

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ястребов Олег Александрович

Должность: Ректор

Дата подписания: 25.05.2026 12:25:52

Уникальный программный ключ:

ca953a01204891083f939673078ef1a989dae18a

Федеральное государственное автономное образовательное учреждение высшего образования

«Российский университет дружбы народов имени Патриса Лумумбы»

Факультет искусственного интеллекта

(наименование основного учебного подразделения (ОУП)-разработчика ОП ВО)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ПРАКТИКУМ ПО ОБРАБОТКЕ ЕСТЕСТВЕННОГО ЯЗЫКА (NLP)

(наименование дисциплины/модуля)

Рекомендована МССН для направлений подготовки:

**02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ;**

09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

(код и наименование направления подготовки/специальности)

Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: РАЗРАБОТКА И ОБУЧЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

(наименование (профиль/специализация) ОП ВО)

2026 г.

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Практикум по обработке естественного языка (NLP)» входит в программу бакалавриата «Искусственный интеллект: разработка и обучение интеллектуальных систем» по направлениям подготовки 02.03.02 Фундаментальная информатика и информационные технологии и 09.03.03 Прикладная информатика, и изучается в 6, 7, 8 семестрах 3, 4 курсов. Дисциплину реализует Кафедра прикладного искусственного интеллекта. Дисциплина состоит из 9 разделов и 81 тема и направлена на изучение полного цикла проектирования, разработки и эксплуатации промышленных NLP-систем: анализа требований заказчика и обследования предметной области, проектирования архитектуры NLP-пайплайна (выбор моделей, стратегия fine-tuning, инфраструктура), организации сбора и аннотирования текстовых данных, итеративной разработки и дообучения языковых моделей, тестирования и валидации (функциональное, нагрузочное, приёмочное), оптимизации и развёртывания (контейнеризация, CI/CD, мониторинг), а также практик командной работы (Agile/Scrum, Git workflow, код-ревью, документирование) в контексте многосеместрового проектного цикла.

Целью освоения дисциплины является формирование у студентов навыков командной проектной работы над промышленной NLP-системой в течение трёх семестров: от анализа требований и проектирования архитектуры до развёртывания, мониторинга и передачи заказчику, включая способность декомпозировать задачу, распределять роли в команде, управлять backlog-ом и спринтами, обеспечивать качество кода и данных, формулировать техническую документацию, проводить приёмочное тестирование NLP-систем, а также рефлексировать над процессом и результатами проекта.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Практикум по обработке естественного языка (NLP)» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.2 Умеет анализировать альтернативные варианты решений для достижения намеченных результатов; разрабатывать план, определять целевые этапы и основные направления работ; УК-2.3 Владеет методиками разработки цели и задач проекта; методами оценки продолжительности и стоимости проекта, а также потребности в ресурсах;
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	УК-3.1 Знает типологию и факторы формирования команд, способы социального взаимодействия; УК-3.2 Умеет действовать в духе сотрудничества; принимать решения с соблюдением этических принципов их реализации; проявлять уважение к мнению и культуре других; определять цели и работать в направлении личностного, образовательного и профессионального роста; УК-3.3 Владеет навыками распределения ролей в условиях командного взаимодействия; методами оценки своих действий, планирования и управления временем;
УК-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе	УК-6.2 Умеет демонстрировать умение самоконтроля и рефлексии, позволяющие самостоятельно корректировать обучение по выбранной траектории;

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
	принципов образования в течение всей жизни	
ПК-1	Способен анализировать требования к программному обеспечению систем ИИ, разрабатывать технические спецификации и техническое задание на систему	ПК-1.3 Разрабатывает ТЗ на систему с элементами ИИ, проводит обследование текущей ситуации и выявление требований;
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	ПК-2.1 Проектирует архитектуру ИС с компонентами ИИ, выбирает архитектурные паттерны и технологический стек;
ПК-3	Способен разрабатывать и реализовывать стратегии тестирования и контроля качества программного обеспечения систем ИИ	ПК-3.2 Разрабатывает план тестирования и организационные документы для тестирования ПО систем ИИ;
DL-1	Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	DL-1.9 Разрабатывает, адаптирует и внедряет трансформерные архитектуры для решения задач обработки последовательностей, включая создание новых моделей, оптимизацию обучения и промышленное развертывание; DL-1.12 Применяет, адаптирует и разрабатывает методы дообучения нейронных сетей для эффективной адаптации моделей к новым задачам и доменам;
DL-4	Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка	DL-4.2 Определяет стек технологий, методов и алгоритмов для построения продуктов с обработкой естественного языка (диалоговые системы, вопросно-ответные системы, рекомендательные системы и т.д.); DL-4.3 Имплементирует известные алгоритмы, архитектуры и модели обработки естественного языка на реальных данных, строит пайплайны обучения моделей и развертывания NLP-сервисов в продуктивной среде;
FC-2	Способен проводить передовые исследования в области фундаментальных и генеративных моделей	FC-2.1 Исследует и разрабатывает большие языковые модели (LLM) и другие модели для символьных данных; FC-2.3 Исследует и создает мультимодальные большие языковые модели (LLM);
SS-1	Способен учитывать философские, когнитивные и социальные основания концепций ИИ в профессиональной деятельности	SS-1.2 Применяет методики работы с этическими и социальными рисками, возникающими на разных стадиях жизненного цикла ИИ;
SS-2	Способен к эффективной коммуникации и командной работе в междисциплинарных проектах в области ИИ	SS-2.1 Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы в контексте гибридной команды "Человек+ИИ", включая постановку задач людям и ИИ-агентам, фиксацию договоренностей и критериев качества; SS-2.2 Учитывает профессиональные и ролевые особенности коллег и контур ИИ-компонентов: адаптирует язык под аудиторию (tech/product/C-level), распределяет ответственность (RACI) и представляет результаты в понятном формате;
SS-3	Способен к критическому анализу, метарефлексии и переносу знаний при работе с системами ИИ	SS-3.2 Определяет релевантность применения ИИ для решения конкретных задач, анализирует поведение ИИ в техническом, социальном и правовом контекстах, переносит идеи и методы за пределы исходной предметной области;

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Практикум по обработке естественного языка (NLP)» относится к блоку по выбору блока образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Практикум по обработке естественного языка (NLP)».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	Философия; Искусственный интеллект и когнитивная психология; Этика и безопасность использования искусственного интеллекта;	
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	Эксплуатационная практика (учебная); Правоведение; Этика и безопасность использования искусственного интеллекта; Методы разработки решений на основе искусственного интеллекта (Git, Docker);	
УК-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	<i>Основы программирования HTML - CSS - JavaScript**;</i> <i>Основы программирования на языке NodeJS**;</i> <i>Основы программирования на языке Go**;</i> <i>Основы программирования на языке Julia**;</i>	
ПК-1	Способен анализировать требования к программному обеспечению систем ИИ, разрабатывать технические спецификации и техническое задание на систему	<i>Правоведение;</i> <i>Параллельное и распределенное программирование;</i> <i>Введение в искусственный интеллект;</i> <i>Искусственный интеллект и когнитивная психология;</i> <i>Этика и безопасность использования искусственного интеллекта;</i> <i>Методы машинного обучения;</i> <i>История и теория программирования;</i> <i>Программирование на языке C++;</i> <i>Методы разработки решений на основе искусственного интеллекта (Git, Docker);</i>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
		<i>Введение в базы данных; Онтология и графы знаний; Эксплуатационная практика (учебная); Технологическая (проектно-технологическая) практика (учебная);</i>	
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	<i>Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная); Программирование на языке C++; Параллельное и распределенное программирование; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Алгоритмы и структуры данных; Hadoop, SPARK; Программирование на языке Python; Введение в базы данных; Онтология и графы знаний;</i>	
ПК-3	Способен разрабатывать и реализовывать стратегии тестирования и контроля качества программного обеспечения систем ИИ	<i>Теория вероятностей и математическая статистика; Этика и безопасность использования искусственного интеллекта; Статистические методы и первичный анализ данных; Методы машинного обучения; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта; Анализ естественного языка с помощью методов искусственного интеллекта; Программирование на языке Python; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная);</i>	
SS-1	Способен учитывать философские, когнитивные и социальные основания концепций ИИ в профессиональной деятельности	<i>Философия; История и теория программирования; Введение в искусственный интеллект; Искусственный интеллект и когнитивная психология;</i>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
		<p><i>Онтология и графы знаний;</i> <i>Методы машинного обучения;</i> <i>Лингвистические основы анализа естественного языка;</i> <i>Этика и безопасность использования искусственного интеллекта;</i></p>	
SS-2	<p>Способен к эффективной коммуникации и командной работе в междисциплинарных проектах в области ИИ</p>	<p><i>Программирование на языке Python;</i> <i>Методы разработки решений на основе искусственного интеллекта (Git, Docker);</i> <i>Иностранный язык**;</i> <i>Русский язык (как иностранный)**;</i> <i>Иностранный язык в профессиональной деятельности**;</i> <i>Русский язык (как иностранный) в профессиональной деятельности**;</i> <i>Технологическая (проектно-технологическая) практика (учебная);</i> <i>Эксплуатационная практика (учебная);</i></p>	
SS-3	<p>Способен к критическому анализу, метарефлексии и переносу знаний при работе с системами ИИ</p>	<p><i>Эксплуатационная практика (учебная);</i> <i>Теория вероятностей и математическая статистика;</i> <i>Искусственный интеллект и когнитивная психология;</i> <i>Этика и безопасность использования искусственного интеллекта;</i> <i>Статистические методы и первичный анализ данных;</i> <i>Методы машинного обучения;</i> <i>Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i> <i>Анализ естественного языка с помощью методов искусственного интеллекта;</i> <i>Правоведение;</i> <i>Введение в искусственный интеллект;</i> <i>Введение в компьютерное зрение;</i> <i>Основы программирования HTML - CSS - JavaScript**;</i> <i>Основы программирования на языке NodeJS**;</i> <i>Основы программирования на языке Go**;</i> <i>Основы программирования на языке Julia**;</i> <i>Философия;</i></p>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
DL-1	Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	<i>Анализ естественного языка с помощью методов искусственного интеллекта; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i>	
DL-4	Способен применять и (или) разрабатывать алгоритмы, методы и технологии обработки естественного языка	<i>Лингвистические основы анализа естественного языка; Анализ естественного языка с помощью методов искусственного интеллекта;</i>	
FC-2	Способен проводить передовые исследования в области фундаментальных и генеративных моделей	<i>Онтология и графы знаний; Лингвистические основы анализа естественного языка; Анализ естественного языка с помощью методов искусственного интеллекта; Введение в компьютерное зрение; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i>	

* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

** - элективные дисциплины /практики

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Практикум по обработке естественного языка (NLP)» составляет «11» зачетных единиц.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.		Семестр(-ы)		
			6	7	8
<i>Контактная работа, ак.ч.</i>	162		68	52	42
Лекции (ЛК)	0		0	0	0
Лабораторные работы (ЛР)	0		0	0	0
Практически/семинарские занятия (СЗ)	162		68	52	42
<i>Самостоятельная работа обучающихся, ак.ч.</i>	153		85	65	3
<i>Контроль (экзамен/зачет с оценкой), ак.ч.</i>	81		27	27	27
Общая трудоемкость дисциплины	ак.ч.	396	180	144	72
	зач.ед.	11	5	4	2

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
Раздел 1	Инициация проекта: анализ требований, данные, архитектура	1.1	Введение в проектную разработку NLP-систем	Дисциплина как проектный цикл: 3 семестра, командная работа, реальная задача. Жизненный цикл NLP-проекта: постановка → данные → моделирование → тестирование → развёртывание → мониторинг. Формирование команд (3–5 человек). Распределение ролей: Team Lead, ML/NLP Engineer, Data Engineer, QA/DevOps. Agile/Scrum: спринты, daily standup, ретроспективы. Инструменты: GitHub Projects, Issues, Kanban	СЗ	УК-3.1, УК-3.2, SS-2.1
		1.2	Выбор задачи и анализ предметной области	Предложенные домены: автоматическая классификация обращений клиентов, извлечение информации из документов (NER + RE), вопросно-ответная система по базе знаний (RAG), суммаризация новостей/протоколов, чат-бот для службы поддержки, анализ тональности и репутации, модерация контента. Команды выбирают задачу. Обследование: интервью с «заказчиком», анализ аналогов. Документирование: описание предметной области	СЗ	ПК-1.3, SS-1.2, УК-2.2
		1.3	Формулирование требований и анализ осуществимости	Функциональные требования: что система должна делать (классифицировать, извлекать, генерировать, отвечать). Нефункциональные: latency, throughput, accuracy, поддерживаемые языки, максимальная длина текста. Анализ осуществимости: доступность данных, существующие модели, GPU-ресурсы. User Stories. Критерии приёмки. Документ: ТЗ v1.0	СЗ	ПК-1.3, УК-2.3, ПК-3.2
		1.4	Стратегия сбора и аннотирования текстовых данных	Источники текстов: открытые корпуса, веб-скрейпинг, API (Twitter, Reddit, новостные), корпоративные данные (обезличенные). Аннотирование: инструменты (Label Studio, Prodigy, Doccano), схема разметки, инструкция для аннотаторов, контроль качества (inter-annotator agreement, Cohen's κ). Формат данных: JSONL, CoNLL, BIO. Документ: Data Collection Plan	СЗ	DL-4.2, SS-2.1
		1.5	Сбор и аннотирование данных — итерация 1	Практическая работа: команды собирают и аннотируют первую порцию данных (200–500 текстов). Распределение	СЗ	DL-4.2, SS-2.1,

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			задач через GitHub Issues. Проверка качества: перекрёстная валидация разметки, вычисление κ. Анализ: баланс классов, длина текстов, покрытие сценариев. EDA по собранным данным. Обновление Data Collection Plan		УК-3.3
		1.6 Проектирование архитектуры NLP-системы	Компоненты: источник текста → предобработка (токенизация, очистка) → модель → постобработка → API → хранилище → мониторинг. Выбор модели: BERT vs. RoBERTa vs. T5, моноязычная vs. мультязычная. Стратегия: full fine-tuning vs. LoRA vs. prompt-tuning. Архитектурная диаграмма. Документ: Architecture Decision Records (ADR)	СЗ	ПК-2.1, DL-4.2, FC-2.1
		1.7 Организация репозитория и CI/CD	Структура: data/, src/, configs/, models/, tests/, docs/, notebooks/. Git workflow: feature branches, PR, код-ревью. DVC для данных и моделей. CI: GitHub Actions (lint, тесты). Pre-commit hooks (black, flake8, isort). README.md. Практика: настройка инфраструктуры проекта	СЗ	SS-2.1, ПК-3.2, SS-2.2
		1.8 Baseline модель — быстрый прототип	Обучение baseline: TF-IDF + LogisticRegression или предобученная модель с минимальным fine-tuning. Оценка: целевые метрики (F1, accuracy, ROUGE). Фиксация baseline как «точка отсчёта». Анализ ошибок: на каких примерах модель ошибается, какие данные нужны дополнительно. Документирование baseline	СЗ	DL-4.2, DL-1.9
		1.9 Планирование спринтов и Definition of Done	Декомпозиция работы на спринты (2 недели). Backlog: User Stories → задачи (Issues) → оценка (story points). Kanban-доска в GitHub Projects. Definition of Done для NLP-задач: код, тесты, код-ревью, документация, метрики. Ретроспектива первых недель	СЗ	УК-2.2, УК-3.2, SS-2.1
		1.10 Итерация 2: расширение данных и улучшение модели	Сбор дополнительных данных по выявленным пробелам. Аугментация текстов: back-translation, synonym replacement, EDA (Easy Data Augmentation). Fine-tuning предобученной модели (Hugging Face Trainer). Логирование экспериментов (W&B / MLflow). Сравнение с baseline. Код-ревью PR внутри команды	СЗ	DL-4.2, DL-1.9, SS-2.2
		1.11 Промежуточная презентация: результаты инициации	Подготовка промежуточного отчёта: описание задачи, данных, архитектуры, baseline, текущие результаты.	СЗ	УК-3.3, SS-3.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				Формат: 10-минутная презентация + демонстрация. Обратная связь от преподавателя и других команд. Фиксация action items		
		1.12	Ретроспектива и планирование следующей фазы	Ретроспектива команды: что сработало, что нет, какие процессы улучшить. Обновление ТЗ по итогам feedback. Планирование: приоритеты на фазу разработки. Рефлексия индивидуального вклада. Обновление README и документации	СЗ	УК-6.2, УК-3.2, SS-1.2
Раздел 2	Разработка: итеративное улучшение модели и данных	2.1	Итерация 3: целенаправленное улучшение данных	Анализ ошибок модели: confusion matrix по категориям, примеры типичных ошибок. Целенаправленный сбор данных для слабых мест (hard examples, edge cases). Очистка данных: удаление некачественной разметки, дубликатов, шумных примеров. Active learning: приоритизация наиболее информативных примеров для разметки (обзор)	СЗ	DL-4.2, DL-4.3, SS-3.2
		2.2	Итерация 4: продвинутая аугментация и предобработка	Аугментация текстов: back-translation (MarianMT), contextual augmentation (вставка слов через MLM), paraphrase generation (T5). Domain-specific предобработка: очистка HTML, нормализация, обработка специальных сущностей (email, URL, числа). Практика: разработка custom аугментации для проекта	СЗ	DL-4.2, DL-1.9
		2.3	Эксперименты с моделями и стратегиями fine-tuning	Сравнение моделей: BERT-base vs. RoBERTa vs. DeBERTa, русскоязычные варианты. Full fine-tuning vs. LoRA vs. adapter-tuning. Влияние размера модели: base vs. large. Ablation studies. Практика: таблица сравнения, обоснование финального выбора	СЗ	DL-1.9, FC-2.1, FC-2.3
		2.4	Оптимизация гиперпараметров и стратегии обучения	Learning rate: warmup + linear decay vs. cosine. Batch size и gradient accumulation. Mixed precision. Early stopping. Подбор гиперпараметров: Optuna + Hugging Face Trainer. Практика: систематический подбор, логирование, визуализация	СЗ	DL-4.2, DL-1.9
		2.5	Постобработка и бизнес-логика	Постобработка для NER: объединение субтокенов, фильтрация по confidence. Для классификации: калибровка вероятностей, threshold tuning. Для генерации: фильтрация toxicity, fact checking (обзор). Бизнес-логика:	СЗ	DL-4.3, ПК-2.1

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				маршрутизация обращений, формирование отчёта. Практика: реализация постобработки для проекта		
		2.6	Тестирование NLP-моделей: стратегия и план	Уровни тестирования: unit (токенизация, постобработка), integration (пайплайн end-to-end), model (метрики на test set), system (API, latency). Тестирование по подгруппам: длинные/короткие тексты, разные домены, разные языки. Edge cases: пустой текст, спецсимволы, adversarial inputs. Behavioral testing: CheckList (обзор). Документ: Test Plan	СЗ	ПК-3.2, DL-1.12
		2.7	Реализация тестов	Pytest для предобработки и постобработки. Тесты пайплайна: вход → ожидаемый формат выхода. Регрессионные тесты: $F1 \geq$ порог на фиксированном test set. Behavioral tests: invariance (перефразировка не меняет результат), directional expectations. Интеграция в CI. Практика: покрытие ключевых компонентов	СЗ	ПК-3.2, SS-2.2
		2.8	Код-ревью и качество кода	Чек-лист код-ревью: корректность, читаемость, тесты, документация, отсутствие утечки данных. Рефакторинг: выделение переиспользуемых компонентов (data processing, model training, evaluation). Type hints, docstrings. Практика: перекрёстное ревью между командами	СЗ	SS-2.2, SS-2.1
		2.9	Итерация 5: финальное улучшение модели в семестре	Применение всех наработок: расширенные данные, лучшая модель, оптимальные гиперпараметры, постобработка. Финальная оценка на held-out test set. Сравнение с baseline: таблица прогресса по итерациям. Замораживание модели v1.0	СЗ	DL-4.2, DL-1.9
		2.10	Промежуточный отчёт: прогресс по модели	Обновление документации: текущие метрики, описание экспериментов, сравнение с baseline. Демонстрация: примеры предсказаний, примеры ошибок. Обратная связь. Планирование развёртывания	СЗ	УК-3.3, SS-3.2
		2.11	Ретроспектива и планирование развёртывания	Ретроспектива итераций 3–5: что дало наибольший прирост. Обновление backlog: задачи на развёртывание, тестирование, документирование. Оценка технического долга. Распределение ответственности на фазу развёртывания	СЗ	УК-6.2, УК-2.2
Раздел 3	Развёртывание, мониторинг и сдача MVP	3.1	Контейнеризация NLP-сервиса	Dockerfile: базовый образ, установка зависимостей, копирование модели и кода. Оптимизация размера образа:	СЗ	ПК-2.1, DL-4.3

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			multi-stage build, исключение training-зависимостей. Docker Compose: сервис инференса + Redis (кэш) + PostgreSQL (логи). Health check. Практика: контейнеризация NLP-пайплайна		
		3.2 API инференса: проектирование и реализация	FastAPI: эндпоинты /predict (single text), /predict_batch, /health. Pydantic-модели: TextRequest, PredictionResponse. Обработка ошибок: слишком длинный текст, невалидная кодировка, таймауты. Swagger-документация. Практика: реализация API для проекта	СЗ	ПК-2.1, DL-4.3
		3.3 Оптимизация инференса NLP-моделей	Экспорт: ONNX (optimum). Quantization: dynamic (torch.quantization), ONNX Runtime INT8. Distillation (DistilBERT). Batched inference. Кэширование повторяющихся запросов (Redis). Замеры: latency по длине текста, throughput. Практика: оптимизация, таблица «метод → latency → accuracy»	СЗ	DL-4.3, FC-2.3
		3.4 CI/CD для NLP-проекта	CI: lint → unit tests → model regression tests → build Docker image. CD: публикация образа в registry при merge в main. Тегирование версий. Автоматический запуск evaluation при обновлении модели. Практика: настройка полного CI/CD	СЗ	ПК-3.2, SS-2.1
		3.5 Мониторинг и логирование NLP-сервиса	Логирование запросов и предсказаний (structured logging). Мониторинг: latency, throughput, error rate. Специфика NLP: распределение длин входных текстов, распределение confidence, распределение классов предсказаний. Обнаружение дрейфа: изменение vocabulary, topic shift. Практика: настройка логирования и мониторинга	СЗ	DL-4.3, ПК-3.2
		3.6 Нагрузочное тестирование	Locust: сценарии для NLP (разная длина текстов, burst traffic). Определение пропускной способности. Bottleneck: tokenization vs. inference vs. postprocessing. Влияние batch size на throughput. Практика: нагрузочное тестирование, отчёт с рекомендациями	СЗ	ПК-3.2, DL-4.3
		3.7 Техническая документация проекта v1.0	Документы: ТЗ, Architecture Decision Records, Model Card, Datasheet, API спецификация (OpenAPI), план тестирования, инструкция по развёртыванию. README. Практика: финализация документации для MVP	СЗ	ПК-1.3, SS-1.2
		3.8 Приёмочное тестирование MVP	Тестирование по критериям из ТЗ: метрики ≥ пороги,	СЗ	ПК-3.2,

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				latency \leq SLA, edge cases (пустой текст, спецсимволы, длинный текст), API корректность. Протокол тестирования. Практика: перекрёстное приёмочное тестирование (одна команда тестирует другую)		DL-1.12
		3.9	Демонстрация MVP заказчику	Презентация MVP: работающая система, ключевые метрики, архитектура, ограничения. Формат: 15 мин + вопросы. Обратная связь от «заказчика». Фиксация замечаний и пожеланий к доработке	СЗ	УК-3.3, SS-3.2
		3.10	Ретроспектива семестра 6	Ретроспектива: процессы, техническое качество, коммуникация. Velocity: запланировано vs. выполнено. Технический долг. Индивидуальная рефлексия	СЗ	УК-6.2, УК-3.2
		3.11	Планирование семестра 7	Анализ feedback от «заказчика» и перекрёстного тестирования. Приоритизация доработок. Обновление backlog. Перераспределение ролей (при необходимости). Milestones для семестра 7	СЗ	УК-2.2, УК-2.3
Раздел 4	Доработка и расширение функциональности	4.1	Анализ feedback и планирование доработок	Детальный разбор замечаний по MVP: недостаточная точность на подкатегориях, проблемы с длинными текстами, edge cases. Декомпозиция доработок на задачи. Sprint planning. Обновление T3 v2.0	СЗ	УК-2.3, ПК-1.3
		4.2	Расширение датасета и переобучение	Целенаправленный сбор данных по слабым местам MVP. Pseudo-labeling: использование текущей модели для разметки новых данных с ручной верификацией. Переобучение на расширенном датасете. Сравнение метрик до/после. Обновление Datasheet	СЗ	DL-4.2, DL-1.9
		4.3	Дополнительная функциональность: RAG и интеграция с LLM	Реализация RAG-компонента: индексация документов (FAISS/Chroma), retrieval по запросу, формирование промпта, генерация ответа. Или: интеграция с LLM API для расширения функциональности (суммаризация, генерация ответов). Практика: добавление нового функционала, тесты, код-ревью	СЗ	FC-2.1, FC-2.3
		4.4	Дополнительная функциональность: интерфейс и визуализация	Dashboard: Streamlit / Gradio для демонстрации. Визуализация предсказаний: подсветка сущностей, attention weights, confidence scores. Генерация отчётов: агрегированная статистика. Практика: реализация интерфейса для проекта	СЗ	DL-4.3, ПК-2.1

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
		4.5	Мультиязычность и кросс-доменный перенос	Расширение на дополнительный язык (русский/английский): мультиязычная модель (XLM-RoBERTa), zero-shot cross-lingual transfer. Или: адаптация к новому домену (domain adaptation). Практика: эксперимент с мультиязычностью/domain adaptation, оценка потерь качества	СЗ	FC-2.1, DL-4.2
		4.6	Обработка длинных документов	Проблема: трансформеры ограничены 512 токенами. Стратегии: chunking + aggregation, sliding window, Longformer/BigBird (обзор). Hierarchical approach: предложения → документ. Практика: реализация обработки длинных документов для проекта	СЗ	DL-4.2, DL-4.3
		4.7	Интеграционное тестирование доработок	Полное интеграционное тестирование: API → модель → постобработка → хранилище → визуализация. Behavioral tests для новой функциональности. Регрессионные тесты: старая функциональность не сломана. Обновление Test Plan	СЗ	ПК-3.2, SS-2.2
		4.8	Промежуточная демонстрация: прогресс доработок	Демонстрация улучшений: новые метрики, новая функциональность, результаты тестирования. Сравнение с MVP. Обратная связь. Корректировка плана	СЗ	УК-3.3, SS-3.2
		4.9	Ретроспектива и подготовка к финализации	Ретроспектива: что улучшилось, какие проблемы остались. Технический долг. План финализации: документация, тестирование, презентация. Распределение ответственности за финальные deliverables	СЗ	УК-6.2, УК-2.2
Раздел 5	Финализация продукта и документации	5.1	Финальная оптимизация модели	Последние улучшения: подбор порогов, ensemble (при необходимости), distillation для production. Экспорт финальной модели в production-формат. Фиксация версий: модель, данные, код. Тегирование release v2.0	СЗ	DL-4.2, DL-1.9
		5.2	Финальное приёмочное тестирование	Полное тестирование по ТЗ v2.0. Регрессионные тесты. Нагрузочные тесты. Behavioral tests (CheckList). Edge cases. Протокол приёмки: PASS/FAIL по каждому критерию	СЗ	ПК-3.2, DL-1.12
		5.3	Оценка предвзятостей и этических аспектов	Анализ по подгруппам: тональность по полу, NER для разных национальностей, fairness метрики. Toxicity testing: проверка генерации на toxicity. Формулирование ограничений использования. Обновление Model Card: «Ethical Considerations» и «Limitations»	СЗ	SS-3.2, SS-1.2
		5.4	Финализация технической документации	Ревизия: ТЗ v2.0, ADR, Model Card, Datasheet, API spec, Test	СЗ	ПК-1.3,

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				Plan, инструкция по развёртыванию. Consistency check. Генерация Swagger. Практика: финальная вычитка всеми участниками		SS-1.2
		5.5	Подготовка delivery package	Docker image (финальный), исходный код (тег release), документация, модель (DVC / Model Registry), данные (DVC), тесты, CI/CD. Инструкция по развёртыванию с нуля. Проверка: другая команда может развернуть по инструкции	СЗ	ПК-2.1, SS-2.1
		5.6	Подготовка финальной презентации	Структура: бизнес-задача → требования → архитектура → данные → модель → результаты → demo → ограничения → дальнейшие шаги. Live demo. Распределение между участниками. Репетиция	СЗ	УК-3.3, SS-3.2
		5.7	Индивидуальная рефлексия по итогам двух семестров	Индивидуальный отчёт: мой вклад, чему научился, какие навыки развил, что сделал бы иначе. Анализ эволюции роли в команде. Связь с профессиональным ростом	СЗ	УК-6.2, SS-1.2
		5.8	Ретроспектива семестра 7	Командная ретроспектива: сравнение с семестром 6, эволюция процессов. Velocity. Формулирование «памятки» для следующей итерации	СЗ	УК-3.2, УК-6.2
		5.9	Планирование семестра 8	Планирование финальной фазы: исследовательское расширение, научная статья, защита. Оставшиеся доработки. Индивидуальные цели на семестр	СЗ	УК-2.2, УК-2.3
Раздел 6	Защита проекта и peer review	6.1	Защита проекта — команды 1 и 2	Презентация (20 мин) + live demo (10 мин) + вопросы (10 мин). Критерии: качество модели, архитектура, код, тесты, документация, командная работа, обоснованность решений. Оценка преподавателем и peer review	СЗ	УК-3.3, DL-4.2
		6.2	Защита проекта — команды 3 и 4	Аналогичный формат. Перекрестное оценивание по единому чек-листу	СЗ	УК-3.3, DL-4.2
		6.3	Защита проекта — оставшиеся команды	Аналогичный формат. Обсуждение общих паттернов и лучших находок	СЗ	УК-3.3, SS-3.2
		6.4	Перекрытое ревью кода и системы	Команды обмениваются репозиториями. Код-ревью по чек-листу. Попытка развёртывания чужого проекта по инструкции. Отчёт ревью с рекомендациями	СЗ	SS-2.2, ПК-3.2
		6.5	Перекрытое приёмочное тестирование	Тестирование чужого проекта по его ТЗ: метрики, edge cases, API, документация. Протокол приёмки. Обсуждение расхождений	СЗ	ПК-3.2, DL-1.12
		6.6	Общая ретроспектива и lessons learned	Командные ретроспективы за 2 семестра. Общие уроки.	СЗ	УК-6.2,

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				Формулирование «памятки разработчика NLP-систем». Связь с промышленной практикой		УК-3.2
		6.7	Анализ feedback и планирование семестра 8	Разбор обратной связи. Приоритизация финальных доработок. Планирование: исследование, статья, передача	СЗ	УК-2.3, SS-1.2
		6.8	Фиксация промежуточных результатов	Обновление артефактов по итогам feedback. Тегирование v2.1. Подготовка backlog семестра 8	СЗ	SS-2.1, УК-2.2
Раздел 7	Исследовательское расширение и продвинутые техники	7.1	Исследование новых подходов для проекта	Обзор литературы: новейшие методы для задачи проекта (последние 1–2 года). LLM для NLP-задач: GPT-4, LLaMA, Mistral — применимость к задаче. Промпт-инженерия vs. fine-tuning. Формулирование гипотез, планирование экспериментов	СЗ	FC-2.1, FC-2.3
		7.2	Эксперимент с LLM и промпт-инженерией	Использование LLM API (OpenAI / open-source) для задачи проекта: zero-shot, few-shot, chain-of-thought. Сравнение с fine-tuned моделью: качество, стоимость, latency. Документирование результатов. Обсуждение: когда LLM оправданы, когда fine-tuned модель лучше	СЗ	FC-2.3, FC-2.1
		7.3	Продвинутый RAG и knowledge grounding	Улучшение RAG: re-ranking (cross-encoder), query decomposition, multi-hop reasoning. Hybrid search: dense + sparse retrieval. Evaluation: retrieval accuracy, answer faithfulness. Практика: улучшение RAG-компонента проекта (если применимо) или другое исследовательское расширение	СЗ	FC-2.1, DL-4.3
		7.4	Continuous learning и адаптация модели	Инкрементальное обучение: partial fit, fine-tuning на новых данных без деградации на старых. Мониторинг дрейфа языка: изменение тем, появление новых терминов. Стратегия переобучения: scheduled vs. triggered. Практика: реализация пайплайна обновления модели	СЗ	DL-4.3, DL-1.12
		7.5	Объяснимость NLP-моделей	Методы: attention visualization, LIME для текстов, SHAP для текстов (обзор), integrated gradients. Counterfactual explanations: «какое слово изменить, чтобы изменился результат». Практика: добавление модуля объяснимости в систему	СЗ	DL-1.12, SS-3.2
		7.6	Мониторинг дрейфа текстовых данных в production	Специфика текстового дрейфа: изменение словаря, topic shift, стилистические изменения. Методы: embedding drift (косинусное расстояние между средними эмбедами),	СЗ	DL-4.3, ПК-3.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				распределение длин, OOV rate. Практика: реализация мониторинга для проекта		
		7.7	Интеграция исследовательских результатов в production	Выбор лучших наработок. Интеграция в основную ветку: рефакторинг, тесты, код-ревью. Обновление модели (версионирование). Обновление документации и Model Card. Тегирование release v3.0	СЗ	DL-4.2, SS-2.1
Раздел 8	Финальная доработка и передача проекта	8.1	Финальная оптимизация и замораживание	Последний цикл оптимизации: финальная модель, оптимальные пороги, оптимизация инференса (ONNX, quantization). Замораживание: модель, данные, код. Фиксация всех версий	СЗ	DL-4.3, DL-4.2
		8.2	Полное регрессионное и приёмочное тестирование	Финальный раунд: unit, integration, model regression, нагрузочные, behavioral, приёмочные. Все тесты зелёные. Протокол финального тестирования. Сравнение с ТЗ v2.0	СЗ	ПК-3.2, DL-1.12
		8.3	Написание технического отчёта / научной статьи	Технический отчёт (10–15 стр.) или краткая статья (4–6 стр., формат конференции): введение, обзор, описание системы, эксперименты, результаты, заключение. Практика: написание черновика, взаимное ревью	СЗ	SS-1.2, SS-3.2
		8.4	Финализация всей документации	Полная ревизия: ТЗ, ADR, Model Card v3.0, Datasheet, API spec, Test Plan, Test Report, инструкция, отчёт, README. Consistency check. Публикация (GitHub Pages / Wiki)	СЗ	ПК-1.3, SS-1.2
		8.5	Формирование финального delivery package	Пакет: Docker image, исходный код (тег v3.0), документация, модель, данные, тесты, CI/CD, отчёт. Проверка: третье лицо может развернуть. Формальная передача	СЗ	ПК-2.1, SS-2.1
		8.6	Подготовка финальной презентации	Структура: эволюция за 3 семестра → финальная архитектура → результаты → исследовательские находки → live demo → ограничения → рекомендации. Распределение. Репетиция	СЗ	УК-3.3, SS-3.2
		8.7	Индивидуальная итоговая рефлексия	Итоговый отчёт: путь за 3 семестра, эволюция навыков, ключевые уроки, связь с карьерными целями. Формат: 3–4 страницы	СЗ	УК-6.2, SS-1.2
Раздел 9	Финальная защита и подведение итогов	9.1	Финальная защита — команды 1 и 2	Итоговая презентация (25 мин) + live demo (10 мин) + вопросы (15 мин). Критерии: эволюция за 3 семестра, качество системы, код, тесты, документация, командная работа, исследовательский вклад. Оценка преподавателем и	СЗ	УК-3.3, DL-4.2

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			внешним экспертом		
		9.2	Финальная защита — команды 3 и 4	СЗ	УК-3.3, DL-4.2
		9.3	Финальная защита — оставшиеся команды	СЗ	УК-3.3, SS-3.2
		9.4	Финальное перекрёстное ревью	СЗ	SS-2.2, ПК-3.2
		9.5	Итоговая ретроспектива: 3 семестра проекта	СЗ	УК-6.2, УК-3.2
		9.6	Карта профессионального развития	СЗ	УК-6.2, SS-1.2
		9.7	Подведение итогов дисциплины	СЗ	УК-3.1, SS-1.2

* - заполняется только по **ОЧНОЙ** форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	Персональные компьютеры, необходимое ПО
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	Персональные компьютеры, необходимое ПО

* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература:

1. Язык и искусственный интеллект: проблемы, вызовы, решения [Электронный ресурс] : монография / Н. Ю. Бородулина, Е. Ю. Воякина, В. С. Григорьева и др. ; под общ. ред. Л. Г. Поповой. – Тамбов : Издательский центр ФГБОУ ВО «ТГТУ», 2025. ISBN 978-5-8265-2940-9

2. Лейн, Х. Обработка естественного языка в действии : практическое руководство для разработчиков / Х. Лейн, М. Дышель. — 2-е изд. — Санкт-Петербург : Питер, 2026. — 736 с. — (Для профессионалов). — ISBN 978-5-4461-4385-6.

Дополнительная литература:

1. Гольдберг Й. Нейросетевые методы в обработке естественного языка / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2019. – 282 с.: ил. ISBN 978-5-97060-754-1

Ресурсы информационно-телекоммуникационной сети «Интернет»:

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН

<https://mega.rudn.ru/MegaPro/Web>

- ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>

- ЭБС «Юрайт» <http://www.biblio-online.ru>

- ЭБС «Консультант студента» www.studentlibrary.ru

- ЭБС «Знаниум» <https://znaniyum.ru/>

2. Базы данных и поисковые системы

- Sage <https://journals.sagepub.com/>

- Springer Nature Link <https://link.springer.com/>

- Wiley Journal Database <https://onlinelibrary.wiley.com/>

- Научнометрическая база данных Lens.org <https://www.lens.org>

Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля:*

* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**