

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Романчук Иван Сергеевич
Должность: Ректор
Дата подписания: 30.01.2025 11:22:30
Уникальный программный ключ:
6319edc2b582ffda443f01d5779368d0957ac34f5cd074d81181530452479

Приложение к рабочей
программе дисциплины

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Наименование дисциплины	<i>Глубинное обучение</i>
Направление подготовки / Специальность	<i>38.03.01 Экономика</i>
Направленность (профиль) / Специализация	<i>Экономика и анализ данных</i> <i>ОП ВО</i>
Форма обучения	<i>очная</i>

Разработчик Капелюк С.Д., профессор научно-учебной лаборатории исследований рынка труда

1. Темы дисциплины для самостоятельного освоения обучающимися
Отсутствуют.

2. План самостоятельной работы:

№ п/п	Учебные встречи	Виды самостоятельной работы	Форма отчетности / контроля	Количество баллов	Рекомендуемый бюджет времени на выполнение (ак.ч.)
1.	Введение в нейронные сети	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	8
2.	Работа с изображениями и сверточные архитектуры	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	8
3.	Современные сверточные архитектуры и их использование	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	23
4.	Работа с последовательностями и рекуррентные архитектуры	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	23
5.	Механизм внимания и Трансформер	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	23
6.	Нейронные сети в реальных условиях	1. Подготовка к тестированию	1. Тестирование	-	6
		2. Выполнение задания для самостоятельной работы	2. Выполнение работы	3	22
7.	Подготовка к	Изучение материалов	-	-	35

	экзамену	по дисциплине по вопросам к экзамену			
	Итого			18	178

3. Требования и рекомендации по выполнению самостоятельных работ обучающихся, критерии оценивания

Вид: Подготовка к тестированию

Краткая характеристика: вид проверки знаний и умений учащихся, который направлен на выявление степени усвоения изученного материала. Оно содержит обобщенный материал по основным изученным темам, требует от учащихся хорошей ориентировки в явлениях и фактах.

Рекомендации для подготовки:

- просмотр видеолекций;
- чтение основной и дополнительной литературы;
- повтор изученного на лекционных и практических занятиях.

Пример теста:

1. Какая основная цель использования двунаправленной рекуррентной нейронной сети (Bidirectional RNN)?
 - а) Для обработки последовательных данных как в прямом, так и в обратном направлении
 - б) Для параллельной обработки данных
 - в) Для последовательной обработки данных
 - г) Для иерархической обработки данных
2. Какие методы сжатия всегда применяются к уже обученной нейронной сети, то есть изменение обучения нейросети НЕ требуется? (Выберите все верные варианты ответов)
 - а) Дистилляция
 - б) Разреживание
 - в) Статическая квантизация
 - г) Динамическая квантизация
3. Какие способы используют для улучшения сходимости при обучении нейросетей? (Выберите ВСЕ правильные варианты ответов)
 - а) Периодическое сбрасывание весов в случайные значения
 - б) Использование мини-батча размера
 - в) Постепенное уменьшение длины шага
 - г) Продвинутое алгоритмы оптимизации, например SGD-Momentum и Adam
4. В каком случае бинарный дропаут (случайное отключение нейронов) можно считать способом получения ансамбля нейронных сетей?
 - а) Если на этапе выполнения предсказания не использовать дропаут и использовать веса, разделенные на один минус вероятность отключения нейронов (дропаута)
 - б) Если на этапе выполнения предсказания использовать дропаут и выполнить проход вперед несколько раз
 - в) Если на этапе выполнения предсказания использовать дропаут и выполнить проход вперед один раз
 - г) Если на этапе выполнения предсказания не использовать дропаут и использовать веса, разделенные на вероятность отключения нейронов (дропаута)
5. Какая цель у использования сверточных слоев в задачах классификации текста?
 - а) Извлечь признаки из текста
 - б) Понизить размерность представлений текстовых данных
 - в) Предсказать автора текста
 - г) Классифицировать текст по длине
6. Какие величины вычисляются на проходе назад?

- а) Значение финальной вершины вычислительного графа
 - б) Значения производных всех вершин по финальной вершине
 - в) Значения всех вершин вычислительного графа
 - г) Значения производных финальной вершины по остальным вершинам
7. Какими из перечисленных способов можно получить набор нейросетей для ансамблирования?
- а) Обучить несколько нейросетей различной архитектуры
 - б) Обучить нейросети одной архитектуры из нескольких различных начальных приближений
 - в) Использовать зигзагообразное расписание длины шага и сохранять веса нейронной сети в нескольких точках процесса обучения
 - г) Обучить нейросети для решения разных задач на разных данных
8. В чем основной принцип дистилляции?
- а) Нейросеть-"ученик" учится предсказывать такие же вероятности классов, как нейросети-"учитель"
 - б) Нейросеть-"ученик" учится предсказывать такой же класс для каждого объекта, как нейросети-"учитель"
 - в) Нейросеть-"ученик" учится повторять часть весов нейросети-"учителя"
 - г) Нейросеть-"ученик" учится повторять промежуточные активации нейросети-"учителя"
9. Вы хотите выполнить разреживание своей нейронной сети, сохранив качество предсказаний. Какие действия для этого стоит предпринять?
- а) Использовать Lasso- и/или Group Lasso-регуляризацию при обучении
 - б) Обнулять веса по порогу или исходя из желаемого уровня разреженности
 - в) Дообучать нейросеть после обнуления весов
 - г) Обнулять веса по одному с помощью жадного алгоритма
10. Какие из перечисленных величин обязательно нужно квантизировать, чтобы не только сжать обученную полносвязную нейросеть, но и ускорить ее предсказания? Если ускорить предсказание можно и без квантизации величины, не выделяйте ее.
- а) параметры линейных слоев
 - б) входы линейных слоев
 - в) выходы линейных слоев
 - г) обучающие данные
11. Дано одноканальное изображение размером 32 x 32 пикселей. К нему применили операцию свертки с размером ядра 5 и шагом свертки 3. Какой размер изображения после свертки?
- а) 28 x 28 пикселей
 - б) 18 x 18 пикселей
 - в) 25 x 25 пикселей
 - г) 10 x 10 пикселей
12. Чему равно поле восприятия после применения двух операций свертки с размером ядра 5 x 5 и шагом 1?
- а) 24 x 24 пикселей
 - б) 5 x 5 пикселей
 - в) 10 x 10 пикселей
 - г) 9 x 9 пикселей
13. Пусть даны изображения А и В. Мы пропускаем их через сверточную сеть. Как измеряется близость этих изображений по содержанию?
- а) Суммой квадратов разностей значений пикселей изображений после некоторых сверточных слоев
 - б) По квадрату разности выхода нейронной сети для изображений
 - в) Сравнивая попарные произведения значений пикселей в разных каналах изображений
 - г) Сравнивая суммы значений всех пикселей изображений

14. Пусть даны изображения А и В. Мы пропускаем их через сверточную сеть. Как измеряется близость этих изображений по стилю?
- Сравнивая попарные произведения значений пикселей в разных каналах изображений
 - Суммой квадратов разностей значений пикселей изображений после некоторых сверточных слоев
 - По квадрату разности выхода нейронной сети для изображений
 - Сравнивая суммы значений всех пикселей изображений
15. Как отличается контекст у моделей BERT и GPT?
- Никак
 - У GPT есть полный контекст, у BERT только левый
 - У BERT есть полный контекст, у GPT только левый
 - У GPT есть только левый контекст, у BERT только правый
16. Как работает механизм внимания в рекуррентной нейронной сети?
- Применяя полносвязный слой
 - Применяя рекуррентный слой
 - Путем вычисления взвешенной суммы скрытых состояний
 - Применяя сверточный слой
17. Что такое модель трансформер в глубинном обучении?
- Тип рекуррентной нейронной сети
 - Тип сверточной нейронной сети
 - Тип нейронной сети, использующий механизмы self-attention
 - Тип нейронной сети, использующий сверточные механизмы внутреннего внимания
18. Как модель трансформер обрабатывает последовательные входные данные?
- Применяя сверточные слои
 - Применяя рекуррентные слои
 - Применяя механизмы само-внимания
 - Применяя комбинацию сверточных и рекуррентных слоев
19. В чем разница между энкодером и декодером в модели трансформер?
- энкодер генерирует выходные данные, а декодер обрабатывает входные данные
 - энкодер и декодер имеют одинаковую функцию
 - энкодер обрабатывает входные данные, а декодер формирует выходные данные
 - энкодер и декодер взаимозаменяемы
20. В чем основное различие между моделями BERT и GPT в глубинном обучении?
- BERT - декодер от трансформера, а GPT - энкодер от трансформера
 - BERT - энкодер от трансформера, а GPT - рекуррентная нейронная сеть
 - BERT - энкодер от трансформера, а GPT - декодер от трансформера
 - BERT - энкодер от трансформера, а GPT – трансформер

Вид: Выполнение задания для самостоятельной работы

Краткая характеристика заданий: Задания курса направлены на развитие у студентов практических навыков в области глубокого обучения, включая работу с нейронными сетями, обработку изображений, рекуррентные и трансформерные архитектуры, а также работу с реальными наборами данных. Студенты будут осваивать как теоретические аспекты, так и практическую реализацию различных моделей и алгоритмов глубокого обучения, таких как нейронные сети, сверточные сети, механизмы внимания и трансформеры.

Рекомендации для подготовки: При выполнении заданий важно следовать точным указаниям задания и проводить тщательную проверку корректности реализованных решений. Рекомендуется изучить необходимые библиотеки и алгоритмы, такие как PyTorch и TensorFlow, а также уделить внимание основам работы с изображениями и текстами. Для успешного выполнения заданий стоит внимательно работать с параметрами моделей, оптимизацией и проверкой результатов на тестовых данных.

Пример задания для самостоятельной работы к встрече «Введение в нейронные сети»: Запишите функцию $y=(x+a)(x+b)(x+c)$ в виде вычислительного графа, где:

- $z1=x+a$
- $z2=x+b$
- $z3=x+c$

1. С помощью метода автоматического дифференцирования найдите производную y по $z2$.
2. Продолжите алгоритм автоматического дифференцирования и найдите производную y по x . Для этого вычислите производную y по $z1$, а затем искомую y по x .

Пример задания для самостоятельной работы к встрече «Работа с изображениями и сверточные архитектуры»:

Дано трехканальное изображение размером 64×64 пикселей (например, цветное изображение RGB). К нему последовательно применили три операции свертки с различными параметрами:

1. Первая свертка: размер ядра 7×7 , шаг свертки 2, padding (дополнение) 3.
2. Вторая свертка: размер ядра 5×5 , шаг свертки 1, padding 2.
3. Третья свертка: размер ядра 3×3 , шаг свертки 1, padding 1.

Необходимо определить:

1. Размер изображения после каждой свертки.
2. Чему равно поле восприятия после применения всех трех операций свертки?

Пример задания для самостоятельной работы к встрече «Современные сверточные архитектуры и их использование»:

Реализуйте алгоритм переноса стиля (neural style transfer) с использованием предобученной сверточной нейронной сети VGG19 для объединения содержания одного изображения со стилем другого изображения. Для этого:

1. Используйте два входных изображения - одно в качестве источника содержания (content image) и второе в качестве источника стиля (style image). В качестве начального изображения для оптимизации возьмите копию первого изображения. Измените размер всех изображений до 512×512 пикселей (или 128×128 , если GPU недоступен).
2. Извлеките признаки изображений с помощью предобученной сети VGG19, используя выходы определенных сверточных слоев. Для захвата содержания используйте слой 'conv_4', а для захвата стилистических особенностей - слои 'conv_1' через 'conv_5'. Реализуйте функцию потерь, состоящую из двух компонент: функции потерь содержания (content loss) как среднеквадратичного отклонения признаков промежуточного изображения от признаков изображения-источника содержания, и функции потерь стиля (style loss) как среднеквадратичного отклонения матриц Грама признаков промежуточного изображения от матриц Грама признаков изображения-источника стиля.
3. Оптимизируйте полученную функцию потерь по пикселям промежуточного изображения с помощью алгоритма L-BFGS в течение 300 итераций. При этом для функции потерь стиля используйте веса [1, 0.75, 0.2, 0.2, 0.2] для разных слоев и общий коэффициент $style_weight=500000$, а для функции потерь содержания - коэффициент $content_weight=1$.

В результате должно получиться изображение, сочетающее содержание первого изображения со стилистическими особенностями второго. Визуализируйте исходные изображения и результат.

Пример задания для самостоятельной работы к встрече «Работа с последовательностями и рекуррентные архитектуры»:

Создайте и обучите языковую модель с использованием вероятностного подхода и

рекуррентной нейронной сети (RNN).

1. Скачайте и загрузите данные с названиями динозавров из файла `dinos.txt`. Преобразуйте каждое название динозавра в список символов. Создайте биграммы для каждого названия динозавра и добавьте символы для обозначения начала и конца слова.
2. Используйте метод `padded_everygram_pipeline` из библиотеки `nltk` для создания данных и словаря для обучения модели.
3. Разделите данные на обучающую и тестовую выборки. Обучите вероятностную модель MLE с использованием биграмм на обучающей выборке. Вычислите и выведите перплексию модели на тестовой выборке.
4. Усложните модель до триграмм и повторите процесс обучения и вычисления перплексии.
5. Подготовьте данные для обучения, преобразовав названия динозавров в подходящий формат для RNN.
6. Создайте датасет `DinosDataset`, который будет возвращать `one-hot` векторы для каждого символа в названии динозавра.
7. Определите модель RNN с заданным размером скрытого слоя и функцией активации. Опишите функцию потерь и алгоритм оптимизации для обучения модели.
8. Обучите модель на подготовленных данных и сохраните результаты обучения.

Пример задания для самостоятельной работы к встрече «Механизм внимания и Трансформер»:

Реализуйте классификатор настроения текстов (`sentiment analysis`) на основе архитектуры `Transformer` для набора данных `IMDB`. Необходимо создать модель, которая будет определять положительные и отрицательные отзывы. Для этого:

1. Создайте класс `Vocabulary` для обработки текстов, который должен выполнять следующие функции:
 - удалять `URL`-адреса из текста,
 - оставлять только буквенные символы,
 - приводить текст к нижнему регистру,
 - учитывать только слова, встречающиеся не менее 10 раз,
 - добавлять специальные токены `<unk>` для неизвестных слов и `<pad>` для выравнивания последовательностей.
2. Выполните предобработку текстов из набора данных `IMDB`:
 - удалите стоп-слова,
 - выполните лемматизацию,
 - токенизируйте тексты,
 - ограничьте длину последовательностей до 350 токенов,
 - создайте батчи размером 128 элементов.
3. Реализуйте архитектуру `Transformer` для классификации текстов, которая должна включать:
 - позиционное кодирование,
 - механизм многоголового внимания с 4 головами,
 - 4 слоя трансформера,
 - размерность эмбедингов 64,
 - полносвязный слой размерности 64,
 - `dropout` 0.2.
4. Обучите модель в течение 25 эпох с использованием оптимизатора `AdamW`, функции потерь `BCEWithLogitsLoss` и `scheduler'a` в виде полиномиального затухания с разогревом. После обучения визуализируйте карты внимания (`attention maps`) для анализа работы модели. Итоговая модель должна достичь высокой точности классификации на тестовом наборе данных `IMDB`.

Пример задания для самостоятельной работы к встрече «Нейронные сети в реальных условиях»:

1. Используйте набор данных MNIST для обучения сверточной нейронной сети (CNN) распознаванию рукописных цифр. Данные следует разделить на обучающую (60000 изображений) и тестовую (10000 изображений) выборки. Выполните предварительную обработку данных с помощью StandardScaler и преобразуйте их в формат, подходящий для работы с изображениями размером 28x28 пикселей.
2. Реализуйте CNN с двумя сверточными слоями (Conv2d с 10 и 20 фильтрами соответственно, размер ядра 5x5) и одним полносвязным слоем. После каждого сверточного слоя используйте функцию активации ReLU и максимальное объединение (max pooling).
3. Создайте собственную функцию потерь, которая объединяет кросс-энтропию с L1-регуляризацией (коэффициент регуляризации 0.0003) для предотвращения переобучения. Обучите модель в течение 15 эпох с размером мини-батча 64 и скоростью обучения 0.0001, используя оптимизатор Adam.
4. После обучения примените технику прореживания (pruning) к весам сети, обнулив 50% наименьших по модулю весов в каждом слое. Проведите дополнительное обучение (fine-tuning) прореженной сети в течение 5 эпох. Оцените точность модели на тестовой выборке до и после прореживания.
5. Сохраните веса обученной модели в файл и продемонстрируйте, как представить веса полносвязного слоя в разреженном формате с помощью библиотеки scipy.sparse.

4. Рекомендации по самоподготовке к промежуточной аттестации по дисциплине

Вопросы для самопроверки к экзамену

1. Нейронные сети: определение и основные понятия.
2. Общая идея графа вычислений.
3. Автоматическое дифференцирование.
4. Полносвязные нейронные сети.
5. Функции потерь и алгоритм обратного распространения ошибки.
6. Обучение нейронных сетей.
7. Теорема об универсальном аппроксиматоре.
8. Введение в PyTorch.
9. Обучение нейросетей в PyTorch.
10. Свёртки изображений.
11. Объединение пикселей.
12. Свёрточные нейронные сети.
13. Инициализация весов сети.
14. Переобучение сетей.
15. L1 и L2 регуляризации.
16. Отключение нейронов.
17. Аргументация данных.
18. Пакетная нормализация.
19. Методы регуляризации.
20. Современные архитектуры для работы с изображениями.
21. Visual Geometry Group (VGG).
22. GoogLeNet.
23. ResNet.
24. MobileNet.
25. EfficientNet.
26. Перенос обучения (transfer learning).
27. Перенос стиля изображения. Ускорение переноса стиля.
28. Основы рекуррентных нейронных сетей.

29. Работа с последовательностями и рекуррентные архитектуры: определение и основные понятия.
30. Векторные представления слов.
31. Токенизация на части слов.
32. Затухающие и взрывающиеся градиенты.
33. Варианты рекуррентных архитектур.
34. Классификация текста с помощью сверточных нейронных сетей.
35. Классификация текста с помощью рекуррентных нейронных сетей.
36. Механизм внимания: основные понятия.
37. Модель Трансформер: структура и применение.
38. Слои с механизмом внимания.
39. Контекстозависимые эмбединги слов: BERT, GPT.
40. Классификация текста с помощью Трансформера.
41. Нейронные сети на практике.
42. Ансамблирование нейронных сетей.
43. Оценка неопределенности нейронных сетей.
44. Атаки на нейронные сети.
45. Сжатие нейронных сетей.
46. Дистилляция.
47. Разреживание нейронных сетей.
48. Квантизация нейронных сетей.