

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Национальный исследовательский университет  
"Высшая школа экономики"**

Факультет Компьютерных наук  
Департамент больших данных и информационного поиска  
Базовая кафедра Яндекс

**Рабочая программа дисциплины «Глубинное обучение»**

для образовательной программы «Науки о данных»  
направления подготовки 01.04.02 "Прикладная математика и информатика"  
уровень магистра

Разработчик(и) программы  
Ратников Ф. Д., к.ф.-м.н. ([fratnikov@hse.ru](mailto:fratnikov@hse.ru))

Одобрена на заседании базовой кафедры Яндекс  
«\_\_» \_\_\_\_\_ 2017 г.

Заведующий Кафедрой  
М.А. Бабенко \_\_\_\_\_

Утверждена Академическим советом образовательной программы  
«\_\_» \_\_\_\_\_ 2017 г., № протокола \_\_\_\_\_

Академический руководитель образовательной программы  
С. О. Кузнецов \_\_\_\_\_

Москва, 2017

*Настоящая программа не может быть использована другими подразделениями  
университета и другими вузами без разрешения подразделения-разработчика программы*



## 1. Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает требования к образовательным результатам и результатам обучения студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих дисциплину «Глубинное обучение», учебных ассистентов и студентов направления подготовки 01.04.02 «Прикладная математика и информатика», обучающихся по образовательной программе «Науки о данных».

Программа учебной дисциплины разработана в соответствии с:

- Образовательным стандартом федерального государственного автономного образовательного учреждения высшего профессионального образования «Национального исследовательского университета «Высшая школа экономики»;
- Образовательной программой подготовки магистра по направлению 01.04.02 «Прикладная математика и информатика» по образовательной программе «Науки о данных».
- Объединенным учебным планом университета по образовательной программе «Науки о данных», утвержденным в 2017 г.

## 2. Цели освоения дисциплины

Предлагаемый курс посвящён методам «глубинного обучения» - нового поколения нейросетевых методов машинного обучения, вызвавших бурный всплеск развития в ряде прикладных областей. В первую очередь курс направлен на формирования у студентов навыков решения прикладных задач при помощи глубоких нейронных сетей.

За последние несколько лет, методы глубокого обучения прочно закрепились в таких прикладных областях, как

- Компьютерное зрение: распознавание визуальных образов, сегментация, восстановление цвета по ч/б картинкам, описание картинки тэгами и многое другое,
- Обработка текстов: анализ тональности, вопросно-ответные системы, синтез текста, машинный перевод, семантический анализ и ряд других задач,
- Обработка речи: распознавание, синтез, классификация акцентов, тональностей и пр.,
- Робототехника и игрушки: оптимизация стратегии поведения, Reinforcement Learning

Успешное решение задач при помощи глубоких нейронных сетей также появляется в других прикладных областях: прикладные задачи Информационного Поиска, Физика Высоких Энергий и Астрофизика, а также ещё множество примеров неожиданного применения.

Будучи молодым и бурно развивающимся ремеслом, Глубинное Обучение зачастую не имеет достаточно обоснованного теоретического аппарата и устоявшихся и всюду



действенных методов. Возможности же привести работоспособную архитектуру нейронной сети в негодность одной строкой кода, напротив, изобилуют. Таким образом, эффективность использования глубоких нейронных сетей определяется в первую очередь практическим опытом использования глубоких нейронных сетей.

Не менее важно, что реальные прикладные задачи, с которыми вероятнее всего столкнётся разработчик или исследователь Глубокого Обучения, как правило имеют множество особенностей, которые можно эксплуатировать теми или иными приёмами и эвристиками. Иными словами, наивное сведение прикладной задачи к абстрактному “распознаванию образов” почти всегда можно улучшить, используя детали этой задачи.

К сожалению, множество эвристических приёмов такого характера ещё менее структурировано, чем Глубокое Обучение, и даже именуется за это “чёрной магией” среди практиков.

Тем не менее, специалист с достаточным опытом может во многих задачах прийти к более эффективному решению посредством использования методов Глубинного Обучения вместе или вместо классических методов и моделей Машинного Обучения.

### **3. Компетенции обучающегося, формируемые в результате освоения дисциплины**

#### **Развиваемые компетенции**

Дисциплина формирует следующие компетенции.

1. ПК-8: способность решать задачи производственной и технологической деятельности с использованием методов глубокого обучения на профессиональном уровне, включая постановку задачи, базовое решение, прототипирование и внедрение.
2. Представление об основных сферах применения методов глубокого обучения, особенностям и трюкам в каждой такой сфере.
3. Углублённое понимание одной из основных сфер применения глубокого обучения.

#### **Учебные задачи дисциплины**

Предлагаемый курс нацелен в первую очередь на формирование у студента таких навыков и получения ими опыта решения прикладных задач с использованием Глубинного Обучения.

В рамках курса планируется осветить:

- Основы Deep Learning
  1. Основные архитектуры и слои,
  2. Специфичные методы оптимизации (Momentum SGD, RMSprop, ada\*, etc.)
  3. Общий pipeline решения прикладных задач,
  4. Решение простейших задач на практике (NotMnist)
- Глубокое обучение в анализе изображений



1. Обзор основных задач
  2. Свёрточные сети, Pooling
  3. Специфические архитектуры – AutoEncoders, Siamese, Stacked
  4. Прикладные эвристики: Batch Normalization, Dropout, DropConnect, added noise, 1x1 convolution
  5. “Зоопарк моделей”, дообучение сетей, Fine-tuning
- Глубокое обучение в обработке текстов:
    1. Краткий обзор предметной области
    2. Сведение прикладных задач к Deep Learning
    3. Свёрточные и рекуррентные сети в обработке текстов, GRU, LSTM
    4. Специфические архитектуры: Text Convolution, Recurrent , Encoder-Decoder,
    5. Ситуативные эвристики, nudging dataset
  - Решение прикладной задачи из анализа изображений, обработки текстов или (для мазохистов) другой не покрытой в курсе прикладной области.
    2. Предполагается решение задачи в semi-supervised режиме в течение всего курса с периодическими выступлениями о том, что удалось сделать, что не заработало и обсуждений, как с этим бороться.
    3. Основные этапы – (1) исследование особенностей задачи, (2) формирование базового решения и (3,4,...) итерации улучшения качества решения.
    4. После каждого этапа следует мини-конференция (1 семинар), где студенты кратко рассказывают о своих задачах и ходе исследований, слушают рассказы других студентов и предлагают возможные улучшения.
    5. К концу курса из таких шагов формируется решение выбранной прикладной задачи. Если в ходе работы удалось добиться научно актуальных результатов, на их основе публикуется статья в одном из тематических журналов.



#### 4. Тематический план дисциплины «Глубинное обучение»

№	Название темы	Аудиторные часы		Самосто- ятельная работа
		Лекции	Сем. и практика	
1	Введение. Что такое глубинное обучение представлений. Основные абстракции. Знакомство с Lasagne.	2	2	10
2	Краткий обзор сфер применения и решение простых задач анализа изображений и текстов. Выбор тем проектов.	2	2	16
3	Практические задачи компьютерного зрения и особенности их решения.	8	4	16
4	Анализ проектных задач и способов их решения. Краткий доклад по итогам.	0	4	14
5	Практические задачи обработки последовательностей (текст, речь и т.д.), приёмы для их решения. Поддержка в разработке базового решения проектных задач.	8	4	16
6	Промежуточный отчёт по проектным задачам — представление базового решения.	0	4	16
7	Задачи на обучение с подкреплением. Методы решения при помощи глубоких нейронных сетей.	4	4	14
8	Вычислительная оптимизация обученных нейронных сетей. Особенности применения в production.	2	2	12
9	Финальный отчёт по проектным задачам.	4	4	16
	Итого	30	30	130



## 5. Формы контроля и структура итоговой оценки

Оценка успеваемости зависит исключительно от суммы баллов, набранных студентом за время прохождения курса. Сумма максимальных баллов за все задания заведомо существенно выше порога отличной оценки — это сделано для того, чтобы студент мог самостоятельно выбрать задания, выполнение которых будет ему наиболее полезно.

Небольшая часть заданий (первые 4), затрагивающих основы глубокого обучения, являются обязательными к сдаче. Игнорировать их можно только при наличии (и демонстрации) опыта решения аналогичных задач в прошлом.

Суммарный балл складывается из:

- 10 Домашних заданий по 10 баллов
- Проектной задачи, разбитой на 3 этапа
- ↘ Предварительный анализ - 10 баллов
- ↘ Формирование базового решения - 20 баллов
- ↘ Итерации улучшения решения — 5 баллов за итерацию, еще 5 — при улучшении качества
- Дополнительных возможностей набора баллов:
- ↘ Статей, постов, коммитов, докладов — если они публично доступны и произошли не раньше начала курса.
- ↘ Участия в обсуждении проектов однокурсников.

Д/з сдаются в виде электронного отчёта (по умолчанию — ipython-тетрадки). Для каждого д/з установлен дедлайн, после которого работа штрафует на  $\log_2(\text{число\_просроченных\_дней}+1)$  баллов с округлением в пользу студента. Штраф за опоздание фиксируется в момент первой посылки, после чего студент может исправлять недочёты и повторно присылать д/з. Сдавать д/з можно до окончания курса.

На каждый этап проекта также будет установлен дедлайн, при этом используется та же система штрафов.

Экзамен по курсу сводится к презентации решения проектной задачи. В случае, если студент набрал достаточный для желаемой оценки балл без защиты проекта, формально за курс выставляется оценка автоматом.

Оценка по десятибальной системе равна сумме баллов, делённой на 10 с округлением вниз, за 2 исключениями:

- Оценки менее первичных 10 баллов отображаются в 1 балл по десятичной системе.
- Оценки в 110 и более баллов отображаются в 10 баллов по десятичной системе.



**Таблица соответствия оценок по десятибалльной и системе зачет/незачет**

Оценка по 10-балльной шкале	Оценка по 5-балльной шкале
1	незачет
2	
3	
4	зачет
5	
6	
7	
8	
9	
10	

**Таблица соответствия оценок по десятибалльной и пятибалльной системе**

По десятибалльной шкале	По пятибалльной системе
1 – неудовлетворительно 2 – очень плохо 3 – плохо	неудовлетворительно – 2
4 – удовлетворительно 5 – весьма удовлетворительно	удовлетворительно – 3
6 – хорошо 7 – очень хорошо	хорошо – 4
8 – почти отлично 9 – отлично 10 – блестяще	отлично – 5



## **6. Программа дисциплины «Глубокое обучение в примерах и задачах»**

### **Тема 1. Введение. Основные абстракции глубинное обучение. Обучение представлений.**

Данная тема посвящена закреплению знаний студентов о линейных моделях, введению в глубокое обучение, основным абстракциям и базовым приёмам, необходимым для решения простых задач при помощи нейросетей.

К теоретическому материалу прилагается знакомство с фреймворками, используемыми во всех дальнейших заданиях, и решению задачи на классификацию распадов (HiggsML) из Физики Высоких Энергий.

1. Напоминание про линейные модели классификации и регрессии. Модели, функции потерь, алгоритмы обучения, регуляризация. Ограничения линейных моделей.
2. Обучение признаков без учителя (unsupervised feature learning). Метод главных компонент (PCA), Ограниченные Машины Больцмана (RBM) (без подробного описания CD), DBM и DBN.
3. Использование RBM для классификации лиц (openml-faces). Преимущества и недостатки обучения признаков без учителя.
4. Многослойный персептрон. Абстракция слоя. Полносвязный слой. Нелинейности. Backpropagation (и почему chain rule). Регуляризация (L1-L2; Dropout). Адаптивные стохастические методы оптимизации.
5. Знакомство с theano и lasagne. Описание простой логистической регрессии в Theano. Разработка архитектуры глубокой сети для решения задачи HiggsML.

Обе практические работы из этого раздела являются обязательными для сдачи курса.

### **Тема 2. Основные сферы применения глубинного обучения.**

Данный раздел знакомит студентов с классами задач и тем, как их решать при помощи глубокого обучения. После ознакомления со всеми областями студентам предлагается определиться с задачей на проект.

Попутно объясняются базовые принципы работы свёрточных сетей для изображений и текстов. В данной теме намеренно не рассматриваются подробно рекуррентные сети для текстов, поскольку их объяснение заняло бы непозволительно много времени.

1. Задачи машинного зрения. Классификация, сегментация, генерация изображений. Свёрточные сети, Pooling, Batch Normalization. Стандартный пайплайн решения задачи при помощи нейронных сетей.
2. Решение задачи классификации NotMnist при помощи свёрточных сетей, или, быть может, любых других методов, которыми студент попытается улучшить точность.
3. Задачи автоматической обработки текстов. Классификация, тэгирование, языковые





модели, машинный перевод, диалоговые системы, тематическое моделирование. Word Embeddings (w2v, glove, trained embedding), текстовые свёрточные сети.

4. Решение задачи классификации нежелательного контента в объявлениях Avito.ru при помощи свёрточных сетей над обучаемым словарём word embeddings.

5. Другие сферы приложения. Распознавание и синтез речи. Поиск по текстам и изображениям (Information Retrieval, Object Retrieval) Оптимизация стратегии поведения в робототехнике. Задачи на стыках сфер (естественно-языковое описание объектов на изображении)

6. Мини-семинар про Inceptionism и попытки понять, что же прячется в чёрной коробке, и почему нейронка приняла своё решение в той или иной ситуации.

7. Дедлайн выбора тем проектных задач и формирования команд (if any). После выбора задачи, студенты должны проанализировать несколько статей на данную тему.

Обе домашние задачи из данной темы являются обязательными.

### **Тема 3. Практические задачи компьютерного зрения**

1. Сегментация изображений. Раскрашивание. Предсказание bounding box. Полносвёрточные сети. Deconvolution. Maxout и прочие хитрые нелинейности. Fine-tuning и дообучение сетей. Net2net.

2. Использование сети VGG19, предобученной на ImageNet, для классификации объектов на небольшой выборке изображений.

3. Генеративные сети. Generative adversarial networks (GAN). Gradient reversal trick. Lap-GAN, DC-GAN. Domain adaptation via gradient reversal.

4. Art Style Transfer — применение «фильтра Ван Гога» к изображению при помощи всё той же VGG19.

5. Автокодировщики. Sparse autoencoders, denoizing autoencoders. Вариационные автокодировщики (VAE). Использование автокодировщиков для изменения изображения. Прочие применения автокодировщиков. Common Semantic Space Embedding.

6. Имплементация VAE, обучение на лицах. Image Morphing, добавление улыбки.

Обязательным условием сдачи курса является решение хотя бы одного д/з из этого блока или демонстрация результатов работы над схожей задачей.

### **Тема 4. Промежуточный дедлайн проектных задач: анализ**

В рамках этих занятий студентам предлагается прочитать некоторое количество статей по выбранной задаче, сформировать предположительный план решения и рассказать своим коллегам по курсу обо всём этом.

Часы, отведённые на самостоятельную работу, предполагается потратить в удобное для студента время во время прохождения предыдущего раздела.



## **Тема 5,6 Работа с последовательностями и проекты**

1. Рекуррентные нейронные сети. Backpropagation through time. GRU, LSTM. Gradient clipping. Применение для Language Modelling.
2. Character Level RNN для генерации статей законодательной системы РФ.
3. Encoder-decoder. Машинный перевод и диалоговые системы. Character level Vs word level. Hierarchical softmax & Negative sampling. Pairwise learning functions. Attention. DRAW. Альтернативные структуры рекуррентных сетей.
4. [промежуточный дедлайн по базовому решению проектных задач]
5. Задачи распознавания и синтеза речи
6. Архитектуры с долгосрочной памятью. Stack/List augmented RNN. Neural Turing Machines. RAM-machines. Neural programmer-interpreter. Немногие практические юз-кейсы.
7. Реализация GRU и Stack-Augmented RNN, сравнение эффективности на простых задачах.

Обязательным минимумом является выполнение хотя бы одной задачи из этого или среднего раздела (reinforcement learning).

## **Тема 7 Deep Reinforcement Learning**

1. Формализм MDP. POMDP. Классические способы решения. Проблемы классических способов. Deep Q-learning. Авторреляции и другие проблемы DQN и способы их нивелировать. Experience Replay и асинхронность.
2. Реализация Deep Recurrent Q-learning для игры в Atari. По умолчанию предлагается использовать yandexdataschool.AgentNet + OpenAI.Gym.
3. Advantage Actor-Critic.. Континуальное пространство действий в POMDP. DDPG. Сферы применения.
4. Обучение сети оптимальному управлению роботом-манипулятором при помощи DDPG. По умолчанию предлагается использовать yandexdataschool.AgentNet + OpenAI.Gym.MuJoCo.

Обязательным минимумом является выполнение хотя бы одной задачи из этого или предыдущего раздела (работа с последовательностями).

## **Темы 8,9. Оптимизация обученной сети, финал проектной работы**

1. Оптимизация структуры сети. Optimal Brain Damage. Разложения матриц весов. Soft-targets. Битовая магия и float8. Имплементация в железе и BinaryNet.
2. Финальные презентации решений проектных задач студентов. Обсуждение результатов.

Ко всем проектам должен прилагаться код, который воспроизводит его результаты.



## **7. Учебно-методическое и информационное обеспечение дисциплины**

Поскольку курс нацелен в первую очередь на погружение в практический аспект глубокого обучения, а сама она до сих пор быстро развивается, любые достаточно проработанные учебники становятся устаревшими ещё до публикации. Вместо этого, в рамках проектной работы студентам будет необходимо проанализировать несколько свежих публикаций по теме своего проекта и смежным темам.

Тем не менее, существует некоторое количество материала, который будет любопытно прочесть вне зависимости от научных интересов студента.

## **8. Список литературы**

**Базовые учебники отсутствуют**

**Дополнительная литература**

1. Bengio, Y.; Courville, A.; Vincent, P. (2013). "Representation Learning: A Review and New Perspectives". IEEE Transactions on Pattern Analysis and Machine Intelligence 35(8)
2. Efficient BackProp. Yann LeCun, Leon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. 1 Image Processing Research Department AT& T Labs
3. Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012: Neural Information Processing Systems, Lake Tahoe, Nevada
4. Karpathy A. (2015), The Unreasonable Effectiveness of Recurrent Neural Networks, Hacker's Guide to Neural Networks
5. Alex Graves, Santiago Fernandez, Faustino Gomez, and Jürgen Schmidhuber (2006). Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural nets. Proceedings of ICML'06,

## **9. Тематика заданий по формам текущего контроля**

Темы домашних работ указаны в предыдущем разделе.

## **10. Вопросы для оценки качества освоения дисциплины**

Вопросы служат скорее для самопроверки. Формальный контроль знаний будет ориентирован на навыки применения нейронных сетей, а не на теоретические вопросы.  
Тема 1.

1. Из чего должна состоять простая нейронная сеть?
2. Как обучение представлений связано с глубокими нейросетями?



3. Как обучать нейронную сеть? Какие адаптивные методы оптимизации вы знаете?
4. Чем `expit` отличается от `tanh`? Что такое ReLU и LeakyReLU? Зачем нужен SoftMax?

#### Тема 2.

5. Вам нужно во что бы то ни стало найти все фотографии котиков в интернете. Ваши действия?
3. Кто такие свёрточные нейронные сети зачем они нужны?
4. На вход подаётся картинка  $3 \times 100 \times 100$ . К нему применяются подряд 3 свёртки  $3 \times 3$  с 64 фильтрами, `max-pooling`  $3 \times 3$ , еще 3 свёртки  $3 \times 3$  с 128 нейронами. Какой будет размер у последнего слоя? Какой слой труднее всего вычислять? `Padding` не используется, `stride` тоже.
5. Вы должны определить пол, возраст, религиозные убеждения и ожидаемую продолжительность жизни человека по истории комментариев в YouTube. Как же быть? (Считайте, что у вас есть толпа ассессоров и свой маленький youtube)

#### Тема 3.

6. У вас завалилась коллекция старых немецких фильмов известного содержания. Всё бы хорошо, но они бесцветные (сепия), а фильмы хочется наблюдать в цвете. Ваши действия?
7. Вы сняли с полки школьный фотоальбом с фотографиями одноклассников. Где они? Кто они? Что они делают? Можно ли научиться находить их на фотографиях в новостях, VK, fb и так далее? Как бы вы это сделали?
8. Есть чёрно-белая картинка  $32 \times 32$  подаётся на вход однослойному автоэнкодеру со скрытым состоянием в 900 нейронов. Автоэнкодер обучается по среднеквадратичной ошибке между картинкой на входе и её же — на выходе. Что произойдёт? Что выучит автоэнкодер?
9. У вас есть большой и толстый файлообменник с тэгированными картинками. Хочется находить в нём картинки похожие на референсную картинку по смыслу. Алсо хочется уметь искать по ключевой фразе. Важи действия?
10. Есть много неразмеченных фотографий лиц. Мы хотим заставить их всех улыбаться. Что делать?
11. Котики — вечный источник лайков. Нужно научиться порождать их в неограниченных количествах. Как?

#### Тема 4.

[по теме проекта]

1. Так что там у вас за задача?
2. Как её можно формализовать?
3. С какими данными будем работать?
4. Что об этом думает мир?
5. Как вы планируете её решать?

#### Тема 5.

1. Пушкин написал великие вещи, но слишком рано умер. Можно ли дописать за него ещё стихотворений? Ещё лучше — как научиться их зачитывать?
2. Как сделать свой маленький Google Translate?
3. Зачем нейронкам внимание?
4. А зачем им аугментации(стэк, NTM, RAM)?

#### Тема 6.

[по теме проекта]

1. Так что вы всё это время делали?
2. Опишите архитектуру нейронки? Почему она такая?
3. Как вы её учите?
4. Как эту штуку запускать?



5. Тема 7.

1. Что такое POMDP? Приведите примеры.
2. Как бы вы учили нейронку водить машину?
3. Чем будет отличаться нейронка, играющая в Го (опишите её) от нейронки, управляющей собакой-роботом Aibo при беге по прямой(тоже опишите её).
4. Опишите архитектуру и алгоритм обучения нейронной сети, играющей в Atari. А теперь придумайте игру, которую ей будет максимально трудно выучить. Почему?

Тема 8.

1. Вам отдали доученный VGG19, который умеет предсказывать ОченьВажнуюВещь. Вы хотите поставить её на квадрокоптер, который будет искать ОченьВажнуюВещь с высоты птичьего полёта. На борту квадрокоптера стоит RaspberryPi 3B с 1GB памяти, при этом 512mb уже съела операционка. Прогонять поиск ОБВ хочется максимально часто. Как можно максимально ускорить сеть? Потребуется ли дополнительная память и сколько? Предложите варианты.
2. На всё том же VGG19 вместо последнего слоя приделали блок, который умеет тэгировать изображения. Есть файлообменник с  $10^9$  фоток, хочется их всех протэгировать. Как будем это делать?

## I. Темы вопросов на экзамене

Фактического экзамена с вопросами в рамках курса не предусмотрено, см. систему оценок.

## I. Методические указания студентам

Сдавать этот курс можно с использованием любого известного вам фреймворка. В качестве формата “по-умолчанию” предлагается использовать Ipython-тетрадки (Ipython-notebook). Если вы используете или создаёте какие либо дополнительные файлы – их тоже нужно приложить, либо рассказать, как их можно достать. Любое присланное домашнее задание или решение проекта должно воспроизводиться по вашим указаниям (например, открыть тетрадку и нажать “Run All”).

Способ установки необходимых библиотек вручную будет описан в программе курса. Автоматически установить Theano+Lasagne также можно при помощи Docker-контейнера <https://hub.docker.com/r/justtheuristic/agentnet/>.

Если вы решите использовать другой фреймворк для проектной задачи, будьте готовы самостоятельно его поддерживать. Семинарист не несёт ответственность за то, что в Caffee собранного в ручную по инструкции годовалой давности не собирается модель, написанная под другую его версию контрибутором-любителем. Сдавать д/з на своём фреймворке также можно, однако будьте готовы потратить на него намного больше усилий.

Идеальной стратегией будет предварительно согласовать использование фреймворка с преподавателем, чтобы избежать возможных проблем с совместимостью.

Командное решение проектной задачи допускается только по предварительному согласованию (чем раньше, тем выше вероятность одобрения) и только в случае, если задача бьётся на несколько подзадач (по числу участников), решение каждой из которых по сложности и полезности сравнимо с индивидуальными проектами. В таком случае каждый участник отчитывается за свою часть проекта.



Рабочая программа дисциплины “Глубинное обучение” для образовательной программы  
«Науки о данных» направления подготовки 01.04.02 "Прикладная математика и информатика"  
уровень магистра

Использование сторонних материалов и чужого кода при решении д/з и в работе над проектом допускается только при явном указании этого и возможности рассказать про то, как он работает.