

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таскаев Сергей Валерьевич
Должность: Ректор
Дата подписания: 2025.02.16 09:00:59
Уникальный программный ключ:
054c0182970293149c2169910009940292896864

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю) «Искусственные нейронные сети» по направлению подготовки (специальности) 02.03.02 «Фундаментальная информатика и информационные технологии» направленности (профилю) «Математические и алгоритмические основы интеллектуальных систем» ФГБОУ ВО «ЧелГУ»

стр. 1

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)

Искусственные нейронные сети

Направление подготовки (специальность)

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль)

Математические и алгоритмические основы интеллектуальных систем

Присваиваемая квалификация (степень)
бакалавр

Форма обучения
очная

Троицк, 2025 г.



Содержание

1. Паспорт фонда оценочных средств
2. Перечень формируемых компетенций
 - 2.1. Компетенции, закреплённые за дисциплиной
3. Содержание оценочных средств по дисциплине
 - 3.1. Виды оценочных средств
 - 3.2. Содержание оценочных средств
4. Порядок проведения и критерии оценивания промежуточной аттестации
 - 4.1. Порядок проведения промежуточной аттестации
 - 4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств
 - 4.3. Результаты промежуточной аттестации и уровни сформированности компетенций



1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Направление подготовки: *02.03.02 «Фундаментальная информатика и информационные технологии»*

Направленность (профиль): *Математические и алгоритмические основы интеллектуальных систем*

Дисциплина: *Искусственные нейронные сети.*

Семестры изучения: *8.*

Формы промежуточной аттестации: *зачет – 8 с.*

Для оценивания результатов используется балльно-рейтинговая система

2. ПЕРЕЧЕНЬ ФОРМИРУЕМЫХ КОМПЕТЕНЦИЙ

2.1. Компетенции, закреплённые за дисциплиной

Изучение дисциплины «Искусственные нейронные сети» направлено на формирование следующих компетенций:

Коды компетенции согласно ФГОС (ОПОП ВО)	Содержание компетенций согласно ФГОС (ОПОП ВО)	Индикаторы достижения компетенции согласно ОПОП	Перечень планируемых результатов обучения по дисциплине
1	2	3	4
УК-4	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах)	УК -4.1. Имеет представление о правилах и принципах деловой устной и письменной коммуникации на государственном языке Российской Федерации и иностранном (ых) языке(ах) УК -4.2. Демонстрирует умение осуществлять деловую коммуникацию в устной и письменной формах, использовать методы и навыки делового общения УК -4.3. Имеет навыки делового общения на государственном языке Российской Федерации и иностранном (ых) языке(ах)	Знать: основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации. (для достижения УК-4.1) Уметь: составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений. (для достижения УК-4.2) Владеть: навыками вербальной коммуникации на техническом иностранном языке; навыками организации запросов в информационно-поисковых системах на



			русском и английском языках. (для достижения УК-4.3)
ПК-2	Способность к осуществлению интеграции программных модулей и компонент и проверки работоспособности программного продукта на основе международных и профессиональных стандартов ИТ, современных парадигм и методологий, инструментальных и вычислительных средств, методов и механизмов оценки и анализа функционирования средств и систем информационных технологий.	ПК-2.1. Обладает знаниями о методах и средствах сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах инструментальных и вычислительных средств ПК-2.2. Демонстрирует умения: применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; проводить проверку и оценку работоспособности программного продукта. ПК-2.3. Имеет практический опыт (навыки): сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; оценки работоспособности программного продукта.	Знать: методы и средства сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах; основы теории нейронных сетей. (для достижения ПК-2.1) Уметь: применять методы и средства сборки модулей и компонент программного обеспечения; проводить исследования задач классификации с применением нейронных сетей. (для достижения ПК-2.2) Владеть: навыками создания программных интерфейсов; оценки работоспособности программного продукта; навыком конструирования нейронных сетей. (для достижения ПК-2.3)
ПК-3	Способность к	ПК-3.1. Обладает знаниями	Знать: методы и средства



	разработке требований и проектированию программного обеспечения на основе применения базовых математических знаний и информационных технологий при решении проектно-технических и прикладных задач.	о методах и средствах проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. ПК-3.2. Демонстрирует умения: разрабатывать требования к программному продукту, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. ПК-3.3. Имеет практический опыт (навыки): проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.	проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, виды систем поддержки принятия решений, вопросы практического использования экспертных и интеллектуальных информационных систем. (для достижения ПК-3.1) Уметь: использовать программные средства разработки систем искусственного интеллекта; классифицировать решаемые задачи, анализировать архитектуру экспертных систем. (для достижения ПК-3.2) Владеть: навыками работы с базами данных и базами знаний; быть способным оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств; навыки реализации простейших ЭС. (для достижения ПК-3.3)
--	---	--	---

3. СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

3.1 Виды оценочных средств

№ п/п	Контролируемые темы/разделы	Код компетенции/планируемые результаты обучения	Наименование оценочного средства для текущего контроля	Наименование оценочного средства на промежуточной аттестации
1	Методы представления и обработки знаний.	УК-4, ПК-2, ПК-3 (знания, умения, навыки)	Домашняя работа, лабораторная работа, тестирование, устный опрос	Практическое задание для зачета Устный опрос (вопросы 1-10), тестирование



2	Нейронные сети	УК-4, ПК-2, ПК-3 (знания, умения, навыки)	Домашняя работа, лабораторная работа, тестирование, устный опрос	Практическое задание для зачета Устный опрос (вопросы 11-19), тестирование
3	Нечеткая логика	УК-4, ПК-2, ПК-3 (знания, умения, навыки)	Домашняя работа, лабораторная работа, тестирование, устный опрос	Устный опрос (вопросы 20-29), тестирование Экзаменационное тестирование
4	Генетические алгоритмы	УК-4, ПК-2, ПК-3 (знания, умения, навыки)	Домашняя работа, лабораторная работа, тестирование, устный опрос	Устный опрос (вопросы 30-41), тестирование
5	Интеллектуальные системы	УК-4, ПК-2, ПК-3 (знания, умения, навыки)	Домашняя работа, лабораторная работа, тестирование, устный опрос	Устный опрос (вопросы 42-52), тестирование

Типовые задания, критерии и показатели оценивания в рамках текущего контроля представлены в рабочей программе по дисциплине. Полные комплекты оценочных средств и контрольно-измерительных материалов хранятся на кафедре и являются учебно-методическими материалами ограниченного (конфиденциального) пользования.

3.2 Содержание оценочных средств

Оценочные средства для промежуточной аттестации представлены базой контрольных вопросов, тестовых заданий к зачету и практическими заданиями для зачета.

3.2.1. База контрольных вопросов к зачету

1. Архитектура и классификация интеллектуальных систем. Этапы разработки систем искусственного интеллекта.
2. Данные и знания. Сравнительная характеристика.
3. Приобретение и формализация знаний.
4. Организация и представление знаний. Модели представления знаний.
5. Логические модели представления знаний.
6. Продукционное представление знаний.
7. Семантические сети.
8. Фреймы.
9. Моделирование человеческих рассуждений в ИС. Логика Д.С.Милль.



10. ДСМ-метод выделения признаков для описания ситуации. Рассуждения по аналогии.
11. Модели и механизмы вывода на знаниях.
12. Прямая цепочка рассуждений. Технология разработки программы.
13. Обратная цепочка рассуждений. Технология разработки программы.
14. Выбор в условиях неопределенности (формула Байеса, коэффициенты уверенности).
15. Методы оценки субъективной вероятности.
16. Стратегии и методы поиска решений. Общие характеристики.
17. Поиск в пространстве состояний. Эвристический поиск.
18. Поиск в иерархии пространств.
19. Понятие нечеткой логики и отношений. Лингвистическая переменная.
20. Способы построения функции принадлежности. Метод парных сравнений.
21. Построение функции принадлежности лингвистических термов с использованием статистических данных.
22. Параметрический подход к построению функции принадлежности.
23. Выбор альтернатив на основе НМ. Принятие решений в условиях определенности.
24. Нейронные сети. Основы проектирования и сферы применения. Модель формального нейрона. Реализация нелинейной зависимости в нейронной сети.
25. Методы обучения нейронных сетей (с учителем и без учителя). Метод обратного распространения ошибки.
26. Алгоритм обучения сети методом обратного распространения ошибки.
27. Генетические алгоритмы. Концепция. Генетические операторы. Примеры применения.
28. Детерминированные конечные автоматы и преобразователи. Варианты их использования в задачах обработки естественного языка.
29. N-граммы. Определение, виды. Использование в задаче предсказания следующего слова.
30. Сглаживание данных о N-граммах. Цель, виды.
31. Классы слов и теггирование. Растановка тегов на основе правил.
32. Стохастическое теггирование.
33. Комбинированное теггирование. Подход на основе трансформационно-ориентированного обучения.
34. Синтаксический анализ. Контекстно-свободные грамматики.
35. Алгоритм парсинга Кока-Янгера-Касами.



36. Статистический парсинг. Вероятностные контекстно-свободные грамматики.

37. Понятие семантики. Требования к представлению смысла.

38. Модель представления смысла. Варианты представления моделей (логика предикатов первого порядка, фреймы, семантические сети и т. п.)

39. Семантический анализ на основе синтаксиса. Подход на основе контекстно-свободных грамматик.

40. Расширение функционала на основе матриц атрибут-значение (AVM). Операция унификации.

41. Лексическая семантика. Понятие смысла. Варианты задания смысла.

42. Задача устранения противоречий смысла слова (Word sense disambiguation). Подходы к решению задачи.

43. Вычисление семантических отношений между словами. Методы на основе использования тезауруса.

44. Вычисление семантических отношений между словами. Дистрибуционные методы.

45. Понятие дискурса. Задача линейной сегментации дискурса. Подходы к сегментации.

46. Когерентные отношения. Задача назначения когерентных отношений.

47. Разрешение ссылок. Подзадачи. Типы ссылающихся выражений.

48. Задача разрешения анафор. Параметры для разрешения анафор. Алгоритм Хоббса.

49. Задача разрешения анафор. Центрирующий алгоритм. Подходы на основе машинного обучения.

50. Разрешение со-ссылок (Coreference resolution). Параметры для разрешения совместных ссылок.

51. Диалоговые системы. Свойства диалогов. Базовая архитектура диалоговых систем.

52. Менеджер диалогов. Подходы к реализации.

3.2.2. Пример тестовых вопросов и заданий для зачета

Вариант 1

Задание 1 Какие функции выполняет входной слой многослойного персептрона?

1. Транслирует сигнал на выходной слой многослойного персептрона.
2. Удаляет "шум" из сигнала.



3. Передает входной вектор сигналов на первый скрытый слой.
4. Вычисляет производную для алгоритма обратного распространения ошибки.

Задание 2. Аксон – это выходной или входной отросток нейрона?

1. Входной
2. Выходной

Задание 3. Что идет сначала – мутация или кроссовер? (в генетических алгоритмах).

1. Мутация
2. Кроссовер
3. Операции кроссовер в генетических алгоритмах не существует.
4. Без разницы.

Задание 4. Нейрон j получил на вход сигнал от четырех других нейронов уровни возбуждения, значения которых равны 10, -20, 5, 4 и соответствующие веса связей равны 0.8, 0.5, 0.7 и -0.5 соответственно. Вычислите сигнал на выходе j -го нейрона в случае если функция активации нейронов есть гиперболический тангенс (\tanh). Выберите правильный ответ:

1. -
2. 0.2449
3. 0.3145
4. 0.5
5. -0.5

Задание 5. Дано: нейрон с функцией активации типа гиперболический тангенс с тремя входами. Входы все равны 1 и все веса также равны 1. Параметр θ в формуле гиперболического тангенса равен 1. Чему будет равен выход нейрона?

Вариант 2

Задание 1. Что обычно длиннее – аксон или дендрит?

1. Аксон
2. Дендрит



Задание 2. Можно ли применять функцию активации типа «ступенька» при методе обучения обратного распространения ошибки?

1. Да
2. Нет

Задание 3. Какая из нижеперечисленных нейронных сетей есть сеть с обратными связями?

1. Сеть Кохонена.
2. Сеть Хемминга.
3. Выходная звезда Гроссберга.
4. Радиально – базисная сеть.

Задание 4. Нейрон j получил на вход сигнал от четырех других нейронов уровни возбуждения, значения которых равны 10, -20, 5, 4 и соответствующие веса связей равны 0.8, 0.5, 0.7 и -0.5 соответственно. Вычислите сигнал на выходе j -го нейрона в случае если функция активации нейронов есть логистическая сигмоида (). Выберите правильный ответ:

1. -4
2. 0.25
3. -3.52
4. 1.7

Задание 5. Дано: нейронная сеть с одним скрытым слоем. У сети 1 вход, 3 нейрона в скрытом слое и один выход. Что будет на выходе сети в случае, если на входе 1, все веса равны 1?

3.2.3. Пример заданий для самостоятельной работы

Задание 2. Можно ли применять функцию активации типа «ступенька» при методе обучения обратного распространения ошибки?

1. Да
2. Нет

Задание 3. Какая из нижеперечисленных нейронных сетей есть сеть с обратными связями?

1. Сеть Кохонена.
2. Сеть Хемминга.



3. Выходная звезда Гроссберга.
4. Радиально – базисная сеть.

Задание 4. Нейрон j получил на вход сигнал от четырех других нейронов уровни возбуждения, значения которых равны 10, -20, 5, 4 и соответствующие веса связей равны 0.8, 0.5, 0.7 и -0.5 соответственно. Вычислите сигнал на выходе j -го нейрона в случае если функция активации нейронов есть логистическая сигмоида (σ). Выберите правильный ответ:

1. -4
2. 0.25
3. -3.52
4. 1.7

Задание 5. Дано: нейронная сеть с одним скрытым слоем. У сети 1 вход, 3 нейрона в скрытом слое и один выход. Что будет на выходе сети в случае, если на входе 1, все веса равны 1?

3.2.3. Примеры кейсов-заданий для практической работы

№ 1 – Эволюция информационных систем.

Охарактеризуйте основные направления исследований, проводимые в области искусственного интеллекта. Приведите известные вам примеры интеллектуальных систем. Назовите основные функции, присущие ИИС. На чем основана их реализация? Дайте краткую характеристику систем с интеллектуальным интерфейсом, экспертных систем, самообучающихся систем и адаптивных информационных систем. Сформулируйте основные отличия систем искусственного интеллекта от обычных программных средств. Перечислите и охарактеризуйте основные компоненты статических экспертных систем. Какого профиля специалисты привлекаются для разработки экспертных систем? Каковы их функции? Чем отличаются динамические экспертные системы от статических? Охарактеризуйте экспертную систему по следующим параметрам: типу приложения, стадии существования, масштабу, типу проблемной среды, типу решаемой задачи. Расскажите об основных характеристиках инструментальных средств, предназначенных для разработки интеллектуальных информационных систем (уровень используемого языка, парадигма программирования; способ представления знаний, механизм вывода и моделирования, средства приобретения знаний, технологии разработки приложений).



Домашнее задание. Найдите в Интернете описания ИИС. Дайте их сравнительную оценку. Подготовьте сообщение и интеллект-карту, посвященную конкретной экспертной системе.

№ 2 – Нейронные сети.

Опишите модель искусственного нейрона. Приведите примеры передаточных функций. Что такое перцептрон? Какие модели нейронных сетей вам известны? Проведите сравнение однослойных и многослойных ИНС. Какими особенностями обладают рекуррентные и самоорганизующиеся сети? Расскажите о моделях ИНС Хопфилда и Кохонена.

Дайте характеристику основных этапов построения нейронной сети. Расскажите о методах обучения ИНС (коррекция по ошибке, обучение Хебба, соревновательное обучение, метод обратного распространения ошибки). Опишите алгоритм обратного распространения ошибки. Сформулируйте его достоинства и недостатки. Расскажите об известных вам способах реализации ИНС. Для каких задач целесообразно применять ИНС? Каковы условия применения моделей этого типа? Сформулируйте основные проблемы, возникающие при применении нейронных сетей.

Сформулируйте постановку прикладной задачи экономического характера, для решения которой возможно и целесообразно применить нейронную сеть. Опишите, как это можно сделать. Опишите методику решения задач определенного класса при поддержке выбранного программного обеспечения. Подготовьте набор содержательных примеров для обучения нейронной сети с заданной целью. Сформулируйте постановку содержательной задачи для решения методами нейронных сетей.

Подготовьте обучающую и тестирующую выборки примеров. Выберите топологию сети, выполните проектирование и реализацию в системе SciLab или Deductor (другой системе). Сформулируйте постановку задачи извлечения знаний для решения с помощью технологии нейронных сетей. Подготовьте необходимые данные. Выберите топологию сети, выполните проектирование и реализацию в системе SciLab или Deductor (NeuroSolutions или другой системе). Составьте задачу классификации (диагностики) для решения с помощью технологии нейронных сетей. Подготовьте необходимые данные. Выберите топологию сети, выполните проектирование и реализацию в системе NeuroSolutions (или другой системе). Сформулируйте задачу прогнозирования для решения с помощью технологии нейронных сетей. Подготовьте необходимые данные. Выберите топологию сети, выполните проектирование и реализацию в системе SciLab, Deductor, NeuroSolutions или другой системе.



Домашнее задание. Подготовьте сообщение или интеллект-карту по разработанной и обученной ИНС в области экономики, бизнесе (другой проблемной области)».

№ 3 – Нечеткие знания и способы их обработки.

Поясните смысл понятия «нечеткость» знаний. Дайте характеристику компонентам нечеткости. Что такое недетерминированность выводов? Какие средства следует использовать в системах, обладающих этим свойством? Проанализируйте какую-либо известную вам игру на предмет ее машинной реализации. Какими приемами необходимо обеспечить компьютерного игрока в этой игре? Расскажите о способах устранения многозначности. Почему ее необходимо устранять? Приведите примеры.

Какими способами можно представлять и обрабатывать ненадежные знания? Приведите примеры. Охарактеризуйте способы обработки неполных знаний в интеллектуальных системах. Приведите собственный пример появления противоречия в логической ЭС при добавлении нового знания. Какие преимущества по сравнению с логическими имеют фреймовые системы? Что такое абдукция? Дайте формальное определение и объясните, чем она отличается от дедукции. Приведите примеры.

Дайте определение понятий «лингвистическая переменная» и «нечеткое множество», поясните их на примере. Какие операции можно выполнять над нечеткими множествами? Рассмотрите математические операции над нечеткими множествами: дополнение, объединение и пересечение множеств.

Дайте определение нечеткого отношения и расскажите о свойствах нечетких отношений. Каким образом и с какой целью используются нечеткие отношения в ИИС? Что такое нечеткая импликация? Какими способами ее можно реализовать для правил с одним выходом и двумя выходами? Приведите примеры. Расскажите о нечетком логическом выводе. Чем он отличается от четкого логического вывода по правилу *Modus Ponendo Ponens*. Дайте определение композиции нечетких отношений.

Приведите пример нечеткой композиции. Расскажите об организации ЭС с нечетким логическим выводом. Какие способы используются в таких системах для представления и обработки знаний? Подготовьте собственные наборы нечетких правил для реализации в системе с нечетким выводом.

Разработайте интеллектуальную систему с нечетким представлением знаний для решения небольшой прикладной экономической или управленческой задачи.



№ 4 – Эволюционные аналогии в искусственных интеллектуальных системах.

Перечислите основные направления эволюционного моделирования и приведите основные факторы, определяющие неизбежность эволюции. Какие алгоритмы называют генетическими? Сформулируйте основные особенности генетических алгоритмов. Охарактеризуйте простой генетический алгоритм. Приведите пример. Опишите операторы репродукции и кроссинговера в простом генетическом алгоритме. Приведите примеры. Приведите примеры использования простого генетического алгоритма для вычисления функции $f(x) = x^4$ на интервале $[0, 1, 2, 3, 4]$.

Составьте примеры, иллюстрирующие работу операторов репродукции, кроссинговера, мутации и инверсии. Дайте характеристику понятию «схема» в простом генетическом алгоритме. Расскажите о назначении и способах использования схем. Приведите примеры. Расскажите о фундаментальной теореме генетического алгоритма. Приведите пример применения фундаментальной теоремы генетического алгоритма. Расскажите о классифицирующих системах Холланда. Приведите пример.

Перечислите основные этапы технологии генетического программирования. В чем особенности эволюционного программирования? Приведите основные шаги обобщенного алгоритма эволюционного программирования. Охарактеризуйте метод эволюционных стратегий. В чем его отличие от эволюционного программирования и от генетических алгоритмов? Расскажите о применении эволюционных вычислений в ИИС. Каким образом применяют ГА для обучения нейронных сетей?

Приведите небольшой содержательный пример, демонстрирующий применение ГА для формирования продукционных правил интеллектуальной системы.

Домашнее задание. Сформулируйте прикладную экономическую или управленческую оптимизационную задачу и опишите ее решение с применением генетического алгоритма.

№ 5 – Интеллектуальные мультиагентные системы.

Расскажите о сущности мультиагентных технологий. Что подразумевается под агентом и как он может быть реализован? Какими свойствами обладают «интеллектуальные агенты»? Дайте характеристику архитектурам мультиагентных систем. Сформулируйте основные проблемы, возникающие при моделировании коллективного поведения интеллектуальных агентов.



Охарактеризуйте основные модели координации поведения агентов в мультиагентных системах: теоретико-игровые, модели коллективного поведения автоматов, модели планирования коллективного поведения, модели на основе VDI- архитектур, модели координации поведения на основе конкуренции. Сформулируйте постановки задач координации поведения агентов на основе модели аукциона.

Проведите сравнительный анализ свойств мобильных и статических агентов. Опишите технологию построения мультиагентных систем. Приведите примеры инструментальных средств, предназначенных для этой цели. Перечислите основные преимущества интеллектуальных поисковых мультиагентных систем перед традиционными средствами поиска информации. Для каких задач актуально применение мультиагентных технологий? Приведите примеры. Сформулируйте содержательный пример задачи кооперации и покажите возможный способ ее решения средствами мультиагентных технологий. Приведите пример задачи координации коллективного поведения, для решения которой актуально применение мультиагентных технологий. Сформулируйте принцип координации и правила нормативного поведения агентов. Спроектируйте структуру мультиагентной системы для реализации конкретного виртуального предприятия. Опишите виды агентов, их функции и способы возможной реализации. Охарактеризуйте механизм координации поведения агентов.

3.2.4. Пример проектных заданий для самостоятельной работы

Проектное задание № 1

Программная реализация нейронной сети на языке Python

Постановка задачи

Даны два изображения. Необходимо используя нейронную сеть распознать на первом изображении руки и лица людей, распознать руки и лица людей на втором изображении, и определить, было ли изменение позиций распознаваемых объектов на первом изображении с позициями на втором изображении соответственно на заданную величину. Если позиция изменилась, оповестить об этом в программе.

Данную задачу можно рассматривать как детектор движения заданных объектов и в дальнейшем улучшить до захвата изображения с камер и сравнением двух следующих друг за другом кадров.



Выбор среды разработки

Для создания данного проекта необходимо использовать язык программирования Python с предустановленной библиотекой Keras. Для его функционирования, необходимо выбрать из уже предоставленных предобученных сетей выбрать наиболее подходящую по скорости и точности распознавания. Так же, предварительно необходимо создать датасет, содержащий метки входных изображений с позиционированием каждого распознаваемого объекта. Для создания датасета использовалась программа labeling, с помощью которой были созданы xml файлы с разметкой рук и лиц людей на изображении. После чего, данные файлы были преобразованы с помощью python скрипта в файлы записи, используемые Tensorflow object detection API. На выбор, данная программа предоставляет большой спектр нейронных сетей, которые отличаются своей архитектурой.

В качестве примера, была использована сверточная сеть, с названием “ssd_mobilenet_v2_coco”. Она является самой быстрой из всех имеющихся по скорости, но самой неточной по распознаванию объектов. В большинстве случаев, ее используют для распознавания объектов в реальном времени на камерах мобильных телефонов. К данной сети, прилагается конфигурационный файл для ее настройки, где можно указать такие параметры как скорость обучения, функции активации нейронов, способ обучения нейронной сети.

После настройки, через командную строку был запущен процесс обучения на данных, которые были подготовлены ранее с помощью labeling. Несмотря на то, что все параметры для обучения были установлены для минимальной затраты ресурсов компьютера, процесс обучения нейронной сети по подсчетам мог бы занять продолжительное время.

На каждый шаг обучения нейронной сети в среднем уходило по 60 секунд. Для нормального функционирования нейронной сети необходимо было примерно 20000 шагов, что заняло бы приблизительно 330 часов, поэтому, данный способ реализации не подходил для выполнения задачи.

Для создания программы будет использован фреймворк Keras, для Python, позволяющий создать нейронную сеть и избежать проблем с большим периодом ее обучения.

Описание алгоритма реализации



Любое изображение, может быть представлено в виде двумерного массива, индексами которого являются координаты пикселей изображения, а их значениями массив, состоящий из 3 элементов (R, G, B), в диапазоне 0-255. Таким образом, каждой точке изображения однозначно соответствует три числа, определяющие ее цвет.

В дальнейшем, с помощью библиотеки Keras планируется создать модель сверточной нейронной сети Sequential, которая будет обучена на обучающей выборке созданной в программе labeling, которая упрощает работу выделения объектов на изображении. После обучения нейронной сети, с помощью библиотеки OpenCV (библиотека компьютерного зрения) будет производиться преобразование и загрузка данных изображений в оперативную память компьютера. Изображения будут передаваться на вход обученной нейронной сети, которая будет выделять заданные объекты. Данные объекты будут храниться в массиве, каждый элемент которого будет иметь позицию x и y, высоту и ширину h и w соответственно.

Проектное задание № 2

Принципы создания и обучения ИНС в системе Deductor Studio Academic

Цель работы - познакомиться с принципами создания и обучения искусственных нейронных сетей (ИНС) на примере решения практической задачи кластеризации данных в программной системе интеллектуального анализа данных Deductor Studio фирмы BaseGroup Labs. Это позволит совершенствовать навыки учащихся в решении задач с привлечением методов научного направления «Искусственный интеллект», а также при создании интеллектуальных информационных систем.

Задачи практической работы:

- освоить принципы работы в системе Deductor;
- научиться создавать и обучать самоорганизующуюся ИНС для решения задачи кластеризации данных;
- самостоятельно решить задания лабораторной работы в соответствии с изложенными теоретическими и практическими положениями.

Продолжительность лабораторной работы - 4 часа. Первые два часа отводятся на ознакомление с методическими указаниями (п.1-3), оставшиеся часы рассчитаны на решение индивидуальных заданий к лабораторной работе (п.4) и оформление отчета по результатам их выполнения.



Результаты, полученные при выполнении лабораторной работы, также могут быть использованы студентами данных специальностей при курсовом и дипломном проектировании.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Последовательность выполнения лабораторной работы:

1. Изучить теоретические положения и разобрать предлагаемые в методических указаниях примеры.
2. Решить прилагаемые индивидуальные задания.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Нейронные сети, использующие принципы обучения без учителя, часто относят к самоорганизующимся, а именно способным особым образом группировать и обобщать информацию. Примером такой сети является самоорганизующаяся карта Кохонена.

Процесс обучения без учителя, как и в случае обучения с учителем, заключается в подстраивании весов синапсов. При этом их подстройка может проводиться только на основании информации о состоянии нейрона и уже имеющихся значениях весовых коэффициентов.

Среди алгоритмов, предназначенных для данного типа обучения, можно выделить два - предложенных Хеббом и Кохоненом. Каждый из них имеет строгую математическую теорию, а также ряд модификаций. Однако именно модель обучения без учителя, предложенная Т. Кохоненом в 1984 году и являющаяся классической, получила наибольшее распространение при решении практических задач. Причинами этого являются более точная работа, а также близость заложенных в ней идей к реальному функционированию мозга.

Алгоритм Кохонена дает возможность строить нейронную сеть для разделения векторов входных сигналов на подгруппы. Сеть состоит из n нейронов, образующих прямоугольную решетку на плоскости (рис. 1). Элементы входных сигналов подаются на входы всех нейронов сети. В процессе работы алгоритма настраиваются синаптические веса нейронов.

Входные сигналы (вектора действительных чисел) последовательно предъявляются сети. Желаемые выходные сигналы не определяются. После того как было предъявлено достаточное число входных векторов, синаптические веса сети определяют кластеры.



Для отнесения объекта к кластеру необходимо найти кластер с минимальным расстоянием от этого объекта до центроида кластера. Как правило, для этого используется расчет Евклидова расстояния или расстояния Хемминга.

В результате решения задачи кластеризации получается карта кластеров, где близко расположенным кластерам соответствуют более близкие друг к другу входные вектора нейросети. Все это позволяет выполнять наглядное упорядочивание многопараметрической информации. Важно, что при этом могут быть обнаружены неожиданные скопления «близких» данных, последующая интерпретация которых пользователем может привести к получению нового знания об исследуемом объекте.

Рассмотрим шаги процесса обучения нейросети по алгоритму, предложенному Кохоненом. Отметим, что в данном случае под нейросетью понимается однослойный персептрон – слой Кохонена.

Шаг 1. Инициализация сети.

Шаг 2. Сети предъявляется новый входной вектор.

Шаг 3. Вычисляется Евклидово расстояние до всех нейронов слоя Кохонена

Шаг 4. Настраивается «нейрон – победитель» и его соседи.

Шаг 5. Повторяются шаги со 2 по 4 для каждого входного вектора.

После обучения кластеризация выполняется посредством подачи на вход сети испытываемого вектора. Нейрон со значением, равным единице, является индикатором кластера.

Для определения выходных значений слоя Кохонена используется правило «победитель получает все». Согласно ему для данного входного вектора только один нейрон слоя Кохонена выдает единицу, все остальные – ноль.

Максимальным значением S объявляется «победителем» и его выход устанавливается равным единице.

Обучающий алгоритм настраивает синаптические связи в окрестности «победившего» нейрона таким образом, чтобы они были похожими на входной вектор. Сначала в эту окрестность могут входить все нейроны слоя Кохонена, однако в процессе обучения сети такие зоны соседства будут сужаться, образуя топологические области (карты), которые соответствуют определенным входным векторам. Так, на рис. 2 показаны зоны соседства (S_j) в различные моменты времени для нейрона j , которые уменьшаются в процессе обучения

сети. В результате фактически возникает группировка входных векторов в кластеры, каждый из которых ассоциируется с определенным нейроном.



Подобный принцип функционирования ИНС известен как самоорганизующиеся карты Кохонена.

Существуют различные программные средства, реализующие алгоритм Кохонена для кластеризации данных. Одной из них, отличающейся удобным интерфейсом и простой использования, является программная система интеллектуального анализа данных Deductor Studio фирмы BaseGroup Labs.

3. ПОРЯДОК РАБОТЫ В СИСТЕМЕ DEDUCTOR

В качестве примера применения самоорганизующихся карт рассмотрим решение задачи кластеризации типа цветков ириса на основе их характеристик. Эта задача, известная как «ирисы Фишера», относится к тестовым и входит в группу задач для исследования возможностей нейросетевого моделирования (проект ELENA - Enhanced Learning for Evolutive Neural Architectures). Для доступа к данному примеру следует открыть файл «C:\Program Files\BaseGroup\ Deductor\Samples\Demo.ded», а затем выбрать «Пример кластеризации с помощью карты Кохонена».

Характеристики цветков соответствуют названиям столбцов таблицы. Задача состоит в том, чтобы определить по различным параметрам цветка его класс. Предполагается, что цветы одного класса имеют схожие значения параметров, следовательно, они должны располагаться в одном кластере.

Процесс формирования карты этих кластеров заключается в выполнении следующей последовательности действий.

На первом шаге настраивается назначение столбцов таблицы, при этом входными данными считаются «Длина чашелистика», «Ширина чашелистика», «Длина лепестка», «Ширина лепестка», выходными – «Класс цветка».

На втором шаге необходимо настроить способ разделения исходного множества данных на тестовое и обучающее, а также количество примеров в том и другом множестве.

Следующий шаг предлагает настроить параметры карты (количество ячеек по X и по Y, их форму) и параметры обучения (способ начальной инициализации, тип функции соседства, перемешивать ли строки обучающего множества и количество эпох, через которые необходимо перемешивание).

На четвертом шаге настраиваются параметры остановки обучения

На пятом шаге настраиваются остальные параметры обучения – способ начальной инициализации, тип функции соседства и также параметры кластеризации – автоматическое определение числа кластеров с



соответствующим уровнем значимости либо фиксированное количество кластеров. Предоставляется возможность настроить интервалы обучения. Каждый интервал задается количеством эпох, радиусом и скоростью обучения. Число кластеров для рассматриваемой задачи устанавливается равное трем.

На последнем шаге запускается сам процесс обучения. При этом во время обучения могут быть просмотрены количество распознанных примеров и текущие значения ошибок.

Далее после настройки параметров отображения могут быть просмотрены результаты кластеризации данных, на основе которых можно решить и задачу прогнозирования. В этом случае результирующее поле (то, которое необходимо спрогнозировать) в построении карты не участвует. После кластеризации, используя диаграмму «Что-если» можно провести эксперимент. Алгоритм определяет точку пространства, где расположены введенные для прогноза данные, затем определяет, к какому кластеру принадлежит данная точка, и подсчитывает среднее по результирующему полю всех точек этого кластера, что и будет результатом прогноза (для дискретных данных результатом прогноза является значение, больше всего встречающееся в результирующем поле всех ячеек кластера).

Анализ результатов может быть выполнен не только в целом по конкретному цветку, но и по каждой из его характеристик. Качество кластеризации можно оценить, просмотрев карту «КЛАСС ЦВЕТКА». На ней видно, что большинство цветов были классифицированы правильно. Заметим, что все цветы класса *Setosa* попали в один кластер. Это говорит о значительном отличии параметров цветов этого класса от других. Явное различие наблюдается по длине и ширине лепестка. То, что часть примеров *Virginica* попала в класс *Versicolor* и наоборот, говорит о меньшем различии этих классов. На картах, в отличие от *Setosa*, не видны резкие отличия параметров цветов этих двух классов. Этим как раз и объясняется «проникновение» некоторой части примеров в другой кластер.

Таким образом, применение самоорганизующихся карт позволило визуально представить начальное многомерное (четырёхмерное) пространство исходных данных в двумерной форме, удобной для восприятия и анализа. Информация, описывающая цветки, была сгруппирована в три кластера, соответствующие типам цветков «*Setosa*», «*Versicolor*» и «*Virginica*». Вследствие этого в дальнейшем может быть достаточно точно определен тип цветка на основе его характеристик, отсутствующих в данных кластеризации.

ВАРИАНТЫ ЗАДАНИЙ для самостоятельного выполнения



Данные к выполнению заданий сгруппированы в таблице. При выполнении индивидуальных заданий необходимо учитывать, что число параметров кластеризации, а также классов объектов должно быть не менее 5. Число строк в таблице с исходными данными кластеризации должно быть не менее 50.

Таблица Исходные данные для выполнения индивидуального задания

№ варианта Задание

- 1 Разработка ИНС для кластеризации данных об автомобилях
- 2 Разработка ИНС для классификации данных о квартирах
- 3 Разработка ИНС для классификации данных о компьютерах
- 4 Разработка ИНС для классификации данных о телевизорах
- 5 Разработка ИНС для кластеризации данных о высших учебных заведениях
- 6 Разработка ИНС для кластеризации данных о мобильных телефонах
- 7 Разработка ИНС для кластеризации данных о студентах
- 8 Разработка ИНС для кластеризации данных о музыкальных инструментах
- 9 Разработка ИНС для кластеризации данных о странах мира
- 10 Разработка ИНС для кластеризации данных о банковских кредитах
- 11 Разработка ИНС для кластеризации данных о фильмах
- 12 Разработка ИНС для кластеризации данных о книгах
- 13 Разработка ИНС для кластеризации данных о видеокамерах
- 14 Разработка ИНС для кластеризации данных о фотоаппаратах
- 15 Разработка ИНС для кластеризации данных о стиральных машинах
- 16 Разработка ИНС для кластеризации данных о санаториях и пансионатах
- 17 Разработка ИНС для кластеризации данных о городах

ОФОРМЛЕНИЕ И ПОРЯДОК КОНТРОЛЯ РЕЗУЛЬТАТОВ

Результатом выполнения проектной работы является проект, созданный в системе Deductor и содержащий результаты выполнения лабораторной работы.

Отчет по лабораторной работе оформляется на листах формата А4 и сдается в распечатанном виде. Отчет должен содержать задание к лабораторной работе, таблицу исходных данных кластеризации, описание конфигураций ИНС, результаты их обучения и тестирования. Преподаватель проверяет корректность конфигураций ИНС, а также правильность полученных результатов. При защите лабораторной работы преподаватель вправе задать вопросы по теоретической части.



6. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Опишите, в чем состоит процесс решения задачи кластеризации данных.
2. Охарактеризуйте особенности обучения самоорганизующейся карты Кохонена.
3. Назовите основные шаги алгоритма обучения ИНС без учителя.
4. Расскажите, как происходит настройка ИНС для решения задачи кластеризации в системе Deductor.
5. Опишите, как происходит процесс обучения ИНС в системе Deductor.
6. Назовите доступные в системе Deductor способы оценки качества кластеризации данных.

4. ПОРЯДОК ПРОВЕДЕНИЯ И КРИТЕРИИ ОЦЕНИВАНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

4.1. Порядок проведения промежуточной аттестации

Промежуточная аттестация в 8 семестре проводится в форме зачета. Студент получает тестовое и практическое задание для выполнения на компьютере. Продолжительность – до 80 минут.

При дистанционном обучении устный опрос, в том числе защита курсовых работ, проводятся в Microsoft Teams. Практические задания и письменные ответы размещаются в системе Moodle. Тестирование осуществляется в системе Moodle.

4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств

4.2.1. Критерии оценивания на зачете

«Зачтено» (45-60 баллов) – выставляется, если студент в полном объеме выполнил предложенное задание, программа работает без ошибок, корректно обрабатывает запросы пользователя, либо работает с незначительными легко устранимыми ошибками. «Зачтено» соответствует критериям «отлично», «хорошо», либо «удовлетворительно» таблицы п. 4.3.

«Не зачтено» (до 45 баллов) – выставляется, если студент не смог выполнить предложенное задание, не умеет создавать и работать с базами
© ФГБОУ ВО «ЧелГУ»



данных, допускает значительные ошибки в написании запросов. «Не зачтено» соответствует критерию «неудовлетворительно» таблицы п. 4.3.

Критерии оценивания теоретического вопроса

Максимальный балл за ответ на теоретический вопрос — 10 баллов.

Отлично/ зачтено/ 9-10 баллов	Хорошо/ зачтено/ 7-8 баллов	Удовлетворительно/ зачтено/ 5-6 баллов	Неудовлетворительно /незачтено/ 0-4 балла
Высокий уровень освоения проверяемых компетенций	Средний уровень освоения проверяемых компетенций	Базовый уровень освоения проверяемых компетенций	Недостаточный уровень освоения проверяемых компетенций
Обучающийся отлично знает материал, умеет анализировать проблему и аргументировано изложить свою точку зрения, владеет достаточным для высказывания лексическим запасом, грамотно изъясняется на иностранном языке с использованием точных терминов и названий. Обучающийся практически не допускает ошибок.	Обучающийся хорошо знает материал, умеет анализировать проблему и аргументировано изложить свою точку зрения, владеет достаточным для высказывания лексическим запасом, грамотно изъясняется на иностранном языке с использованием точных терминов и названий. Обучающийся допускает незначительные ошибки.	Обучающийся знаком с материалом, владеет достаточным для высказывания лексическим запасом. Обучающийся допускает фактические и языковые ошибки, не оперирует лексическим запасом по теме.	Обучающийся не знает основных положений вопроса, не ориентируется в основных понятиях, излагает материал с трудом, с грубыми фактическими и языковыми ошибками, либо отказывается от ответов на вопросы.

Критерии оценивания решения задачи (разработка кода)

Максимальный балл за тест — 20 баллов.

Оценка	Отлично/ зачтено	Хорошо/ зачтено	Удовлетворитель но/зачтено	Неудовлетворительно/ не зачтено
Баллы	18-20 баллов	15-17 баллов	10-14 баллов	0-9 баллов
Характеристика	Работоспособный и оптимизированный код, тестовые значения приводят к корректному результату, присутствует комментарий.	Работоспособный, но не оптимизированный код, тестовые значения приводят к корректному результату, нет комментариев.	Частично работоспособный, и не оптимизированный код, не все тестовые значения приводят к корректному результату, нет комментариев.	Полностью неработоспособный, и не оптимизированный код, либо все тестовые значения приводят к некорректному результату, нет комментариев.
Уровень	высокий	средний	базовый	низкий

Критерии оценивания теста

Максимальный балл за тест — 20 баллов.



Оценка	Отлично/ зачтено	Хорошо/ зачтено	Удовлетворитель но/зачтено	Неудовлетворительно/ не зачтено
Уровень	Высокий уровень освоения проверяемых компетенций	Средний уровень освоения проверяемых компетенций	Базовый уровень уровня освоения проверяемых компетенций	Низкий уровень освоения проверяемых компетенций
Баллы	18-20 баллов	15-17 баллов	10-14 баллов	0-9 баллов

Критерии оценивания практического вопроса в билете

Максимальный балл за решение задачи — 15 баллов.

Отлично 14-15 баллов	Хорошо 10-13 баллов	Удовлетворительно 5-9 баллов	Неудовлетворительно 0-4 балла
Высокий уровень освоения проверяемых компетенций	Средний уровень освоения проверяемых компетенций	Базовый уровень уровня освоения проверяемых компетенций	Низкий уровень освоения проверяемых компетенций
Полное и верное решение. Содержит необходимые формулы, определения и ссылки на применяемые свойства, утверждения.	Пояснения частично отсутствуют или пояснения полные, но допущены арифметические или другого рода непринципиальные ошибки.	Решение имеет существенные и принципиальные ошибки (влияющие на логику кода), но содержит верную часть с пояснениями.	Решение неверное или отсутствует.

При подведении итогов учитываются результаты текущей аттестации. Полученные за текущую аттестацию баллы суммируются с баллами, полученными при прохождении промежуточной аттестации. При этом допускается получение студентами автоматической оценки (зачет) только по результатам работы в семестре:

При ответе на экзаменационный билет студент может получить следующие результаты по баллам за экзаменационный билет, с учетом накопительной системы баллов, полученных за текущий контроль, не включая баллы, набранные при выполнении теста.

При оценке знаний студента учитываются также:

- результаты текущего контроля;
- посещаемость учебных занятий;
- активность во время занятий;
- участие в научной работе;
- наличие навыков самостоятельной и исследовательской работы.



4.3. Результаты промежуточной аттестации и уровни сформированности компетенций

Код компетенции	Планируемые результаты обучения по дисциплине	Критерии оценивания			
		Отлично Высокий уровень освоения проверяемых компетенций	Хорошо Средний уровень освоения проверяемых компетенций	Удовлетворительно Базовый уровень освоения проверяемых компетенций	Неудовлетворительно Недостаточный уровень освоения проверяемых компетенций
УК-4	<p><i>Знать:</i> основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации.</p> <p><i>Уметь:</i> составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений.</p> <p><i>Владеть:</i> навыками вербальной коммуникации на техническом иностранном языке; навыками организации запросов в информационно-поисковых</p>	<p><i>Знает:</i> основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации.</p> <p><i>Умеет:</i> составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений.</p> <p><i>Владеет:</i> навыками вербальной коммуникации на техническом иностранном языке; навыками организации запросов в информационно-поисковых системах на</p>	<p><i>Знает:</i> основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации, но допускает несущественные ошибки.</p> <p><i>Умеет:</i> составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений, но допускает несущественные ошибки.</p> <p><i>Владеет:</i> навыками вербальной коммуникации на техническом иностранном языке, но</p>	<p><i>Знает:</i> в ограниченном объеме основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации.</p> <p><i>Умеет:</i> в ограниченном объеме составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений.</p> <p><i>Владеет:</i> в ограниченном объеме навыками вербальной коммуникации на техническом иностранном языке;</p>	<p><i>Не знает:</i> основные термины, употребляющиеся в сфере компьютерных технологий; технологии обмена информацией на основе интернет-сервисов; методы систематизации актуальной информации.</p> <p><i>Не умеет:</i> составлять тексты и сообщения с описанием технологических и программных характеристик разрабатываемых продуктов; осуществлять документирование программных решений.</p> <p><i>Не владеет:</i> навыками вербальной коммуникации на техническом иностранном языке; навыками организации запросов в информационно-поисковых</p>



	системах на русском и английском языках.	русском и английском языках.	допускает несущественные ошибки		системах на русском и английском языках.
ПК-2	<p><i>Знать:</i> методы и средства сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах; основы теории нейронных сетей.</p> <p><i>Уметь:</i> применять методы и средства сборки модулей и компонент программного обеспечения; проводить</p>	<p><i>Знает:</i> методы и средства сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах; основы теории нейронных сетей.</p> <p><i>Умеет:</i> применять методы и средства сборки модулей и компонент программного обеспечения; проводить исследования задач классификации с применением нейронных сетей.</p>	<p><i>Знает:</i> методы и средства сборки модулей и компонент программного обеспечения, о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах; основы теории нейронных сетей, но допускает несущественные ошибки.</p> <p><i>Умеет:</i> применять методы и средства сборки модулей и компонент программного обеспечения; проводить исследования задач классификации с применением нейронных сетей, но допускает несущественные ошибки.</p> <p><i>Владеет:</i> навыками создания программных интерфейсов;</p>	<p><i>Знает:</i> в ограниченном объеме методы и средства сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о международных и профессиональных стандартах информационных технологий; основы теории нейронных сетей.</p> <p><i>Умеет:</i> в ограниченном объеме применять методы и средства сборки модулей и компонент программного обеспечения; проводить исследования задач классификации с применением нейронных сетей.</p> <p><i>Владеет:</i> в ограниченном объеме навыками создания программных интерфейсов; оценка работоспособности и программного</p>	<p><i>Не знает:</i> методы и средства сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах; основы теории нейронных сетей.</p> <p><i>Не умеет:</i> применять методы и средства сборки модулей и компонент программного обеспечения; проводить исследования задач классификации с применением нейронных сетей.</p>



	<p>исследования задач классификации с применением нейронных сетей.</p> <p><i>Владеть:</i> навыками создания программных интерфейсов; оценки работоспособности программного продукта; навыком конструирования нейронных сетей.</p>	<p><i>Владеет:</i> навыками создания программных интерфейсов; оценки работоспособности и программного продукта; навыком конструирования нейронных сетей.</p>	<p>навыком конструирования нейронных сетей, но допускает несущественные ошибки.</p>	<p>продукта; навыком конструирования нейронных сетей.</p>	<p><i>Не владеет:</i> навыками создания программных интерфейсов; оценки работоспособности и программного продукта; навыком конструирования нейронных сетей.</p>
ПК-3	<p><i>Знать:</i> методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, виды систем поддержки принятия решений, вопросы практического использования экспертных и интеллектуальных информационных систем.</p>	<p><i>Знает:</i> методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, виды систем поддержки принятия решений, вопросы практического использования экспертных и интеллектуальных информационных систем.</p> <p><i>Умеет:</i> использовать программные средства разработки систем искусственного</p>	<p><i>Знает:</i> методы и средства проектирования программного обеспечения, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, но допускает несущественные ошибки.</p> <p><i>Умеет:</i> использовать программные средства разработки систем искусственного интеллекта; классифицировать решаемые задачи, но допускает несущественные ошибки</p> <p><i>Владеет:</i> навыками работы с базами данных и</p>	<p><i>Знает:</i> в ограниченном объеме методы и средства проектирования программного обеспечения, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, виды систем поддержки принятия решений, вопросы практического использования экспертных и интеллектуальных информационных систем.</p> <p><i>Умеет:</i> в ограниченном объеме и самостоятельно использовать программные средства разработки систем</p>	<p><i>Не знает:</i> методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, основные задачи, решаемые системами искусственного интеллекта, основные модели представления знаний, методы инженерии знаний, виды систем поддержки принятия решений, вопросы практического использования экспертных и интеллектуальных информационных систем.</p> <p><i>Не умеет:</i> использовать программные средства разработки систем искусственного</p>



	<p><i>Уметь:</i> использовать программные средства разработки систем искусственного интеллекта; классифицировать решаемые задачи, анализировать архитектуру экспертных систем.</p> <p><i>Владеть:</i> навыками работы с базами данных и базами знаний; быть способным оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств; навыки реализации простейших ЭС.</p>	<p>интеллекта; классифицировать решаемые задачи, анализировать архитектуру экспертных систем.</p> <p><i>Владеет:</i> навыками работы с базами данных и базами знаний; быть способным оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств; навыки реализации простейших ЭС.</p>	<p>базами знаний; способен оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств, но допускает несущественные ошибки</p>	<p>искусственного интеллекта; классифицировать решаемые задачи.</p> <p><i>Владеет:</i> в ограниченном объеме навыками работы с базами данных и базами знаний; способен на базовом уровне оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств; навыки реализации простейших ЭС.</p>	<p>интеллекта; классифицировать решаемые задачи, анализировать архитектуру экспертных систем.</p> <p><i>Не владеет:</i> навыками работы с базами данных и базами знаний; быть способным оценивать возможность применения конкретной ЭС для решения задач заданного класса; иметь навыки представления знаний с помощью инструментальных средств; навыки реализации простейших ЭС.</p>
--	---	--	--	--	---

Уровни сформированности компетенций определяются следующим образом:

1. Высокий уровень соответствует оценке “отлично” (“зачтено”), и предполагает:

- готовность к самостоятельной профессиональной деятельности;
- глубокое и правильное усвоение программного материала, последовательное, грамотное и логически стройное его изложение;
- владение основными методами и алгоритмами решения задач;
- умение строить математические модели, увязывать теорию с практикой, применять знания.



2. Средний уровень соответствует оценке “хорошо” (“зачтено”) и предполагает:

- твердое знание программного материала, его изложение грамотное и по существу;
- владение основными методами;
- отсутствие существенных ошибок, но затруднения в выводах и доказательствах;
- умение применять основные положения для решения задач.

3. Базовый уровень соответствует оценке “удовлетворительно” (“зачтено”), и предполагает:

- знания только основного материала, неумение делать выводы и проводить доказательства;
- ошибки, недостаточно правильные формулировки;
- трудное увязывание основных положений с практикой.

4. Низкий уровень соответствует оценке “неудовлетворительно” (“не зачтено”) и предполагает:

- незнание основополагающих вопросов изучаемого курса или значительной части программного материала;
- ошибки, неумение их исправлять;
- неумение увязать теорию с практикой.

