

Документ подписан простой электронной подписью
Информация о владельце:

ФИО: Макаренко Елена Николаевна

Должность: Ректор

Дата подписания: 29.07.2022 17:56:09

Уникальный программный ключ:

c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ

Директор Института магистратуры

Иванова Е.А.

« 22 » февраля 2022 г.

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

Направление 01.04.02 Прикладная математика и информатика
магистерская программа 01.04.02.04 "Искусственный интеллект: математические модели и прикладные решения"

Для набора 2022 года

Квалификация
Магистр

Составитель(и) программы:

Богачев Т.В., к.ф.-м.н., доц кафедры фундаментальной и прикладной математики
Программа одобрена на заседании кафедры высшей фундаментальной и прикладной математики
«22» февраля 2022 г., протокол №6

ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цели:

Изучение полного цикла построения и эффективного внедрения нейронных сетей в приложения для мобильных устройств

Задачи:

Обзор популярных фреймворков для обучения нейронных сетей;

Изучение методов внедрения нейронных сетей в веб-приложения;

Изучение методов внедрения нейронных сетей в iOS и Android приложения;

Обзор методов конвертации нейронных сетей для эффективного использования ресурсов мобильного устройства;

Изучение методов внедрения нейронных сетей для использования во встраиваемых системах;

Изучение технологии WebGL и ускорение нейронных сетей с помощью данной технологии.

МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

2.1. Учебная дисциплина «Нейронные сети для мобильных приложений» (2 курс магистратуры, 3 семестр) относится к вариативной части блока дисциплин (модулей) и является дисциплиной по выбору.

2.2. Для изучения данной учебной дисциплины необходимы знания, умения и навыки, формируемые дисциплинами магистратуры «Основы нейронных сетей» и «Прикладное машинное обучение».

2.3. Знания и навыки, полученные в ходе изучения данной дисциплины, могут использоваться для решения профессиональных задач в научно-исследовательской, научно-производственной и проектной деятельности, в частности, при разработке мобильных приложений, использующих нейронные сети, в рамках подготовки выпускной квалификационной работы.

ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ОП ВО по данному направлению подготовки (специальности):

Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы:

Компетенция	Индикаторы компетенций	Результаты обучения
ПК-5. Способен руководить проектами по созданию, поддержке и использованию системы искусственного интеллекта на основе нейросетевых моделей и методов	ПК-5.1. Руководит работами по оценке и выбору моделей искусственных нейронных сетей и инструментальных средств для решения поставленной задачи	<p>Знать: функциональность современных инструментальных средств и систем программирования в области создания моделей искусственных нейронных сетей</p> <p>Уметь: проводить оценку и выбор моделей искусственных нейронных сетей и инструментальных средств для решения задач машинного обучения</p> <p>Уметь: применять современные инструментальные средства и системы программирования для разработки и обучения моделей искусственных нейронных сетей</p>
	ПК-5.2. Руководит созданием систем искусственного интеллекта на основе моделей искусственных нейронных сетей и инструментальных средств	<p>Знать: принципы построения систем искусственного интеллекта на основе искусственных нейронных сетей, методы и подходы к планированию и реализации проектов по созданию систем искусственного интеллекта</p> <p>Уметь: руководить выполнением коллективной проектной деятельности для создания, поддержки и использования систем искусственного интеллекта на основе искусственных нейронных сетей</p>
	ПК-5.3. Руководит проектами по разработке систем искусственного интеллекта на основе моделей глубоких нейронных сетей и нечетких моделей и методов	<p>Знать: принципы построения моделей глубоких нейронных сетей и глубокого машинного обучения (с подкреплением и без)</p> <p>Знать: подходы к применению моделей на основе нечеткой логики в системах искусственного интеллекта</p> <p>Уметь: руководить выполнением коллективной проектной деятельности для создания, поддержки и использования систем искусственного интеллекта на основе моделей глубоких нейронных сетей и нечетких моделей и методов</p>

СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов, из них 34 часа лекционных занятий, 34 часа лабораторных занятий, 76 часов на самостоятельную работу и 36 часов на подготовку к экзамену.

Форма отчетности: экзамен в 3 семестре

4.1 Содержание дисциплины, структурированное по темам, с указанием видов учебных занятий и отведенного на них количества академических часов

№ п/п	Раздел дисциплины/темы	Семестр	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам)
			Контактная работа преподавателя с обучающимися					
			Лекции	Семинарские (практические занятия)	Лабораторные занятия			
1	Раздел 1. Знакомство с множеством фреймворков для обучения и работы нейронных сетей	1	4	0	4	15	Тестирование.	
1.1	Знакомство с фреймворками Tensorflow и PyTorch. Виды представления графа нейронной сети.	1	2		2	5		
1.2	Знакомство с фреймворками для внедрения нейронных сетей в мобильные приложения	1	2		2	10		
2	Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения	1	6	4	4	25	Выполнение лабораторных работ. Выполнение дополнит. заданий	
2.1	Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях.	1	3	2	2	15		
2.2	Технологии для ускорения вычислений нейронных сетей на мобильных устройствах	1	3	2	2	10		

№ п/п	Раздел дисциплины/темы	Семестр	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)			Самостоятельная работа	Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам)
			Контактная работа преподавателя с обучающимися				
2	Раздел 3. Примеры внедрения нейронных сетей в мобильных приложения	1	10	4	18	43	Выполнение лабораторных работ. Выполнение дополнит. заданий
2.1	Реализация приложения для классификации изображений для Android и iOS устройств	1	3	2	4	14	
2.2	Реализация приложения для нахождения и сегментации объектов на изображении для Android и iOS устройств	1	3		7	14	
2.3	Реализация приложения с нейросетевой обработкой изображений для серверных решений	1	4	2	7	15	
3	Раздел 4. Внедрение нейронных сетей во встраиваемые устройства и игровые движки	1	4	4	10	25	Выполнение лабораторных работ. Выполнение дополнит. заданий
3.1	Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano	1	2	2	5	15	
3.2	Реализация приложения для нахождения и сегментации объектов на изображении для фреймворка Barracuda (Unity3D)	1	2	2	5	10	
Итого часов по курсу			24	12	36	108	
Подготовка к экзамену						52	

4.2 План внеаудиторной самостоятельной работы обучающихся по дисциплине

Се- местр	Название раздела, темы	Самостоятельная работа обучающихся			Оценочное средство	Учебно- методическое обес- печение самостоя- тельной работы
		Вид самостоятельной работы	Сроки выпол- нения	Затраты времени (час.)		
1	Знакомство с множеством фреймворков для обучения и работы нейронных сетей.	Изучение лекции и учебной литературы	4 недели	15	Тестирование	Материалы лекций, рекомендованная учебная литература, материалы, выложенные на интернет-странице курса на сайте edu.mmcs.sfedu.ru
1	Технологии внедрения нейронных сетей в мобильные приложения.	Изучение лекции и учебной литературы, выполнение дополнительных заданий	4 недели	25	Проверка выполненных заданий	Материалы лекций, рекомендованная учебная литература, материалы, выложенные на интернет-странице курса на сайте edu.mmcs.sfedu.ru

Се- местр	Название раздела, темы	Самостоятельная работа обучающихся			Оценочное средство	Учебно- методическое обес- печение самостоя- тельной работы
		Вид самостоятельной работы	Сроки выпол- нения	Затраты времени (час.)		
1	Примеры внедрения нейронных сетей в мобильных приложения.	Изучение лекции и учебной литературы, выполнение дополнительных заданий	4 недели	43	Проверка выпол- ненных заданий	Материалы лек- ций, рекомендо- ванная учебная ли- тература, материа- лы, выложенные на интернет- странице курса на сайте edu.mmcs.sfedu.ru
1	Внедрение нейронных сетей во встраиваемые устройства и игровые движки.	Изучение лекции и учебной литературы, выполнение дополнительных заданий	4 недели	25	Проверка выпол- ненных заданий	Материалы лек- ций, рекомендо- ванная учебная ли- тература, материа- лы, выложенные на интернет- странице курса на сайте edu.mmcs.sfedu.ru
Общая трудоемкость самостоятельной работы по дисциплине (час)				108		
Бюджет времени самостоятельной работы, предусмотренный учебным планом для дан- ной дисциплины (час)				108		
Подготовка к экзамену				52		

Се- местр	Название раздела, темы	Самостоятельная работа обучающихся			Оценочное средство	Учебно- методическое обес- печение самостоя- тельной работы
		Вид самостоятельной работы	Сроки выпол- нения	Затраты времени (час.)		
	Общая трудоемкость самостоятельной работы в течение семестра и на подготовку к эк- замену по дисциплине (час)			112		
	Бюджет времени самостоятельной работы в течение семестра и на подготовку к экзамену по дисциплине, предусмотренный учебным планом для данной дисциплины (час)			112		

4.3 Содержание учебного материала

Раздел 1. Знакомство с множеством фреймворков для обучения и работы нейронных сетей

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.

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.

Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения

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.

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.

Раздел 3. Примеры внедрения нейронных сетей в мобильных приложения

Тема 3.1. 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.

Тема 3.2. 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.

Тема 3.3. 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.

Раздел 4. Внедрение нейронных сетей во встраиваемые устройства и игровые движки

Тема 4.1. 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.

Тема 4.2. 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.

Вопросы, выносимые на самостоятельное изучение

Архитектуры нейросетей для классификации изображений. Архитектуры нейросетей на сегментации изображений. Архитектуры нейросетей для нахождения объектов на изображении. Библиотека OpenCV. Использование библиотеки OpenCV на языке Python. Использование библиотеки OpenCV на языке C++.

ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При проведении лекций и лабораторных занятий используются следующие образовательные технологии:

- мультимедийные лекции
- электронные формы контроля
- самотестирование студентов

Учебный процесс базируется на концепции компетентностного обучения, ориентированного на формирование конкретного перечня профессиональных компетенций, актуализацию получаемых теоретических знаний. Развертывание компетентностной модели обучения предполагает широкое применение инновационных способов организации учебного процесса, в т.ч. применение метода проектного обучения, технологий управляемого самостоятельного обучения в том числе балльно-рейтинговой системы, а также внедрение системы онлайн-поддержки внеаудиторной работы студентов.

Дисциплина может быть реализована частично или полностью с использованием ЭИОС Университета (ЭО и ДОТ). Аудиторные занятия и другие формы контактной работы обучающихся с преподавателем могут проводиться с использованием платформ Microsoft Teams и MOODLE, в том числе, в режиме онлайн-лекций и онлайн-семинаров.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Полный комплект контрольно-оценочных материалов (Фонд оценочных средств) оформляется в виде приложения к рабочей программе дисциплины.

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

1. Основная литература.

1. Гудфеллоу, Я. Глубокое обучение / Я. Гудфеллоу, И. Бенджио, А. Курвилль ; перевод с английского А. А. Слинкина. — 2-е изд. — Москва : ДМК Пресс, 2018. — 652 с. — ISBN 978-5-97060-618-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/107901> (дата обращения: 10.10.2021). — Режим доступа: для авториз. пользователей.

2. Теофили, Т. Глубокое обучение для поисковых систем : руководство / Т. Теофили ; перевод с английского Д. А. Беликова. — Москва : ДМК Пресс, 2020. — 318 с. — ISBN 978-5-97060-776-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140574> (дата обращения: 10.10.2021). — Режим доступа: для авториз. пользователей.

2. Дополнительная литература.

1. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для вузов / Л. А. Станкевич. — Москва : Издательство Юрайт, 2021. — 397 с. — (Высшее образование). — ISBN 978-5-534-02126-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/469517> (дата обращения: 10.10.2021).

2. Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных [Электронный ресурс] / П. Флах. — Электрон. дан. — Москва: ДМК Пресс, 2015. — 400 с. — Режим доступа: <https://e.lanbook.com/book/69955>.

3. Список авторских методических разработок.

нет

4. Периодические издания

нет

5. Перечень ресурсов сети Интернет, необходимых для освоения дисциплины

1. Научная электронная библиотека. <http://elibrary.ru/>

2. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов» <http://school-collection.edu.ru/>

3. Университетская библиотека online: http://biblioclub.ru/index.php?page=main_ub_red
MathWorld. <http://mathworld.wolfram.com> 4. Электронно-библиотечная система (ЭБС) ЮРАЙТ www.biblio-online.ru.

6. Программное обеспечение информационно-коммуникационных технологий

Операционная система Microsoft Windows (любая версия), пакет Microsoft Office (любая версия).

МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

1. Учебно-лабораторное оборудование

При проведении дисциплины учащиеся должны быть обеспечены:

1. Лекционной аудиторией с мультимедийным презентационным оборудованием для демонстрации презентаций и иллюстративного материала.

2. Аудиторией для лабораторных занятий с аппаратными и программными средствами в соответствии с реализуемой учебной тематикой.

2. Программные средства

Microsoft Windows, Microsoft Office, Windows CAL's - Договор 232.02.02.03-16/60 от 10.08.2018 г., с 10.08.2018 г. по 10.08.2019 г.; Договор №232.02.02.03-16/46 от 30.08.2019 г., с 31.07.2019 г. по 30.07.2020 г.; Государственный контракт № SC-P/5679-01/07 от 04.12.2007 г., с 21.12.2007 г. (срок использования ПО неограничен).

Среда программирования PyCharm (любая версия).

Среда программирования Xcode (любая версия).

Среда программирования AndroidStudio (любая версия).

**Методические указания для обучающихся
по освоению дисциплины**

Методические указания приведены в учебных пособиях, перечисленных в разделе VII

УЧЕБНАЯ КАРТА ДИСЦИПЛИНЫ**Нейронные сети для мобильных приложений**

5 зач.ед.; ак.ч всего: 180, в т.ч.: 34 лекций, 34 лабор., 76 самостоят., 36 на подготовку к экзамену

Преподаватель Демяненко Яна Михайловна Алексей Сергеевич

Кафедра прикладной математики и информатики

Курс 2 (магистратура), семестр 3

Код и наименование направления подготовки (специальности): 01.04.02 «Прикладная математика и информатика»

Магистерская программа: «Искусственный интеллект: математические модели и прикладные решения»

№	Виды контрольных мероприятий	Текущий контроль	Рубежный контроль (при наличии)
	Раздел 1. Знакомство с множеством фреймворков для обучения и работы нейронных сетей	0	15
1.	Тестирование		15
	Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения	10	0
1.	Выполнение лабораторных работ	10	
	Раздел 3. Примеры внедрения нейронных сетей в мобильных приложения	20	0
1.	Выполнение лабораторных работ	20	
	Раздел 4. Внедрение нейронных сетей во встраиваемые устройства и игровые движки	15	0
1.	Выполнение лабораторных работ	15	
	Всего	45	15
	Бонусные баллы (до 10)	За выполнение дополнительных заданий	Бонусные баллы (до 10)
	Промежуточная аттестация в форме экзамена (40 баллов)	1 часть – письменная работа (30 баллов) 2 часть – устный ответ (10 баллов)	

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»
Факультет компьютерных технологий и защиты информации
Кафедра фундаментальной и прикладной математики

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

НЕЙРОННЫЕ СЕТИ ДЛЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Код и наименование направления подготовки/специальности:
01.04.02 «Прикладная математика и информатика»

Уровень образования:
Магистратура

Магистерская программа:
«Искусственный интеллект: математические модели и прикладные решения»

Форма обучения:
Очная

Ростов-на-Дону, 2022

ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ, ФОРМИРУЕМЫХ ДИСЦИПЛИНОЙ Нейронные сети для мобильных приложений

Код компетенции и	Формулировка компетенции
1	2
ПК	ПРОФЕССИОНАЛЬНЫЕ КОМПЕТЕНЦИИ
ПК-5	Способность руководить проектами по созданию, поддержке и использованию системы искусственного интеллекта на основе нейросетевых моделей и методов

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ Нейронные сети для мобильных приложений

№ n/n	<i>Контролируемые разделы дисциплины*</i>	<i>Индикатор контролируемой компетенции</i>	<i>Наименование оценочного средства**</i>
1.	Раздел 1. Знакомство с множеством фреймворков для обучения и работы нейронных сетей	ПК-5.1. Руководит работами по оценке и выбору моделей искусственных нейронных сетей и инструментальных средств для решения поставленной задачи	Тестирование. Выполнение лабораторных работ.
2.	Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения	ПК-5.1. Руководит работами по оценке и выбору моделей искусственных нейронных сетей и инструментальных средств для решения поставленной задачи	Выполнение лабораторных работ. Выполнение дополнительных заданий.
3.	Раздел 3. Примеры внедрения нейронных сетей в мобильных приложения	ПК-5.2. Руководит созданием систем искусственного интеллекта на основе моделей искусственных нейронных сетей и инструментальных средств	Выполнение лабораторных работ. Выполнение дополнительных заданий.
4.	Раздел 4. Внедрение нейронных сетей во встраиваемые устройства и игровые движки	ПК-5.3. Руководит проектами по разработке систем искусственного интеллекта на основе моделей глубоких нейронных сетей и нечетких моделей и методов	Выполнение лабораторных работ. Выполнение дополнительных заданий.

* Наименование раздела указывается в соответствии с рабочей программой дисциплины.

**Наименование оценочного средства указывается в соответствии с учебной картой дисциплины.

Кафедра прикладной математики и программирования

Письменные тесты

по дисциплине «Нейронные сети для мобильных приложений»

1. Банк тестов по разделам

Раздел 1 «**Знакомство с множеством фреймворков для обучения и работы нейронных сетей**»

В квадратных скобках указывается число баллов за правильный ответ.

Какой из фреймворков использует динамическое представление графа вычислений?

[1]

PyTorch

Tensorflow

Tensorflow-Lite

Ни один из перечисленных

Какой из фреймворков нельзя использовать для внедрения в iOS приложение? [2]

PyTorch Mobile

Tensorflo-Lite

CoreML

Все вышеперечисленные применимы

Чем фреймворк Tensorflow-Lite отличается от Tensorflow? [2]

Имеет динамическое представление графа вычислений

Ориентирован для запуска моделей на мобильных устройствах

Используется для запуска нейронных сетей только на Android устройствах

Является промежуточным фреймворком для экспорта модели с фреймворка PyTorch на CoreML

Какой фреймворк обычно используется для экспорта моделей с фреймворка PyTorch на CoreML и Tensorflow? [2]

Tensorflow-Lite

MXNet

ONNX

PyTorch Mobile

Какой фреймворк является высокоуровневой надстройкой над фреймворком Tensorflow? [1]

Keras

Tensorflow-Lite

MXNet

Gluon

Можно ли web-ориентированные модели с фреймворка Tensorflow-JS использовать в мобильных приложениях Android и iOS? [2]

Нет

Можно, если конвертировать модели в поддерживаемые данными платформами фреймворки

Любая ли архитектура нейронной сети поддерживается GPU ускорителем в Android приложении при использовании фреймворка Tensorflow-Lite? [2]

Да

Нет, так как ускорителем GPU в Android приложениях поддерживается ограниченный набор операций

Какое одно из основных преимуществ динамического графа вычислений? [1]

Скорость вычислений

Гибкая отладка разрабатываемой архитектуры сети

Лучшая переносимость модели

Какой из фреймворков нельзя использовать для приложений на Android? [1]

Tensorflow-Lite

PyTorch Mobile

CoreML

Все из выше перечисленных

Модель на каком из перечисленных фреймворках даст большую производительность при использовании в приложении для iOS?

Tensorflow-Lite

PyTorch Mobile

CoreML

MXNet

2. Критерии оценки:

- оценка «отлично» (15 итоговых баллов) выставляется студенту, если он набрал от 12 до 15 баллов;

- оценка «хорошо» (12 итоговых баллов) выставляется студенту, если он набрал от 10 до 12 баллов;
- оценка «удовлетворительно» (8 итоговых баллов) выставляется студенту, если он набрал от 7 до 8 баллов;
- оценка «удовлетворительно» (6 итоговых балла) выставляется студенту, если он набрал от 1 до 6 баллов;
- оценка «неудовлетворительно» (0 итоговых баллов) выставляется студенту, если он набрал менее 1 балла.

К комплекту тестов прилагаются ключи.

1	a
2	d
3	b
4	c
5	a
6	b
7	b
8	b
9	c
10	c

Состав

Лабораторные работы по дисциплине «Нейронные сети для мобильных приложений»

1. Тематика лабораторных работ по разделам и темам

Приведен один из вариантов лабораторных работ. Другие варианты содержат такое же количество однотипных заданий.

Раздел 2. Технологии внедрения нейронных сетей в мобильные приложения

Тема 2.1. Способы эффективного переноса архитектур нейронных сетей на фреймворки, используемые в мобильных приложениях

Лабораторная работа 1 (2 балла)

Требуется перенести одну из предобученных моделей для классификации из зоопарка моделей фреймворка PyTorch на фреймворк для мобильных приложений Tensorflow-Lite и CoreML используя фреймворки ONNX и Keras.

Лабораторная работа 2 (3 балла)

Требуется перенести одну из предобученных моделей для сегментации из зоопарка моделей фреймворка PyTorch на фреймворк для мобильных приложений Tensorflow-Lite и CoreML используя фреймворки ONNX и Keras.

Лабораторная работа 3 (3 балла)

Требуется перенести одну из предобученных моделей для обнаружения объектов из зоопарка моделей фреймворка PyTorch на фреймворк для мобильных приложений Tensorflow-Lite и CoreML используя фреймворки ONNX и Keras.

Тема 2.2. Технологии для ускорения вычислений нейронных сетей на мобильных устройствах

Лабораторная работа 4 (2 балла)

Требуется ускорить нейронную сеть из первой лабораторной работы на фреймворке Tensorflow-Lite используя квантизацию.

Использование дополнительных оптимизаций нейронной сети при конвертации на фреймворк Tensorflow-Lite. (необязательное дополнительное задание)

Раздел 3. Примеры внедрения нейронных сетей в мобильных приложения

Тема 3.1. Реализация приложения для классификации изображений для Android и iOS устройств

Лабораторная работа 5 (5 баллов)

Требуется запустить классифицирующую модель на фреймворке Tensorflow-Lite или PyTorch Mobile в приложении для Android устройства.

Разработка и обучение собственной архитектуры нейронной сети и её интеграция в мобильное приложение на системах Android или iOS. (необязательное дополнительное задание)

Тема 3.2. Реализация приложения для нахождения и сегментации объектов на изображении для Android и iOS устройств

Лабораторная работа 6 (5 баллов)

Требуется запустить YOLOv5 модель на фреймворке Tensorflow-Lite или PyTorch Mobile в приложении для Android устройства.

Тема 3.3. Реализация приложения с нейросетевой обработкой изображений для серверных решений

Лабораторная работа 7 (10 баллов)

Требуется реализовать серверное решение на фреймворке Flask с использованием нейронных сетей на фреймворках PyTorch или Tensorflow.

Раздел 4. Внедрение нейронных сетей во встраиваемые устройства и игровые движки

Тема 4.1. Реализация приложения с нейросетевой обработкой изображений для Nvidia Jetson Nano

Лабораторная работа 8 (5 баллов)

Требуется внедрить модель на фреймворке Caffe в OpenCV приложение с помощью модуля DNN.

Лабораторная работа 9 (10 баллов)

Требуется реализовать и развернуть решение для обнаружения лиц на встраиваемом устройстве Nvidia Jetson Nano.

2. Методические рекомендации по выполнению лабораторных работ

Цель лабораторных работ

Целью лабораторных работ является самостоятельная реализация проекта мобильного приложения или серверного решения, использующего фреймворки глубокого обучения.

Организация самостоятельной работы

- При подготовке к лабораторной работе необходимо изучить лекционные материалы по темам из соответствующей лабораторной работе главы.
- Все лабораторные работы выполняются на операционной системе Ubuntu в среде программирования PyCharm, Android Studio или AppCode.

3. Критерии оценки:

Общее количество баллов:

- раздел 2: 10
- раздел 3: 20
- раздел 4: 15

**Экзаменационная программа по дисциплине
«Нейронные сети для мобильных приложений»**

1. 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.
2. Универсальный фреймворк Tensorflow-Lite. Расширение фреймворка PyTorch Mobile. Фреймворк Tensorflow-JS. Сравнение Tensorflow-Lite и PyTorch Mobile. Фреймворк CoreML.
3. 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.
4. Оптимизация модели в Tensorflow-Lite. Квантизация модели в Tensorflow-Lite. Квантизация модели в PyTorch Mobile. Технология WebGL.
5. 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.
6. Архитектура сегментирующей нейронной сети YOLACT. Преобразование YOLACT для поддержки PyTorch Mobile. Внедрение YOLACT в Android приложение. Внедрение YOLACT в iOS приложение.
7. 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.
8. 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.
9. 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.

10. Архитектуры нейросетей для классификации изображений. Архитектуры нейросетей на сегментации изображений. Архитектуры нейросетей для нахождения объектов на изображении. Библиотека OpenCV. Использование библиотеки OpenCV на языке Python. Использование библиотеки OpenCV на языке C++.

Экзаменационный билет № 1

по дисциплине «Нейронные сети для мобильных приложений»

1. Квантизация модели в PyTorch Mobile.
2. Перенос модели с PyTorch на Tensorflow-Lite.
3. Использование нейросетей в игровом движке Unity.

Критерии оценки

Экзамен проводится в двух частях: 1 часть – письменная работа (30 баллов), 2 часть – устный ответ (10 баллов).

За каждый вопрос письменной части можно получить до 6 баллов, количество баллов зависит от полноты ответа:

10 баллов - ответ полный и правильный; материал изложен в определенной логической последовательности, литературным языком; ответ самостоятельный;

7-9 баллов - ответ полный и правильный; материал изложен в определенной логической последовательности, при этом допущены две-три несущественные ошибки, исправленные по требованию преподавателя;

4-6 баллов - ответ полный, но при этом допущена существенная ошибка, или неполный, несвязный;

1-3 балла - при ответе обнаружено непонимание обучающимся основного содержания учебного материала, или допущены существенные ошибки, которые обучающийся не смог исправить при наводящих вопросах преподавателя;

0 баллов - ответ отсутствует.

В устную часть входят 5 вопросов по программе, предполагающих короткие ответы. За устную часть можно получить до 10 баллов. За каждый вопрос можно получить до 2-х баллов, количество баллов зависит от полноты ответа:

2 балла - ответ полный и правильный; ответ самостоятельный;

1 балл - ответ полный, но при этом допущена существенная ошибка, или неполный, несвязный;

0 баллов - при ответе обнаружено непонимание обучающимся основного содержания учебного материала, или допущены существенные ошибки, которые обучающийся не смог исправить при наводящих вопросах преподавателя или ответ отсутствует