

**ГОУ ВПО Российско-Армянский (Славянский)
университет**

Утверждено
Директор Института математики и информатики
Арамян Р.Г.

«21» мая 2025, протокол №9.1

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

Наименование дисциплины: Введение в большие языковые модели (LLM)

Автор (ы) Аветисян К.И.
Ф.И.О, ученое звание (при наличии), ученая степень (при наличии)

Направление подготовки: 01.04.02 Прикладная математика и информатика
Наименование образовательной программы: «Системное программирование»

1. АННОТАЦИЯ

1.1. Краткое описание содержания данной дисциплины;

Языковые модели — основа современных приложений обработки естественного языка (NLP), позволяющая использовать одну универсальную систему для решения множества задач. По мере развития искусственного интеллекта (ИИ), машинного обучения (ML) и NLP важно глубоко понимать, как работают такие модели. Этот курс поможет студентам всесторонне разобраться в языковых моделях, пройдя весь путь их создания: от сбора и очистки данных для предобучения до построения трансформеров, обучения модели, её оценки и подготовки к использованию.

1.2. Трудоемкость в академических кредитах и часах, формы итогового контроля (экзамен/зачет);

Виды учебной работы	В акад. часах
Общее количество часов	64
Лекция	32
Практика	
Итоговый контроль	Экзамен

1.3. Взаимосвязь дисциплины с другими дисциплинами учебного плана специальности (направления)

Данная дисциплина связана с дисциплиной «Вступление в машинное обучение»

1.4. Результаты освоения программы дисциплины:

2. Код 3. компетенции	Наименование компетенции	Код индикатора достижения компетенций	Наименование индикатора достижений компетенций
УК-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели	1	Разрабатывает командную стратегию для достижения поставленной цели
		2	Умеет организовывать и руководить работой команды
		3	Демонстрирует понимание

			результатов работы команды и личных действий в ней
ОПК-1	Способен решать актуальные задачи фундаментальной и прикладной математики	1	Имеет представление об основных подходах к решению актуальных задач фундаментальной и прикладной математики
		2	Демонстрирует умение применять математический аппарат для решения задач.
		3	Имеет навыки выбора подходящих методов решения задач фундаментальной и прикладной математики
ОПК-2	Способен совершенствовать и реализовывать новые математические методы решения прикладных задач	1	Обладает знаниями о существующих математических методах, применяемых для решения прикладных задач.
		2	Демонстрирует умение использования математического языка и математической символики, построения цепочки рассуждений, формулировки математических утверждений для решения прикладных задач.
		3	Имеет практический опыт совершенствования и реализации различных

			математических методов решения прикладных задач
ОПК-3	Способен разрабатывать математические модели и проводить их анализ при решении задач в области профессиональной деятельности	1	Формулирует основные теоретические положения в области математического моделирования.
		2	Демонстрирует умения давать содержательную интерпретацию полученных результатов при проведении анализа математических моделей.
		3	Имеет практический опыт разработки и проведения анализа математических моделей при решении задач
ОПК-4	Способен комбинировать и адаптировать существующие информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности	1	Обладает знаниями о существующих информационно-коммуникационных технологиях и основных требованиях информационной безопасности.
		2	Демонстрирует умения комбинировать и адаптировать существующие информационно-коммуникационные технологии, а также умение учитывать основные требования информационной

			безопасности при решении прикладных задач
		3	Имеет практический опыт комбинирования и адаптирования существующих информационно коммуникационных технологий и учета основных требований информационной безопасности при решении прикладных задач
УК-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий	1	Критически анализирует проблемную ситуацию с целью выработки стратегии действий, аргументировано формулирует собственные суждения и оценки
		2	Использует критический анализ, систематизацию и обобщение информации для решения проблемной ситуации
		3	Владеет навыками критического анализа проблемных ситуаций на основе системного подхода, вырабатывания стратегии действий
УК-2	Способен управлять проектом на всех этапах его жизненного цикла	1	Определяет этапы жизненного цикла проекта и выстраивает

			последовательность их реализации
		2	Формулирует проблему, на решение которой направлен проект, грамотно определяет цель проекта.
		3	Проектирует решение конкретных задач проекта, выбирая оптимальный способ их решения.
УК-4	Способен применять современные коммуникативные технологии, в том числе на иностранном(ых) языке(ах), для академического и профессионального взаимодействия	1	Обладает знаниями особенностей и правил личной и профессиональной устной и письменной коммуникации, в том числе на иностранном(ых) языке(ах)
		2	Демонстрирует умение применять современные коммуникативные технологии для академического и профессионального взаимодействия в ситуации устной и письменной коммуникации, в том числе на иностранном(ых) языке(ах)
		3	Имеет навыки академического и профессионального взаимодействия, в том числе

			на иностранном(ых) языке(ах)
УК-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия	1	Обладает необходимыми знаниями о разнообразии культур и об основных принципах межкультурного взаимодействия
		2	Демонстрирует умение самостоятельно добывать профессиональные знания с использованием иностранного языка для развития способности межкультурного взаимодействия
		3	Имеет навыки межкультурного взаимодействия при выполнении профессиональных задач
УК-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	1	Применяет рефлексивные методы в процессе оценки разнообразных ресурсов, используемых для решения задач самоорганизации и саморазвития
		2	Определяет цели и приоритеты собственной деятельности и способы их достижения
		3	Планирует результаты собственной деятельности с

			учетом ресурсов	необходимых
--	--	--	--------------------	-------------

4. УЧЕБНАЯ ПРОГРАММА

4.1. Цели и задачи дисциплины

Дать студентам глубокое и практическое понимание языковых моделей через пошаговое создание собственной модели с нуля.

Задачи курса:

- Ознакомить студентов с архитектурой трансформеров и принципами работы языковых моделей.
- Научить собирать и очищать данные для предобучения.
- Дать навыки построения и обучения языковой модели.
- Показать методы оценки качества модели и подготовки её к разворачиванию.
- Развить инженерное мышление через практический проект полного цикла.

4.2. Трудоемкость дисциплины и виды учебной работы (в академических часах и зачетных единицах)

Виды учебной работы	Всего, в акад. часах	Распределение по семестрам					
		— сем	3 сем	— сем	— сем.	— сем	— сем.
1	2	3	4	5	6	7	8
1.Общая трудоемкость изучения дисциплины по семестрам, в т. ч.:	180						
1.1. Аудиторные занятия, в т. ч.:	32		32				
1.1.1.Лекции	32		32				
1.2. Самостоятельная работа, в т. ч.:	108		108				
1.3. Другие методы и формы занятий	40		40				
Итоговый контроль (Экзамен, Зачет, диф. зачет - указать)			Экзам ен				

4.3. Содержание дисциплины

4.3.1. Тематический план и трудоемкость аудиторных занятий (модули, разделы дисциплины и виды занятий) по рабочему учебному плану

Разделы и темы дисциплины	Всего (ак. часов)	Лекции (ак. часов)	Практ. Занятия (ак. часов)	Семинары (ак. часов)	Лабор. (ак. часов)
---------------------------	----------------------	-----------------------	----------------------------------	-------------------------	-----------------------

1	2=3+4+5+6 +7	3	4	5	6
Тема 1. Механизм внимания.	1	1			
Тема 2. Энкодер-декодер архитектуры.	1	1			
Тема 3. Реализация механизма	1	1			
Тема 4. Архитектура трансформера.	1	1			
Тема 5. Токенизация	1	1			
Тема 6. Тонкая настройка языковых	1	1			
Тема 7. Мультиязычные языковые	1	1			
Тема 8. Авторегрессивные модели.	1	1			
Тема 9. Тонкая настройка	1	1			
Тема 10. Модели семейства GPT.	1	1			
Тема 11. Модели семейства GPT.	1	1			
Тема 12. Реализация модели GPT.	1	1			
Тема 13. Модель Llama.	1	1			
Тема 14. Модели Mistral.	1	1			
Тема 15. Реализация модели GPT.	1	1			
Тема 16. Расчет вычислительных	1	1			
Тема 17. Подбор гиперпараметров	1	1			
Тема 18. Смесь экспертов	1	1			
Тема 19. Видеокарты	1	1			
Тема 20. CUDA ядра	1	1			
Тема 21. Параллелизм	1	1			
Тема 22. Параллелизм	1	1			
Тема 23. Законы увеличения	1	1			
Тема 24. Законы увеличения	1	1			
Тема 25. Инференс	1	1			
Тема 26. Оценка моделей	1	1			
Тема 27. Данные	1	1			
Тема 28. Данные	1	1			
Тема 29. Состязательное обучение	1	1			
Тема 30. Состязательное обучение	1	1			
Тема 31. Разные виды внимания	1	1			
Тема 32. Модель DeepSeek	1	1			
ИТОГО	32	32			

4.3.2. Краткое содержание разделов дисциплины в виде тематического плана

Тема 1. Механизм внимания.

Рассматривается механизм внимания (attention mechanism) в нейронных сетях, который позволяет моделям фокусироваться на наиболее значимых частях входных данных при

обработке последовательностей. Объясняется, как механизм внимания взвешивает различные части входных данных, чтобы улучшить точность и интерпретируемость модели. Описываются различные виды механизмов внимания, такие как self-attention и multi-head attention.

Тема 2. Энкодер-декодер архитектуры.

Рассматриваются энкодер-декодер архитектуры, которые широко применяются в задачах преобразования последовательностей, таких как машинный перевод и обработка естественного языка. Объясняется принцип работы энкодера, который считывает и кодирует входную последовательность в фиксированное представление, и декодера, который преобразует это представление обратно в выходную последовательность.

Тема 3. Реализация механизма внимания.

Рассматривается реализация механизма внимания в нейронных сетях для улучшения обработки последовательных данных. Объясняется процесс вычисления весов внимания, который включает в себя оценку значимости каждой части входных данных и их взвешивание для создания контекстного представления. Описываются основные шаги реализации, такие как вычисление сходства, нормализация с использованием softmax функции и вычисление взвешенной суммы. Приводятся примеры реализации различных типов внимания, включая self-attention и multi-head attention.

Тема 4. Архитектура трансформера.

Рассматривается архитектура трансформера, которая используется для обработки последовательных данных и является основой многих современных моделей, таких как BERT и GPT. Объясняется принцип работы трансформера, основанный на механизмах внимания, который позволяет параллельно обрабатывать элементы последовательности, что значительно ускоряет обучение и повышает производительность. Описывается архитектура, у которой ключевыми компонентами являются механизмы multi-head attention и позиционные кодировки, которые помогают моделировать последовательные зависимости.

Тема 5. Токенизация

Рассматривается процесс токенизации — преобразование текстовых данных в формат, пригодный для обработки языковыми моделями. Объясняется, как текст разбивается на меньшие единицы (токены), такие как слова, подслова или символы, в зависимости от выбранной стратегии. Описываются различные методы токенизации, включая whitespace-токенизацию, byte-pair encoding (BPE), WordPiece и SentencePiece. Также обсуждается влияние выбора токенизатора на качество и эффективность модели, а также роль специальной разметки (например, [CLS], [SEP]) при работе с моделями трансформерного типа.

Тема 6. Тонкая настройка языковых моделей.

Рассматривается тонкая настройка (fine-tuning) языковых моделей, используемая для адаптации предобученных моделей к специфическим задачам и данным. Объясняется процесс тонкой настройки, включающий дообучение модели на целевом датасете с использованием небольшого количества эпох, что позволяет модели более точно решать заданные задачи, такие как классификация текста, извлечение информации или машинный перевод. Описываются преимущества тонкой настройки, включая экономию вычислительных ресурсов и повышение точности на специфических задачах, а также примеры успешного применения этого метода в различных областях.

Тема 7. Мультиязычные языковые модели.

Рассматриваются мультиязычные языковые модели, предназначенные для обработки текста на различных языках с использованием единой модели. Объясняется принцип работы таких моделей, которые обучаются на многоязычных корпусах данных и могут понимать и

генерировать текст на нескольких языках. Описываются ключевые особенности мультязычных моделей, такие как их способность к перекрестному обучению, что позволяет моделям переносить знания с одного языка на другой, улучшая производительность. Приводятся примеры популярных мультязычных моделей, таких как mBERT и XLM-R, и их применение в задачах машинного перевода, мультязычного поиска и других многоязычных приложениях.

Тема 8. Авторегрессивные модели.

Рассматриваются авторегрессивные модели, используемые для генерации последовательных данных, таких как текст, аудио или временные ряды. Объясняется принцип работы авторегрессивных моделей, которые предсказывают следующий элемент последовательности на основе предыдущих элементов, что позволяет моделировать временные зависимости в данных. Описываются ключевые преимущества этих моделей, включая их способность к генерации последовательностей высокой точности и гибкость в применении к различным типам данных. Приводятся примеры популярных авторегрессивных моделей, таких как GPT (Generative Pre-trained Transformer) и их успешное использование в задачах генерации текста и других приложениях.

Тема 9. Тонкая настройка мультязычных языковых моделей.

Рассматривается тонкая настройка мультязычных языковых моделей для адаптации их к специфическим задачам на нескольких языках. Объясняется процесс тонкой настройки, который включает дообучение предобученных моделей на целевых многоязычных датасетах, что позволяет улучшить производительность моделей в различных языковых контекстах. Описываются преимущества такого подхода, включая возможность эффективного использования одной модели для множества языков и улучшение качества обработки языка в специфических задачах.

Тема 10. Модели семейства GPT.

Рассматриваются модели семейства GPT-1 (Generative Pre-trained Transformer 1), которые являются первыми в серии генеративных предобученных трансформеров. Объясняется принцип работы GPT-1, основанный на архитектуре трансформера, и его способность генерировать текст на основе предшествующего контекста. Описываются ключевые особенности GPT-1, включая его предобучение на большом корпусе данных и последующую тонкую настройку для выполнения специфических задач, таких как машинный перевод и текстовая генерация.

Тема 11. Модели семейства GPT.

Рассматриваются модели семейства GPT-2 (Generative Pre-trained Transformer 2), которые представляют собой улучшенную версию оригинальной модели GPT. Объясняется принцип работы GPT-2, основанный на масштабной архитектуре трансформера с большим количеством параметров, что позволяет генерировать более связный и осмысленный текст. Описываются ключевые особенности GPT-2, такие как его способность продолжать текст, задаваемый пользователем, и высокая точность в различных задачах, включая ответ на вопросы, перевод текста и создание творческого контента. Приводятся примеры применения GPT-2 в различных областях, демонстрирующие её мощные возможности в генерации естественного языка.

Тема 12. Реализация модели GPT.

Рассматривается реализация модели GPT-1 (Generative Pre-trained Transformer 1). Объясняется архитектура многослойного трансформера с механизмом внимания для генерации текста. Описываются этапы предобучения на большом корпусе текстов и тонкой настройки для специфических задач.

Тема 13. Модель Llama.

Рассматривается модель Llama, предназначенная для обработки и генерации текста. Объясняется архитектура модели, основанная на многослойном трансформере с механизмом внимания, что позволяет эффективно работать с последовательными данными. Описываются основные особенности и преимущества Llama, такие как улучшенная производительность и точность в сравнении с предыдущими моделями. Приводятся примеры применения модели Llama в различных задачах обработки естественного языка, включая генерацию текста и перевод.

Тема 14. Модели Mistral.

Рассматриваются модели Mistral, разработанные для обработки и генерации текста с высокой точностью. Объясняется их архитектура, основанная на улучшенных трансформерах, что обеспечивает эффективную работу с большими объемами данных. Описываются ключевые особенности моделей Mistral, такие как повышенная производительность и оптимизация для различных задач обработки естественного языка. Приводятся примеры применения моделей Mistral в задачах, таких как генерация текста, машинный перевод и анализ текста.

Тема 15. Реализация модели GPT.

Рассматривается реализация модели GPT-2 (Generative Pre-trained Transformer 2). Объясняется архитектура многослойного трансформера с механизмом внимания, что позволяет генерировать связный и осмысленный текст. Описываются этапы предобучения модели на большом корпусе данных и тонкой настройки для конкретных задач. Приводятся примеры кода для реализации GPT-2.

Тема 16. Расчёт вычислительных ресурсов.

Рассматривается процесс оценки и планирования вычислительных ресурсов, необходимых для обучения и использования языковых моделей. Объясняется, как определить потребности в оперативной памяти (RAM), видеопамати (VRAM), объёме хранилища и времени обучения в зависимости от размера модели, объёма обучающего датасета и параметров обучения (batch size, sequence length, количество эпох и т.д.). Описываются подходы к оценке потребления ресурсов на различных этапах — предобработка данных, обучение модели, инференс. Также обсуждаются инструменты мониторинга (например, NVIDIA SMI, TensorBoard) и методы оптимизации использования ресурсов, включая градиентный аккумулятор, mixed-precision training и распределённое обучение.

Тема 17. Подбор гиперпараметров

Рассматриваются методы выбора оптимальных гиперпараметров для эффективного обучения нейронных сетей. Объясняется роль ключевых гиперпараметров, таких как скорость обучения (learning rate), размер батча (batch size), число слоёв и голов внимания, dropout, весовые коэффициенты регуляризации и др. Описываются различные стратегии подбора, включая ручной перебор (manual search), поиск по сетке (grid search), случайный поиск (random search) и байесовскую оптимизацию. Особое внимание уделяется важности валидационного набора данных для оценки качества моделей при разных конфигурациях. Также обсуждаются инструменты автоматического тюнинга, такие как Optuna и Ray Tune.

Тема 18. Смесь экспертов

Рассматривается архитектурный подход "смесь экспертов", который позволяет масштабировать языковые модели, увеличивая их ёмкость без пропорционального роста вычислительных затрат. Объясняется принцип работы MoE: входные данные направляются только к подмножеству экспертов (нейронных блоков), выбранных с помощью управляющего механизма (gating). Это позволяет использовать разные подмодели для обработки разных типов входов, улучшая обобщающую способность модели. Описываются ключевые компоненты: эксперты, механизм маршрутизации и методы балансировки нагрузки.

между экспертами. Приводятся примеры использования MoE в современных масштабных моделях, таких как GShard и Switch Transformer.

Тема 19. Видеокарты

Рассматривается роль видеокарт (GPU) в обучении и инференсе нейронных сетей, особенно в задачах обработки естественного языка. Объясняется, почему именно GPU, благодаря своей архитектуре с тысячами параллельных ядер, значительно ускоряют вычисления по сравнению с CPU. Описываются основные характеристики видеокарт, важные для работы с моделями: объём видеопамати, количество CUDA-ядер, пропускная способность памяти и поддержка библиотек (например, CUDA, cuDNN). Приводятся примеры популярных GPU для задач машинного обучения (например, NVIDIA A100, RTX 4090, V100) и обсуждается, как выбрать подходящую видеокарту в зависимости от размера модели и бюджета. Также рассматриваются возможности распределённого обучения на нескольких GPU.

Тема 20. CUDA ядра

Рассматриваются CUDA-ядра — базовые вычислительные элементы видеокарт NVIDIA, предназначенные для параллельной обработки данных. Объясняется, как архитектура CUDA позволяет эффективно выполнять операции с большими тензорами, характерные для обучения нейронных сетей. Описываются принципы работы CUDA-программирования: блоки и нити (threads), сетки блоков, управление памятью и синхронизация. Приводятся примеры того, как CUDA-ядра ускоряют матричные умножения, свёртки и другие операции, лежащие в основе глубокого обучения. Также обсуждаются специализированные ядра — Tensor Cores, которые ещё больше увеличивают скорость вычислений при использовании смешанной точности (mixed precision).

Тема 21. Параллелизм

Рассматривается параллелизм данных (data parallelism) — подход, при котором одни и те же операции выполняются параллельно над разными частями данных. Объясняется, как модель копируется на несколько устройств (обычно GPU), каждое из которых обрабатывает свой мини-батч, после чего градиенты объединяются и синхронизируются. Описываются основные реализации: `DataParallel`, `DistributedDataParallel` в PyTorch, а также преимущества и ограничения такого подхода при масштабировании моделей.

Тема 22. Параллелизм

Рассматривается параллелизм модели (model parallelism) — метод, при котором разные части модели размещаются на разных устройствах. Объясняется, как слои модели или даже отдельные матричные операции могут быть распределены между несколькими GPU для экономии памяти и повышения вычислительной эффективности. Обсуждаются такие техники, как вертикальный и горизонтальный параллелизм, `pipeline parallelism`, а также их применение в обучении больших моделей, включая Megatron-LM и DeepSpeed.

Тема 23. Законы увеличения

Рассматриваются эмпирические закономерности, выявленные при обучении языковых моделей разного масштаба. Объясняется, как метрики качества (например, перплексия или точность) изменяются в зависимости от роста параметров модели, размера обучающего датасета и объёма вычислений. Приводятся графики и исследования (например, от OpenAI и DeepMind), показывающие логарифмическое улучшение качества по мере увеличения ресурсов. Обсуждаются ограничения этих закономерностей и их практическое значение при планировании обучения моделей.

Тема 24. Законы увеличения

Рассматривается, как законы масштабирования используются на практике для прогнозирования производительности моделей и оптимального распределения ресурсов. Объясняется, как, зная зависимости между качеством, размером модели и объёмом данных,

можно предсказать, какие параметры модели дадут наилучший результат при фиксированном бюджете. Обсуждаются примеры использования scaling laws при проектировании крупных моделей (например, GPT-3, Chinchilla) и подборе баланса между объёмом модели и количеством обучающих шагов.

Тема 25. Инференс

Рассматривается процесс инференса — применение обученной модели для генерации или предсказания результатов на новых данных. Объясняется, как происходит прохождение входной информации через нейронную сеть без обновления весов, и какие параметры (например, размер контекста, режим генерации) влияют на скорость и качество вывода. Описываются различные методы генерации текста, включая жадный поиск (greedy decoding), выбор с температурой (sampling with temperature), top-k и top-p sampling, а также beam search. Обсуждаются вопросы оптимизации инференса: кэширование, квантование, компиляция модели (например, с помощью ONNX, TensorRT) и распределённый вывод на нескольких устройствах.

Тема 26. Оценка моделей

Рассматриваются методы оценки качества языковых моделей после обучения. Объясняется значение различных метрик, таких как перплексия (perplexity), точность (accuracy), полнота (recall), F1-мера, BLEU, ROUGE и другие, в зависимости от задачи (генерация текста, классификация, перевод и т.д.). Описывается важность разбиения данных на обучающую, валидационную и тестовую выборки. Также обсуждаются методы человеческой оценки (human evaluation), включая оценку связности, грамматичности и соответствия контексту. Поднимаются вопросы переобучения и обобщающей способности модели, а также практики создания надёжного и справедливого бенчмарка для сравнения моделей.

Тема 27. Данные

Рассматривается ключевая роль данных в обучении языковых моделей. Объясняется, почему качество, разнообразие и репрезентативность данных напрямую влияют на способность модели к обобщению. Обсуждаются типы шума в данных (опечатки, дубли, нерелевантные тексты) и их влияние на результат. Приводятся практики отбора и фильтрации данных, включая автоматические и ручные методы очистки, а также важность соблюдения этических норм и лицензирования.

Тема 28. Данные

Рассматриваются технические и организационные аспекты подготовки масштабных датасетов для обучения языковых моделей. Объясняется, как осуществляется сбор, агрегация, очистка и разметка данных из различных источников (веб, книги, Википедия, форумы и др.). Описываются подходы к дедупликации, токенизации, балансировке тем и доменов. Также обсуждается создание специализированных подмножеств данных (например, для дообучения или адаптации к задаче), а также автоматизация pipeline'a подготовки данных.

Тема 29. Состязательное обучение

Рассматриваются базовые принципы состязательного обучения — подхода, при котором две модели обучаются одновременно в противоборстве: генератор пытается создавать правдоподобные данные, а дискриминатор — отличать реальные от сгенерированных. Объясняется, как такой процесс помогает улучшить качество генерации и повысить устойчивость модели к ошибкам. Приводится пример архитектуры GAN (Generative Adversarial Networks) и её применение в различных задачах.

Тема 30. Состязательное обучение

Обсуждается применение состязательного обучения в задачах обработки естественного языка и обучении языковых моделей. Описываются методы улучшения качества генерации текста с помощью состязательных подходов, такие как обучение с подкреплением с использованием

дискриминатора (adversarial RL), генерация ответов и оценка их качества. Рассматриваются вызовы и методы борьбы с нестабильностью обучения и режимом коллапса. Приводятся современные исследования и примеры использования в GPT и других трансформерах.

Тема 31. Разные виды внимания

Рассматриваются различные типы механизмов внимания, применяемых в нейронных сетях для улучшения обработки последовательностей. Объясняются отличия и особенности таких видов внимания.

Тема 32. Модель DeepSeek R1

Рассматривается архитектура и ключевые особенности модели DeepSeek R1 — современной языковой модели, ориентированной на задачи поиска и понимания естественного языка. Объясняется, как DeepSeek R1 сочетает методы глубокого обучения и механизмы внимания для эффективного извлечения релевантной информации из больших текстовых массивов. Описываются её компоненты: энкодеры, специальные слои для семантического сопоставления запросов и документов, а также методы обучения с использованием пар «вопрос–ответ». Приводятся примеры применения DeepSeek R1 в системах поиска, рекомендаций и анализа текстов.

4.4. Модульная структура дисциплины с распределением весов по формам контролей

Формы контролей	Вес формы (форм) текущего контроля в результирующей оценке текущего контроля (по модулям)		Вес формы промежуточного контроля в итоговой оценке промежуточного контроля		Вес итоговой оценки промежуточного контроля в результирующей оценке промежуточных контролей		Вес итоговой оценки промежуточного контроля в результирующей оценке промежуточных контролей (семестровой оценке)		Весы результирующей оценки промежуточных контролей и оценки итогового контроля в результирующей оценке итогового контроля	
	M1 ¹	M2	M1	M2	M1	M2				
Вид учебной работы/контроля										
Контрольная работа (при наличии)										
Устный опрос (при наличии)										
Тест (при наличии)										
Лабораторные работы (при наличии)										

¹ Учебный Модуль

Письменные домашние задания (при наличии)								
Реферат (при наличии)								
Эссе (при наличии)								
Проект (при наличии)								
Другие формы (при наличии)								
Веса результирующих оценок текущих контролей в итоговых оценках промежуточных контролей								
Веса оценок промежуточных контролей в итоговых оценках промежуточных контролей								
Вес итоговой оценки 1-го промежуточного контроля в результирующей оценке промежуточных контролей								
Вес итоговой оценки 2-го промежуточного контроля в результирующей оценке промежуточных контролей								
Вес результирующей оценки промежуточных контролей в результирующей оценке итогового контроля								
Вес итогового контроля (Экзамен/зачет) в результирующей оценке итогового контроля								
	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$

5. Теоретический блок

5.1. Материалы по теоретической части курса

5.1.1. “Deep Learning” by Ian Goodfellow, Yoshua Bengio, Aaron Courville

5.1.2. “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow” by Geron Aurelien

5.1.3. “Natural Language Processing in Action” by Lane, Howard, and Hapke

5.1.4. “Generative Deep Learning” by David Foster

6. Фонды оценочных средств

6.3. Материалы по практической части курса

6.3.1. Учебно-методические пособия

- «Архитектура трансформеров», авторский учебник
- Методичка «Fine-tuning моделей GPT»

6.3.2. Учебные справочники

- HuggingFace Transformers Docs
- PyTorch Documentation
- NVIDIA CUDA Toolkit Guide

6.3.3. Задачники (практикумы)

- Сборник задач по реализации self-attention
- Практикум: оптимизация моделей и inference

6.3.4. Наглядно-иллюстративные материалы

- Слайды: трансформеры, GPT, LLaMA
- Инфографика: архитектура моделей (GPT, Mistral, DeepSeek)

6.3.5. Прочие материалы

- Лабораторные ноутбуки Jupyter
- YouTube-подборка видеолекций

6.4. Вопросы и задания для самостоятельной работы студентов

- Объясните разницу между self-attention и cross-attention.
- Напишите код токенизации на основе HuggingFace.
- Подберите гиперпараметры для мини-GPT модели.
- Оцените эффективность модели при изменении batch size.

6.6. Образцы контрольных работ, тестов и др. форм контроля

Контрольная работа (фрагмент):

1. Опишите архитектуру GPT.
2. Реализуйте self-attention слой.
3. Чем отличаются autoregressive и masked модели?

Тест (фрагмент):

- ☐ Внимание используется только в энкодере

- ☒ LLaMA — это autoregressive модель

6.7. Перечень экзаменационных вопросов

1. Объясните архитектуру трансформера.
2. В чем разница между GPT и BERT?
3. Что такое positional encoding?
4. Как устроена модель Mistral?
5. Как выбрать гиперпараметры?

6.8. Образцы экзаменационных билетов

Билет №3

1. Опишите процесс токенизации текста.
2. Реализуйте attention-механизм в коде.
3. Какие GPU подходят для инференса больших моделей?

6.9. Образцы практических заданий на экзамене

- Напишите код для расчета attention scores
- Постройте пайплайн тонкой настройки GPT на датасете новостей
- Рассчитайте память, требуемую для обучения LLaMA-2 на 1 GPU

6.10. Банк тестовых заданий для самоконтроля

- В чем преимущество механизма внимания над RNN?
- Укажите, какие задачи решаются autoregressive моделями
- Что такое CUDA ядра?

6.11. Методики решения и ответы к тестам

Пример:

- Вопрос: Что такое self-attention?
- Методика: Рассмотреть входной тензор, вычислить Query, Key, Value, затем Attention scores, применить softmax и итоговую агрегацию.

- Ответ: механизм, при котором каждый элемент входа взаимодействует с остальными через веса внимания.

7. Методический блок

7.1. Методика преподавания

Курс ориентирован на подготовку студентов к практической работе с современными нейросетевыми архитектурами, в первую очередь трансформерными и языковыми моделями, включая их реализацию, настройку и оценку. Преподавание построено по принципам интеграции теории, практики и исследовательского подхода.