

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
прикладной математики и
информатики**

А.М. Райгородский

	Рабочая программа дисциплины (модуля)
по дисциплине:	Современные технологии разработки сервисов анализа данных
по направлению:	Прикладная математика и информатика
профиль подготовки:	А1360: Передовые методы искусственного интеллекта Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 8 (весенний) - Дифференцированный зачет

Аудиторных часов: 30 всего, в том числе:

лекции: 15 час.

семинары: 15 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 15 час.

Всего часов: 45, всего зач. ед.: 1

Программу составил: П.И. Ахтямов, старший преподаватель

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 12.02.2024

Аннотация

В этом курсе мы поговорим об аспектах, которые обычно приобретаются только на практике в процессе первой профессии в IT. Мы рассмотрим, как вести коммуникацию между сотрудниками посредством кода, автоматизируем рутинные задачи при помощи командной строки, научимся создавать окружение проекта. Немаловажным аспектом курса является повествование об автоматизации тестирования в сфере анализа данных. После окончания курса вы сможете построить автоматический конвейер сборки, тестирования проекта машинного обучения и подготовки использования этого проекта в рабочей среде.

1. Цели и задачи

Цель дисциплины

- изучение широкого спектра инструментария для разработки и поддержки приложений в сфере анализа данных;
- формирование базовых принципов разработки сервисов анализа данных;
- разбор принципов работы популярных фреймворков для реализации микросервисов искусственного интеллекта.

Задачи дисциплины

- углубленное изучение архитектур приложений машинного обучения;
- ознакомление с принципами контейнеризации приложений;
- изучение парадигм тестирования ПО;
- выработка методики воспроизводимого анализа данных;
- ознакомление с процессом командной интеграции в проектах анализа данных.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.3 Знает основные требования информационной безопасности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

- основные архитектуры для создания проектов машинного обучения;
- основы организации поставки конечного продукта для проектов искусственного интеллекта.

уметь:

- создавать инфраструктуру для работы проектов машинного обучения с последующим воспроизведением;
- реализовывать микросервисы, грамотно организовывать пересылку данных между сервисами;
- настраивать организацию рабочего процесса разработки в проектах машинного обучения.

владеть:

- навыками проектирования тест-кейсов для задач машинного обучения;
- навыком быстрого реагирования на экстренные ситуации по данным метрик из систем мониторинга;
- навыком отслеживания прогресса статуса проекта по системе контроля версий.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Командная строка, написание скриптов	1	1		1
2	Текстовые редакторы написания кода	1	1		1
3	Разработка интерфейсов командной строки	1	1		1
4	Системы контроля версий	1	1		1
5	Системы непрерывной интеграции и поставки	1	1		1
6	Основы сборки образов и контейнеризации	1	1		1
7	Основы работы в средах разработки	1	1		1
8	Тестирование: парадигмы, специфика для машинного обучения	1	1		1
9	Микросервисы	1	1		1
10	Системы оперативного реагирования для задач машинного обучения	2	2		2
11	Воспроизводимость экспериментов в машинном обучении	2	2		2
12	Проектирование архитектуры проектов искусственного интеллекта	2	2		2
Итого часов		15	15		15
Подготовка к экзамену		0 час.			

Общая трудоёмкость	45 час., 1 зач.ед.
--------------------	--------------------

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 8 (Весенний)

1. Командная строка, написание скриптов

Bash как скриптовый язык: переменные, условные переменные, циклы. Запуск скриптов, разграничение прав в Unix-like операционных системах.

2. Текстовые редакторы написания кода

Vim, Sublime Text. Автодополнение, сниппеты.

3. Разработка интерфейсов командной строки

Позиционные и непозиционные аргументы, переменные окружения. Виртуальные окружения.

4. Системы контроля версий

История. Отличие систем Git и SVN. Понятие ветвлений, фиксаций, репозитория. Процесс организации рабочего процесса в репозиториях.

5. Системы непрерывной интеграции и поставки

Основные принципы. Отличие между системами непрерывной интеграции.

6. Основы сборки образов и контейнеризации

Понятие образа, контейнера. Различие между виртуализацией и контейнеризацией. Основные инструменты контейнеризации.

7. Основы работы в средах разработки

Кодогенерация, статический анализ кода, удаленное управление ресурсами, отладка, профилировка.

8. Тестирование: парадигмы, специфика для машинного обучения

Пирамида тестирования + специфика для ML. V-model. Дизайн тестирования для ML.

9. Микросервисы

Основные понятия и фреймворки для их реализации. Прикладной уровень стека сетевого сообщения, протокол REST API. Пересылка метаданных по сети.

10. Системы оперативного реагирования для задач машинного обучения

Аналитика данных и системы мониторинга.

11. Воспроизводимость экспериментов в машинном обучении

Это открытая проблема в вычислительных науках, и она становится все более актуальной, поскольку все больше областей полагаются на результаты вычислений экспериментов. В этом разделе мы рассмотрим эту открытую проблему.

12. Проектирование архитектуры проектов искусственного интеллекта

Проектирование здания, разработка конструктивной модели на основе проекта или разработка способов построения сложной модели – все эти задачи уже содержат некоторую степень автоматизации. Недавние достижения в генеративном дизайне, анализе безопасности и 5D-планировании – это только первые намеки на то, какие алгоритмы высокой сложности и методы глубокого обучения, используемые в ИИ, можно внедрить в сферу проектирования и строительства.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная компьютерами и проектором.

6. Перечень рекомендуемой литературы

Основная литература

Фонд кафедры

1. Гленфорд М. Искусство тестирования программ/Гленфорд Майерс, Том Баджетт, Кори Сандлер /М.: Диалектика. – 2012.
2. Хамбл Д., Фарли Д. Непрерывное развертывание ПО: автоматизация процессов сборки, тестирования и внедрения новых версий программ. – Litres, 2019.
3. Мартин Р. Чистая архитектура. Искусство разработки программного обеспечения. – "Издательский дом "Питер", 2018.

Дополнительная литература

Фонд кафедры

1. Арнольд Роббинс: Bash. Карманный справочник системного администратора. Вильямс, 2017 г.
2. Крис Р. Микросервисы. Паттерны разработки и рефакторинга. – Издательский дом "Питер", 2019.
3. Дрю Нейл: Практическое использование Vim. ДМК-Пресс, 2017 г.
4. Моуэт Э. Использование Docker. – Litres, 2019.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не используются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

На занятиях используются мультимедийные технологии, включая демонстрацию презентаций. Для контроля и коррекции знаний обучающиеся могут использовать компьютерное тестирование.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Студент, изучающий дисциплину, должен с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания на практике.

Успешное освоение дисциплины требует:

- посещения студентом всех видов аудиторных занятий;
- ведения конспекта в ходе лекционных занятий;
- качественной самостоятельной подготовки к практическим занятиям, активной работы на них;
- активной самостоятельной и аудиторной работы студента;
- своевременной сдачи преподавателю заданий по аудиторным видам работ.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению: Прикладная математика и информатика
профиль подготовки: АІ360: Передовые методы искусственного интеллекта
Физтех-школа Прикладной Математики и Информатики
кафедра алгоритмов и технологий программирования
курс: 4
квалификация: бакалавр

Семестр, формы промежуточной аттестации: 8 (весенний) - Дифференцированный зачет

Разработчик: П.И. Ахтямов, старший преподаватель

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.3 Знает основные требования информационной безопасности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

2. Показатели оценивания компетенций

В результате изучения дисциплины «Современные технологии разработки сервисов анализа данных» обучающийся должен:

знать:

- основные архитектуры для создания проектов машинного обучения;
- основы организации поставки конечного продукта для проектов искусственного интеллекта.

уметь:

- создавать инфраструктуру для работы проектов машинного обучения с последующим воспроизведением;
- реализовывать микросервисы, грамотно организовывать пересылку данных между сервисами;
- настраивать организацию рабочего процесса разработки в проектах машинного обучения.

владеть:

- навыками проектирования тест-кейсов для задач машинного обучения;
- навыком быстрого реагирования на экстренные ситуации по данным метрик из систем мониторинга;
- навыком отслеживания прогресса статуса проекта по системе контроля версий.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

1. Напишите скрипт при помощи `bash` и `awk`, который строит гистограммы распределений по классам из датасета в формате `csv` файла.
2. Напишите скрипт, который будет параллельно выкачивать данные для большого датасета.

3. Внедрите в репозиторий проверку кода методами статического анализа перед фиксацией изменений.
4. Напишите сервис, который получает изображение по сети и возвращает список ключевых точек на изображении.
5. Настройте при помощи инструментов непрерывной интеграции проверку модели на переобучение на малом количестве данных.
6. Настройте при помощи инструментов непрерывной поставки пакет модели машинного обучения на языке программирования Python.
7. Протестируйте сервис подготовки признаков для сервиса обучения моделей машинного обучения.
8. Настройте систему уведомлений при падении микросервисов предсказания будущих прогнозов.
9. Воспроизводимость экспериментов в машинном обучении: фиксирование семени, сохранение конфигурации запуска эксперимента.
10. Проектирование архитектуры проектов искусственного интеллекта.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Командная строка, написание скриптов.
2. Текстовые редакторы написания кода: vim, sublime text, VS Code.
3. Разработка интерфейсов командной строки: ArgumentParser, getopt.
4. Системы контроля версий: git, svn. Git: основные команды: add, commit, push. Git: продвинутые команды: rebase, squash, reset.
5. Системы непрерывной интеграции и поставки: отличия. Примеры систем непрерывной интеграции и поставки.
6. Основы сборки образов и контейнеризации: Docker, rkt.
7. Основы работы в средах разработки: автодополнение, рефакторинг, профилировка, покрытие кода тестами
8. Тестирование: парадигмы, специфика для машинного обучения. Пирамида тестирования + специфика для ML. V-model. 1 Дизайн тестирования для ML.
9. Микросервисы: основные понятия и фреймворки для их реализации. Прикладной уровень стека сетевого сообщения, протокол REST API. Пересылка метаданных по сети.
10. Системы оперативного реагирования для задач машинного обучения: аналитика данных и системы мониторинга.
11. Примеры систем мониторинга: Grafana, Prometheus.

Критерии оценивания

Оценка "Отлично" (10) - полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы, код оформлен в едином удобочитаемом стиле.

Оценка "Отлично" (9) - полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы.

Оценка "Отлично" (8) - полностью и вовремя решены все задачи без ошибок. Продemonстрирован грамотный подход к решению задач.

Оценка "Хорошо" (7) - полностью решены все задачи. Допущены несущественные ошибки.

Оценка "Хорошо" (6) - полностью решено большинство задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Хорошо" (5) - полностью решено две трети задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Удовлетворительно" (4) - полностью решено более половины задач. В остальных задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Удовлетворительно" (3) - полностью решено более половины задач.

Оценка "Неудовлетворительно" (2) - решено менее половины задач.

Оценка "Неудовлетворительно" (1) - не решено ни одной задачи.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет может проводиться по итогам текущей успеваемости и сдачи заданий и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме.

При проведении устного дифференцированного зачета обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося не должен превышать одного астрономического часа.

Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины, а также справочной литературой, конспектами лекций или другими материалами.