

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

Институт прикладной математики и компьютерных наук

УТВЕРЖДЕНО:
Директор
А. В. Замятин

Оценочные материалы по дисциплине

Глубинное обучение

по направлению подготовки

01.04.02 Прикладная математика и информатика

Направленность (профиль) подготовки:
Информационная безопасность

Форма обучения
Очная

Квалификация
Магистр

Год приема
2024

СОГЛАСОВАНО:
Руководитель ОП
А.Ю. Матророва

Председатель УМК
С.П. Сущенко

Томск – 2024

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

Целью освоения дисциплины является формирование следующих компетенций:

ПК-5 Способен управлять получением, хранением, передачей, обработкой больших данных.

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

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

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

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

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

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

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- тесты;
- контрольная работа;
- лабораторные работы.

Пример теста (ИУК-1.1, ИУК-1.2)

1. Какая нейросетевая модель из перечисленных в лучшей степени подходит для прогнозирования временных последовательностей?
 - а) Single-Layer Perceptron
 - б) CNN
 - в) LSTM
 - г) Multi-layer Perceptron
2. Почему модели на сверточных нейронных сетях показывают наилучшие показатели по классификации объектов на изображениях по сравнению с другими моделями?
 - а) Они в высокой степени оптимизированы для обработки векторов с числовыми, а не категориальными признаками
 - б) Они обладают широким набором инструментов преобразования признаков пространства, которые может варьировать разработчик в модели
 - в) Они учитывают корреляцию смежных компонент вектора
 - г) Они используют существенно большее число настраиваемых параметров, по сравнению с другими моделями
3. Каким главным недостатком обладает рекуррентная нейронная сеть?
 - а) Длительная процедура обучения
 - б) Невозможность обучения на категориальных данных
 - в) Сложность запоминания длительных последовательностей
 - г) Использование существенных вычислительных ресурсов
4. Какие меры не приводят к уменьшению переобучения нейросетевой модели?

- а) Установка штрафов за большие значения весов нейронов сети
 - б) Увеличение количества слоев сети
 - в) Добавление шума в выборку
 - г) Уменьшение количества нейронов сети
5. От чего в большей степени зависит успешное решение задачи классификации однослойным персептроном?
- а) от размера выборки
 - б) от размерности признакового пространства
 - в) соотношения разделения выборки на обучающую и тестовую
 - г) от распределения объектов в пространстве признаков

Ключи: 1 в), 2 в), 3 в), 4 б), 5 г).

Критерии оценивания: тест считается пройденным, если обучающий ответил правильно как минимум на половину вопросов.

Темы лабораторных работ по дисциплине (ИПК-5.1, ИПК-5.2, ИПК-5.3, ИУК-1.1, ИУК-1.2, ИУК-1.3)

Лабораторная работа №1. «Реализация глубоких полносвязных нейросетевых моделей»

Цель работы – написать программу на языках Python и R, выполняющую построение и обучение нейронных сетей прямого распространения, решающих задачи классификации и регрессии (выборки получены от преподавателя), требуется подобрать безизбыточную архитектуру сети, работающей с допустимым уровнем ошибки и визуализировать процесс обучения моделей. Результаты работы привести в отчете.

Лабораторная работа №2 «Разработка и развертывание приложений с глубокими нейросетевыми моделями»

Цель работы – разработать веб-приложение на языке Python, использующее обученную нейронную сеть, полученную в предыдущей работе, предоставляющее пользователю сервис для ввода исследуемых данных и вывода результата классификации данных. Результаты работы привести в отчете.

Лабораторная работа №3 «Инструменты повышения эффективности проектирования и обучения глубоких нейронных сетей»

Цель работы – исследовать на языках Python и R методы настройки параметров и гиперпараметров нейронных сетей с помощью разных оптимизаторов, перебора архитектур для решения задач многоклассовой, бинарной классификации, а также регрессии для выборок, предоставленных преподавателем. Результаты работы привести в отчете.

Лабораторная работа №4 «Исследование текстов глубокими нейросетевыми моделями»

Цель работы – написать программу на языке Python, выполняющую классификацию текстовых данных на основе библиотеки Keras с использованием инструментов NLTK: токенизации и лемматизации, векторного преобразования текста для выборки, полученной от преподавателя. Результаты работы привести в отчете.

Лабораторная работа №5 «Сегментация и классификация объектов на изображениях с помощью глубоких нейронных сетей»

Цель работы – написать программу на языке Python, использующую (на выбор) или библиотеку PyTorch или Keras, выполняющую две задачи: 1) классификацию изображений, 2) локализацию значимых областей на изображении, с применением

сверточных нейронных сетей. Выборки изображений получены от преподавателя. В работе следует использовать процедуру аугментации данных и провести тестирование с сетью, полученной путем переноса обучения. Результаты работы привести в отчете.

Лабораторная работа №6 «Реализация чат-бот систем с помощью инструментов глубокого обучения»

Цель работы – написать программу чат-бота на языке Python, выполняющую задачу взаимодействия человека с ботом, использующий нейросетевой классификатор высказываний по одной из тем, которую предложил преподаватель. Результаты работы привести в отчете.

Лабораторная работа №7 «Генерация изображений и временных последовательностей»

Цель работы – написать программы на языке Python, выполняющую построение и обучение модели генерации временных последовательностей и изображений (выборки изображений и временные последовательности, используемые для настройки, получены от преподавателя), требуется подобрать архитектуры GAN, использующихся для создания результата и оценки его качества, визуализировать процесс обучения моделей и привести метрики качества. Результаты работы привести в отчете.

Лабораторная работа №8 «Исследование процедуры дообучения нейронных сетей»

Цель работы – написать программу на языке Python, выполняющую построение и обучение модели классификатора (выборка изображений получена от преподавателя), а также привести эксперименты по дообучению нейронной сети путем добавления новых обучающих данных, в процессе функционирования модели. Показать на каких наборах данных модель сохраняет знания, полученные ранее, а на каких начинает терять память о зависимостях. Результаты работы привести в отчете.

Критерии оценивания лабораторных работ:

«Зачтено» – студент в целом хорошо разбирается в задаче, знает и использует методы решения практически самостоятельно или при небольшой подсказке преподавателя, отвечает на вопросы, возможно с негрубыми ошибками. Представляет работу на защите в целом хорошо, с несущественными замечаниями.

«Не зачтено» – студент слабо разбирается в задаче, не знает или знает плохо методы решения, не отвечает, либо отвечает, но с грубыми ошибками на вопросы преподавателя.

Контрольная работа (ИПК-5.1, ИПК-5.2, ИПК-5.3, ИУК-1.1, ИУК-1.2, ИУК-1.3)

Контрольная работа состоит из 2 теоретических вопросов и 3 задач.

Перечень теоретических вопросов:

1. На что влияет размер эмбединга в LLM?
2. Что такое позиционное кодирование?
3. Из каких блоков состоит трансформер?
4. Реализация механизма внимания.
5. Классы задач обработки последовательностей.
6. Вычисление градиента в рекуррентной нейронной сети.
7. Штрафы по норме параметров.
8. Ранняя остановка.
9. Разряженные представления.
10. Использование прореживания.
11. Пакетная нормализация.
12. Усреднение Поляка.
13. Рекурсивные нейронные сети.

14. Нейронные эхо-сети.
15. Репрезентативная способность в автокодировщиках.
16. Жадное послойное предобучение без учителя.
17. Ограниченные машины Больцмана.
18. Глубокие сети доверия.
19. Глубокие машины Больцмана.
20. Ориентированные порождающие сети.
21. Проблемы оптимизации нейронных сетей.
22. Алгоритмы оптимизации с адаптивной скоростью обучения.
23. Приближенные методы оптимизации второго порядка.
24. Алгоритм итерации нейронной ценности
25. Архитектура глубокой Q-сети
26. Настройка двойной глубокой Q-сети
27. Дуэльные архитектуры в ценностно-ориентированном глубоком обучении с подкреплением.
28. Методы приоритетного воспроизведения полезного опыта.
29. Параллельное обновление политики в A3C.
30. Надежное прогнозирование преимущества GAE.
31. Синхронное обновление политик в A2C.
32. Формирование функции ценности в VPG.
33. Максимизация ожидаемой выгоды и энтропии в «мягком актёре-критике».
34. Отличия методов градиента политик и «актёр-критик».
35. Генеративно-состязательная сеть Вассерштайна.
36. Особенности CycleGAN
37. Мультимодальные трансформеры
38. Диффузионные модели
39. Графовые нейронные сети.
40. Получение объяснений в глубоких моделях. Примеры задач:

Задача 1.

Построить регрессор, использующий слой RNN в качестве скрытого слоя для прогноза добычи жидкости и обводненности для каждой скважины на 120 суток вперед по предыдущим значениям признаков из выборки.

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество нейронов в слоях (ячеек), процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными. Сделать подбор количества предыдущих значений, используемых для прогнозной модели, для более точной оценки целевой переменной. Сделать эксперименты с прореживанием (Dropout).

Вычислить следующие метрики работы: R2, MSE, MAE для всех исследованных моделей.

Задача 2.

Построить нейросетевые модели – бинарный и многоклассовый классификатор, согласно Вашему варианту задачи.

Сначала построить бинарный классификатор для классов 0 и 1.

Набор данных: <https://www.kaggle.com/datasets/emmarex/plantdisease> Класс 0: Tomato, Класс 1: Potato, Класс 2: Pepper.

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество карт признаков, распределение слоёв, процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными для следующих задач.

Выполнить эксперименты с прореживанием (Dropout).

Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах.

Выполнить аугментацию данных и провести повторное обучение на новом расширенном наборе.

Сохранить лучшую архитектуру нейронной сети – бинарного классификатора.

Создать нейронную сеть – многоклассовый классификатор с использованием сверточной основы, обученной ранее на ImageNet (ResNet, VGG, Inception, Xception и т.д.). Выполнить дообучение модели в двух вариантах: заключительные полносвязные слои и fine-tuning. Сохранить лучшие модели. Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах из трех классов.

Задача 3.

Обучить нейросетевые модели BERT, стека LSTM, двунаправленную LSTM, предсказывающие авторство текстов по фрагменту текстов Некрасова Н.А., Лермонтова М.Ю., Тютчева Ф.И.,

Для каждого выбранного поэта взять не менее 50 фрагментов из произведений, каждый фрагмент содержит не менее 100 символов.

Получить токенизатор и обученную модель BERT с HuggingFace.

Выполнить загрузку и предварительную обработку данных из текстов. Разделить каждую выборку на обучающую, тестовую и валидационную.

Добавить к архитектуре BERT полносвязный (или два полносвязных) слой(я) и обучить полученную модель в следующих вариантах:

1) Заморозить веса в BERT и настроить добавленный(е) полносвязный(ые) слой(и). (Это – модель 1a)

2) Выполнить Fine-tuning полученной архитектуры. (Это – модель 1b)

Обучить вторую модель - стек LSTM (две последовательные ячейки LSTM, добавить Dropout) для классификации текстовых данных.

Обучить третью модель – двунаправленную ячейку LSTM для классификации текстовых данных.

Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах.

Критерии оценивания:

Результаты контрольной работы определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка «отлично» выставляется, если даны правильные ответы на все теоретические вопросы и все задачи решены без ошибок.

Оценка «хорошо» выставляется, если корректные ответы даны на большую часть вопросов, но были отмечены неуверенность в ответе и информация представлена фрагментарно, также задачи были решены правильно с небольшими замечаниями.

Оценка «удовлетворительно» выставляется, если даны правильные ответы на половину теоретические вопросы и одна из задач решена с негрубой ошибкой.

Оценка «неудовлетворительно» выставляется, если корректные ответов меньше половины и задачи были решены неправильно.

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

Экзаменационный билет состоит из трех частей.

Первая часть представляет собой тест из 5 вопросов, проверяющих ИУК-1.1, ИУК-1.2. Ответы на вопросы первой части даются путем выбора из списка предложенных.

Вторая часть содержит один вопрос, проверяющий ИПК-5.1, ИПК-5.3, ИУК-1.3. Ответ на вопрос второй части дается в развернутой форме.

Третья часть содержит 2 вопроса, проверяющих ИПК-5.2, ИУК-1.3 и оформленные в виде практических задач. Ответы на вопросы третьей части предполагают решение задач и краткую интерпретацию полученных результатов.

Перечень теоретических вопросов:

1. На что влияет размер эмбединга в LLM?
2. Что такое позиционное кодирование?
3. Из каких блоков состоит трансформер?
4. Реализация механизма внимания.
5. Классы задач обработки последовательностей.
6. Вычисление градиента в рекуррентной нейронной сети.
7. Штрафы по норме параметров.
8. Ранняя остановка.
9. Разряженные представления.
10. Использование прореживания.
11. Пакетная нормализация.
12. Усреднение Поляка.
13. Рекурсивные нейронные сети.
14. Нейронные эхо-сети.
15. Репрезентативная способность в автокодировщиках.
16. Жадное послойное предобучение без учителя.
17. Ограниченные машины Больцмана.
18. Глубокие сети доверия.
19. Глубокие машины Больцмана.
20. Ориентированные порождающие сети.
21. Проблемы оптимизации нейронных сетей.
22. Алгоритмы оптимизации с адаптивной скоростью обучения.
23. Приближенные методы оптимизации второго порядка.
24. Алгоритм итерации нейронной ценности
25. Архитектура глубокой Q-сети
26. Настройка двойной глубокой Q-сети
27. Дуэльные архитектуры в ценностно-ориентированном глубоком обучении с подкреплением.
28. Методы приоритетного воспроизведения полезного опыта.
29. Параллельное обновление политики в A3C.
30. Надежное прогнозирование преимущества GAE.
31. Синхронное обновление политик в A2C.
32. Формирование функции ценности в VPG.
33. Максимизация ожидаемой выгоды и энтропии в «мягком актёре-критике».
34. Отличия методов градиента политик и «актёр-критик».
35. Генеративно-состязательная сеть Вассерштайна.
36. Особенности CycleGAN

37. Мультимодальные трансформеры
38. Диффузионные модели
39. Графовые нейронные сети.
40. Получение объяснений в глубоких моделях.

Примеры задач:

Задача 1.

Построить регрессор, ячейку LSTM (или GRU) в скрытом слое, для прогноза добычи жидкости и обводненности для каждой скважины на 120 суток вперед по предыдущим значениям признаков из выборки.

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество нейронов в слоях (ячеек), процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными. Сделать подбор количества предыдущих значений, используемых для прогнозной модели, для более точной оценки целевой переменной. Сделать эксперименты с прореживанием (Dropout).

Вычислить следующие метрики работы: R2, MSE, MAE для всех исследованных моделей.

Задача 2.

Построить нейросетевые модели – бинарный и многоклассовый классификатор, согласно Вашему варианту задачи.

Сначала построить бинарный классификатор для классов 0 и 1.

Набор данных: <https://www.kaggle.com/datasets/fpeccia/weed-detection-in-soybean-crops>
Класс 0: soybean, Класс 1: soil, Класс 2: grass

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество карт признаков, распределение слоёв, процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными для следующих задач.

Выполнить эксперименты с прореживанием (Dropout).

Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах.

Выполнить аугментацию данных и провести повторное обучение на новом расширенном наборе.

Сохранить лучшую архитектуру нейронной сети – бинарного классификатора.

Создать нейронную сеть – многоклассовый классификатор с использованием сверточной основы, обученной ранее на ImageNet (ResNet, VGG, Inception, Xception и т.д.). Выполнить дообучение модели в двух вариантах: заключительные полносвязные слои и fine-tuning. Сохранить лучшие модели. Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах из трех классов.

Задача 3.

Обучить нейросетевые модели BERT, стека LSTM, двунаправленную LSTM, предсказывающие авторство текстов по фрагменту текстов Фета А.А., Бунина И.А., Есенина С.А.

Для каждого выбранного поэта взять не менее 50 фрагментов из произведений, каждый фрагмент содержит не менее 100 символов.

Получить токенизатор и обученную модель BERT с HuggingFace.

Выполнить загрузку и предварительную обработку данных из текстов. Разделить каждую выборку на обучающую, тестовую и валидационную.

Добавить к архитектуре BERT полносвязный (или два полносвязных) слой(я) и обучить полученную модель в следующих вариантах:

1) Заморозить веса в BERT и настроить добавленный(е) полносвязный(ые) слой(и). (Это – модель 1a)

2) Выполнить Fine-tuning полученной архитектуры. (Это – модель 1b)

Обучить вторую модель - стек LSTM (две последовательные ячейки LSTM, добавить Dropout) для классификации текстовых данных.

Обучить третью модель – двунаправленную ячейку LSTM для классификации текстовых данных.

Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах.

Критерии оценивания:

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка «отлично» выставляется, если даны правильные ответы на все теоретические вопросы и все задачи решены без ошибок.

Оценка «хорошо» выставляется, если корректные ответы даны на большую часть вопросов, но были отмечены неуверенность в ответе и информация представлена фрагментарно, также задачи были решены правильно с небольшими замечаниями.

Оценка «удовлетворительно» выставляется, если даны правильные ответы на половину теоретические вопросы и одна из задач решена с негрубой ошибкой.

Оценка «неудовлетворительно» выставляется, если корректных ответов меньше половины и задачи были решены неправильно.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Тест (ИУК-1.1, ИУК-1.2)

1. Какая нейросетевая модель из перечисленных в лучшей степени подходит для прогнозирования временных последовательностей?
 - а) Single-Layer Perceptron
 - б) CNN
 - в) LSTM
 - г) Multi-layer Perceptron
2. Почему модели на сверточных нейронных сетях показывают наилучшие показатели по классификации объектов на изображениях по сравнению с другими моделями?
 - а) Они в высокой степени оптимизированы для обработки векторов с числовыми, а не категориальными признаками
 - б) Они обладают широким набором инструментов преобразования признакового пространства, которые может варьировать разработчик в модели
 - в) Они учитывают корреляцию смежных компонент вектора
 - г) Они используют существенно *большее* число настраиваемых параметров, по сравнению с другими моделями
3. Каким главным недостатком обладает рекуррентная нейронная сеть?
 - а) Длительная процедура обучения
 - б) Невозможность обучения на категориальных данных
 - в) Сложность запоминания длительных последовательностей
 - г) Использование существенных вычислительных ресурсов

4. Какие меры не приводят к уменьшению переобучения нейросетевой модели?
 - а) Установка штрафов за большие значения весов нейронов сети
 - б) Увеличение количества слоев сети
 - в) Добавление шума в выборку
 - г) Уменьшение количества нейронов сети
5. От чего в большей степени зависит успешное решение задачи классификации однослойным персептроном?
 - а) от размера выборки
 - б) от размерности признакового пространства
 - в) соотношения разделения выборки на обучающую и тестовую
 - г) от распределения объектов в пространстве признаков

Ключи: 1 в), 2 в), 3 в), 4 б), 5 г).

Задачи (ИПК-5.1, ИПК-5.2, ИУК-1.3)

Задача 1. Построить регрессор – стек двух слоёв с LSTM (или GRU) для прогноза добычи жидкости и обводненности для каждой скважины на 120 суток вперед по предыдущим значениям признаков из выборки.

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество нейронов в слоях (ячеек), процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными. Сделать подбор количества предыдущих значений, используемых для прогнозной модели, для более точной оценки целевой переменной. Сделать эксперименты с прореживанием (Dropout).

Вычислить следующие метрики работы: R2, MSE, MAE для всех исследованных моделей.

Задача 2. Построить нейросетевые модели – бинарный и многоклассовый классификатор, согласно Вашему варианту задачи.

Сначала построить бинарный классификатор для классов 0 и 1.

Набор данных: <https://www.kaggle.com/datasets/emmarex/plantdisease> Класс 0: Tomato_YellowLeaf_Curl_virus, Класс 1: Tomato_healthy, Класс 2: Tomato_Septoria_Leaf_spot

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество карт признаков, распределение слоёв, процедур оптимизации:

Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными для следующих задач.

Выполнить эксперименты с прореживанием (Dropout).

Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах.

Выполнить аугментацию данных и провести повторное обучение на новом расширенном наборе.

Сохранить лучшую архитектуру нейронной сети – бинарного классификатора.

Создать нейронную сеть – многоклассовый классификатор с использованием сверточной основы, обученной ранее на ImageNet (ResNet, VGG, Inception, Xception и т.д.). Выполнить дообучение модели в двух вариантах: заключительные полносвязные

слои и fine-tuning. Сохранить лучшие модели. Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах из трех классов.

Теоретические вопросы (ИУК-1.1, ИУК-1.2, ИПК-5.3)

1. На что влияет размер эмбединга в LLM?
2. Что такое позиционное кодирование?
3. Из каких блоков состоит трансформер?
4. Реализация механизма внимания.
5. Классы задач обработки последовательностей.
6. Вычисление градиента в рекуррентной нейронной сети.
7. Штрафы по норме параметров.
8. Ранняя остановка.
9. Разряженные представления.
10. Использование прореживания.
11. Пакетная нормализация.
12. Усреднение Поляка.
13. Рекурсивные нейронные сети.
14. Нейронные эхо-сети.
15. Репрезентативная способность в автокодировщиках.
16. Жадное послойное предобучение без учителя.
17. Ограниченные машины Больцмана.
18. Глубокие сети доверия.
19. Глубокие машины Больцмана.
20. Ориентированные порождающие сети.
21. Проблемы оптимизации нейронных сетей.
22. Алгоритмы оптимизации с адаптивной скоростью обучения.
23. Приближенные методы оптимизации второго порядка.
24. Алгоритм итерации нейронной ценности
25. Архитектура глубокой Q-сети
26. Настройка двойной глубокой Q-сети
27. Дуэльные архитектуры в ценностно-ориентированном глубоком обучении с подкреплением.
28. Методы приоритетного воспроизведения полезного опыта.
29. Параллельное обновление политики в A3C.
30. Надежное прогнозирование преимущества GAE.
31. Синхронное обновление политик в A2C.
32. Формирование функции ценности в VPG.
33. Максимизация ожидаемой выгоды и энтропии в «мягком актёре-критике».
34. Отличия методов градиента политик и «актёр-критик».
35. Генеративно-состязательная сеть Вассерштайна.
36. Особенности CycleGAN
37. Мультимодальные трансформеры
38. Диффузионные модели
39. Графовые нейронные сети.
40. Получение объяснений в глубоких моделях.

Информация о разработчиках

Аксёнов Сергей Владимирович, к.т.н., кафедра теоретических основ информатики (ТОИ) Института прикладной математики и компьютерных наук (ИПМКН) Национальный исследовательский Томский государственный университет (НИ ТГУ), доцент каф. ТОИ.