

**МАШИННОЕ ОБУЧЕНИЕ**

для школьников

**11 класс, 64 часа.**

Курс «Машинное обучение для школьников» предназначен для учащихся 10-11 классов в качестве основного или дополнительного модуля информатики.

Изучаются методы работы с данными, методы машинного обучения и нейронных сетей. D 11 изучаются различные архитектуры нейронных сетей и их применение для решения задач обработки текстов, звуков, изображений и др. Также изучаются отдельные методы «мягких» вычислений.

Курс практический и представлен в виде «тетрадок» Jupyter Notebook на языке Python, содержащих теоретическую и программную часть.

Для изучения курса необходимо:

- знать базовые конструкции языка Python, уметь устанавливать, запускать «тетрадки» Jupyter Notebook, необходимые библиотеки Python (при работе со своего компьютера). Рекомендуется установить расширения nbextension для Jupyter Notebook.

- знать английский язык для чтения документации, уметь находить необходимую информацию в документации, сети Интернет.

- для работы с видео и звуком нужны установленные микрофон, динамики и web-камера.

**СОДЕРЖАНИЕ КУРСА**

| **№ п\п** | **Модуль** | **Наименование раздела** | **Коли-чество часов** | |
| --- | --- | --- | --- | --- |
| ***1 полугодие 11 класса*** | | | | | |
| 01 | Сверточные нейронные сети | Повторение основных понятий нейронных сетей из 10 класса, многослойный персептрон, сверточные сети. | 2 | |
| 02 | Виды и параметры сверток. | 2 | |
| 03 | Методы оптимизации. | 2 | |
| 04 | Способы повышения обобщающей способности: dropout, нормализация и регуляризация. | 2 | |
| 05 | Аугментация данных | 2 | |
| 06 | Обработка изображений | Перенос обучения. Векторные представления изображений | 2 | |
| 07 | Детекция\локализация объектов на изображении. | 2 | |
| 08 | Попиксельная классификация (сегментация) | 2 | |
| 09 | Генеративно – состязательный подход | Состязательные атаки на нейронные сети классификации изображений | 2 | |
| 10 | Генеративно – состязательный подход, теория | 2 | |
| 11 | Генерация изображений | 2 | |
| 12 | Перенос стиля изображения | 2 | |
| 13 | Изменение изображений | 2 | |
| 14 |  | Сверточные сети для обработки речи | 2 | |
| 15 |  | Реализация итогового проекта. | 2 | |
| 16 |  | Презентация результатов итогового проекта | 2 | |
| ИТОГО: | | | | 32 | |
| ***2 полугодие 11 класса*** | | | | | |
| 17 | Рекуррентные нейронные сети. | Рекуррентные нейронные сети | 2 | |
| 18 | Вентильные сети. LSTM, GRU | 2 | |
| 19 | Генерация музыки вентильными сетями | 2 | |
| 20 | Обработка текста и речи | Векторные представления слов | 2 | |
| 21 | Механизм внимания. Сети Transformer, BERT | 2 | |
| 22 | Векторные представления речи Wav2Vec. Клонирование голоса. | 2 | |
| 23 | Генерация текста. Пишем стихи. | 2 | |
| 24 | Аннотирование изображений. | 2 | |
| 25 | «Мягкие» вычисления. | Эволюционные и генетические вычисления «Почему вымерли динозавры» | 2 | |
| 26 | Нечеткие числа и системы. «Ну примерно половинка» | 2 | |
| 27 | Нейрокриптография. «Играем в шпионов» | 2 | |
| 28 | Чат-боты «Поговори со мной» | 2 | |
| 29 | Обучение с подкреплением «Не мешает подкрепиться» | 2 | |
| 30 | Подходы к работе с большими данными «Ох, как много данных» | 2 | |
| 31 |  | Реализация итогового проекта. | 2 | |
| 32 |  | Презентация результатов итогового проекта | 2 | |
| ИТОГО: | | | | 32 | |
| ВСЕГО | | | | 64 | |

**Краткое содержание разделов**

**1 полугодие 11 класса:**

**01. Повторение основных понятий нейронных сетей из 10 класса, многослойный персептрон, сверточные сети.**

Повтор основных положений нейронных сетей. Слоистые сети, полносвязные и не полносвязные, сверточные сети. Обучающее, проверочное, тестовое множество. Градиентный спуск. Обратное распространение ошибки. И др.

**02. Виды и параметры сверток.**

Углубленное изучение сверток. Двумерные свертки. Реализация в Tensorflow (Keras): принятые соглашения при работе со свертками, тензоры. Понятие ядра (kernel) свертки и размера ядра; понятие фильтра, количество фильтров в слое; понятие сдвига (stride), его параметры; понятие набивки (padding), ее параметры; понятие групп (groups) и их параметры. Распределенная (dilated) свертка, аргумент dilation\_rate. Транспонированная (transposed) свертка, ее аргументы. Разделимые (separable) свертки, идея, реализация и сравнение. Контрольные задания.

**03. Методы оптимизации.**

Краткое напоминание метода градиентного спуска. Проблемы градиентного спуска. Модификации градиентного спуска. Метод знака градиента. Моменты. Стохастический градиентный спуск с [первым] моментом. Автоматическая регулировка шага обучения, RMSProp. Метод Adam. Сравнение методов на простом примере (распознавание типа одежды). Идеи других методов оптимизации: моменты Нестерова, методы второго порядка.

**04. Способы повышения обобщающей способности: dropout, нормализация и регуляризация.**

Явления недообучения и переобучения, борьба с ними. Регуляризация параметров, L1 и L2 регуляризация, о разреженности параметров при L1 регуляризации. Добавление случайности, метод dropout. Нормализация, пакетная нормализация batchNormalization. Использование изучаемых методов на примере полносвязных сетей определения элементарных частиц (Higgs dataset); работа с объектом Dataset. Контрольные задания. Домашняя работа.

**05. Аугментация данных**

Понятие аугментации данных. Объект ImageDataGenerator для аугментации изображений. Реализация аугментаций: повороты, сдвиги, масштабирование, отражение и др. Пример классификации кошек и собак с и без аугментации. Создание собственного генератора данных. О других аугментациях.

**06. Перенос обучения. Векторные представления изображений**

О сложностях обучения нейронных сетей. Идея переноса обучения (transfer learning). Пример использования предобученной сети VGG для набора данных CalTech-101 и сравнение с обучением с нуля. Векторные представления изображений, библиотека Img2Vec. TSNE проекция векторов изображений. Сравнение изображений по их векторам, поиск похожих и не похожих (по косинусному расстоянию). Задания.

**07. Детекция\локализация объектов на изображении.**

Задачи классификации, локализации, детекции и сегментации объектов на изображении. Идеи и архитектура сети YOLO. Оценка похожести ограничивающих окон, метрика «пересечения над объединением» IoU. Пример детекции сетью YOLOv4 из библиотеки Darknet. Пример детекции объектов в видео. Задания.

**08. Попиксельная классификация (сегментация)**

Понятие маски изображения. Идеи и архитектуры сетей типа R-CNN (R-CNN, Fast-R-CNN, Faster-R-CNN), Mask-RCNN. Пример работы Mask-RCNN для изображения. Пример работы Mask-RCNN для изображения с веб-камеры. Задания.

**09. Состязательные атаки на нейронные сети классификации изображений**

Обман нейронной сети, состязательные атаки (adversarial attack), идея. Направленные и не направленные атаки. Атака методом быстрого знака градиента. Пример в библиотеке CleverHans. Обсуждение других атак, физических атак, проблем и способов защиты. Задания.

**10. Генеративно – состязательный подход, теория**

Генеративно – состязательный подход. Сети дискриминатора и генератора их задачи. Общий вид генеративно-состязательной сети (Generative-Adversarial Network, GAN), латентные входы. Пример генерации синусоид, проблемы самостоятельного обучения GAN, предобучение дискриминатора и генератора. Задания.

**11. Генерация изображений**

Использование обученных генераторов изображений. Идеи и архитектура сети StyleGAN2. Пример генерации изображений рыбок. Изучения влияния латентных переменных на генерацию изображений лиц, морфинг. Задания.

**12. Перенос стиля изображения**

Задача переноса стиля изображения. Расчет ошибок стиля и содержания. Представление стиля – матрица Грамма. Генерация изображения с контролем ошибок стиля и содержания. Реализация процедуры генерации в Keras, использование автодифференцирования. Задания.

**13. Изменение изображений**

Пример задачи изменения изображения – создание мультяшного изображения. Идея и архитектура сети CartoonGAN, ошибка генерации и содержания. Пример. Задания.

**14. Сверточные сети для обработки речи**

О представлении звуков. Одномерные свертки. Задача распознавания речи. Ошибка CTC (Connectionist Temporal Classification): понятие символа «промежуток», поиск подходящих последовательностей, расчет ошибки. О модификациях ошибки CTC. Сверточная сеть Wav2letter для распознавания речи. Пример распознавания речи сетью Wav2Letter+. Задания.

**15. Реализация итогового проекта.**

Распознаем заболевания томатов по изображению их листьев – поможем аграриям!

Варианты:

а) индивидуальный – каждый обучающийся выполняет проект самостоятельно, заполняя пропуски в прездаполненном файле-задании. Учитель проверяет правильность.

б) групповой – группы обучающихся соревнуются между собой за достижение лучшего результата, изменяя параметры методов, представленные в файле-решении. Оценка по занятому месту.

**16. Презентация результатов итогового проекта**

Обучающиеся презентуют свои решения, сравнивают с предложенным в файле-решении.

**2 полугодие 11 класса:**

**17. Рекуррентные нейронные сети**

Добавление времени в нейронную сеть. Нейронные сети с временными задержками. Рекуррентные сети, понятие обратной связи. Рекуррентный нейрон, проблема обучения. Разворачивание рекуррентных сетей во времени. Примеры рекуррентных сетей, нейронная сеть NARX (Nonlinear AutoRegression with eXternal input). Архитектурные ограничения NARX, обучение NARX по разомкнутому циклу. Библиотека Sysidentpy, пример моделирования процесса. Задания.

**18. Вентильные сети. LSTM, GRU**

Проблема долгосрочных зависимостей. Понятие вентиля (gate). Устройство LSTM-нейрона (Long Short Term Memory) и LSTM-сети. Устройство GRU (Gated Recurrent Unit) нейрона. О создании других вентильных сетей. Пример прогноза температуры с помощью вентильной нейронной сети. Задания.

**19. Генерация музыки вентильными сетями**

Формат MIDI для представления музыки. Архитектура LSTM генератора музыки. Пример генерации музыки.

**20. Векторные представления слов.**

Векторные представления слов. Унитарный код (one-hot). Контекст слов, идея и архитектура Word2vec представления. Геометрия слов. Косинусное расстояние. Библиотека gensim: сравнение слов, поиск похожих слов и др. Работа с моделями для русского языка. Векторные представления слов GloVe (Global Vectors), примеры. Векторные представления FastText, использование n-грамм символов, примеры. Задания.

**21. Механизм внимания. Сети Transformer, BERT**

Проблемы учета влияния слов друг на друга. Двунаправленный LSTM. Механизм внимания (attention). Множественное (multihead) внимание. Идея и архитектура сети Transformer, различия работы в процессе обучения и генерации. Идея и архитектура сети BERT: предсказание маски, предсказание следующей последовательности. Пример работы BERT (DistilBERT) для классификации отзывов на фильм.

**22. Векторные представления речи Wav2Vec. Клонирование голоса.**

BERT-подобная сеть Wav2Vec для векторного представления речи: квантование, маскирование, функции ошибки, самообучение и обучение с учителем. Пример распознавания речи. Пример клонирования голоса с библиотекой CorentinJ/Real-Time-Voice-Cloning.

**23. Генерация текста. Пишем стихи.**

Генеративные сети для обработки текста. Проблемы стихосложения. Идеи и архитектуры сетей для генерации стихов. Пример. Задания.

**24. Аннотирование изображений**

Совместные векторные представления изображений и текста. Задача аннотирования изображений. Примеры сетей для аннотации изображений. Об обратной задаче – генерации изображения по аннотации, пример.

**25. Эволюционные и генетические вычисления «Почему вымерли динозавры»**

Понятие эволюционных вычислений. Метод «роя частиц»: устройство, пример в библиотеке PySwarms. Генетические алгоритмы: генетический код, функция приспособленности, процедуры отбора, скрещивания, мутации. Онлайн-пример «генетических машинок». Пример решения задачи о рюкзаке с библиотекой pyeasyga. Задания.

**26. Нечеткие числа и системы. «Ну примерно половинка»**

Понятие нечетких множеств и нечеткой логики (fuzzy logic). Функция принадлежности. Нечеткие логические операции: объединение (ИЛИ), пересечение (И), дополнение (НЕ). Пример нечеткого логического вывода, библиотека skfuzzy. Задания. Нейро-нечеткая сеть ANFIS, пример прогноза временного ряда.

**27. Нейрокриптография. «Играем в шпионов»**

Понятия криптографии. Генерация секретных ключей, протокол Диффи-Хеллмана. Архитектура древовидных машин четности (tree parity machines). Генерация секретного ключа с помощью древовидных машин четности. Методы обновления весов. Реализация протокола генерации секретного ключа. Имитация злоумышленника. Исследование влияния параметров на процесс синхронизации. Задания.

**28. Чат-боты «Поговори со мной»**

Задача построения чат-бота, виды чат-ботов. Подходы к построению чат-ботов. Пример самостоятельной реализации чат-бота.

**29. Обучение с подкреплением «Не мешает подкрепиться»**

Понятие обучения с подкреплением (reinforcement learning). Q-обучение, идея и формулы. Эпсилон-жадный выбор. Пример управления тележкой (библиотека gym) с Q-обучением. Задания. Онлайн-пример.

**30. Подходы к работе с большими данными «Ох, как много данных»**

Понятие больших данных. Вычислительные проблемы больших данных. Подход map-reduce. Самостоятельная реализация подхода. Другие примеры. Задания.

**31. Реализация итогового проекта.**

Игра угадывания слова по ассоциациям.

Варианты:

а) индивидуальный – каждый обучающийся выполняет проект самостоятельно, заполняя пропуски в прездаполненном файле-задании. Учитель проверяет правильность.

б) групповой – группы обучающихся соревнуются между собой за достижение лучшего результата, изменяя параметры методов, представленные в файле-решении. Оценка по занятому месту.

**32. Презентация результатов итогового проекта**

Обучающиеся презентуют свои решения, сравнивают с предложенным в файле-решении.

**Порядок проведения занятий**

Рекомендуется проводить занятия в порядке нумерации уроков. В случае сложностей освоения допускается опускать некоторые темы (например работа со звуком, если таковая не предполагается или отсутствует техническая возможность воспроизводить и записывать звук, сложные темы и т.п.).

Время изучения темы преподаватель регулирует самостоятельно, в зависимости от ожидаемого уровня освоения. Предполагается три уровня освоения:

- Начальный – ученик способен читать материалы, запускать коды, наблюдать решения и объяснять их, не внося модификации в код. Рекомендуемая оценка не выше «хорошо».

- Основной – ученик способен читать материалы, запускать коды, наблюдать решения, вносить необходимые изменения в код по аналогии (по заданиям из материалов, заданиям учителя или самостоятельно).

- Продвинутый – ученик способен читать материалы, запускать коды, наблюдать решения, вносить необходимые изменения по аналогии и предлагать свои решения: изменять постановки задач, изменять данные, изменять методы решения, может объяснить результаты. Предполагается интенсивная самостоятельная работа (чтение документации, эксперименты с кодом).