

Специальная цена

# Нейронные сети на Python

Курс о нейронных сетях, глубоком обучении и их применении для решения задач CV, NLP и обучения с подкреплением

Длительность курса: 130 академических часов

1 Первые шаги

- 1 **Искусственный интеллект и другие задачи которые можно решать с помощью нейронных сетей**
- участники узнают что такое искусственные нейроны для того, чтобы понимать из чего состоят искусственные нейронные сети, познакомятся с историей развития искусственного интеллекта и актуальными задачами, которые стоят перед учеными в области Data Science

Домашние задания

1 Введение в PyTorch

1. Используя скрипт `salt.py` посчитать хэш от своего имени. Суть задания подтвердить запуск докера.
  2. Сделать свой `transform`, переводящий данные в распределение  $N(0, 1)$  и выложить код в Slack. Проверить код 3х коллег и отписаться в ветку к выложенному коду.
- 

- 2 **Стохастический градиентный спуск и метод обратного распространения**
- на занятии мы разберем математическую основу обучения нейронной сети и базовые идеи работы с данными
- 

- 3 **Первая нейронная сеть на PyTorch**
- участники познакомятся с фреймворком PyTorch и научатся реализовывать простейшие нейронные сети

Домашние задания

1 Логистическая регрессия на pyTorch

В файле `log_reg.ipynb` изменить код таким образом, чтобы решить ту же самую задачу, а именно обучить логистическую регрессию, на двумерных данных

---

- |   |                                                     |                                                                                                                                                               |
|---|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4 | <b>Переобучение и регуляризация нейронных сетей</b> | участники узнают что такое переобучение нейронной сети, и как с ним бороться для повышения эффективности своих моделей                                        |
|   |                                                     | Домашние задания                                                                                                                                              |
|   |                                                     | 1 Переобучение 1                                                                                                                                              |
|   |                                                     | Добиться от классификатора fashionminst переобучения за счет изменения архитектуры и гиперпараметров.                                                         |
| 5 | <b>Первая нейронная сеть на Tensorflow</b>          | участники научатся реализовывать нейронные сети с помощью одного из самых популярных фреймворков                                                              |
| 6 | <b>Взрыв и затухание градиентов</b>                 | участники узнают в чем заключаются проблемы взрыва и затухания градиентов и познакомятся с методами борьбы позволяющими обучать более глубокие нейронные сети |
| 7 | <b>Погружение в Tensorflow</b>                      | на занятии мы разберем продвинутые возможности фреймворка Tensorflow                                                                                          |
| 8 | <b>Адаптивные методы градиентного спуска</b>        | участники научатся применять современные модификации метода градиентного спуска для ускорения обучения нейронных сетей                                        |

## 2 Основные архитектуры и методы обучения нейронных сетей

### 1 Основные архитектуры нейронных сетей: Автокодировщики

разбор базовых архитектур мы начнем с универсального фреймворка кодировщик-декодировщик и базовой архитектуры в этом фреймворке — автокодировщик. Участники узнают как создавать автокодировщики и смогут модифицировать основную структуру под конкретную задачу.

Домашние задания

#### 1 Автокодировщик

Обновить формулу KL-дивергенции в соответствии со слайдами

Необязательное д/з на поэкспериментировать:

1. Обучить шумный автокодировщик:

а. Добавить ко входным данным нормальный шум

б. Обнулить % случайных пикселей картинки (30-50% должно работать хорошо)

в. Обнулить левую/правую половину картинки

### 2 Практическое занятие на TensorFlow. Разреженный Автокодировщик

на занятии участники решат задачи удаления шума с изображения и разреживания представления данных.

### 3 **Основные архитектуры нейронных сетей: Сверточные сети**

на этом занятии мы научимся использовать наиболее популярную архитектуру нейронной сети для решения задач компьютерного зрения, а также операцию обратную свертке для реализации сверточных автокодировщиков.

Домашние задания

#### 1 Сверточные сети

Необязательное домашнее задание

1. Реализовать сверточный автокодировщик
2. Сделать оптимизацию изображения под активацию сверточных нейронов классификатора
3. Сделать оптимизацию изображения под активацию сверточных нейронов автокодировщика

---

### 4 **Практическое занятие на Keras. Transfer Learning.**

Участники познакомятся на практике с одним из ключевых методов решения задач компьютерного зрения.

---

## 5 Основные архитектуры нейронных сетей: Рекуррентные сети

на примере задачи классификации коротких текстов, участники познакомятся еще с одной важнейшей архитектурой — рекуррентной нейронной сетью. Помимо обработки естественных языков мы рассмотрим общий подход к работе с последовательными данными с помощью рекуррентной сети.

Домашние задания

### 1 Создаем Википедию

1. Используя подход аналогичный torchvision, сделать свой класс датасета.

Необязательное д/з:

1. Поэкспериментировать с разными архитектурами рекурренток: тип ячеек, слои, нормализация, методы оптимизации

---

## 6 Практическое занятие на PyTorch. Генерация Википедии.

на этом занятии мы перейдем от предсказания свойств текста к порождению собственных текстов.

---

**7 Введение в обучение с подкреплением**

участники научатся применять методы классического обучения с подкреплением для таких задач как A/B тестирование или игра в шахматы. Мы разберемся с тем кто такие многорукие бандиты и что такое TD-обучение.

Домашние задания

**1** Подготовка проекта

Нужно выбрать тематику проекта и сделать первичное описание:

1. Формулировка задачи
  2. Предполагаемые методы
  3. Ожидаемый результат
- 

**8 Практическое занятие. Крестики-нолики.**

на примере игры в крестики-нолики участники сравнят решение с помощью нейронной сети с “табличными” методами обучения с подкреплением.

# 3 Современные нейронные сети для задач компьютерного зрения и обработки естественных языков

## 1 Современные сверточные сети. Блоки и шорткаты.

на этом занятии участники познакомятся с историей развития современных сверточных сетей, узнают что позволило сделать их по-настоящему глубокими и как это повлияло на область компьютерного зрения.

Архитектуры: AlexNet, GoogLeNet, Inception, ResNet.

---

## 2 Современные сверточные сети. Transfer learning и fine-tuning

участники узнают что делать если данных недостаточно для обучения глубокой нейронной сети, как использовать знания из уже обученной нейронной сети и подстраивать такую сеть к конкретной задаче.

Домашние задания

### 1 Transfer Learning vs Fine Tuning

Цель: Используя датасет занятия (<https://dropmefiles.com/u1iLk>) реализовать подходы Transfer Learning и Fine Tuning с использованием VGG16 модели на Pytorch

---

## 3 Современные рекуррентные сети. Внимание.

тема посвящена тому, как “научить” нейросеть обращать внимание на наиболее важную информацию на изображении или в тексте. Мы узнаем, как использовать эту способность, чтобы искать логически связанные блоки информации, чем это поможет при машинном переводе, поиске ответов на вопросы, аннотации изображений.

Архитектуры: Attention, Seq2Seq, PointerNetworks

---



- |                                                              |                                                                                                                                                                                                                                                                                          |
|--------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>4</b> <b>Современные рекуррентные сети. Трансформеры.</b> | мы разберемся, как можно использовать только attention-слои для Natural Language Inference, а также разберемся, как трансформер позволяет создать эмбединги, учитывающими контекст при кодировании многозначных слов.<br>Архитектуры: Attention is all you need(transformer), ELMo, BERT |
| <b>5</b> <b>Катастрофическое забывание.</b>                  | архитектуры: <a href="https://arxiv.org/pdf/1612.00796.pdf">https://arxiv.org/pdf/1612.00796.pdf</a>                                                                                                                                                                                     |
| <b>6</b> <b>Metric-learning и обучение без примеров</b>      | мы научимся отображать объекты в пространство с метрикой и использовать такое представление для поиска похожих объектов и решения задачи классификации при отсутствии данных.<br>Архитектуры: DSSM                                                                                       |
| <b>7</b> <b>Глубокие сверточные сети. Сегментация</b>        | на занятии мы разберемся, как при помощи нейронной сети сегментировать изображение, какие для этого есть архитектуры, что за принципы лежат в основе UNet и DeepLab.<br>Архитектуры: UNet, DeepLab                                                                                       |
| <b>8</b> <b>Глубокие сверточные сети. Детекция.</b>          | мы разберем принципы работы нейронных сетей для детекции объектов: от эволюцию от первых “чисто” нейросетевых алгоритмов до современных архитектур, работающих за “один взгляд” - YOLO, SSD<br>Архитектуры: R-CNN, Faster-RCNN, YOLO, Single Shot Detector                               |

## 1 Генеративные состязательные сети

на занятии мы разберем, как нейронная сеть учится обманывать другую нейронную сеть, и как от этого они обе становятся лучше, а мы получаем генератор качественных данных

Архитектуры: GAN (Images)

Домашние задания

### 1 GANs

1. Сделать условный GANs — класс подается на вход Генератору и Дискриминатору
  2. Сделать Adversarial Autoencoder — автокодировщик с лоссом от Дискриминатора вместо dKL
- 

## 2 Вариационный и состязательный автокодировщики

Домашние задания

### 1 Вариационный автокодировщик

Реализовать VAE в pytorch для MNIST датасета. Визуализировать полученный результат.

---

### 3 Доменная адаптация и условная генерация

мы разберем подходы к обучению нейронных сетей на многомодальных данных, способы целенаправленного забывания информации и применение состязательного фреймворка для "честной" условной генерации.

Архитектуры: Conditional GAN

Домашние задания

#### 1 Условная генерация

Обучить две версии СААЕ — "частную" и "нечестную", преобразовать с их помощью исходные данные в латентное представление и обучить 3 классификатора — один на исходных данных и два на кодах, полученных из "честного" и "нечестного" СААЕ. Сравнить качество классификации.

Необязательная часть:

Сделать перенос стиля с одной цифры на другие с помощью "честного" СААЕ

---

### 4 Генеративные модели для текстов

на этом занятии мы познакомимся с методом Professor forcing, научимся использовать идеи обучения с подкреплением для генерации текстов, а также, узнаем как пропускать градиент через дискретные объекты с помощью репараметризации

Архитектуры: Professor forcing, ORGAN, Gumbel trick

---

### 5 Обратное обучение с подкреплением

на этом занятии мы разберем как имитировать поведение эксперта, что делать если награды неизвестны и можно ли назначать награды нейронной сетью.

---

- 6 **Глубокое обучение с подкреплением** участники узнают как же Google удалось победить чемпиона мира в игру го и что еще можно сделать таким же способом.  
Архитектуры: AlphaGo
- 
- 7 **Сверточные и рекуррентные сети на графах** мы разберем архитектуры позволяющие работать с данными имеющими графовую структуру.  
Архитектуры: MPNN, GCN, DNC
- 
- 8 **Сферические свертки и множества точек** участники узнают как применять непрерывные свертки для данных имеющих пространственную структуру.  
Архитектура: SchNet, PointNet

## 1 консультация по проектной работе

обзор выпускной работы. Постановка задачи. Предоставление необходимых для выполнения выпускной работы материалов

Домашние задания

### 1 Выпускная работа

Цель: В рамках курса предусмотрена защита проекта. Он представляет собой генеративную модель для порождения текста заданного стиля или изображений заданной тематики.

Работа над проектом проходит поэтапно:

1. Выбор тематики.
2. Сбор и подготовка соответствующих данных.
3. Построение и обучение генеративной модели

Проект выполняется программистом в течение двух недель после прохождения курса. Готовое решение должно включать в себя код модели и ее обучения на Python, описание архитектуры модели, отчет об обучении модели и примеры генерации.

Примеры тем проекта:

имплементация универсального класса оптимизатора, обобщающего различные методы градиентного спуска для PyTorch;  
состязательная нейронная сеть для порождения изображений рукописного текста с заданным стилем;  
генеративная модель повышения разрешения изображений;  
рекуррентная состязательная seq2seq-архитектура для порождения стихотворений на

---

**2** консультация  
по проектной  
работе

---

**3** защита  
проектной  
работы