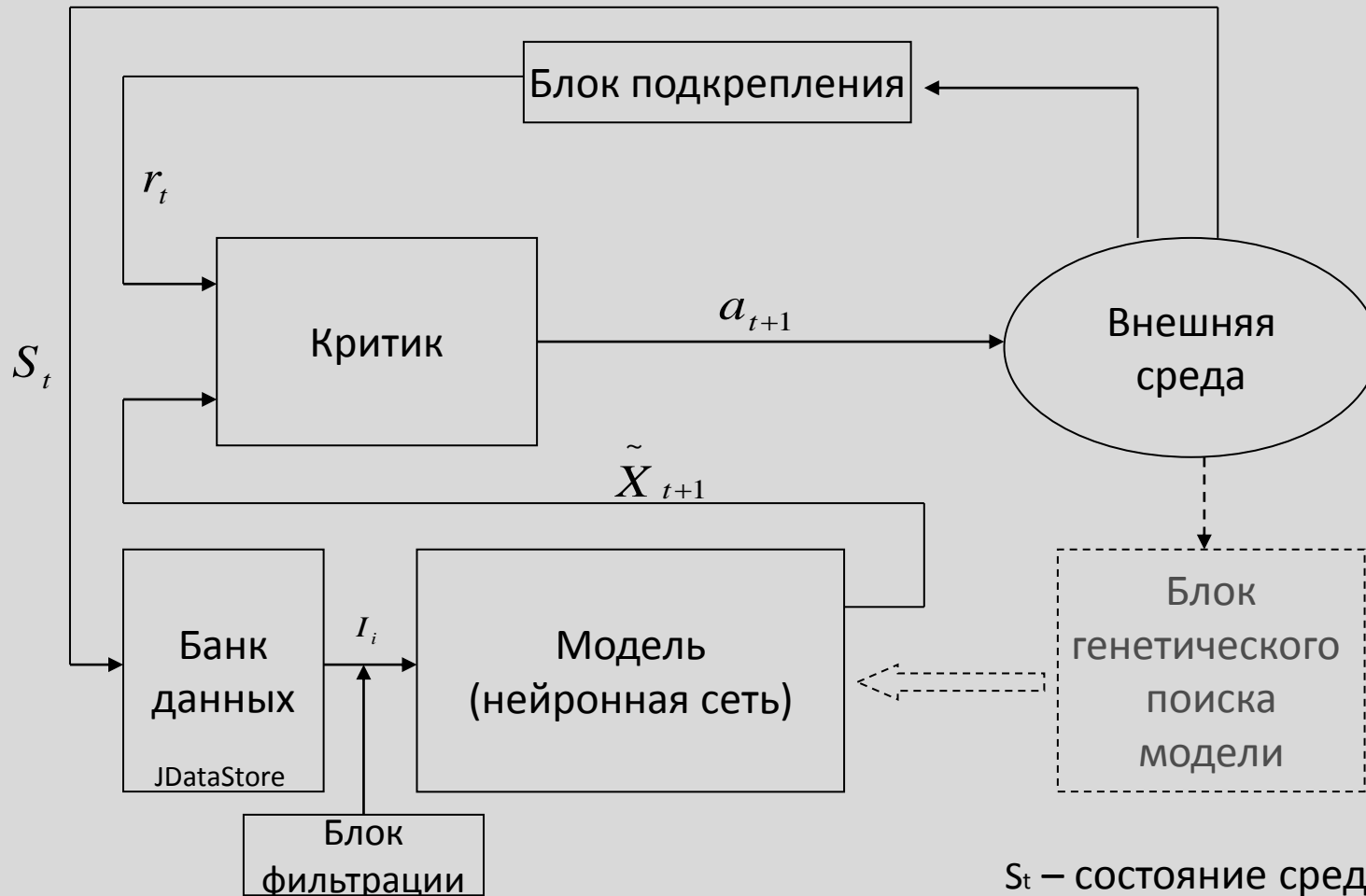
Обобщенная схема адаптивного критика



Шумков Е.А., Ботин В.А.

Еще одна схема Адаптивного критика

Сеть адаптивной критики



Работа модели:

$$X_p = F\left(\sum_{I_z} w_{i_z j_z z} \dots \sum_{I_2} w_{i_2 j_2 2} * F\left(\sum_{I_1} w_{i_1 j_1 1} I_{i_1 j_1 1} - \theta_{j_1 1}\right) - \theta_{j_2 2} - \dots - \theta_{j_z z}\right)$$

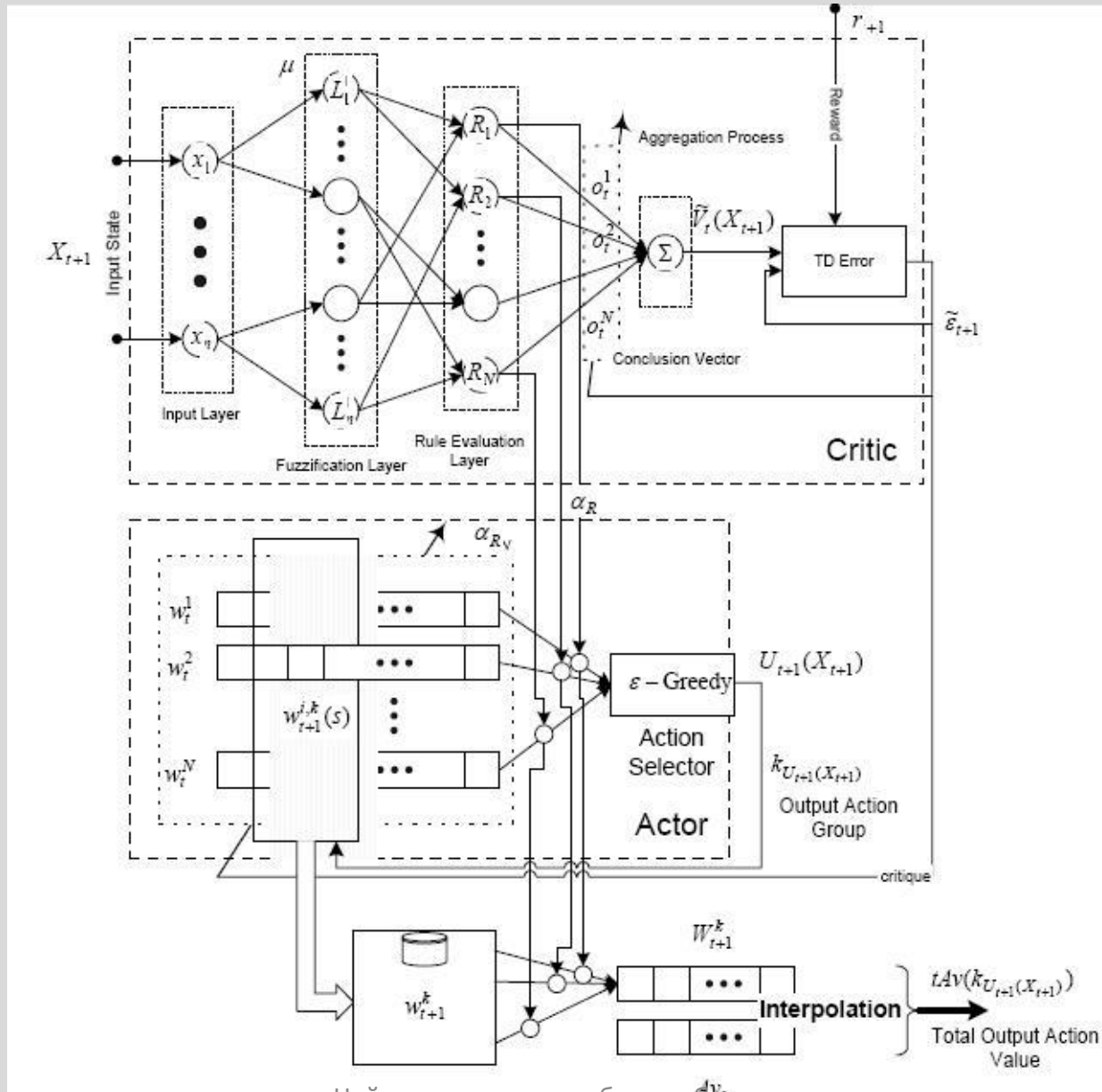
S_t – состояние среды и объекта

a_t – действие агента

X_t – прогноз ситуации

r_t - подкрепление

Адаптивный критик с генетическим алгоритмом



Расшифровка

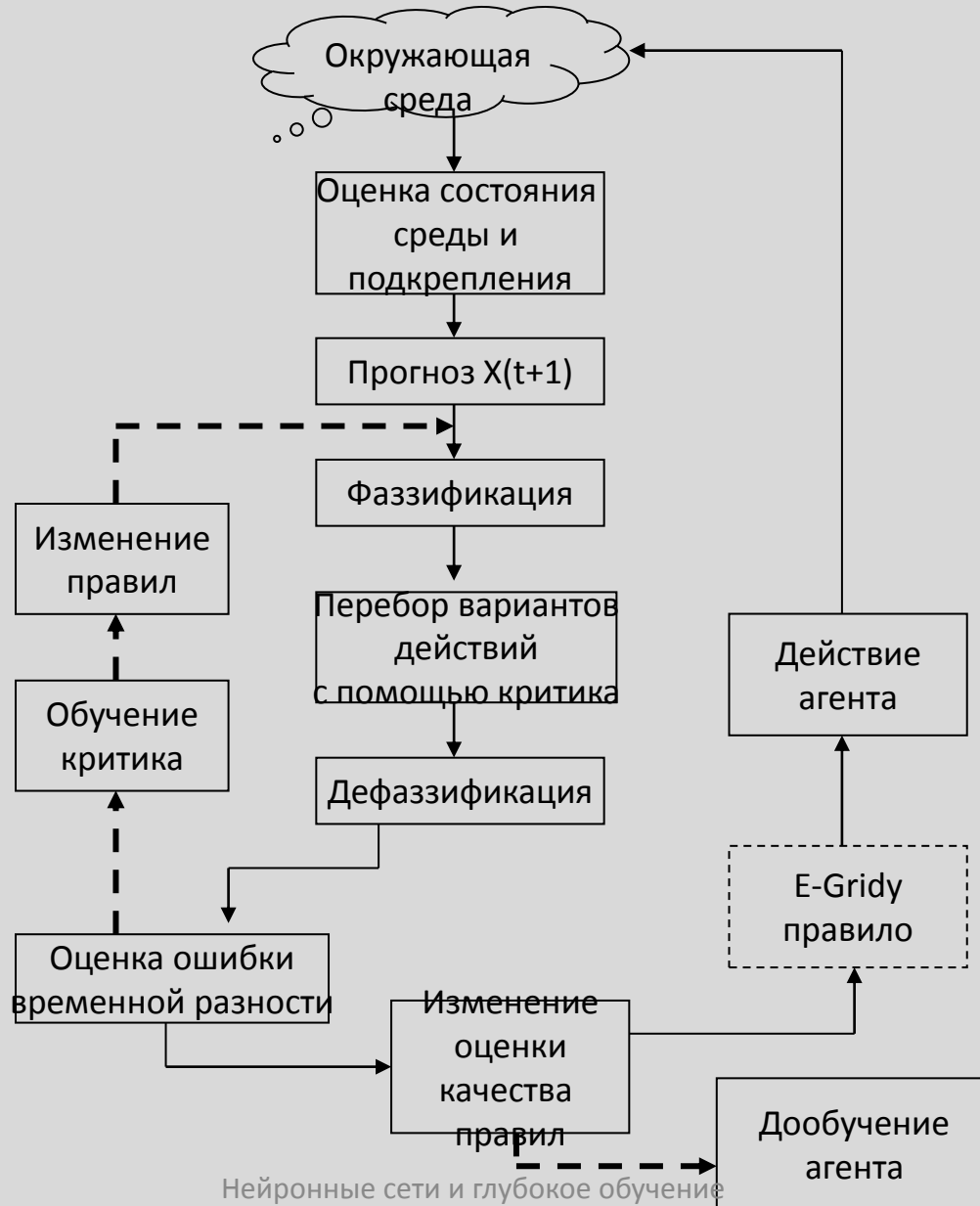
$$X_p = F\left(\sum_{i_z} w_{i_z j_z} z \cdots \sum_{i_2} w_{i_2 j_2} 2 * F\left(\sum_{i_1} w_{i_1 j_1} I_{i_1 j_1} - \theta_{j_1}\right) - \theta_{j_2} - \dots - \theta_{j_z}\right)$$

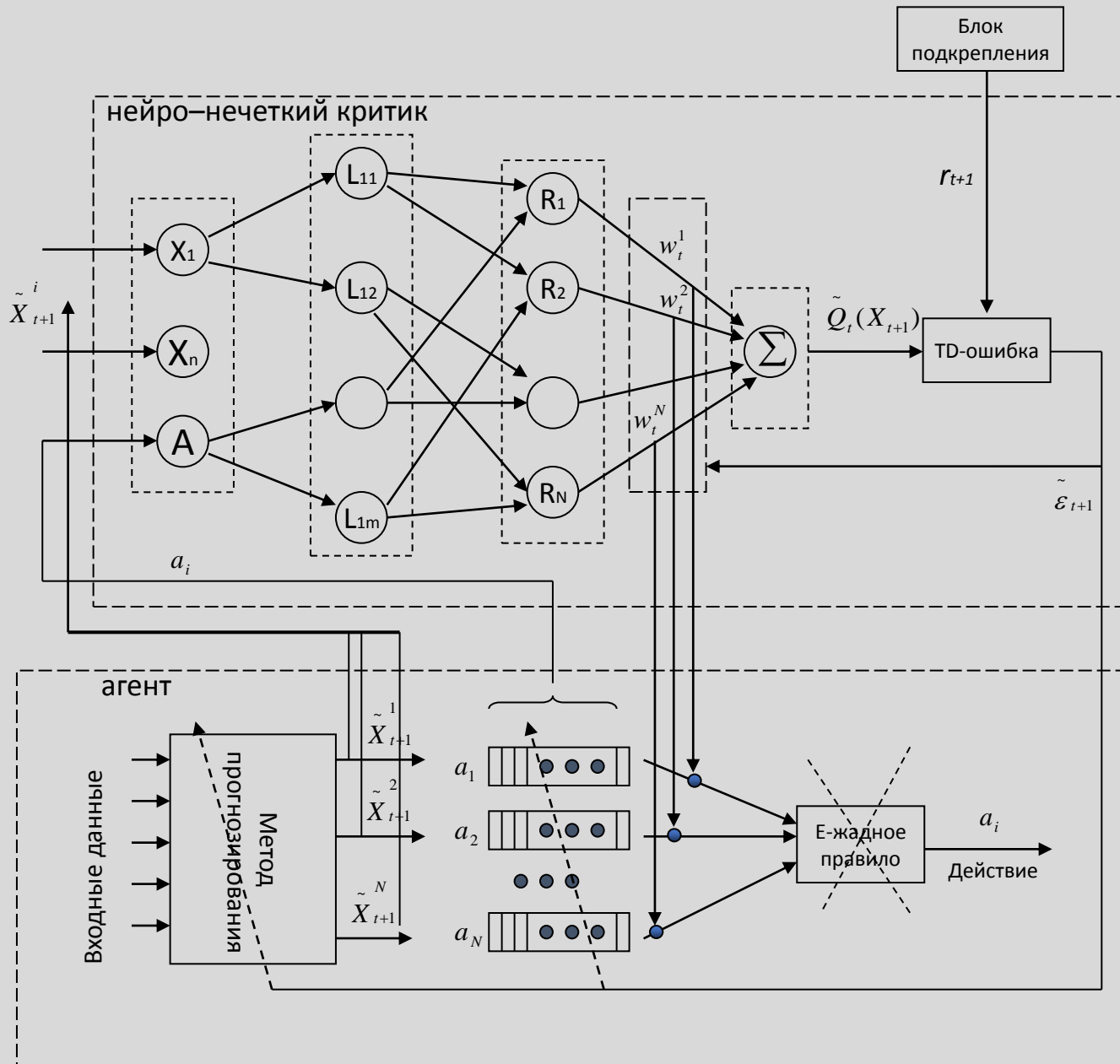
$$Q(S(t), a(t)) = E\{r(t) + \gamma * r(t+1) + \gamma^2 * r(t+2) + \dots + \gamma^{n-1} (t+n)\}$$

$$Q(S(t), a(t)) = E\{r(t) + \gamma * Q(S(t+1), a(t+1))\}$$

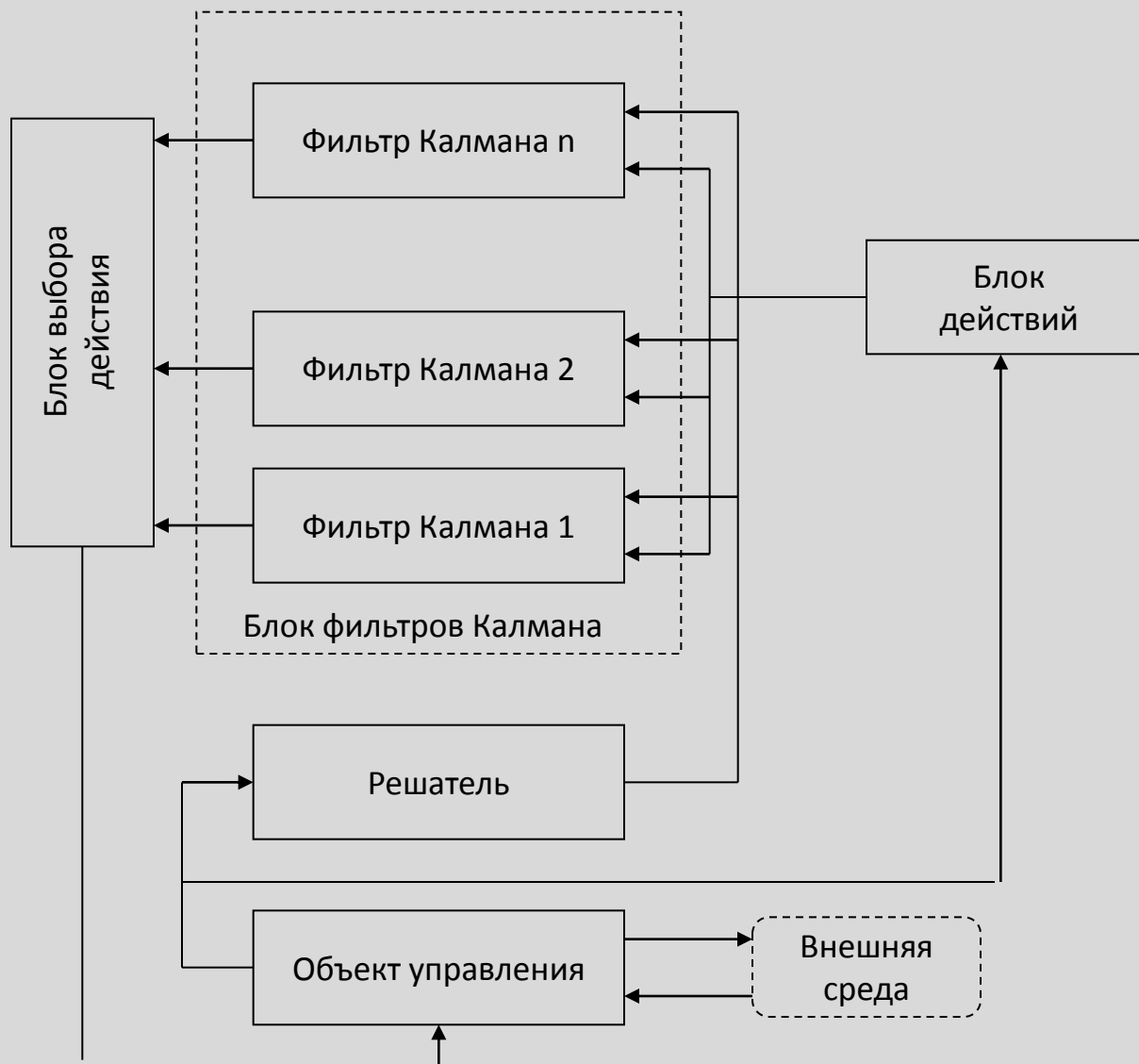
$$\delta = r(t) + \gamma * Q(S(t+1), a(t+1)) - Q(S(t), a(t))$$

Алгоритм работы





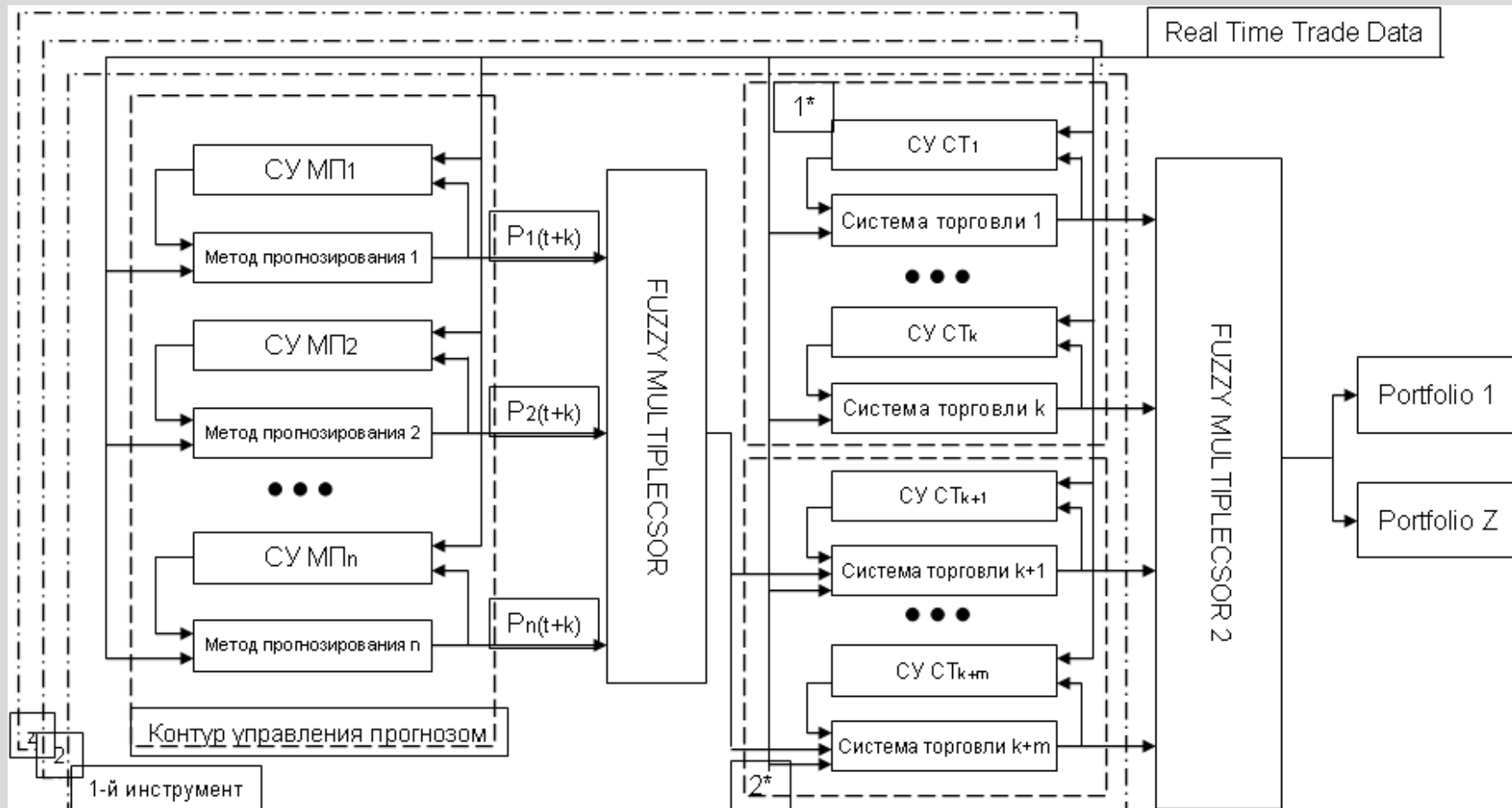
Как вариант – критик на базе фильтра Калмана



Автор – Ботин Валерий (КубГТУ)

ПРИМЕНЕНИЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ

Механические торговые системы (сложная схема МТС – Карлов Д.Н., КубГТУ)



Постановка задачи построения механической торговой системы на адаптивном критике

Пусть:

$C(t)$ - общий капитал игрока в момент времени

$A(t)$ - капитал игрока, вложенный в акции в момент времени

$N(t)$ - капитал игрока, лежащий на его счете в момент времени

Тогда

$$C(t) = N(t) + A(t)$$

Пусть МТС может совершать следующие действия

а) k - купить акций на $k\%$ от капитала $N(t)$

б) 0 - не предпринимать никаких действий

в) $(-k)$ - продать акций на $k\%$ от капитала $N(t)$

Критик предназначен для оценки суммарной награды $V(t) = \sum_K \gamma^K \cdot r(t + k)$

$$\sigma(t) = r(t) + \gamma \cdot V(t) - V(t - 1)$$

γ - Коэффициент забывания (0;1]

ГЛУБОКОЕ ОБУЧЕНИЕ В МЕДИЦИНЕ

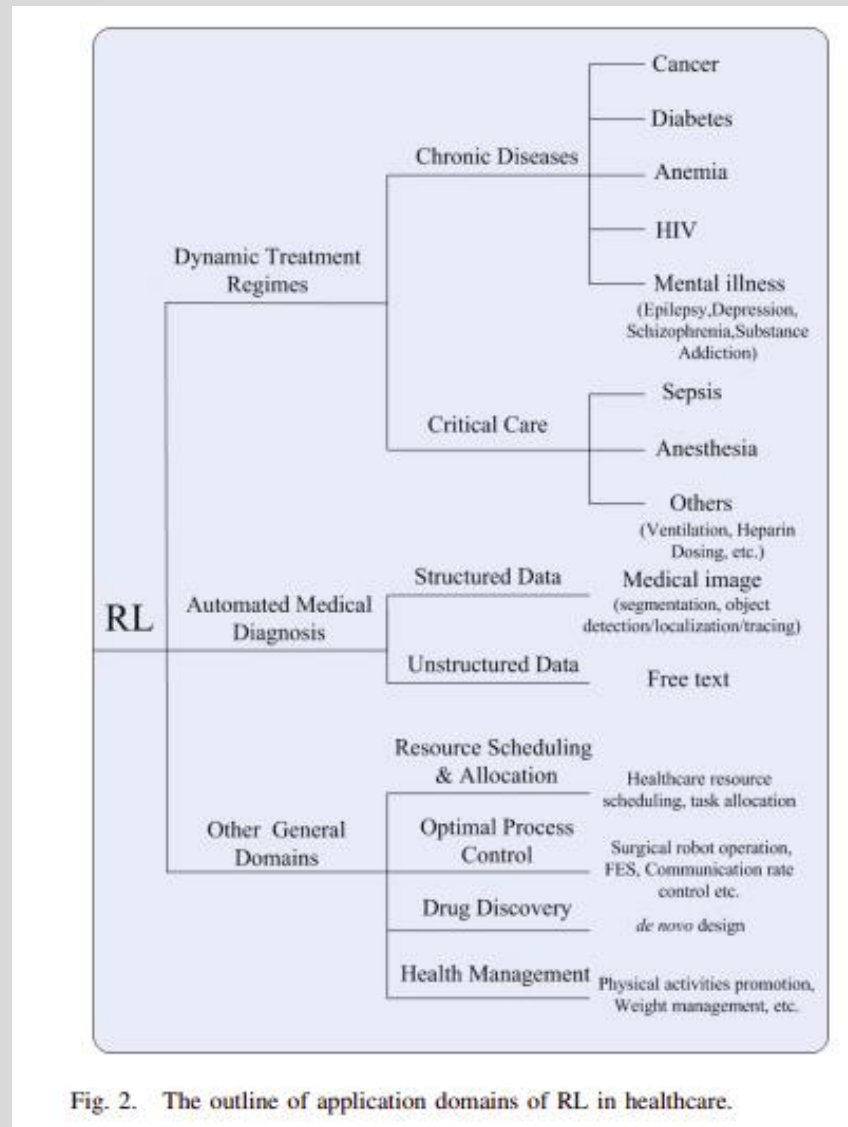
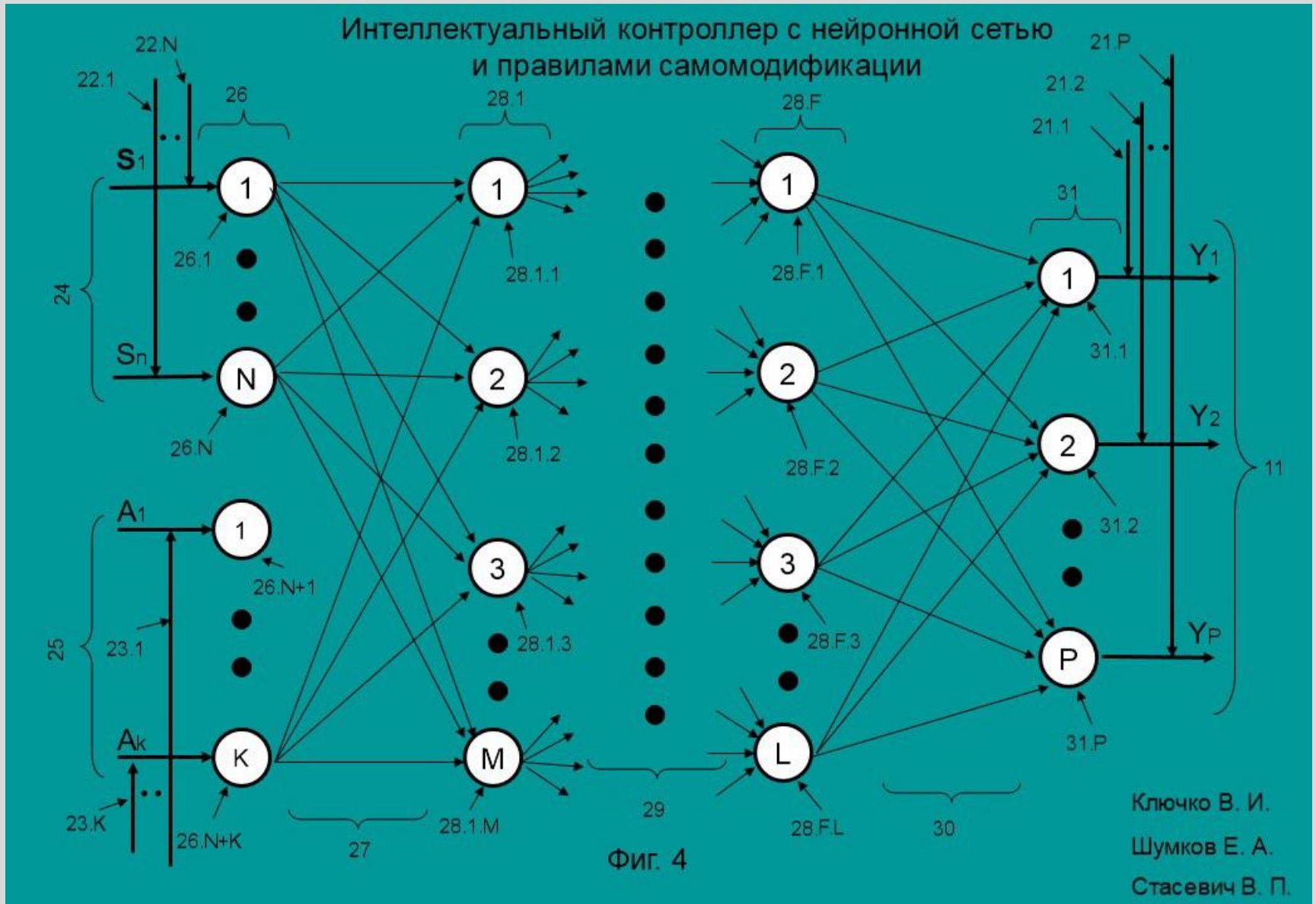


Fig. 2. The outline of application domains of RL in healthcare.

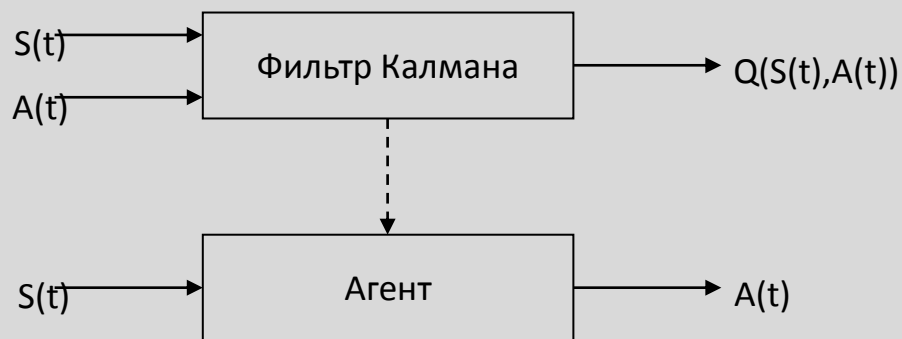
Источник - <https://neptune.ai/blog/reinforcement-learning-applications>

ПРИМЕРЫ ТОПОЛОГИЙ С АДАПТИВНЫМ КРИТИКОМ



Другие варианты реализации адаптивного критика с фильтром Калмана

Q-критик с использованием фильтра Калмана



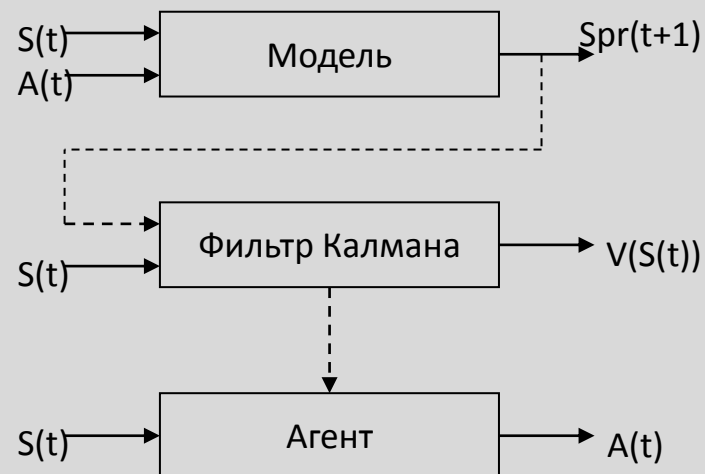
$S(t)$ - состояние

$A(t)$ - действие

$Q(S(t), A(t))$ – оценка качества

Ботин В. (КубГТУ)

V-критик с использованием фильтра Калмана



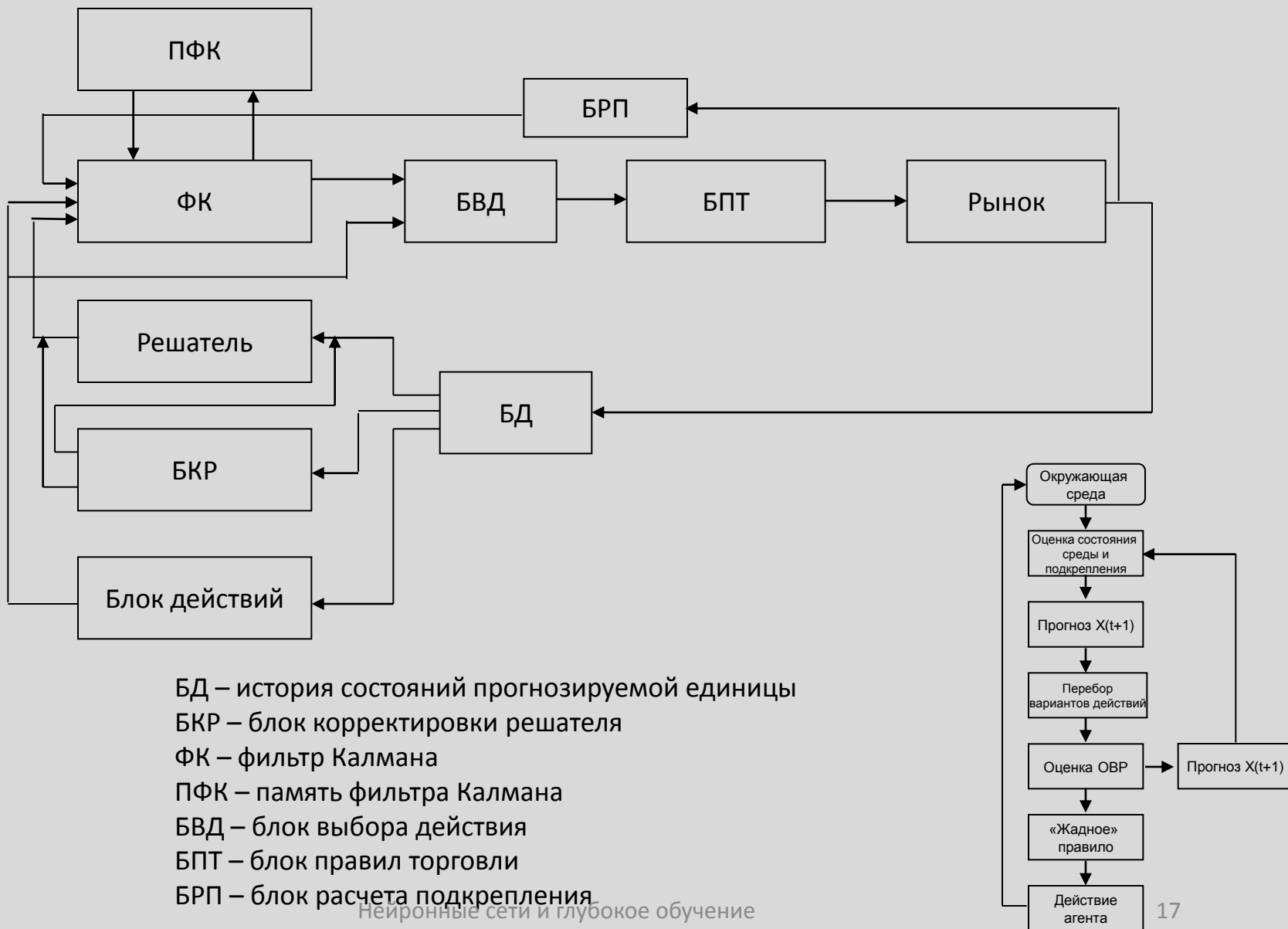
$S(t)$ - состояние

$A(t)$ - действие

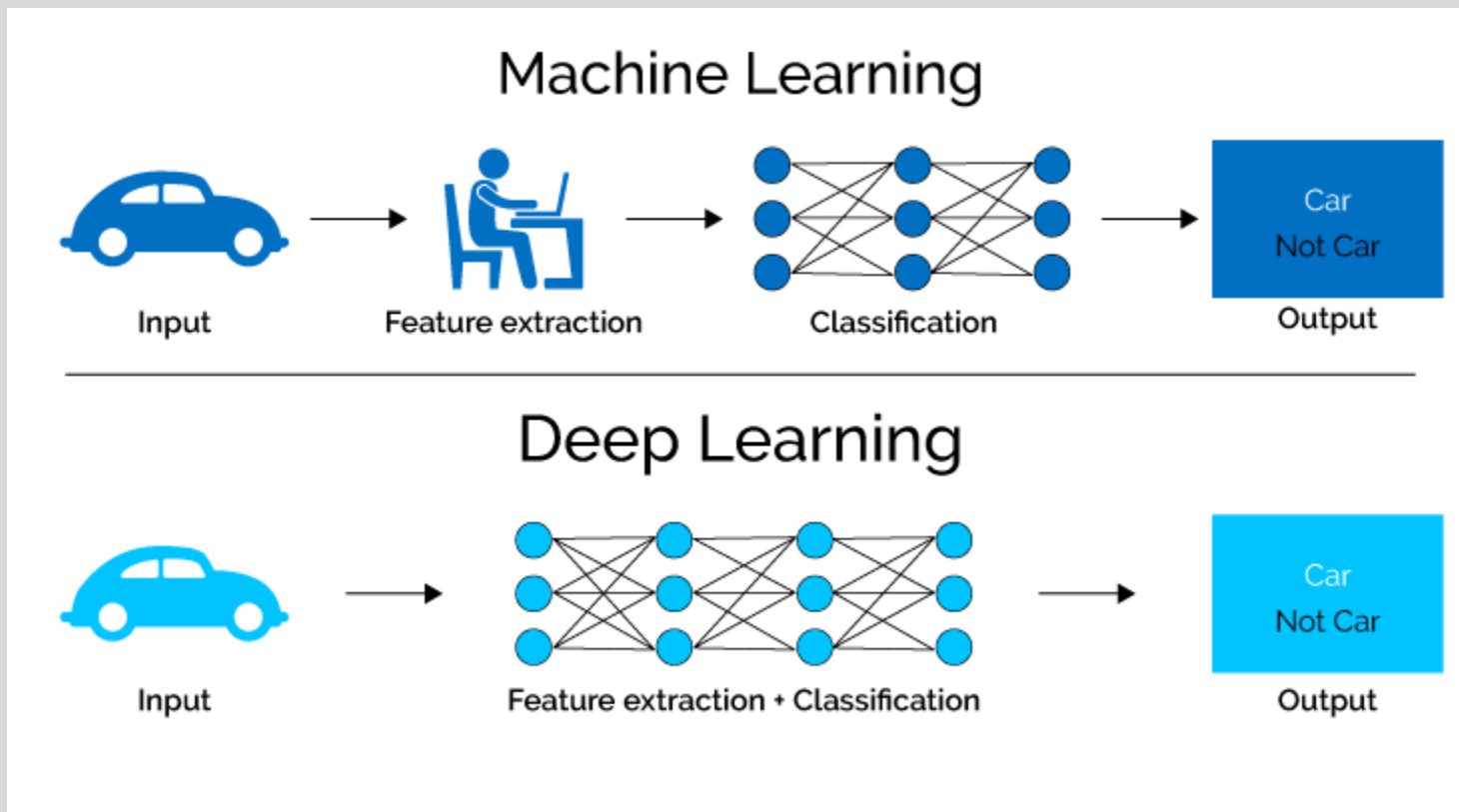
$Spr(t+1)$ - прогноз

$V(S(t))$ – оценка качества

Механическая Торговая Система на базе фильтра Калмана

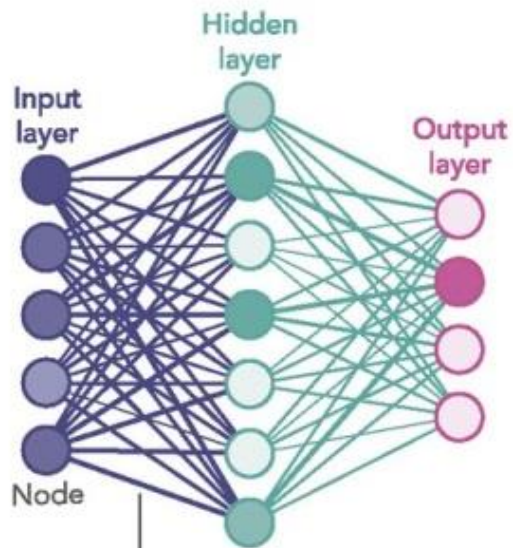


РАЗНИЦА МЕЖДУ МАШИНЫМ И ГЛУБОКИМ ОБУЧЕНИЕМ



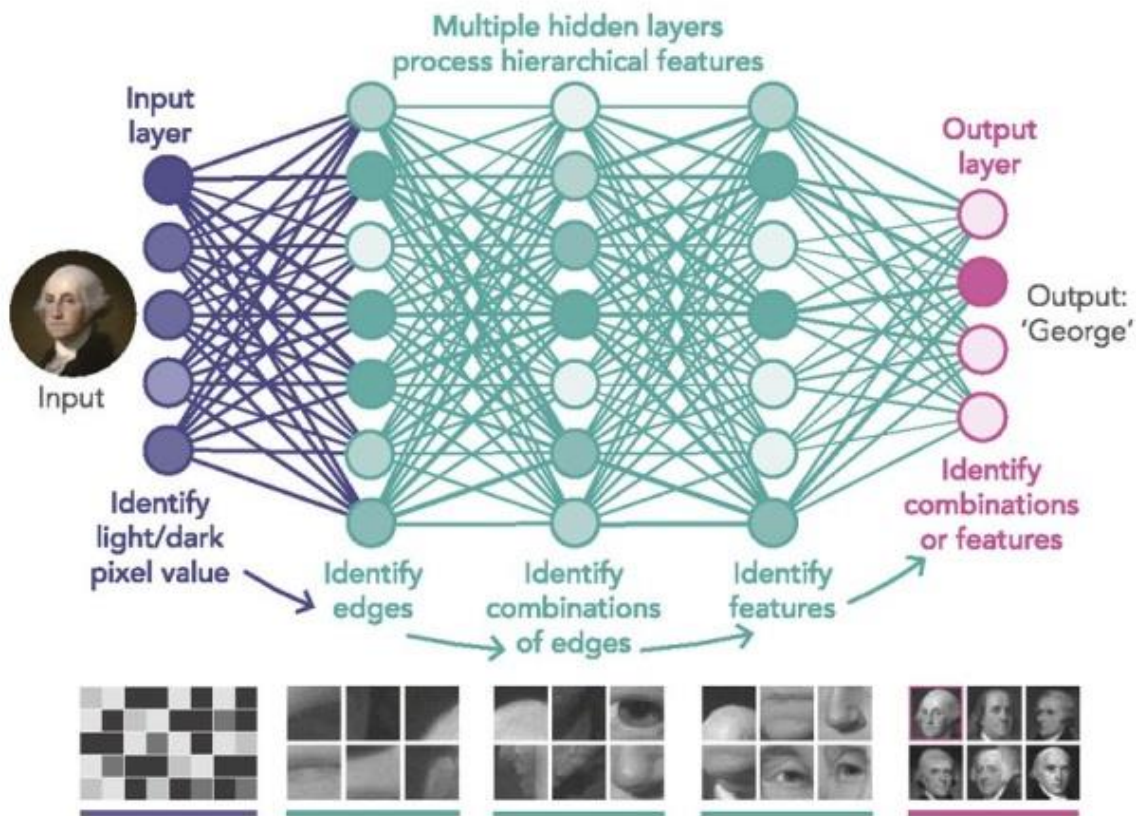
ОБЫЧНОЕ ПОНИМАНИЕ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ

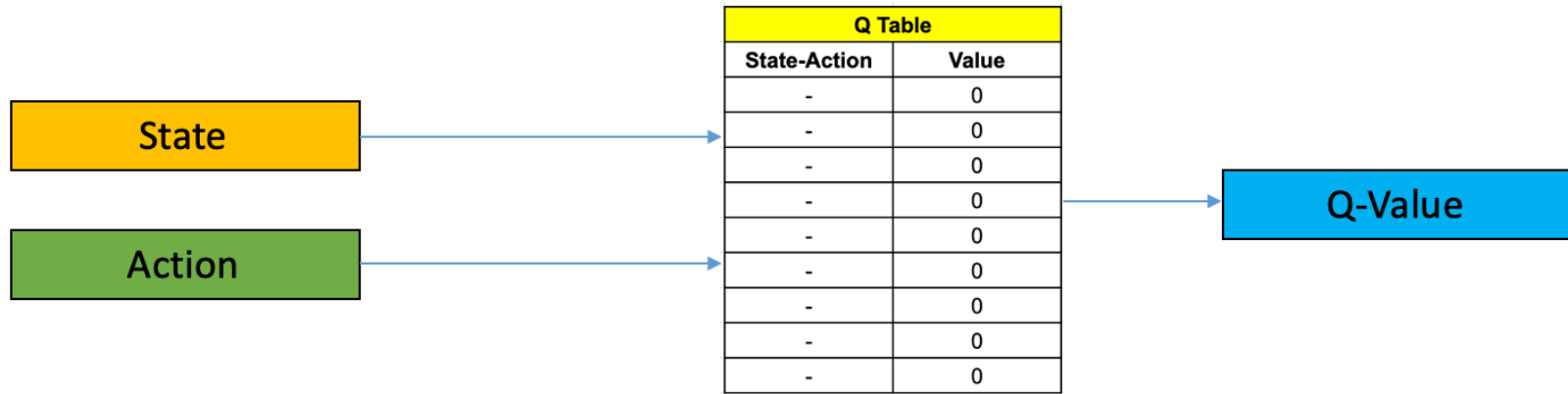
1980S-ERA NEURAL NETWORK



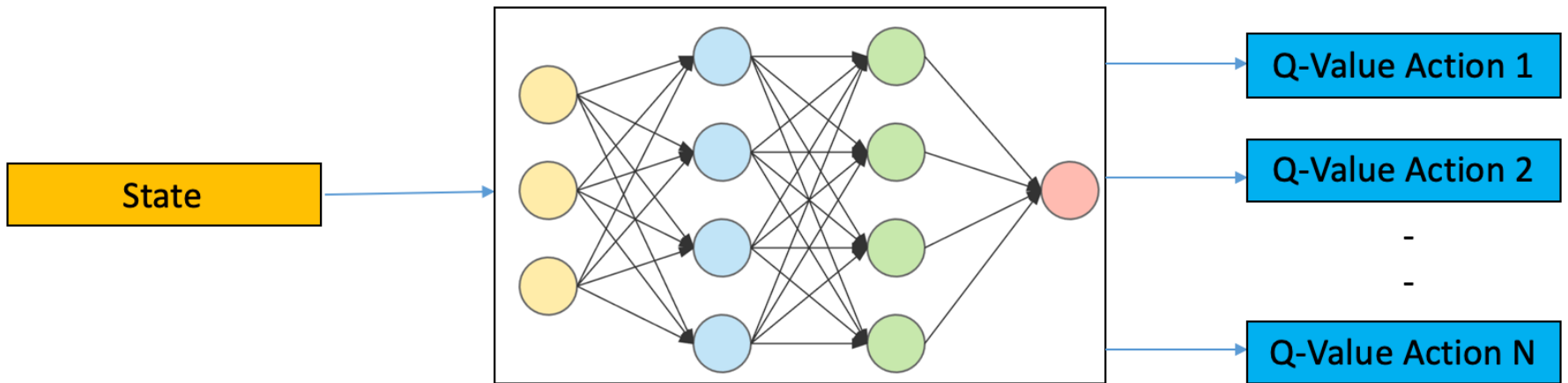
Links carry signals from one node to another, boosting or damping them according to each link's 'weight'.

DEEP LEARNING NEURAL NETWORK





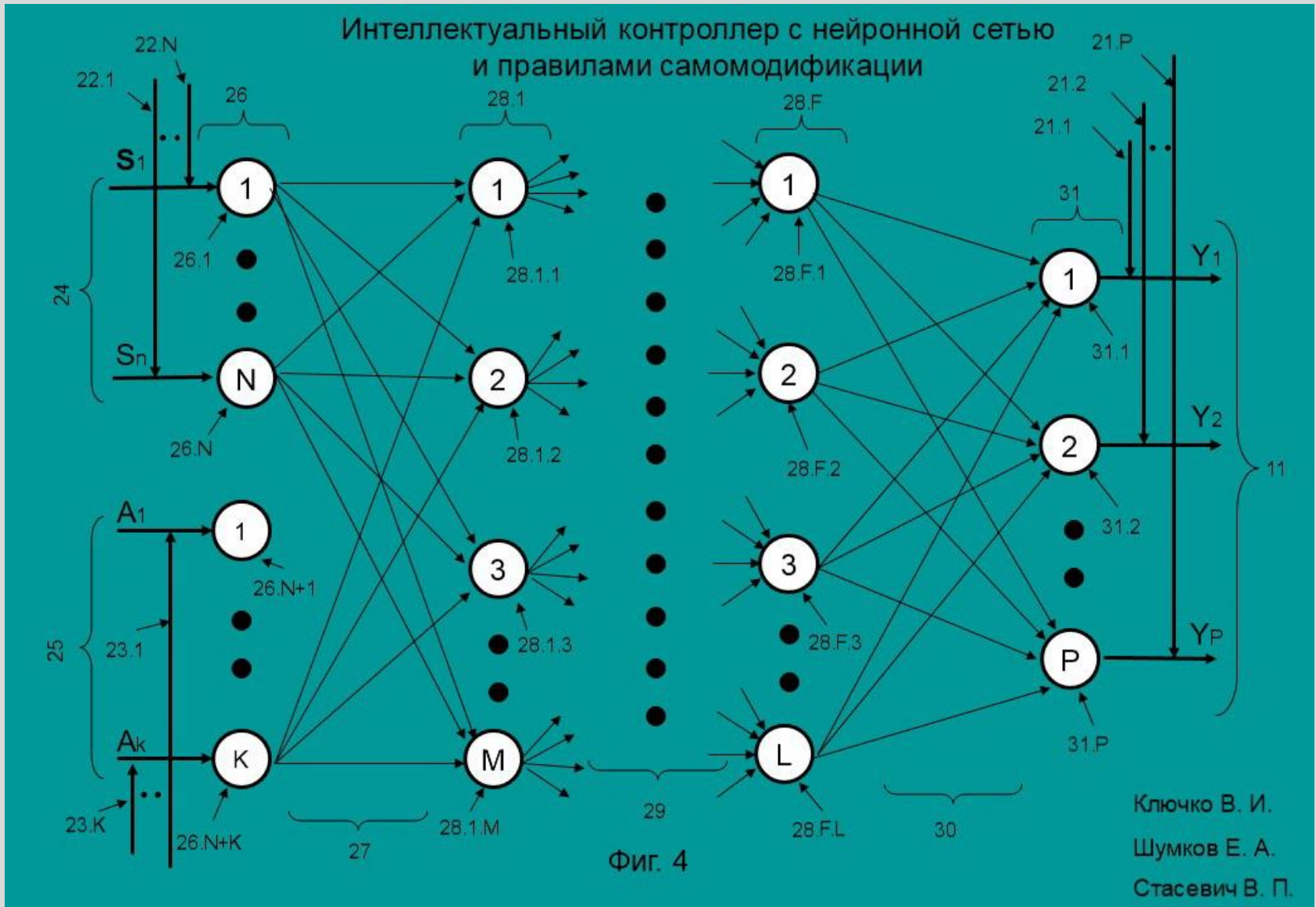
Q Learning



Deep Q Learning

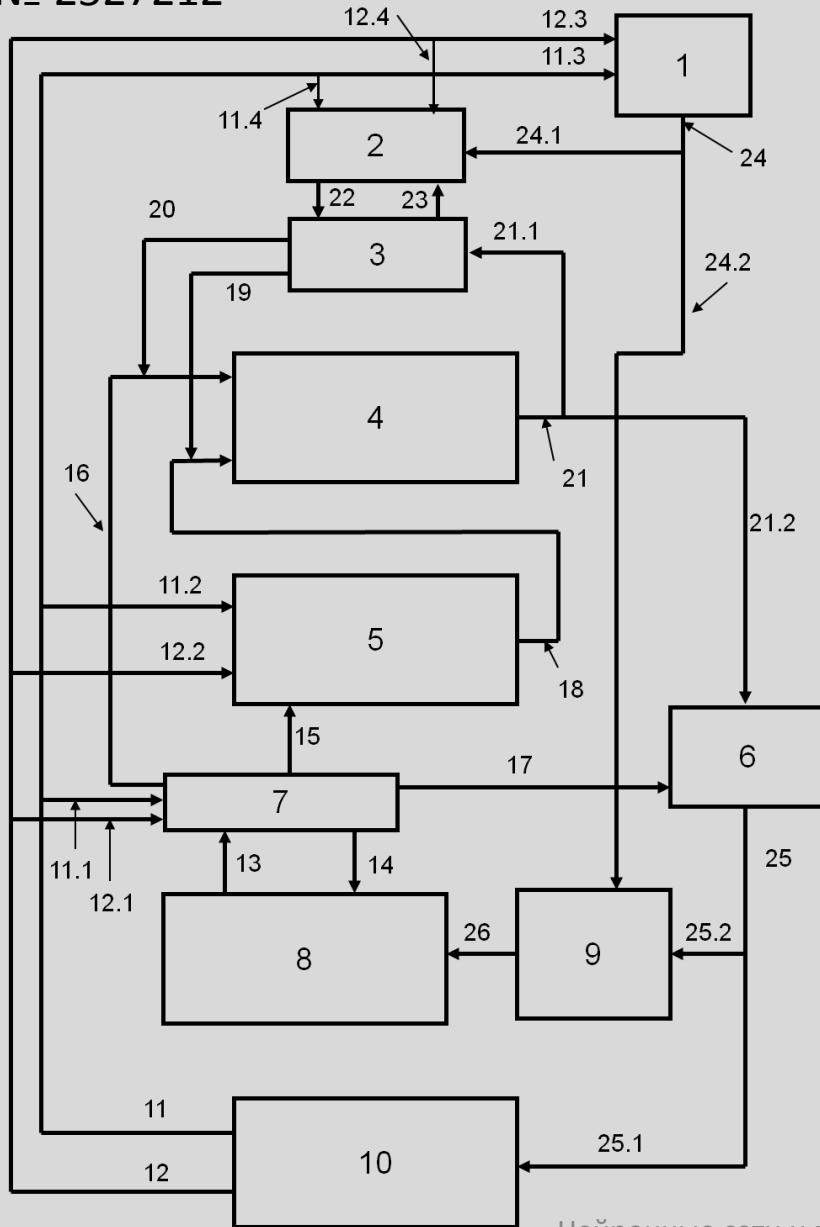
<https://www.v7labs.com/blog/deep-reinforcement-learning-guide>

ПРИМЕРЫ АДАПТИВНЫХ КРИТИКОВ НА НЕЙРОННЫХ СЕТЯХ

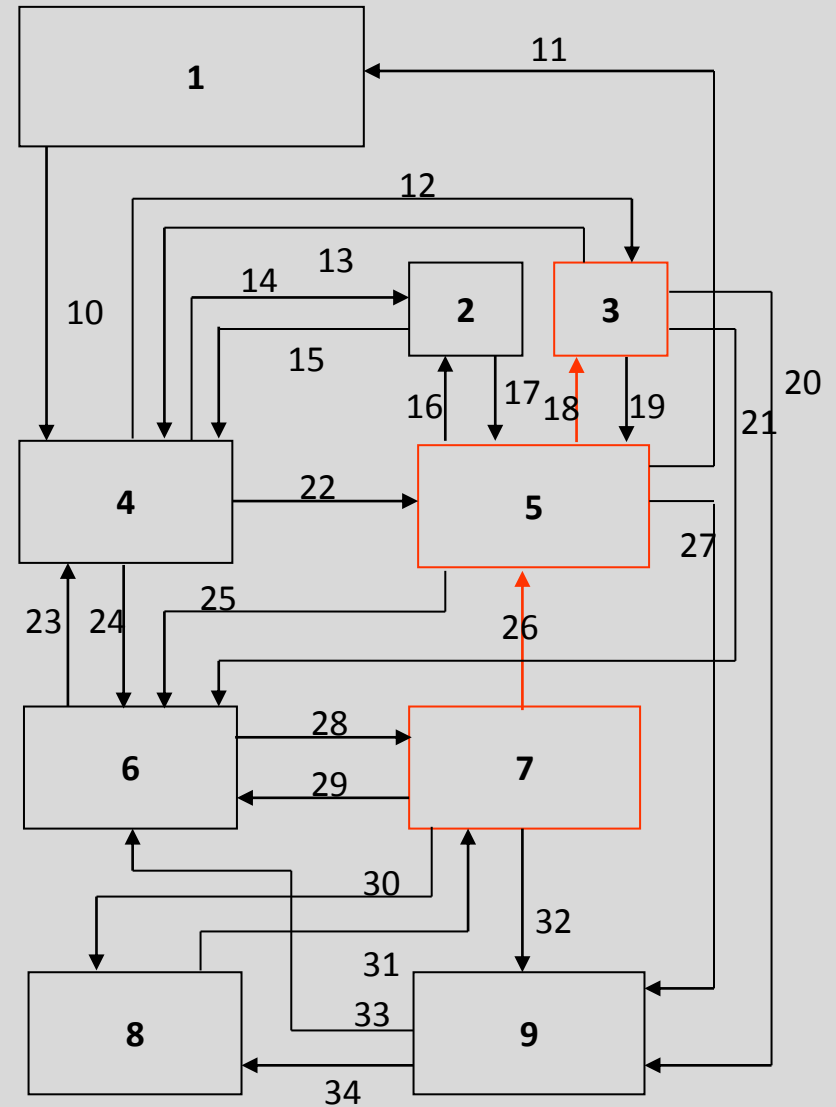


ПРИМЕРЫ АДАПТИВНЫХ КРИТИКОВ НА НЕЙРОННЫХ СЕТЯХ

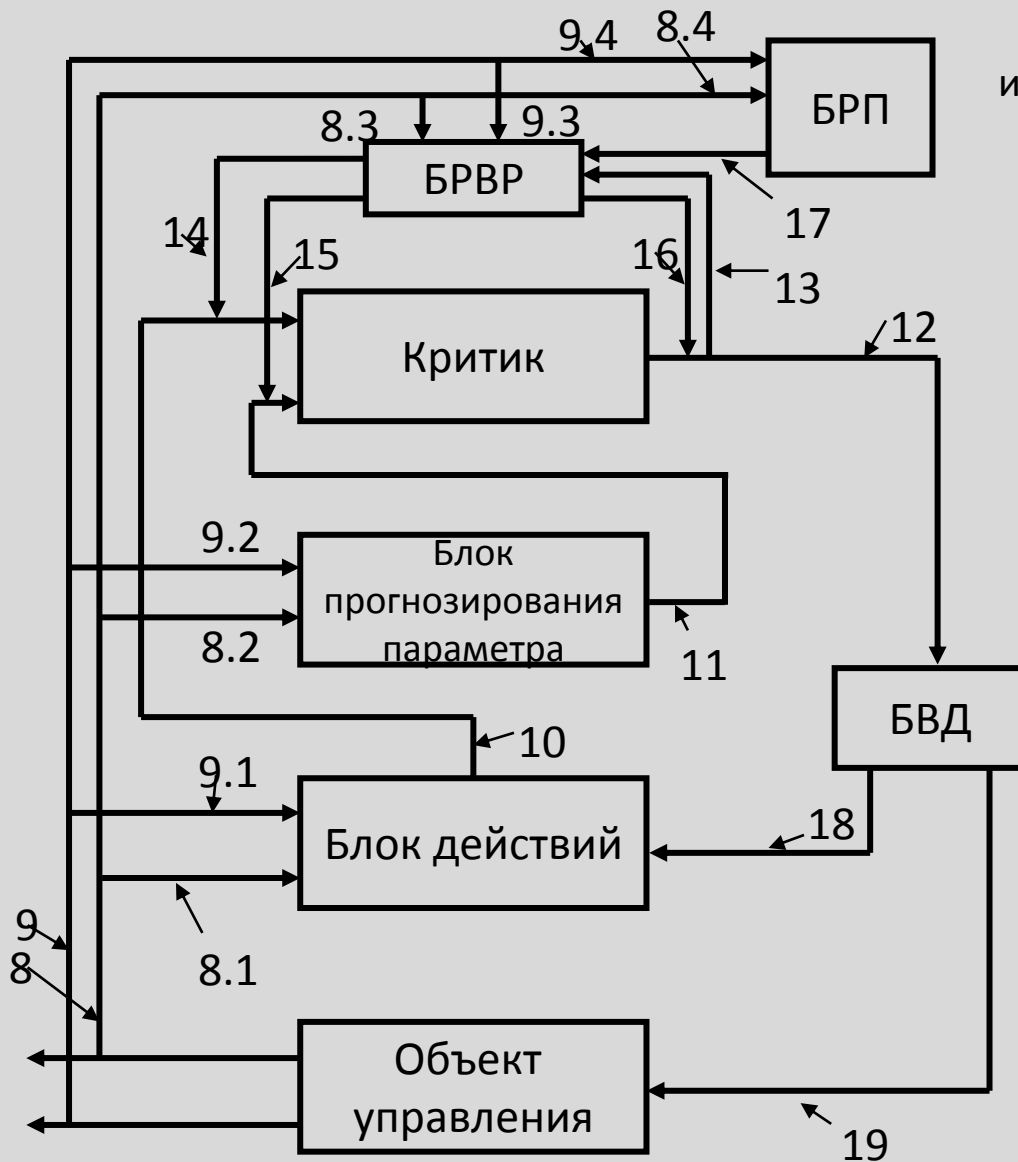
РФ № 2527212



Патент РФ № 2602973

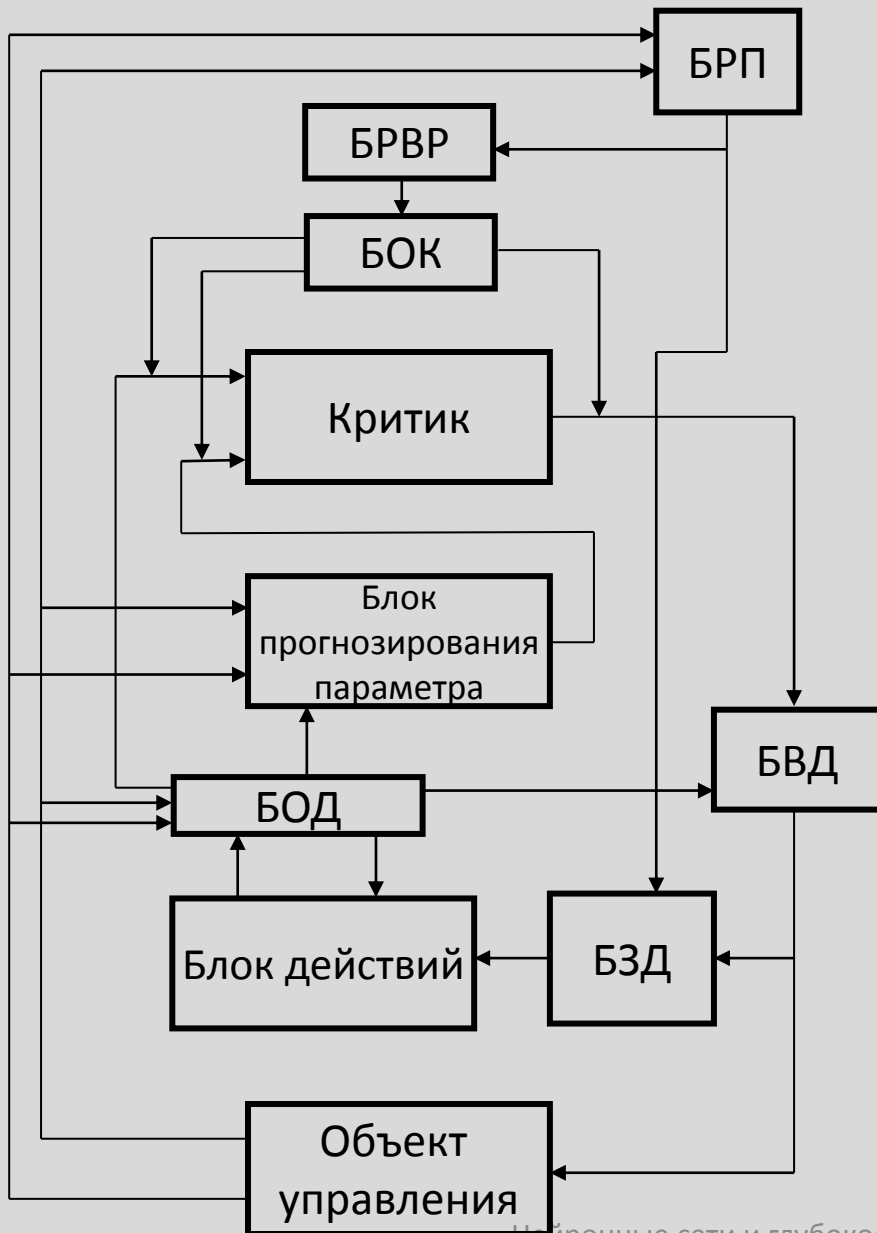


ПРИМЕРЫ АДАПТИВНЫХ КРИТИКОВ НА НЕЙРОННЫХ СЕТЯХ



Модифицированный
интеллектуальный контроллер с
адаптивным критиком
Патент G06F 15/00 № 2450336

ПРИМЕРЫ АДАПТИВНЫХ КРИТИКОВ НА НЕЙРОННЫХ СЕТЯХ



Модифицированный интеллектуальный контроллер с адаптивным критиком
Патент РФ № 2523218

ЛИТЕРАТУРА:

1. Prokhorov D., Santiago R., and Wunsch D. "Adaptive Critic Designs: A Case Study For Neurocontrol". *Neural Networks*, vol. 8, no. 9, pp. 1367-1372, 1995.
2. Sutton R.S., "Learning to Predict by the Methods of Temporal Differences" *Machine Learning*, vol. 3, pp. 9-44, 1988.
3. Watkins C., Dayan P. Q – learning. // *Machine Learning*, vol. 8, pp. 279 – 292. 1992.
4. Саттон, Барто. Обучение с подкреплением
5. Стасевич В. П., Шумков Е. А., Ключко В. И., Воротников С. А. Адаптивные системы на основе самообучающихся нейросетей // *Труды КубГТУ*. - 2002. - Вып.2. - С. 192 - 198.
6. Стасевич В.П. Анализ и адаптивное управление в недетерминированных средах на основе самообучения: дис. канд. техн. наук. – Краснодар, КубГТУ, 2007. – 170 с.
7. Шумков Е.А. Система поддержки принятия решений предприятия // дисс. канд. техн. наук. Краснодар: КубГТУ. 2004.
8. Хайкин С. Нейронные сети: полный курс. 2-е изд.: Пер. с. англ. – М.: Издательский дом «Вильямс», 2006 – 1104 с.

БЛАГОДАРЮ ЗА ВНИМАНИЕ!

Вопросы?