ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«24» июня 2025 г.
Протокол № 2

Рабочая программа дисциплины (модуля)
«Production ML (Машинное обучение в продакшене)»

Направление подготовки: *02.04.01 Математика и компьютерные науки*

Направленность (профиль) подготовки: *Машинное обучение*

Квалификация (степень) выпускника: *магистр*

Форма обучения: *очная (с применением ДОТ)*

Срок освоения программы: *2 года*

Год набора: *2025*

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

Москва
2025

Содержание

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Production ML (Машинное обучение в продакшене)» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по специальности 02.04.01 Математика и компьютерные науки, профиль Машинное обучение, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

Изучение дисциплины (модуля) «Production ML (Машинное обучение в продакшене)» позволяет студентам получить практические навыки, необходимые для успешной интеграции машинного обучения в бизнес-процессы, что критически важно для достижения конкурентных преимуществ. Кроме того, оно способствует пониманию вызовов и решений, связанных с масштабированием и поддержкой ML-моделей в условиях реального времени.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Машинное обучение и входит обязательную часть Блока 1.

Дисциплина (модуль) изучается на 2 курсе в 3 семестре, доступна для прохождения при условии успешного завершения дисциплин (модулей) «Machine Learning (Машинное обучение)», «Основы промышленной разработки».

Цель изучения дисциплины (модуля): освоение студентами методов и практик внедрения, развертывания и поддержки моделей машинного обучения в реальных производственных системах.

Задачи изучения дисциплины (модуля):

- научиться обеспечивать повторяемость результатов экспериментов в машинном обучении через систематизацию процессов;
- освоить последовательность стадий создания и эксплуатации моделей машинного обучения в реальных условиях;
- разработать навыки организации хранения и контроля версий данных для эффективного обучения моделей;
- изучить различия в подходах к обучению и развертыванию моделей в оффлайн и онлайн режимах;
- приобрести умения по адаптации моделей машинного обучения для работы в масштабируемых системах.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- как обеспечить воспроизводимость в машинном обучении;
- основные этапы жизненного цикла моделей;
- как устроен процесс управления данными для обучения, включая версионирование и обеспечение качества;
- отличия между оффлайн и онлайн моделями и как планировать задачи для их обучения и внедрения;
- как подготовить ML модели к масштабированию.

уметь:

- организовывать трекинг кода и результатов экспериментов;
- работать с пайплайнами обработки данных и обучения моделей;
- деплоить модели;
- автоматизировать процесс обучения и деплоя модели;
- разрабатывать тесты для проверки качества моделей и настраивать мониторинг для контроля их работы.

владеть:

- навыком выстраивания MLOps процесса (внедрение и поддержка работы ML моделей в продакшне);
- навыком работы с MLOps инструментами.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
ОПК-1.	Способен находить, формулировать и решать актуальные и значимые проблемы прикладной и компьютерной математики	ОПК-1.1.	Знает основные методы и подходы к решению задач прикладной и компьютерной математики, включая алгоритмы, математическое моделирование и теорию оптимизации, а также современные инструменты и технологии, используемые в этой области
		ОПК-1.2.	Умеет анализировать и формулировать математические задачи, применять соответствующие методы и алгоритмы для их решения, а также интерпретировать и представлять результаты в понятной и доступной форме
		ОПК-1.3.	Имеет практический опыт работы над проектами или исследованиями в области прикладной и компьютерной математики, включая участие в конкурсах, олимпиадах или научных публикациях, где были решены актуальные и значимые задачи
ОПК-3.	Способен самостоятельно создавать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов, в том числе отечественного производства	ОПК-3.1.	Знает основные принципы программирования, архитектуры программного обеспечения и современные языки программирования, а также особенности отечественных информационных технологий и сетевых ресурсов
		ОПК-3.2.	Умеет разрабатывать прикладные программные средства, используя современные инструменты и технологии, а также интегрировать их с сетевыми ресурсами для решения конкретных задач
		ОПК-3.3.	Имеет практический опыт разработки программных средств, используемых при

			построении математических моделей в естественных науках
ПК-1.	Способен определять общие формы и закономерности области машинного обучения	ПК-1.1.	Знает основные теоретические концепции и принципы, относящиеся к области машинного обучения, а также ключевые закономерности и модели, которые помогают в анализе и интерпретации данных
		ПК-1.2.	Умеет проводить систематический анализ области разработки, выявлять и формулировать общие закономерности и тенденции, а также применять методы исследования для получения новых знаний и понимания
		ПК-1.3.	Имеет практический опыт работы в области машинного обучения, включая участие в научных проектах, исследованиях или практических заданиях, где были выявлены и описаны общие формы и закономерности
ПК-3.	Способен решать задачи профессиональной деятельности, формулировать результат, увидеть следствия полученного результата	ПК-3.1.	Знает основные принципы и методы решения задач профессиональной деятельности, а также способы формулирования и представления результатов, включая анализ последствий и их значимость в контексте проекта
		ПК-3.2.	Умеет применять математические и компьютерные методы для решения конкретных задач, формулировать четкие и обоснованные результаты, а также анализировать их последствия для дальнейших действий и решений
		ПК-3.3.	Имеет практический опыт в решении профессиональных задач, включая участие в проектах, где были получены результаты и проанализированы их следствия, что способствовало принятию обоснованных решений

ПК-6.	Способен разрабатывать программное обеспечение для решения прикладных задач в сфере машинного обучения	ПК-6.1.	Знает основные языки программирования, методы разработки программного обеспечения, а также принципы проектирования и архитектуры программных систем, применяемых в машинном обучении
		ПК-6.2.	Умеет анализировать прикладные задачи, разрабатывать алгоритмы и реализовывать их в виде программного обеспечения, используя современные инструменты и технологии, а также проводить тестирование и отладку созданных решений
		ПК-6.3.	Имеет практический опыт разработки программного обеспечения в рамках реальных проектов, включая участие в командах, где были успешно реализованы решения для конкретных прикладных задач в сфере профессиональной деятельности

3. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	ML в индустрии. Базовые инструменты ML инженера	4	4		17	Домашние задания
2	Работа с данными в ML и трекинг ML экспериментов	4	4		17	Домашние задания
3	Подготовка ML моделей к деплою	4	4		17	Домашние задания
4	Тестирование и мониторинг	8	8		32	Домашние задания, Стресс-тест
5	Автоматизация обучения и деплоя ML моделей	4	4		17	Домашние задания
6	LLMOps	2	2		9	Домашние задания
7	Контейнеризация ML-приложений	4	4		17	Домашние задания
	Зачет с оценкой			4		Проект
	Итого:	***30***	***30***	***4***	***126***	
Объем дисциплины (модуля) (в ак. ч.)		***190***				
Объем дисциплины (модуля) (в зач. ед.)		***5***				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	ML в индустрии. Базовые инструменты ML инженера	Введение. Инструменты разработки. Часть 1 Инструменты разработки. Часть 2
2	Работа с данными в ML и трекинг ML экспериментов	Работа с данными в ML Трекинг ML-экспериментов и воспроизводимость
3	Подготовка ML моделей к деплою	Деплой ML-моделей. Часть 1 Деплой ML-моделей. Часть 2
4	Тестирование и мониторинг	Тестирование и валидация кода Мониторинг. Часть 1: стандартные инструменты Предзащита проекта Мониторинг. Часть 2: ML-специфичные инструменты
5	Автоматизация обучения и деплоя ML моделей	Автоматизация пайплайна обучения моделей CI/CD для ML
6	LLMOps	LLMOps
7	Контейнеризация ML-приложений	Контейнеризация ML-приложений

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Лакшманан, В. Машинное обучение. Паттерны проектирования : практическое пособие / В. Лакшманан, С. Робинсон, М. Мунн. - Санкт-Петербург : БХВ-Петербург, 2022. - 448 с. - ISBN 978-5-9775-6797-8. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2140204.

2. Григорьев, А. Машинное обучение. Портфолио реальных проектов : практическое руководство / А. Григорьев. - Санкт-Петербург : Питер, 2023. - 496 с. - (Серия «Библиотека программиста»). - ISBN 978-5-4461-1978-3. - Текст : электронный. - URL: https://znanium.com/catalog/product/2123375.

Дополнительная литература:

1. Кацов, И. Машинное обучение для бизнеса и маркетинга : практическое руководство / И. Кацов. - Санкт-Петербург : Питер, 2019. - 512 с. - (Серия «IT для бизнеса»). - ISBN 978-5-4461-0926-5. - Текст : электронный. - URL: https://znanium.com/catalog/product/1783938.

2. Гифт, Н. Прагматичный ИИ. Машинное обучение и облачные технологии : практическое руководство / Н. Гифт. - Санкт-Петербург : Питер, 2019. - 304 с. - (Серия «Для профессионалов»). - ISBN 978-5-4461-1061-2. - Текст : электронный. - URL: https://znanium.com/catalog/product/1760806.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

— столами и стульями;

— компьютерной техникой;

— механическими калькуляторами;

— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья,

оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование"	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		

Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Production ML (Машинное обучение в продакшене)» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, практические занятия, домашние задания, стресс-тест, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Практическое занятие — это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к практическому занятию рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Стресс-тест — это способ проверить, как система ведёт себя в экстремальных условиях, которые значительно превышают обычную нагрузку или выходят за рамки обучающих сценариев.

Проект – исследовательская работа по дисциплине (модулю) и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

Бонусные баллы — это оценки, которые студенты могут получить за выполнение дополнительных заданий.

Формат бонусных баллов позволяет студентам улучшить общую оценку по дисциплине (модулю) и стимулирует углубленное изучение материала.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Production ML (Машинное обучение в продакшене)»

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***зачета с оценкой,*** при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Production ML (Машинное обучение в продакшене)» оценивается следующим образом:

Активность	Вес	Описание
Домашние задания	60%	Набор задач по темам недели
Стресс-тест	10%	Способ проверки, как система ведёт себя в экстремальных условиях
Зачет с оценкой	30%	Проект – исследовательская работа по дисциплине (модулю) и презентация результатов

В рамках изучения дисциплины (модуля) возможно получение бонусных баллов.

Формула расчёта итоговой оценки по дисциплине (модулю) «Production ML (Машинное обучение в продакшене)»: «0,6 × среднее за домашние задания + 0,1 × тест + 0,3 × зачет с оценкой».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание 1.

Шаг 1. Поднять сервис Evidently AI (1 балл)

В файл docker-compose.yml добавить сервис evidently на порт 8502.

Шаг 2. Добавить создание репортов в train.py (6 баллов)

Написать функцию log_evidenlty(), выполняющую следующее:

- Проверяет наличие проекта в сервисе с названием EV_PROJECT_NAME = “week-10”
- Если проекта нет - создает его.
- Принимает train_df, val_df, создает из них eval’ы (Dataset.from_pandas(...)), описывает с помощью DataDefinition.
- Добавляет в данные признаки (с помощью дескрипторов - длина сообщений и наличие в них слов “bot” и “бот”)
- Создает Report с DataDriftPreset’ом
- Загружает 1 репорт в сервис (при каждом запуске train.py - новый репорт).

Шаг 3. Добавить дэшборд по репортам (3 балла)
Создать дэшборд типа “Line” и size=”full”, визуализирующий Drift Score колонки “text” из репортов (подумать, на каком шаге он должен быть создан).

Как выполнить и сдать задание

В этом задании вы не сдаете ничего в ЛМС, а все проверки вашего домашнего задания осуществляются через ваш репозиторий.

Что необходимо для сдачи задания:

- Используйте для работы и запуска проверки репозиторий из self-service по ссылке:

https://self-service.culab.ru/setup/production-ml

- **Структура репозитория:**
- Обновите docker-compose.y(a)ml, добавив сервис evidently
- Обновите файл train.py
- Убедитесь, что все необходимые и **прошлые** переменные окружения (секреты) добавлены в Gitlab CI/CD Variables:

CI / CD Variables (добавить в Settings → CI/CD → Variables)

Переменная	Значение
EVIDENTLY_PORT	8502
EVIDENTLY_HOST	evidently

Домашнее задание 2.

Шаг 1 (0 балл). Подготовка инфраструктуры

1. На вашей виртуальной машине должны быть установлены Docker и Docker Compose
2. Выберите инструмент для хостинга LLM, доступный в форме Docker-контейнера (например, llama.cpp, text-generation-inference, vLLM и т. д.).
3. Убедитесь, что у вашей ВМ достаточно ресурсов (оперативной памяти и CPU) для выбранной модели, иначе подберите модель поменьше.
4. С помощью docker pull загрузите контейнер инструмента на ВМ.

Шаг 2 (5 баллов). Запуск модели

1. **Выбор модели**: выберите LLM (весом и объёмом, подходящим для вашей ВМ). Загрузите файл модели, поместив его либо на саму ВМ, либо в S3-minio хранилище.
2. **Соберите или возьмите готовый Docker-образ** с выбранным инструментом и запустите модель внутри контейнера, далее поднимите сервис через docker-compose:

- **Обеспечьте**, чтобы модель принимала запросы по OpenAI-like протоколу (см. документацию OpenAI). Обычно инструменты LLM (TGI, vLLM, llama.cpp и т. д.) предоставляют эндпоинты совместимые с POST /v1/chat/completions.
- **Ограничьте** использование CPU и RAM, чтобы исключить перегрузку вашей ВМ. В docker-compose.yaml это можно сделать, задав ресурсы контейнера, используя флаг --compatibility при запуске docker-compose, например

```
version: "3.*"
services:
  gpt_service:
    build:
      context: .
      dockerfile: Dockerfile
    deploy:
          resources:
            limits:
```

12. cpus: "4.0"

memory: 8g

(можете задавать ограничения на количество CPU и памяти — на ваше усмотрение, главное, чтобы ВМ не упала).

Шаг 3 (2 балла). API-key защита

1. **Настройка**

- Модель должна быть доступна по порту PORT=38300.
- Установите переменную окружения API_KEY=cu_prodml_hw_week13. При обращении к API с неправильным ключом доступ должен блокироваться: "Authorization: Bearer $OPENAI_API_KEY"
- Убедитесь, что при запуске контейнера ваш сервис слушает **localhost:38300**.

1. **Проверка**

- Попробуйте curl-запрос вида ниже и проверьте, что получаете валидный JSON-ответ, содержащий сгенерированный текст.

```
curl http://localhost:38300/v1/chat/completions \
  -H "Authorization: Bearer cu_prodml_hw_week13" \
  -H "Content-Type: application/json" \
  -d &apos;{
    "model": "model_identifier",
   "messages": [{"role":"user","content":"Hello!"}] }&apos;
```

Если у вас сервис по умолчанию не слушает /v1, попробуйте без этого суффикса (всё зависит от конкретного фреймворка).

Важно: на шаге 3 мы будем подключать вашу локальную LLM к вашему же боту, так что формат и порт здесь должны оставаться неизменными, чтобы грейдер мог это протестировать.

Шаг 4 (3 балла). Апгрейд бота

1. В предыдущих заданиях вы работали с сервисом классификации сообщений. Теперь необходимо использовать сервис генерации сообщений из репозитория, далее именуемый как *бот*. Его нужно обновить так, чтобы вместо обращения к OpenAI ваш бот ходил **к локальной модели** на порту **38300**.

- Для этого в коде бота нужно понять, как изменить переменные конфигурации OPEN_AI_API_KEY, PROXY_URL, API_PORT, а также какой код добавить, чтобы использовался ваш сервис, например http://gpt_service:38300 (для этого всё должно быть в одной docker-compose-сети).
- Используйте тот же ключ Bearer cu_prodml_hw_week13 при обращении.

1. **Убедитесь**, что ваш бот успешно получает ответы из локальной LLM. Возможно, придётся подстроить JSON-схему ответа.

2. README.md:

- Опишите, какую модель выбрали, какой фреймворк для её развёртывания, почему, и как её задеплоили (docker-compose, ограничение CPU, переменные окружения).
- Укажите точные шаги запуска (если это что-то отличное от стандартного docker-compose up -d).
- Опишите, как ваш бот обращается к локальной LLM через API.

Как выполнить и сдать задание

В этом задании вы не сдаете ничего в ЛМС, а все проверки вашего домашнего задания осуществляются через ваш репозиторий. Для корректной работы автоматической проверки используйте **тот же самый** репозиторий, который вы создавали в первом домашнем задании через self-service.

Что необходимо для сдачи задания:

- **Структура репозитория:**
- В docker-compose.yaml добавьте сервис, запускающий LLM
- Убедитесь, что порт **38300** для LLM не конфликтует с другими сервисами (такими как MLflow, Minio и др.).
- В коде вашего бота замените обращения к OpenAI на вызов http://gpt_service:38300 с нужными заголовками.
- Ответ сервиса будет проверяться на соответствие структуре ответов OpenAI — ваш сервис должен отвечать в нужном формате.
- **Переменные CI/CD**:
- Оставьте все ранее добавленные переменные.
- Добавьте переменную GPT_SERVICE_PORT — порт на котором находится ваш бот.
- Если решите использовать другой ключ, отличный от **cu_prodml_hw_week13**, то обязательно добавьте переменною API_KEY.
- Задайте новый путь до проекта, если он изменился.

Домашнее задание 3.

Шаг 1. Практика по kubectl (10 баллов)

Подготовка:

1. Изучите пример манифеста manifest.yaml из материалов занятия
2. Поднимите локально или на виртуалке minikube и примените этот манифест

```
kubectl apply -f manifest.yaml
```

В своём репозитории подготовьте файл kuber_practice.py (в корне репозитория) и пропишите в нём **по одной команде** kubectl в соответствующую переменную (без |, &&, || и т. д.), **решая следующие пункты**:

1. Выведите все поды в неймспейсе my-namespace для деплоймента app1.

2. Выведите все поды **во всех** неймспейсах для деплоймента app1.
3. Выведите информацию о деплойменте app3 в неймспейсе my-namespace.
4. Выведите логи деплоймента app2 в неймспейсе my-namespace.
5. Увеличьте число реплик у деплоймента app1 в неймспейсе my-namespace до 4.
6. Верните обратно число реплик у деплоймента app1 в неймспейсе my-namespace до 1.
7. Удалите все поды у деплоймента app1 в неймспейсе my-namespace.
8. Сделайте деплоймент app1 доступным извне, чтобы сервис отвечал по порту **33888**.

Примерное описание задания к стресс-тесту

Цель задания:

В этом задании будет производиться тестирование вашего сервиса. Его цель – проверить, насколько ваш сервис устойчив, надежен и готов к выкатке в production-среду. Мы будем имитировать различные виды атак и нештатных ситуаций, которые могут ждать ваш сервис: некорректные запросы, высокий трафик и другие. Ваша задача – сделать сервис максимально устойчивым к различным сценариям поведения его пользователей, используя те инструменты, которые вы изучили в рамках данного курса (использование дополнительных инструментов также приветствуется). Попутно вы научитесь диагностировать проблемы в вашем сервисе, используя логи и мониторинги, настроенные в прошлых заданиях.

По итогам успешного прохождения этого задания ваш сервис станет значительно более надежным и готовым к эксплуатации в production-среде.

Примеры проблем, с которыми может столкнуться ваш сервис:

1. **Некорректные запросы.** Клиенты вашего сервиса – другие разработчики или сервисы – могут допускать ошибки в следовании спецификации вашего API. Ваш сервис должен корректно реагировать на такие запросы и не падать с ошибками.
2. **Аномально высокая нагрузка.** Ваш сервис должен быть готов к резкому росту нагрузки (например, к вашему API подключили новый сервис или кто-то рассказал о нем в своем телеграм-канале). Ожидается, что ваш сервис останется доступным во время и после всплеска нагрузки.
3. **Неожиданные данные.** Что произойдет, если кто-то по ошибке отправит в ваш сервис вместо текста сообщения текст всех диалогов пользователя, которых может быть очень много?
4. **Вредоносные запросы.** Некоторые запросы могут быть не просто ошибочными, а намеренно неправильными, чтобы найти уязвимости в вашем сервисе. Ваша цель – предвидеть такие ситуации и не оставить злоумышленникам возможность влиять на компоненты вашего сервиса произвольным образом.
5. И другие неожиданные проблемы.

Как находить причины проблем:

Основные инструменты для вас – это логи и мониторинги сервиса и ВМ:

- логирование: убедитесь, что ваш сервис логирует входящие запросы, ответы, коды ошибок и любые исключения, которые могут возникнуть внутри него;
- мониторинг: используйте наработки из прошлого задания, чтобы состояние и поведение вашего сервиса были максимально прозрачными для вас;
- состояние ВМ: используйте инструменты командной строки (например, htop) и возможности Yandex Cloud, чтобы отслеживать состояние вашей ВМ.

Как будет оцениваться задание:

Задание состоит из 10 тестов, проверяющих поведение вашего сервиса в различных сценариях. За каждый успешно пройденный тест можно получить 1 балл.

Примерное описание задания и критерии оценивания к проекту

Критерии оценки работы на защите:

1. Техническое качество работы классификатора (latency, uptime, % of code 200 responses, etc)
2. ML качество классификатора (roc-auc)
3. Репозиторий (в ридми описано как разрабатывать, деплоить и поддерживать вашу ML систему; весь код системы закомичен)
4. Презентация на защите

Баллы за пункты 1-2 будут выдаваться линейно (есть некоторый адекватный threshold по метрикам, при преодолении которого будет выставляться полный балл).

- Latency: 0 баллов при 0.5s, полный балл при 0.0s
- Uptime: 0 баллов при 95%, полный балл при 100%
- % of code 200 responses: 0 баллов при 95%, полный балл при 100%
- 0 баллов при пороге ниже sample_submission.csv, полный балл первому месту, далее линейно между 1м и последним местом. Для получения баллов необходимо поменять свое team_name на PROD_ML_FALL_25|LASTNAME1_LASTNAME2, где LASTNAME1, LASTNAME2 фамилии участников команды.

На дисциплине мы учимся не только деплоить ML системы, но и обеспечивать их надежную работу. Поэтому количество баллов за пункты выше будет умножено на min(lifetime, 7days) / 7days. Например, если ваш классификатор был зарегистрирован в системе только 3.5 дня до защиты, то вы получите 50% баллов от ваших результатов. Если ваш классификатор был зарегистрирован в системе 7 дней до дня защиты, вы получите 100% баллов.

О чем нужно рассказать в презентации **(пункт 4)**:

1. Архитектура вашего решения, почему выбраны именно такие инструменты, именно так организован проект. Что изменилось в архитектуре со времени предзащиты?
2. Какие действия предприняли, чтобы обеспечить первые три критерия для защиты?
3. Как бы вы развивали проект дальше, если бы у нас был второй семестр Production ML? Что бы вы переделали, какой техдолг пофиксили, что добавили нового и от чего отказались.
4. Командная работа – вклад каждого участника в командную работу, кто какие домашние задания делал.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какой из следующих инструментов чаще всего используется для разработки ML моделей? A) Excel B) TensorFlow C) Microsoft Word D) Notepad	B	ОПК-1
2.	Какова основная роль ML инженера?	C	ОПК-1

	A) Разработка веб-сайтов B) Оптимизация баз данных C) Создание и внедрение ML моделей D) Администрирование серверов		
3.	Какое из следующих направлений не является областью применения машинного обучения в бизнесе? A) Обработка естественного языка B) Анализ финансовых данных C) Уборка помещений D) Рекомендательные системы	C	ПК-1
4.	Какой метод используется для очистки данных от выбросов? A) Нормализация B) Удаление дубликатов C) Стандартизация D) Логарифмирование	B	ПК-3
5.	Какой из следующих инструментов используется для визуализации данных? A) NumPy B) Pandas C) Matplotlib D) Scikit-learn	C	ПК-6
6.	Назовите одну из основных областей применения машинного обучения в бизнесе.	Рекомендательные системы	ОПК-3
7.	Как называется роль специалиста, который отвечает за внедрение ML моделей в продакшен?	ML инженер	ОПК-3
8.	Какой процесс включает в себя удаление ненужных или ошибочных данных?	Очистка данных	ОПК-1
9.	Какой тип графика часто используется для визуализации распределений данных?	Гистограмма	ОПК-1
10.	Какой метод используется для преобразования категориальных данных в числовые?	Кодирование	ОПК-1
11.	Как называется система, используемая для управления экспериментами в ML?	Система трекинга	ПК-1
12.	Какой метрикой часто оценивается качество классификационной модели?	Точность/accuracy	ПК-1
13.	Какой формат часто используется для сохранения обученных моделей?	Pickle	ПК-3
14.	Какой фреймворк часто используется для создания API для ML моделей?	Flask	ПК-3
15.	Как называется процесс автоматизации развертывания и обучения моделей?	CI/CD	ПК-3
16.	Как называется этап, на котором модель готовится к работе в продакшене?	деплой	ПК-1

17.	Чем обеспечивается управление версиями данных в ML проектах?	версионирование	ПК-1
18.	Какой тип пайплайна объединяет обучение и деплой модели?	автоматизация	ПК-6
19.	Как называется процесс упаковки ML-приложения для деплоя?	контейнеризация	ПК-6
20.	Какой процесс позволяет контролировать работу модели после запуска?	мониторинг	ПК-6