

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Макаренко Елена Николаевна

Должность:

Документ подписан в:

Дата подписания: 20.06.2026 11:42:44

Уникальный программный ключ:

c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ

Начальник

учебно-методического управления

Т.К. Платонова

«25» мая 2026 г.

**Рабочая программа дисциплины
Нейронные сети для мобильных приложений**

Направление подготовки

01.04.02 Прикладная математика и информатика

Направленность (профиль) программы магистратуры

01.04.02.04 Искусственный интеллект: математические модели и прикладные решения

Для набора 2026 года

Квалификация
Магистр

КАФЕДРА Информационных систем и прикладной информатики**Распределение часов дисциплины по семестрам / курсам**

Семестр (<Курс>.<Семестр на курсе>)	3 (2.1)		Итого	
	13			
Неделя	уп	рп	уп	рп
Вид занятий				
Лекции	34	34	34	34
Лабораторные	34	34	34	34
Итого ауд.	68	68	68	68
Контактная работа	68	68	68	68
Сам. работа	76	76	76	76
Часы на контроль	36	36	36	36
Итого	180	180	180	180

ОСНОВАНИЕ

Учебный план утвержден учёным советом Университета (протокол № 9 от 03.03.2026 г.).

Программу составил(и): доцент, Хаймин Е.С.

Зав. кафедрой: д.э.н., профессор С.М. Щербаков

Методический совет направления: д.э.н., доцент Ю.Г. Чернышева

Директор института магистратуры: д.э.н., профессор Е.А. Иванова

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	получение теоретических знаний и практических навыков построения и эффективного внедрения нейронных сетей в приложения для мобильных устройств.
-----	---

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ПК-5. Способен руководить проектами по созданию, поддержке и использованию системы искусственного интеллекта на основе нейросетевых моделей и методов

В результате освоения дисциплины обучающийся должен:

Знать:

нейросетевые методы и модели (соотнесено с индикатором ПК-5.1)

Уметь:

проектировать системы искусственного интеллекта для мобильных приложений с использованием нейросетевых методов и моделей (соотнесено с индикатором ПК-5.2)

Владеть:

навыками разработки и использования систем искусственного интеллекта для мобильных приложений с использованием нейросетевых методов и моделей (соотнесено с индикатором ПК-5.3)

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Раздел 1. Фреймворки для обучения и работы нейронных сетей

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
1.1	Тема 1.1. Фреймворки Tensorflow и PyTorch. Виды представления графа нейронной сети Фреймворки для обучения нейронных сетей. Статический граф вычислений. Динамический граф вычислений. Фреймворк Tensorflow. Фреймворк PyTorch. Фреймворк Keras. Сравнение Tensorflow и PyTorch. Конвертация моделей. Проблемы конвертации моделей. Фреймворк ONNX.	Лекционные занятия	3	4	ПК-5
1.2	Тема 1.1. Фреймворки Tensorflow и PyTorch. Виды представления графа нейронной сети Фреймворки для обучения нейронных сетей. Статический граф вычислений. Динамический граф вычислений. Фреймворк Tensorflow. Фреймворк PyTorch. Фреймворк Keras. Сравнение Tensorflow и PyTorch. Конвертация моделей. Проблемы конвертации моделей. Фреймворк ONNX.	Лабораторные занятия	3	4	ПК-5
1.3	Тема 1.2. Фреймворки для внедрения нейронных сетей в мобильные приложения Универсальный фреймворк Tensorflow-Lite. Расширение фреймворка PyTorch Mobile. Фреймворк Tensorflow-JS. Сравнение Tensorflow-Lite и PyTorch Mobile. Фреймворк CoreML.	Лекционные занятия	3	4	ПК-5
1.4	Тема 1.2. Фреймворки для внедрения нейронных сетей в мобильные приложения Универсальный фреймворк Tensorflow-Lite. Расширение фреймворка PyTorch Mobile. Фреймворк Tensorflow-JS. Сравнение Tensorflow-Lite и PyTorch Mobile. Фреймворк CoreML.	Лабораторные занятия	3	4	ПК-5
1.5	Виды представления графа нейронной сети. Фреймворки для обучения нейронных сетей. Статический граф вычислений. Динамический граф вычислений.	Самостоятельная работа	3	38	ПК-5

Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
2.1	Тема 2.1. Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях Конвертация моделей с фреймворка Tensorflow на Tensorflow-Lite. Конвертация моделей с фреймворка Keras на Tensorflow-Lite. Трассировка модели на фреймворке PyTorch. Использование PyTorch Mobile. Перенос модели с PyTorch на ONNX. Перенос модели с ONNX на Tensorflow-Lite. Перенос модели с PyTorch на Tensorflow-Lite. Перенос модели с PyTorch на CoreML.	Лекционные занятия	3	4	ПК-5

2.2	Тема 2.1. Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях Конвертация моделей с фреймворка Tensorflow на Tensorflow-Lite. Конвертация моделей с фреймворка Keras на Tensorflow-Lite. Трансировка модели на фреймворке PyTorch. Использование PyTorch Mobile. Перенос модели с PyTorch на ONNX. Перенос модели с ONNX на Tensorflow-Lite. Перенос модели с PyTorch на Tensorflow-Lite. Перенос модели с PyTorch на CoreML.	Лабораторные занятия	3	4	ПК-5
2.3	Тема 2.2. Технологии для ускорения вычислений нейронных сетей на мобильных устройствах Оптимизация модели в Tensorflow-Lite. Квантизация модели в Tensorflow-Lite. Квантизация модели в PyTorch Mobile. Технология WebGL.	Лекционные занятия	3	4	ПК-5
2.4	Тема 2.2. Технологии для ускорения вычислений нейронных сетей на мобильных устройствах Оптимизация модели в Tensorflow-Lite. Квантизация модели в Tensorflow-Lite. Квантизация модели в PyTorch Mobile. Технология WebGL.	Лабораторные занятия	3	4	ПК-5
2.5	Тема 2.3. Реализация приложения для классификации изображений для Android и iOS устройств Использование библиотеки Tensorflow-Lite на языке Java. Использование библиотеки Tensorflow-Lite на языке Swift. Загрузка Tensorflow-Lite модели в Android проекте. Загрузка модели в Tensorflow-Lite iOS проекте. Загрузка CoreML модели в iOS проекте. Загрузка PyTorch модели в Android проекте. Загрузка PyTorch модели в iOS проекте. Обработка изображения нейросетью в мобильном приложении.	Лекционные занятия	3	4	ПК-5
2.6	Тема 2.3. Реализация приложения для классификации изображений для Android и iOS устройств Использование библиотеки Tensorflow-Lite на языке Java. Использование библиотеки Tensorflow-Lite на языке Swift. Загрузка Tensorflow-Lite модели в Android проекте. Загрузка модели в Tensorflow-Lite iOS проекте. Загрузка CoreML модели в iOS проекте. Загрузка PyTorch модели в Android проекте. Загрузка PyTorch модели в iOS проекте. Обработка изображения нейросетью в мобильном приложении.	Лабораторные занятия	3	4	ПК-5
2.7	Тема 2.4. Реализация приложения для нахождения и сегментации объектов на изображениях для Android и iOS устройств Архитектура сегментирующей нейронной сети YOLACT. Преобразование YOLACT для поддержки PyTorch Mobile. Внедрение YOLACT в Android приложение. Внедрение YOLACT в iOS приложение.	Лекционные занятия	3	4	ПК-5
2.8	Тема 2.4. Реализация приложения для нахождения и сегментации объектов на изображениях для Android и iOS устройств Архитектура сегментирующей нейронной сети YOLACT. Преобразование YOLACT для поддержки PyTorch Mobile. Внедрение YOLACT в Android приложение. Внедрение YOLACT в iOS приложение.	Лабораторные занятия	3	4	ПК-5
2.9	Тема 2.5. Реализация приложения с нейросетевой обработкой изображений для серверных решений Использование библиотеки PyTorch на языке C++. Использование библиотеки Tensorflow на языке C++. Работа с фреймворком Flask. Работа с Flask и PyTorch. Использование библиотеки Tensorflow-JS на языке JavaScript.	Лекционные занятия	3	4	ПК-5
2.10	Тема 2.5. Реализация приложения с нейросетевой обработкой изображений для серверных решений Использование библиотеки PyTorch на языке C++. Использование библиотеки Tensorflow на языке C++. Работа с фреймворком Flask. Работа с Flask и PyTorch. Использование библиотеки Tensorflow-JS на языке JavaScript.	Лабораторные занятия	3	4	ПК-5
2.11	Тема 2.6. Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano Модуль DNN библиотеки OpenCV. Эффективная компиляция библиотеки OpenCV для устройств Nvidia Jetson. Перенос моделей на фреймворк Caffe. Запуск ONNX модели на OpenCV DNN. Сравнение производительности фреймворков Tensorflow и PyTorch на устройстве Nvidia Jetson Nano. Использование библиотеки Tensorflow-Lite на устройстве Raspberry Pi.	Лекционные занятия	3	4	ПК-5
2.12	Тема 2.6. Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano Модуль DNN библиотеки OpenCV. Эффективная компиляция библиотеки OpenCV для устройств Nvidia Jetson. Перенос моделей на фреймворк Caffe. Запуск ONNX модели на OpenCV DNN. Сравнение производительности фреймворков Tensorflow и PyTorch на устройстве Nvidia Jetson Nano. Использование библиотеки Tensorflow-Lite на устройстве Raspberry Pi.	Лабораторные занятия	3	4	ПК-5
2.13	Тема 2.7. Реализация приложения для нахождения и сегментации	Лекционные	3	2	ПК-5

	объектов на изображении для фреймворка Bgacuda (Unity3D) Использование нейросетей в игровом движке Unity. Фреймворк Bgacuda. Перенос модели с фреймворка Tensorflow на Bgacuda. Перенос модели с фреймворка PyTorch на Bgacuda. Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.	занятия			
2.14	Тема 2.7. Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Bgacuda (Unity3D) Использование нейросетей в игровом движке Unity. Фреймворк Bgacuda. Перенос модели с фреймворка Tensorflow на Bgacuda. Перенос модели с фреймворка PyTorch на Bgacuda. Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.	Лабораторные занятия	3	2	ПК-5
2.15	Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Bgacuda (Unity3D) Использование нейросетей в игровом движке Unity. Фреймворк Bgacuda. Перенос модели с фреймворка Tensorflow на Bgacuda. Перенос модели с фреймворка PyTorch на Bgacuda. Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.	Самостоятельная работа	3	38	ПК-5
2.16	Подготовка к промежуточной аттестации	Экзамен	3	36	ПК-5

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Учебные, научные и методические издания

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
1	Емельянов А. А.	Прикладная информатика: журнал	Москва: Синергия ПРЕСС, 2006	ЭБС «Университетская библиотека онлайн»
2	Горожанина, Е. И.	Нейронные сети: учебное пособие	Самара: Поволжский государственный университет телекоммуникаций и информатики, 2017	ЭБС «IPR SMART»
3	Барский А. Б.	Введение в нейронные сети: практическое пособие	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2011	ЭБС «Университетская библиотека онлайн»
4	Трофимова, Е. А., Мазуров, Вл. Д., Гилёв, Д. В., Трофимовой, Е. А.	Нейронные сети в прикладной экономике: учебное пособие	Екатеринбург: Издательство Уральского университета, 2017	ЭБС «IPR SMART»

5.2. Профессиональные базы данных и информационные справочные системы

Национальная электронная библиотека (НЭБ) - <https://rusneb.ru/>

ИСС «КонсультантПлюс»

ИСС «Гарант» <http://www.internet.garant.ru/>

5.3. Перечень программного обеспечения

Операционная система РЕД ОС

LibreOffice

Python

5.4. Учебно-методические материалы для обучающихся с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной

учебной мебелью и техническими средствами обучения:

- столы, стулья;
- персональный компьютер / ноутбук (переносной);
- проектор;
- экран / интерактивная доска.

Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет, и/или в специализированных лабораториях, предусмотренных образовательной программой.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа к электронной информационно-образовательной среде.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1. Показатели и критерии оценивания компетенций:

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ПК-5: Способен руководить проектами по созданию, поддержке и использованию системы искусственного интеллекта на основе нейросетевых моделей и методов			
3 нейросетевые методы и модели	формулирует и знает основные понятия, определения, алгоритмы и технологии	полнота и содержательность ответа умение приводить примеры	О – опрос (1-18), Э – вопросы к экзамену (1-54)
У проектировать системы искусственного интеллекта для мобильных приложений с использованием нейросетевых методов и моделей	выполняет задания, отвечает на вопросы, применяет техническое и программное обеспечение для решения задач	полнота и содержательность ответа умение приводить примеры и выполнять задания умение самостоятельно находить решение поставленных задач	ЛЗ – лабораторные задания (1-9)
В навыками разработки и использования систем искусственного интеллекта для мобильных приложений с использованием нейросетевых методов и моделей	выполняет задания, проводит анализ данных и их обработку с использованием информационно-коммуникационных технологий	полнота и содержательность ответа умение приводить примеры и выполнять задания умение самостоятельно находить решение поставленных задач	ЛЗ – лабораторные задания (1-9)

1.2 Шкалы оценивания:

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале:

- 84-100 баллов (оценка «отлично»);
- 67-83 баллов (оценка «хорошо»);
- 50-66 баллов (оценка «удовлетворительно»);
- 0-49 баллов (оценка «неудовлетворительно»).

2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Вопросы к экзамену

- 1) Виды представления графа нейронной сети Фреймворки для обучения нейронных сетей.
- 2) Статический граф вычислений.
- 3) Динамический граф вычислений.
- 4) Фреймворк Tensorflow.
- 5) Фреймворк PyTorch.
- 6) Фреймворк Keras.
- 7) Конвертация моделей. Проблемы конвертации моделей.
- 8) Фреймворк ONNX.
- 9) Универсальный фреймворк Tensorflow-Lite.
- 10) Расширение фреймворка PyTorch Mobile.
- 11) Фреймворк Tensorflow-JS.
- 12) Фреймворк CoreML.

- 13) Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях.
- 14) Конвертация моделей с фреймворка Tensorflow на Tensorflow-Lite.
- 15) Конвертация моделей с фреймворка Keras на Tensorflow-Lite.
- 16) Трассировка модели на фреймворке PyTorch.
- 17) Использование PyTorch Mobile.
- 18) Перенос модели с PyTorch на ONNX. Перенос модели с ONNX на Tensorflow-Lite. Перенос модели с PyTorch на Tensorflow-Lite.
- 19) Перенос модели с PyTorch на CoreML.
- 20) Технологии для ускорения вычислений нейронных сетей на мобильных устройствах.
- 21) Оптимизация модели в Tensorflow-Lite.
- 22) Квантизация модели в Tensorflow-Lite.
- 23) Квантизация модели в PyTorch Mobile.
- 24) Технология WebGL.
- 25) Реализация приложения для классификации изображений для Android и iOS устройств.
- 26) Использование библиотеки Tensorflow-Lite на языке Java.
- 27) Использование библиотеки Tensorflow-Lite на языке Swift.
- 28) Загрузка Tensorflow-Lite модели в Android проекте.
- 29) Загрузка модели в Tensorflow-Lite iOS проекте.
- 30) Загрузка CoreML модели в iOS проекте.
- 31) Загрузка PyTorch модели в Android проекте.
- 32) Загрузка PyTorch модели в iOS проекте.
- 33) Обработка изображения нейросетью в мобильном приложении.
- 34) Реализация приложения для нахождения и сегментации объектов на изображении для Android и iOS устройств.
- 35) Архитектура сегментирующей нейронной сети YOLACT.
- 36) Преобразование YOLACT для поддержки PyTorch Mobile.
- 37) Внедрение YOLACT в Android приложение.
- 38) Внедрение YOLACT в iOS приложение.
- 39) Работа с фреймворком Flask.
- 40) Работа с Flask и PyTorch.
- 41) Использование библиотеки Tensorflow-JS на языке JavaScript.
- 42) Внедрение нейронных сетей во встраиваемые устройства и игровые движки
- 43) Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano.
- 44) Модуль DNN библиотеки OpenCV.
- 45) Эффективная компиляция библиотеки OpenCV для устройств Nvidia Jetson.
- 46) Перенос моделей на фреймворк Caffe.
- 47) Запуск ONNX модели на OpenCV DNN.
- 48) Использование библиотеки Tensorflow-Lite на устройстве Raspberry Pi.
- 49) Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Barracuda (Unity3D).
- 50) Использование нейросетей в игровом движке Unity.
- 51) Фреймворк Barracuda.
- 52) Перенос модели с фреймворка Tensorflow на Barracuda.
- 53) Перенос модели с фреймворка PyTorch на Barracuda.
- 54) Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.

Экзаменационное задание включает три вопроса – два теоретических вопроса и одно практико-ориентированное задание из числа приведенных ниже лабораторных заданий.

Критерии оценивания:

- 84-100 баллов (оценка «отлично») – изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины в соответствии с поставленными программой курса целями и задачами обучения; правильные, уверенные действия

по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;

- 67-83 баллов (оценка «хорошо») – наличие твердых и достаточно полных знаний в объеме пройденной программы дисциплины в соответствии с целями обучения, правильные действия по применению знаний на практике, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, обучающийся усвоил основную литературу, рекомендованную в рабочей программе дисциплины;

- 50-66 баллов (оценка «удовлетворительно») – наличие твердых знаний в объеме пройденного курса в соответствии с целями обучения, изложение ответов с отдельными ошибками, уверенно исправленными после дополнительных вопросов; правильные в целом действия по применению знаний на практике;

- 0-49 баллов (оценка «неудовлетворительно») – ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Задания для опроса

Вариант 1

Виды представления графа нейронной сети Фреймворки для обучения нейронных сетей.

Статический граф вычислений.

Динамический граф вычислений.

Вариант 2

Фреймворк Tensorflow.

Фреймворк PyTorch.

Фреймворк Keras.

Вариант 3

Конвертация моделей. Проблемы конвертации моделей.

Фреймворк ONNX.

Универсальный фреймворк Tensorflow-Lite.

Вариант 4

Расширение фреймворка PyTorch Mobile.

Фреймворк Tensorflow-JS.

Фреймворк CoreML.

Вариант 5

Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях.

Конвертация моделей с фреймворка Tensorflow на Tensorflow-Lite.

Конвертация моделей с фреймворка Keras на Tensorflow-Lite.

Вариант 6

Трассировка модели на фреймворке PyTorch.

Использование PyTorch Mobile.

Перенос модели с PyTorch на ONNX. Перенос модели с ONNX на Tensorflow-Lite. Перенос модели с PyTorch на Tensorflow-Lite.

Вариант 7

Перенос модели с PyTorch на CoreML.

Технологии для ускорения вычислений нейронных сетей на мобильных устройствах.

Оптимизация модели в Tensorflow-Lite.

Вариант 8

Квантизация модели в Tensorflow-Lite.

Квантизация модели в PyTorch Mobile.

Технология WebGL.

Вариант 9

Реализация приложения для классификации изображений для Android и iOS устройств.

Использование библиотеки Tensorflow-Lite на языке Java.

Использование библиотеки Tensorflow-Lite на языке Swift.

Вариант 10

Загрузка Tensorflow-Lite модели в Android проекте.

Загрузка модели в Tensorflow-Lite iOS проекте.

Загрузка CoreML модели в iOS проекте.

Вариант 11

Загрузка PyTorch модели в Android проекте.

Загрузка PyTorch модели в iOS проекте.

Обработка изображения нейросетью в мобильном приложении.

Вариант 12

Реализация приложения для нахождения и сегментации объектов на изображении для Android и iOS устройств.

Архитектура сегментирующей нейронной сети YOLACT.

Преобразование YOLACT для поддержки PyTorch Mobile.

Вариант 13

Внедрение YOLACT в Android приложение.

Внедрение YOLACT в iOS приложение.

Работа с фреймворком Flask.

Вариант 14

Работа с Flask и PyTorch.

Использование библиотеки Tensorflow-JS на языке JavaScript.

Внедрение нейронных сетей во встраиваемые устройства и игровые движки

Вариант 15

Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano.

Модуль DNN библиотеки OpenCV.

Эффективная компиляция библиотеки OpenCV для устройств Nvidia Jetson.

Вариант 16

Перенос моделей на фреймворк Caffe.

Запуск ONNX модели на OpenCV DNN.

Использование библиотеки Tensorflow-Lite на устройстве Raspberry Pi.

Вариант 17

Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Barracuda (Unity3D).

Использование нейросетей в игровом движке Unity.

Фреймворк Barracuda.

Вариант 18

Перенос модели с фреймворка Tensorflow на Barracuda.

Перенос модели с фреймворка PyTorch на Barracuda.

Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.

Критерии оценивания (для каждого варианта):

9-10 б. – ответы на все три вопроса варианта даны верно;

7-8 б. – один ответ из 3-х с неточностями;

5-6 б. – 2 ответа из 3-х с неточностями;

3-4 б. – 3 ответа с неточностями;

1-2 б. – нет ответа на один вопрос из 3-х;

Максимальное количество баллов за опрос – 10.

Лабораторные задания

Лабораторное задание 1.

Тема 1.1. Знакомство с фреймворками Tensorflow и PyTorch. Виды представления графа нейронной сети Фреймворки для обучения нейронных сетей. Статический граф вычислений. Динамический граф вычислений. Фреймворк Tensorflow. Фреймворк PyTorch. Фреймворк Keras.

Сравнение Tensorflow и PyTorch. Конвертация моделей. Проблемы конвертации моделей. Фреймворк ONNX.

Лабораторное задание 2.

Тема 1.2. Знакомство с фреймворками для внедрения нейронных сетей в мобильные приложения Универсальный фреймворк Tensorflow-Lite. Расширение фреймворка PyTorch Mobile. Фреймворк Tensorflow-JS. Сравнение Tensorflow-Lite и PyTorch Mobile. Фреймворк CoreML.

Лабораторное задание 3.

Тема 2.1. Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях Конвертация моделей с фреймворка Tensorflow на Tensorflow-Lite. Конвертация моделей с фреймворка Keras на Tensorflow-Lite. Трассировка модели на фреймворке PyTorch. Использование PyTorch Mobile. Перенос модели с PyTorch на ONNX. Перенос модели с ONNX на Tensorflow-Lite. Перенос модели с PyTorch на Tensorflow-Lite. Перенос модели с PyTorch на CoreML.

Лабораторное задание 4.

Тема 2.2. Технологии для ускорения вычислений нейронных сетей на мобильных устройствах Оптимизация модели в Tensorflow-Lite. Квантизация модели в Tensorflow-Lite. Квантизация модели в PyTorch Mobile. Технология WebGL.

Лабораторное задание 5.

Тема 2.3. Реализация приложения для классификации изображений для Android и iOS устройств Использование библиотеки Tensorflow-Lite на языке Java. Использование библиотеки Tensorflow-Lite на языке Swift. Загрузка Tensorflow-Lite модели в Android проекте. Загрузка модели в Tensorflow-Lite iOS проекте. Загрузка CoreML модели в iOS проекте. Загрузка PyTorch модели в Android проекте. Загрузка PyTorch модели в iOS проекте. Обработка изображения нейросетью в мобильном приложении.

Лабораторное задание 6.

Тема 2.4. Реализация приложения для нахождения и сегментации объектов на изображении для Android и iOS устройств Архитектура сегментирующей нейронной сети YOLACT. Преобразование YOLACT для поддержки PyTorch Mobile. Внедрение YOLACT в Android приложение. Внедрение YOLACT в iOS приложение.

Лабораторное задание 7.

Тема 2.5. Реализация приложения с нейросетевой обработкой изображений для серверных решений Использование библиотеки PyTorch на языке C++. Использование библиотеки Tensorflow на языке C++. Работа с фреймворком Flask. Работа с Flask и PyTorch. Использование библиотеки Tensorflow-JS на языке JavaScript.

Лабораторное задание 8.

Тема 2.6. Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano Модуль DNN библиотеки OpenCV. Эффективная компиляция библиотеки OpenCV для устройств Nvidia Jetson. Перенос моделей на фреймворк Caffe. Запуск ONNX модели на OpenCV DNN. Сравнение производительности фреймворков Tensorflow и PyTorch на устройстве Nvidia Jetson Nano. Использование библиотеки Tensorflow-Lite на устройстве Raspberry Pi.

Лабораторное задание 9.

Тема 2.7. Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Barracuda (Unity3D) Использование нейросетей в игровом движке Unity. Фреймворк Barracuda. Перенос модели с фреймворка Tensorflow на Barracuda. Перенос модели с фреймворка PyTorch на Barracuda. Использование нейросетей на фреймворке Tensorflow в игровом движке UnrealEngine 4/5.

Критерии оценивания (для каждого задания):

9-10 б. – задание выполнено верно;

6-8 б. – при выполнении задания были допущены неточности, не влияющие на результат;

3-5 б. – при выполнении задания были допущены ошибки;

0-2 б. – при выполнении задания были допущены существенные ошибки.

Максимальное количество баллов за лабораторные задания – 90 (9 заданий по 10 баллов).

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена.

Экзамен проводится по расписанию промежуточной аттестации. Количество вопросов в экзаменационном задании – 3 (два теоретических вопроса и одно практико-ориентированное задание). Объявление результатов производится в день экзамена. Результаты аттестации заносятся в ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:

- лекции;
- лабораторные занятия.

В ходе лекционных занятий рассматриваются основные теоретические вопросы, даются рекомендации для самостоятельной работы и подготовке к лабораторным занятиям.

В ходе лабораторных занятий углубляются и закрепляются знания студентов по ряду рассмотренных на лекциях вопросов, развиваются навыки практической работы.

При подготовке к лабораторным занятиям каждый студент должен:

- изучить рекомендованную учебную литературу;
- изучить конспекты лекций;
- подготовить ответы на все вопросы по изучаемой теме.

В процессе подготовки к лабораторным занятиям студенты могут воспользоваться консультациями преподавателя.

Вопросы, не рассмотренные на лекциях, лабораторных занятиях, должны быть изучены студентами в ходе самостоятельной работы. Контроль самостоятельной работы студентов над учебной программой курса осуществляется в ходе занятий методом опроса и выполнения лабораторных заданий. В ходе самостоятельной работы каждый студент обязан прочитать основную и по возможности дополнительную литературу по изучаемой теме, дополнить конспекты лекций недостающим материалом, выписками из рекомендованных первоисточников, выделить непонятные термины, найти их значение в энциклопедических словарях.

Студент должен готовиться к предстоящему лабораторному занятию по всем обозначенным в рабочей программе дисциплины вопросам.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации студенты могут воспользоваться электронно-библиотечными системами. Также обучающиеся могут взять на дом необходимую литературу на абонементе университетской библиотеки или воспользоваться читальными залами.