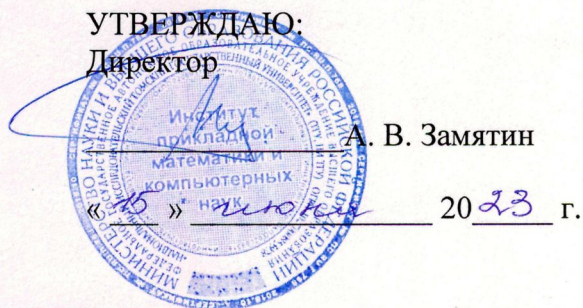


Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ:
Директор



А. В. Замятин

20 23 г.

Рабочая программа дисциплины

Нейронные сети

по направлению подготовки

01.04.02 Прикладная математика и информатика

Направленность (профиль) подготовки :

Интеллектуальный анализ больших данных

Форма обучения

Очная

Квалификация

Магистр

Год приема

2023

Код дисциплины в учебном плане: Б1.В.ДВ.02.01.02

СОГЛАСОВАНО:

Руководитель ОП

А.В. Замятин

Председатель УМК

С.П. Сущенко

Томск – 2023

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

– ПК-5 – способность управлять получением, хранением, передачей, обработкой больших данных;

– УК-1 – способность осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИУК-1.3 Предлагает и обосновывает стратегию действий с учетом ограничений, рисков и возможных последствий.

ИУК-1.2 Осуществляет поиск, отбор и систематизацию информации для определения альтернативных вариантов стратегических решений в проблемной ситуации.

ИУК-1.1 Выявляет проблемную ситуацию, на основе системного подхода осуществляет её многофакторный анализ и диагностику.

ИПК-5.1 Осуществляет мониторинг и оценку производительности обработки больших данных.

ИПК-5.2 Использует методы и инструменты получения, хранения, передачи, обработки больших данных.

ИПК-5.3 Разрабатывает предложения по повышению производительности обработки больших данных.

2. Задачи освоения дисциплины

– Освоить аппарат создания различных топологий искусственных нейронных сетей, а также технологию разработки программ, использующих основные этапы работы искусственных нейронных сетей (создание структур для представления нейросетей, создание и оценка модели сетей, обучение их).

– Научиться применять понятийный аппарат нейронных сетей для решения практических задач профессиональной деятельности.

3. Место дисциплины в структуре образовательной программы

Дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений, предлагается обучающимся на выбор. Дисциплина входит в Прикладной модуль.

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Третий семестр, экзамен

5. Входные требования для освоения дисциплины

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: Интеллектуальный анализ данных.

6. Язык реализации

Русский

7. Объем дисциплины

Общая трудоемкость дисциплины составляет 3 з.е., 108 часов, из которых:

-лекции: 16 ч.

-лабораторные: 32 ч.

в том числе практическая подготовка: 0 ч.

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Тема 1. Введение в нейронные сети

Исторические тенденции в нейронных сетях. Нейронные сети: разные названия и переменчивая фортуна. Увеличение размера набора данных. Увеличение размера моделей. Повышение точности и сложности и расширение круга задач.

Тема 2. Основы машинного обучения для нейронных сетей.

Алгоритмы обучения. Задача Т. Мера качества Р. Опыт Е. Пример: линейная регрессия. Емкость, переобучение и недообучение. Теорема об отсутствии бесплатных завтраков. Регуляризация. Гиперпараметры и контрольные наборы. Перекрестная проверка. Оценки, смещение и дисперсия. Точечное оценивание. Смещение. Дисперсия и стандартная ошибка. Поиск компромисса между смещением и дисперсией для минимизации среднеквадратической ошибки. Состоятельность. Оценка максимального правдоподобия. Условное логарифмическое правдоподобие и среднеквадратическая ошибка. Свойства максимального правдоподобия. Байесовская статистика. Оценка апостериорного максимума (MAP). Алгоритмы обучения с учителем. Вероятностное обучение с учителем. Метод опорных векторов. Другие простые алгоритмы обучения с учителем. Алгоритмы обучения без учителя. Метод главных компонент. Кластеризация методом k средних. Стохастический градиентный спуск. Построение алгоритма машинного обучения. Проблемы, требующие глубокого обучения. Проклятие размерности. Регуляризация для достижения локального постоянства и гладкости. Обучение многообразий.

Тема 3. Нейронные сети прямого распространения.

Пример: обучение XOR. Обучение градиентными методами. Функции стоимости. Выходные блоки. Скрытые блоки. Блоки линейной ректификации и их обобщения. Логистическая сигмоида и гиперболический тангенс. Другие скрытые блоки. Проектирование архитектуры. Свойства универсальной аппроксимации и глубина. Другие архитектурные подходы. Обратное распространение и другие алгоритмы дифференцирования. Графы вычислений. Правило дифференцирования сложной функции. Рекурсивное применение правила дифференцирования сложной функции для получения алгоритма обратного распространения. Вычисление обратного распространения в полносвязном МСП. Символьно-символьные производные. Общий алгоритм обратного распространения. Пример: применение обратного распространения к обучению МСП. Осложнения. Дифференцирование за пределами сообщества глубокого обучения. Производные высшего порядка. Исторические замечания.

Тема 4. Регуляризация для нейронных сетей.

Штрафы по норме параметров. Регуляризация параметров по норме L2. L1-регуляризация. Штраф по норме как оптимизация с ограничениями. Регуляризация и недоопределенные задачи. Пополнение набора данных. Робастность относительно шума. Привнесение шума в выходные метки. Обучение с частичным привлечением учителя. Многозадачное обучение. Ранняя остановка. Связывание и разделение параметров. Сверточные нейронные сети. Разреженные представления. Баггинг и другие ансамблевые методы. Прореживание. Состязательное обучение. Тангенциальное расстояние, алгоритм распространения по касательной и классификатор по касательной к многообразию.

Тема 5. Оптимизация в обучении глубоких моделей

Чем обучение отличается от чистой оптимизации. Минимизация эмпирического риска. Суррогатные функции потерь и ранняя остановка. Пакетные и мини-пакетные

алгоритмы. Проблемы оптимизации нейронных сетей. Плохая обусловленность. Локальные минимумы. Плато, седловые точки и другие плоские участки. Утесы и резко растущие градиенты. Долгосрочные зависимости. Неточные градиенты. Плохое соответствие между локальной и глобальной структурами. Теоретические пределы оптимизации. Основные алгоритмы. Стохастический градиентный спуск. Импульсный метод. Метод Нестерова. Стратегии инициализации параметров. Алгоритмы с адаптивной скоростью обучения. AdaGrad. RMSProp. Adam. Выбор правильного алгоритма оптимизации. Приближенные методы второго порядка. Метод Ньютона. Метод сопряженных градиентов. Алгоритм BFGS. Стратегии оптимизации и метаалгоритмы. Пакетная нормировка. Покоординатный спуск. Усреднение Поляка. Предобучение с учителем. Проектирование моделей с учетом простоты оптимизации. Методы продолжения и обучение по плану.

Тема 6. Сверточные сети.

Операция свертки. Мотивация. Пулинг. Свертка и пулинг как бесконечно сильное априорное распределение. Варианты базовой функции свертки. Структурированный выход. Типы данных. Эффективные алгоритмы свертки. Случайные признаки и признаки, обученные без учителя. Нейробиологические основания сверточных сетей. Сверточные сети и история глубокого обучения.

Тема 7. Моделирование последовательностей: рекуррентные и рекурсивные сети.

Развертка графа вычислений. Рекуррентные нейронные сети. Форсирование учителя и сети с рекурсией на выходе. Вычисление градиента в рекуррентной нейронной сети. Рекуррентные сети как ориентированные графические модели. Моделирование контекстно-обусловленных последовательностей с помощью РНС. Двухнаправленные РНС. Архитектуры кодировщик-декодер или последовательность в последовательность. Глубокие рекуррентные сети. Рекурсивные нейронные сети. Проблема долгосрочных зависимостей. Нейронные эхо-сети. Блоки с утечками и другие стратегии нескольких временных масштабов. Добавление прямых связей сквозь время. Блоки с утечкой и спектр разных временных масштабов. Удаление связей. Долгая краткосрочная память и другие вентильные РНС. Долгая краткосрочная память. Другие вентильные РНС. Оптимизация в контексте долгосрочных зависимостей. Отсечение градиентов. Регуляризация с целью подталкивания информационного потока. Явная память.

Тема 8. Практическая методология.

Показатели качества. Выбор базовой модели по умолчанию. Надо ли собирать дополнительные данные? Выбор гиперпараметров. Ручная настройка гиперпараметров. Алгоритмы автоматической оптимизации гиперпараметров. Поиск на сетке. Случайный поиск. Оптимизация гиперпараметров на основе модели. Стратегии отладки. Пример: распознавание нескольких цифр.

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине проводится путем контроля посещаемости, выполнения лабораторных работ, тестов по лекционному материалу, выполнения домашних заданий и фиксируется в форме контрольной точки не менее одного раза в семестр.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Экзамен в третьем семестре проводится в письменной форме по билетам. Экзаменационный билет состоит из трех вопросов. Продолжительность экзамена 1,5 часа.

Первый и второй вопрос – теоретические вопросы. Третий вопрос – практическая задача. Ответ на третий вопрос предполагает написание короткой программы по изученным алгоритмам.

Примерный перечень теоретических вопросов:

1. Проектирование нейросетевых архитектур. Место нейронных сетей в ИИ. Свойства нейронных сетей.
2. Проектирование нейросетевых архитектур. Модель нейрона. Функции активации. Архитектура нейронных сетей.
3. Процессы обучения. Обучение с учителем. Обучение без учителя. Обучение с подкреплением.
4. Построение архитектуры нейронной сети и ее обучение. Метрики качества. Функции потерь и оптимизаторы обучения. Анализ качества обучения нейронной сети.
5. Модель искусственной нейронной сети. Однослойный персептрон.
6. Проектирование нейросетевых архитектур. Бинарный классификатор.
7. Проектирование нейросетевых архитектур. Многоклассовый классификатор.
8. Проектирование нейросетевых архитектур. Нейросетевой регрессор.
9. Модель искусственной нейронной сети. Модель искусственного нейрона.
10. Градиентный спуск. Метод наименьших квадратов. Стохастический градиентный спуск SGD.
11. Модель полносвязной сети с одним скрытым слоем. Распространение сигнала на слоях сети (входной, скрытый, выходной). Функция ошибки. Веса слоев сети (входной, скрытый, выходной).
12. Метод градиентного спуска. Алгоритм метода градиентного спуска. Цепное правило. Обратное распространение ошибки.
13. Направления модификации алгоритмов градиентного спуска. Градиентный спуск с инерцией SGDmomentum.
14. Направления модификации алгоритмов градиентного спуска. Ускоренный градиентный спуск Нестерова NAG.
15. Направления модификации алгоритмов градиентного спуска. Адаптивный градиентный спуск AdaGrad.
16. Направления модификации алгоритмов градиентного спуска. Метод адаптивного скользящего среднего RMSprop.
17. Направления модификации алгоритмов градиентного спуска. Метод адаптивного шага обучения Adadelta.
18. Направления модификации алгоритмов градиентного спуска. Метод адаптивной инерции Adam.
19. Библиотеки для обучения нейронной сети: TensorFlow, Theano, PyTorch, CNTK, Caffe, Keras. Объекты, функции и параметры объектов нейронных сетей в TensorFlow. Библиотека pandas.
20. Построение полносвязной нейронной сети. Полносвязная нейронная сеть прямого распространения. Пример полносвязной (FullyConnected) нейронной сети.
21. Рекуррентные архитектуры нейронных сетей. Сеть Хопфилда. Архитектура сети Хопфилда. Обучение сети.
22. Особенности процесса обучения сети Хопфилда. Применение обученной сети. Работа сети в режиме фильтрации (восстановление повреждённых образов). Режимы работы сети Хопфилда. Пример восстановления повреждённого изображения. Устойчивость сети в процессе работы. Ассоциативная память.
23. Рекуррентные архитектуры нейронных сетей. Сеть Хопфилда. Задачи минимизации. Решение задачи о коммивояжёре. Ограничения сети.
24. Рекуррентные архитектуры нейронных сетей. Нейронная сеть Хэмминга. Архитектура. Обучение сети. Работа сети. Примеры.
25. Рекуррентные сети на базе персептрона. Персептронная сеть с обратной связью.

26. Алгоритм обучения сети RMLP. Схема включения сети RMLP при решении задачи идентификации.
27. Рекуррентные сети Эльмана. Алгоритм обучения сети Эльмана. Рекуррентные сети RTRN.
28. Сверточные нейронные сети для классификации визуальных образов. Зрительная кора головного мозга. Свертки и сверточные сети.
29. Сверточные нейронные сети для классификации визуальных образов. Сверточные нейронные сети с использованием библиотеки TensorFlow.

Примеры практических задач:

1. Построение нейросетевого регрессора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Лесные пожары (<https://archive.ics.uci.edu/ml/datasets/Forest+Fires>), предсказываемое значение – площадь пожара (Area);
2. Построение нейросетевого регрессора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Качество вина (<https://archive.ics.uci.edu/ml/datasets/Wine+Quality>) предсказываемое значение – качество (Quality), для датасета с красным вином, winequality-red.csv;
3. Построение нейросетевого регрессора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Качество вина (<https://archive.ics.uci.edu/ml/datasets/Wine+Quality>) предсказываемое значение – качество (Quality), для датасета с белым вином, winequality-white.csv;
4. Построение нейросетевого регрессора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Аренда велосипедов (<https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset>), предсказываемое значение – количество аренд велосипедов в сутки (Area), датасет day.csv;
5. Построение нейросетевого регрессора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Аренда велосипедов (<https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset>), предсказываемое значение – количество аренд велосипедов в час (Area), датасет hour.csv;
6. Построение бинарного классификатора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Кот и Собака. (<https://www.microsoft.com/en-us/download/details.aspx?id=54765>)
7. Построение бинарного классификатора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Лошади и Люди. (<http://laurencemoroney.com/horses-or-humans-dataset>)
8. Построение бинарного классификатора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – лошади/зебры. (https://www.tensorflow.org/datasets/catalog/cycle_gan)
9. Построение бинарного классификатора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Яблоко и Апельсин. (https://www.tensorflow.org/datasets/catalog/cycle_gan)
10. Построение бинарного классификатора. Написать программу на Python, которая обучает нейросетевой регрессор, с помощью библиотек scikit-learn и keras (опционально PyTorch). Датасет – Зима и Лето (https://www.tensorflow.org/datasets/catalog/cycle_gan)

11. Исследование рекуррентной нейросети (прогноз температуры). Написать программу на Python, которая обучает регрессионные модели, предсказывающие температуру на завтрашний день в одном из городов и построенных на наборе данных с помощью трёх алгоритмов: простая рекуррентная нейронная сеть, LSTM (или GRU), и стек LSTM (или стек GRU). Датасет – Выборка погода Сочи (2010-2019 годы);

12. Исследование рекуррентной нейросети (прогноз температуры). Написать программу на Python, которая обучает регрессионные модели, предсказывающие температуру на завтрашний день в одном из городов и построенных на наборе данных с помощью трёх алгоритмов: простая рекуррентная нейронная сеть, LSTM (или GRU), и стек LSTM (или стек GRU). Датасет – Выборка погода Хабаровск (2010-2019 годы);

13. Исследование рекуррентной нейросети (прогноз температуры). Написать программу на Python, которая обучает регрессионные модели, предсказывающие температуру на завтрашний день в одном из городов и построенных на наборе данных с помощью трёх алгоритмов: простая рекуррентная нейронная сеть, LSTM (или GRU), и стек LSTM (или стек GRU). Датасет – Выборка погода Калининград (2010-2019 годы);

14. Исследование рекуррентной нейросети (прогноз температуры). Написать программу на Python, которая обучает регрессионные модели, предсказывающие температуру на завтрашний день в одном из городов и построенных на наборе данных с помощью трёх алгоритмов: простая рекуррентная нейронная сеть, LSTM (или GRU), и стек LSTM (или стек GRU). Датасет – Выборка погода Якутск (2010-2019 годы);

15. Исследование рекуррентной нейросети (прогноз температуры). Написать программу на Python, которая обучает регрессионные модели, предсказывающие температуру на завтрашний день в одном из городов и построенных на наборе данных с помощью трёх алгоритмов: простая рекуррентная нейронная сеть, LSTM (или GRU), и стек LSTM (или стек GRU). Датасет – Выборка погода Волгоград (2010-2019 годы).

11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle» - <https://moodle.tsu.ru/course/view.php?id=9649>

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

в) План практических занятий по дисциплине.

г) Методические указания по проведению лабораторных работ.

д) Методические указания по организации самостоятельной работы студентов. («разбор полётов» по результатам выполнения лабораторных работ).

12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

1. Джоэл Грас. Data Science: Наука о данных с нуля. 2-е издание. ISBN 978-5-9775-6731-2, СПб: БХВ-Петербург, 2021

2. Себастьян Рашка, Вахид Мирджалили. Python и машинное обучение. ISBN 978-5-907203-57-0, М.: Диалектика, 2020

3. Ameet V. Joshi. Machine Learning and Artificial Intelligence. ISBN 978-3-030-26621-9. Springer Nature Switzerland AG, 2020

4. Denis Rothman. Artificial Intelligence by Example. Second Edition. ISBN 978-1-83921-153-9. Packt Publishing, 2020

5. Stuart Russel, Peter Norvig. Artificial Intelligence. A Modern Approach. 4th Edition. ISBN: 978-0-13-461099-3. Hoboken: Pearson, 2021

6. Ян Гудфеллоу, Йошуа Бенджио, Аарон Курвилль. Глубокое обучение. Второе цветное издание, исправленное. ISBN 978-5-97060-618-6. М.: ДМК Пресс, 2018

7. Roman Shirkin. Artificial Intelligence. The Complete Beginners' Guide to Artificial Intelligence. ISBN: 9798609154415. Amazon KDP Printing and Publishing, 2020

8. Франсуа Шолле Глубокое обучение на Python. ISBN 978-5-4461-0770-4. СПб: Питер, 2018

б) дополнительная литература:

1. Эндрю Гласснер. Глубокое обучение без математики. Том 1. Основы. ISBN 978-5-97060-701-5. М.: ДМК Пресс, 2020

2. Эндрю Гласснер. Глубокое обучение без математики. Том 2. Практика ISBN 978-5-97060-767-1. М.: ДМК Пресс, 2020

в) ресурсы сети Интернет:

www.MachineLearning.ru — профессиональный вики-ресурс, посвященный машинному обучению и интеллектуальному анализу данных

ММРО — Математические методы распознавания образов

Константин Воронцов. Курс «машинное обучение» школы анализа данных компании Яндекс.

Игорь Кураленок. Курс «машинное обучение» Лекториум.

Роман Шамин. Курс «Машинное обучение и искусственный интеллект в математике и приложениях». НОЦ Математического института им. В. А. Стеклова РАН

13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

- Для приобретения практических навыков – свободно распространяемые среды с открытым кодом Python (<https://www.python.org/>) и RStudio (<https://www.rstudio.com/>);

- Для проектно-групповой и дистанционной работы – российское ПО Mind (<https://mind.com/>).

– публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).

б) информационные справочные системы:

– Электронный каталог Научной библиотеки ТГУ – <http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>

– Электронная библиотека (репозиторий) ТГУ – <http://vital.lib.tsu.ru/vital/access/manager/Index>

14. Материально-техническое обеспечение

При освоении дисциплины используются компьютерные классы ИПМКН ТГУ с доступом к ресурсам Научной библиотеки ТГУ, в том числе отечественным и зарубежным периодическим изданиям и Интернета.

15. Информация о разработчиках

Бакланова Ольга Евгеньевна – кандидат физико-математических наук, доцент, доцент кафедры теоретических основ информатики ТГУ.