Міністерство освіти і науки України
Національний університет «Києво-Могилянська академія»
Факультет інформатики
Кафедра інформатики

Кваліфікаційна робота

освітній ступінь – бакалавр

на тему: «**ТОЧНЕ НАЛАШТУВАННЯ LLM ДЛЯ ОБРАНИХ ПРЕДМЕТНИХ ОБЛАСТЕЙ**»

Виконав: студент 4-го року навчання,
Освітньої програми «Комп'ютерні науки», 122

Остролуцький Андрій Борисович

Керівник Андрощук, М.В_____

старший викладач кафедри інформатики

Рецензент _____

(прізвище та ініціали)

Кваліфікаційна робота захищена

з оцінкою _____

Секретар ЕК _____

«____» _____ 20____ р.

Київ – 2025

ГРАФІК ПІДГОТОВКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

№ з/п	***ПЕРЕЛІК РОБІТ***	Термін виконання	Дата ознайомлення наукового керівника	Підпис наукового керівника	Примітки
1.	Вибір теми, затвердження її на засіданні кафедри та закріплення наукового керівника Узгодження календарного графіка підготовки кваліфікаційної роботи. Ознайомлення студента з критеріями оцінювання кваліфікаційної роботи (п. 8.5).	жовтень			
2.	Вивчення джерел літератури, матеріалів архівів, періодичних видань, збір та узагальнення фактів, даних	жовтень – листопад			
3.	Складання плану каліф. роботи та узгодження з науковим керівником	листопад			
4.	Написання розділів роботи	листопад – березень			
5.	Проміжний контроль виконання роботи	лютий			
6.	Написання кваліфікаційної роботи в цілому, ознайомлення з її першим варіантом наукового керівника	січень – березень			
	Розділ 1 (постановка проблеми, теоретичні основи, огляд літературних джерел)	листопад – грудень			
	Розділ 2 (аналітично-дослідницька частина)	грудень – січень			
	Розділ 3 (обґрунтування вибору предметних областей і опис наборів даних)	січень – лютий			
	Розділ 4 (проектно-рекомендаційна частина)	лютий – березень			
7.	Повне завершення написання кваліфікаційної роботи, оформлення її згідно з вимогами й подання на відгук науковому керівнику	квітень – початок травня			
8.	Подання кваліфікаційної роботи для перевірки письмових робіт студентів НаУКМА на відповідність вимогам академічної доброчесності	середина травня			
9.	Подання на зовнішню рецензію	середина травня			
10.	Підготовка до захисту кваліфікаційної роботи на засіданні кафедри: написання доповіді та виготовлення ілюстративного матеріалу	**до 25 травня**			
11.	Попередній захист кваліфікаційної роботи на засіданні кафедри	**26 травня**			
12.	Подання кваліфікаційної роботи на кафедру з усіма супроводжувальними документами	**до 29 травня**			
13.	Публічний захист кваліфікаційної роботи перед екзаменаційною комісією	згідно з розкладом роботи ЕК			

Графік узгоджено «___» _____ 20___ р.

Науковий керівник Андрощук Максим Віталійович (ПІБ)

Виконавець кваліфікаційної роботи Остролуцький Андрій Борисович (ПІБ)

Національний університет «Києво-Могилянська академія»

Факультет інформатики
Кафедра інформатики
Освітній ступінь Бакалавр
Напрям підготовки 12 – Інформаційні технології
Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри інформатики
к.ф.-м.н., доц. Гороховський С.С
"____" _____2025 року

З А В Д А Н Н Я
ДЛЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ

Остролуцькому Андрію Борисовичу

1. Тема роботи «ТОЧНЕ НАЛАШТУВАННЯ LLM ДЛЯ ОБРАНИХ ПРЕДМЕТНИХ ОБЛАСТЕЙ»

керівник роботи Андрощук, М. В. старший викладач кафедри інформатики

затверджені наказом вищого навчального закладу від «__»_____20__ року №___

2. Строк подання студентом роботи _____

3. План роботи

Зміст
Анотація
Перелік прийнятих скорочень
Вступ
Розділ 1. СТАНОВЛЕННЯ СУЧАСНИХ LLM
Розділ 2. ТЕОРЕТИЧНІ ОСНОВИ МЕТОДУ НИЗЬКОРАНГОВОГО АДАПТУВАННЯ ТА АРХІТЕКТУРИ LLM
Розділ 3. ВИБІР ПРЕДМЕТНИХ ОБЛАСТЕЙ ДЛЯ ТОЧНОГО НАЛАШТУВАННЯ ТА ОПИС ДАНИХ
Розділ 4. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТОЧНОГО НАЛАШТУВАННЯ
Висновки

ЗМІСТ

АНОТАЦІЯ

Кваліфікаційну роботу присвячено точному налаштуванню LLM для обраних предметних областей: науково-теоретична медицина, клінічна медицина та фінанси. В роботі проведено аналіз сучасних методів точного налаштування та огляд архітектури трансформерних моделей. Для практичної частини було використано метод низькорангового адаптування та модель Gemma 2 2B. Запропоновано двоетапний підхід до налаштування для теоретичної медицини, що забезпечив приріст точності на 9,19% (категорія «Professional medicine» MMLU) і покращення на 9,75% (MedQA). У клінічній медицині досягнуто зростання BERTScore Precision на 9,44%. Модель для фінансового аналізу показала 95,38% збігу з анотаціями експертів. Додатково було створено демонстраційні застосунки для клінічної і фінансової сфер як приклад практичного використання моделей.

Ключові слова: великі мовні моделі, точне налаштування, LoRA, медицина, фінанси.

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ

LLM (Large Language Model) – велика мовна модель;

NLP (Natural Language Processing) – обробка природної мови;

SLM (Statistical Language Model) – статистична мовна модель;

NLM (Neural Language Model) – нейронна мовна модель;

FT (Fine-Tuning) – точне налаштування;

PT (Pre-Training) – попереднє тренування;

FFT (Full Fine-Tuning) – повне точне налаштування;

PFT (Partial Fine-Tuning) – часткове точне налаштування;

LoRA (Low-Rank Adaptation) – низькорангове адаптування.

ВСТУП

У сучасному світі інформаційних технологій, LLM набувають все більшого значення завдяки своїй здатності генерувати релевантний і контекстуально збагачений текст та виконувати складні завдання з різноманітних областей. Проте, якщо розглядати LLM у вирішенні вузькопрофільних задач, вони можуть показувати недостатню точність, що обмежує їх використання у сферах, таких як медицина або фінанси, де некоректна або узагальнена відповідь може мати серйозні наслідки.

У сучасних дослідженнях велика увага приділяється точному налаштуванню. Проте, попри всебічний аналіз, питання щодо ефективності FT для реальних, практично значущих задач з використанням обмежених ресурсів залишається відкритим. За таких умов вибір конкретного методу FT є принциповим.

Актуальність дослідження, що було проведено у цій кваліфікаційній роботі, була зумовлена нагальною потребою адаптування LLM до специфіки окремих областей, що здатне забезпечити їх ефективніше використання у практичних задачах.

Метою цієї роботи є розробка та експериментальна реалізація точного налаштування LLM в обраних предметних областях: науково-теоретична медицина, клінічна медицина та фінанси.

Для досягнення мети, було поставлено декілька завдань:

1. Виконати аналіз сучасних підходів до адаптації LLM для спеціалізованих завдань та обрати найбільш доцільний з них.
2. Розглянути теоретичні засади низькорангового адаптування, обґрунтувати вибір LLM для її подальшого використання та детально дослідити архітектуру LLM для усвідомленого підходу до практичного точного налаштування.
3. Обрати та обробити необхідні набори даних, які відповідають зазначеним предметним областям і необхідним критеріям.

4. Провести точне налаштування LLM для кожної предметної області; використати двоетапне FT для однієї з них з метою покращення якості.
5. Провести експериментальне порівняння базової та адаптованої моделей та реалізувати демонстраційні вебзастосунки для наочної оцінки роботи.

Об'єктом даної роботи є процес точного налаштування великої мовної моделі для предметно-орієнтованих завдань.

У кваліфікаційній роботі було використано декілька ключових ***методів***: аналіз літературних джерел, обрання методу FT мовної моделі, обрання показових наборів даних для тренування і тестування моделі, обрання та використання метрик для оцінки продуктивності, інтеграція адаптованої моделі у вебзастосунок як приклад її практичного використання.

Робота складається з чотирьох розділів. **Перший розділ** присвячено аналізу становлення сучасних LLM. Розглянуто типи FT та їх переваги над іншими методами. **Другий розділ** висвітлює теоретичне обґрунтування методу LoRA, причини вибору моделі Gemma 2 2B та теоретичний огляд графіків зниження темпу навчання для покращення тренування. **Третій розділ** охоплює опис обраних предметних областей з детальною характеристикою обраних наборів даних. **Четвертий розділ** являє собою опис проведення практичної частини цієї роботи, представлення результатів FT, а також реалізацію вебзастосунків для наочного представлення результатів адаптування.

Практичне значення одержаних результатів полягає в створені адаптованих мовних моделей, що здатні ефективно працювати в спеціалізованих предметних областях, де базові LLM демонструють недостатню точність. Також, вони можуть стати основою до створення чат-ботів і систем підтримки прийняття рішення у фінансових або медичних установах. В контексті медицини створені моделі можуть допомогти користувачу зробити інформоване рішення та звернутися до відповідного спеціаліста. В контексті фінансів точно налаштована модель може надати користувачу швидкий аналіз поведінки ринку для інформованих інвестиційних рішень.

РОЗДІЛ 1
СТАНОВЛЕННЯ СУЧАСНИХ LLM

1.1. Історичний розвиток мовного моделювання

Щоб краще зрозуміти сучасні підходи до NLP, важливо простежити еволюцію методів мовного моделювання – від статистичних до нейронних та трансформерних архітектур. Кожен з цих етапів сформував важливі концепції, які стали основою для створення LLM. Розгляд цієї еволюції дає можливість усвідомити причини переходу від однієї парадигми до іншої, а також дає ширше уявлення, чому були зроблені відповідні архітектурні рішення в сучасних мовних моделях. Таким чином, в наступних підрозділах буде розглянуто основі етапи розвитку мовних моделей, що мали вирішальний вплив на сучасний стан NLP.

1.1.1. Статистичні мовні моделі

Перші успішні SLM почали з'являтися у 1980-1990-х роках. Ці математичні моделі були спробою виявлення статистичних закономірностей природної мови для покращення її автоматичної обробки. Суть статистичного мовного моделювання полягає в оцінюванні ймовірнісного розподілу різних мовних одиниць **[62]**.

Сфера застосування SLM була широкою і включала такі задачі, як машинний переклад, тегування частин мови, сегментація слів і речень, поверхневий синтаксичний аналіз та виправлення орфографії **[62]**. Крім того, було досліджено використання SLM у сфері інформаційного пошуку, а саме оцінювання ймовірності відповідності документів до запиту користувача **[55]**.

Щоб зрозуміти основні принципи функціонування SLM, необхідно розглянути конкретний приклад. Нехай є речення *s*=«*Київ є столицею України*». Ймовірність появи цього речення визначається як послідовний добуток умовних

ймовірностей слів: $P(s) = P(\text{Київ}) \cdot P(\text{є} \mid \text{Київ}) \cdot P(\text{столицею} \mid \text{Київ}, \text{є}) \cdot P(\text{України} \mid \text{Київ}, \text{є}, \text{столицею})$.

Метод максимальної правдоподібності дозволяє оцінити ці ймовірності на основі частоти фраз у великому корпусі текстів:

$$P(w_i \mid w_1, w_2 \dots w_{i-1}) = \frac{C(w_1, w_2 \dots w_i)}{C(w_1, w_2 \dots w_{i-1})}$$

Формула 1.1 – Оцінка умовної ймовірності методом максимальної правдоподібності

де C позначає частоту появи послідовності $w_1, w_2 \dots w_n$ в датасеті.

SLM, таким чином, оцінює ймовірність появи слова на основі *n* попередніх слів, що дозволяє передбачити слова або оцінювати правдоподібність речень. Такі моделі також називають n-грамними.

Побудови SLM вимагає багато обчислювальних ресурсів та пам'яті, тому що кількість можливих речень зростає експоненційно зі збільшенням кількості слів в словнику. Через це на практиці розмір контексту *n* зазвичай обмежують до 2 або 3 слів, що значно скорочує обсяг даних для зберігання, але водночас унеможливлює врахування довгого контексту.

SLM знайшли своє обмежене застосування в деяких завданнях NLP, але не були спроможними генерувати змістовні тексти природною мовою. Водночас їхнє створення стало важливим поштовхом для подальших досліджень у цьому напрямі.

1.1.2. Нейронні мовні моделі

NLM, незважаючи на не широке використання, зробили свій внесок для сучасних LLM: їх архітектура була ближчою до сьогоднішніх технологій, порівняно з іншими розробками.

NLM використовують нейронні мережі для передбачення ймовірності наступних слів у тексті. Ця інновація забезпечила можливість враховувати довший контекст при генерації тексту, ніж це було можливо для n-грамних моделей.

В основі NLM лежить концепція розподілених векторних представлень слів (векторів-слів). Будучи числовим представленням слова у багатовимірному просторі, вони відіграють ключову роль у кодуванні семантичного значення. Завдяки ним можна легко охопити різні семантичні аспекти слів. Вектори слів, які позначають синоніми, зазвичай розташовані близько один до одного, тоді як слова з різними значеннями знаходяться на більшій відстані. Наприклад, вимірюючи різницю між ними за допомогою косинусної подібності, можна визначити, що синоніми матимуть значення ближчі до 1.

Відомим інструментом для обрахування векторів-слів є Word2Vec **[11]**. Більше того, вектори-слова дозволяють нам знаходити не тільки синоніми, а і слова зі схожими семантичними зв'язками виконуючи прості алгебраїчні операції **[12]**.

Вектори-слова сприяють ефективнішому навчанню NLM. Завдяки розподіленим векторним представленням, нові комбінації слів можуть отримувати високі ймовірності, навіть якщо вони не були присутні в тренувальному корпусі, що підвищує рівень узагальнення **[5]**.

Для прогнозування ймовірностей в NLM на останньому шарі застосовується функція Softmax, яка перетворює необроблені значення (логіти) на ймовірнісні оцінки.

$$Softmax(z_i) = \frac{\exp(z_i)}{\Sigma_{j=1}^{|V|} \exp(z_j)}$$

Формула 1.2 – Функція Softmax для перетворення логітів на ймовірнісні оцінки

де z_i – логіт, пов'язаний з і-тим словом із словника V.

1.1.3. Трансформери

У 2017 році з публікацією наукової статті «Attention Is All You Need» **[2]** світ познайомився з революційною архітектурою під назвою трансформер. Саме на її основі була розроблена GPT-3, яка стала справжнім проривом у галузі NLP і популяризувала ці технології серед широкого загалу. З її появою часто пов'язують початок епохи LLM **[25]**, хоча в наукових роботах цей термін часто застосовуються до моделей, створених раніше. Окрім GPT-3, більшість сучасних LLM мають трансформерну архітектуру (GPT-4 **[47]**, PaLM **[50]**, LLaMa **[40]** та багато інших).

Трансформер продемонстрував проривну продуктивність на таких завданнях як машинний переклад **[2]**, генерація документів **[18]**, синтаксичний аналіз **[32]**. Архітектура трансформерів забезпечує ефективніше опрацювання більш довгих залежностей в тексті, ніж такі альтернативи, як рекурентні мережі **[28]**.

Трансформерна модель Gemma **[16]** була використана в рамках цієї роботи. На її прикладі архітектура трансформерів буде представлена в другому розділі.

1.2. Визначення LLM

Станом на сьогодні найвищим досягненням в мовному моделюванні вважаються LLM. LLM – це модель машинного навчання, створена для задач NLP, головна мета якої полягає в аналізі тексту та генерації нових текстових фрагментів на його основі. Для досягнення цієї функціональності такі моделі проходять довге тренування на великому текстовому корпусу, який може складатися з книг, статей, вебресурсів і т.д. Для зберігання великої кількості інформації LLM зазвичай мають мільярди параметрів, через що в терміні застосовується слово «великий». Більше того, мовні моделі зростають з кожним роком. Наприклад, одна з версій GPT-2, реліз якої відбувся 2019 року, має 1,5 мільярди параметрів, а для її тренування було використано приблизно 40 ГБ текстових даних. Для порівняння, найбільша версія моделі GPT-3 **[34]**, випущена 2020 року, має 175 мільярдів параметрів і використала

570 ГБ даних для навчання **[25]**. Навчання GPT-3 проводилось на 10 000 графічних процесорах V100 протягом 14,8 днів **[7]**, із приблизною вартістю навчання понад 4,6 мільйона доларів **[38]**.

1.3. Парадигма попереднього навчання з точним налаштуванням

Початковий етап навчання LLM, після того як параметри були ініціалізовані випадковими значеннями і модель ще не здатна до змістовної генерації, називається PT. Під час цієї фази модель засвоює структуру мови, взаємозв'язки між словами та інші лінгвістичні особливості, формуючи попередньо навчену модель, також відому як фундаментальна або базова модель, з оновленими вагами **[71]**.

PT авторегресійної мовної моделі з параметрами Ф полягає у максимізації функції правдоподібності, що відповідає умовній ймовірності кожного токена за попередніми в послідовностях взятих з великої тренувальної колекції тексту. Це можна описати за допомогою наступної формули:

$$\max_{\Phi} \sum_{s \in D} \sum_{t=1}^{|s|} \log P_{\Phi}(w_t \mid w_{<t})$$

Формула 1.3 – Цільова функція для PT мовної моделі

де $D = \{s_1, s_2, \dots, s_M\}$ – велика колекція неанотованого тексту, в якому кожне речення s_i – це послідовність токенів $\{w_1, w_2, \dots, w_T\}$. В результаті PT модель отримує початкові параметри Φ_0.

Метод PT використовувався в різних нейронних моделях для різних завдань, таких як класифікація зображень **[75]**, розпізнання мовлення **[73]**, розрізнення сутностей **[35]** і машинний переклад **[57]**.

Втім, функціональність попередньо навчених авторегресійних LLM обмежується здатністю продовжувати наданий текст та їхнє використання на практиці є дуже обмеженим. Для розв'язання цієї проблеми, разом із впровадженням архітектури трансформерів, була запропонована парадигма попереднього навчання з подальшим точним налаштуванням (PT+FT парадигма). FT передбачає додаткове навчання базової моделі на меншому, цільовому наборі даних з метою адаптації її до конкретного завдання в галузі NLP. Прикладами таких завдань є слідування інструкціям користувача (що є ключовим для створення мовних асистентів), аналіз тональності тексту, машинний переклад або класифікація тексту.

Незважаючи на високі результати у виконанні широкого спектра NLP-завдань, такі моделі часто демонструють обмежену якість у спеціалізованих предметних галузях. Основною причиною цього є недостатність специфічних знань у даних, використаних на етапі PT. У таких випадках FT також є корисним, оскільки дає змогу суттєво покращити точність моделі в межах конкретної області **[66, 71]**.

1.4. Основні переваги точного налаштування

На противагу тренуванню повністю нової моделі «з нуля», ключовою перевагою FT є можливість швидко отримати адаптовану модель під потрібний випадок використання на основі базової моделі. Серед інших переваг є потреба в значно меншій кількості обчислювальних ресурсів, що робить FT значно дешевшим ніж PT.

Ще однією важливою проблемою, яку розв'язує PT+FT парадигма, є можливість ефективного використання обмеженої кількості анотованих даних для розв'язання спеціалізованих завдань. До впровадження цієї парадигми дослідники були змушені навчати моделі у дискримінативний спосіб, використовуючи лише наявні дані для керованого тренування. Оскільки таких даних зазвичай недостатньо, дискримінативно натреновані моделі часто демонстрували незадовільну продуктивність. Водночас, PT є некерованим і потребує лише

достатньої кількості неанотованого тексту. Тому навіть за умов відсутності значного обсягу розмічених даних, якісні представлення, отримані за рахунок PT, суттєво підвищують якість LLM після FT.

Також у багатьох індустріях, таких як сфера охорони здоров'я, фінансів або права, існують суворі правила поводження з чутливою інформацією. FT мовної моделі, яка натренована на даних конкретної організації чи галузі, може дати змогу отримати модель, яка буде готова до розгортання на власному сервері. Використовуючи таку модель, організація покращує конфіденційність даних та зменшує ризик їхнього витоку.

Наостанок варто зазначити, що для мовних моделей в науковому просторі довго не було консенсусу щодо найбільш ефективного способу переносити ознаки вивчених представлень на цільове завдання. Ці способи включали внесення специфічних для завдання змін в архітектуру **[63, 9]**. Також для цього використовувались складні схеми навчання **[27]** та додавання допоміжних цілей навчання **[58]**. Згодом, на заміну цим методам прийшла PT+FT парадигма, яка дозволяє отримати універсальну, агностичну до типу завдання модель, яку можна адаптувати до широкого спектру застосувань без зміни її архітектури. Це є ще одною перевагою цієї парадигми.

1.5. Основні підходи до точного налаштування

FT може реалізовуватись за допомогою різних методів. Попри те, що FT концептуально є значно ефективнішим за повноцінне навчання нової моделі, дослідники продовжують шукати способи підвищення його ефективності. Існує низка підходів до FT, які відрізняються між собою архітектурними особливостями. Далі представлено такі основні методи.

1.5.1. Повне точне налаштування

FFT – це процес додаткового навчання всієї моделі, тобто коригування всіх її параметрів, на новому наборі даних після етапу PT **[66]**. Визначальною

характеристикою цього підходу є те, що під час навчання усі вагові коефіцієнти та зсуви всіх шарів моделі оновлюються на етапі зворотного поширення помилки (backpropagation).

Одними з перших, хто застосував FFT для мовної моделі були Говард і Рудер **[27]**, які точно налаштовували всі шари мовної моделі довготривалої короткочасної пам'яті з різною швидкістю навчання для покращення завдання класифікації тексту. Також визначною є робота Радфольда та інших **[28]**, котрі точно налаштували трансформерну модель.

Математично FFT можна представити наступним чином. Нехай дана попередньо тренована авторегресійна мовна модель $P_\Phi(y|x)$ з параметрами Ф. Завдання або дані, під які ми хочемо адаптувати модель, представлені тренувальним датасетом, який складається з пар контексту та цілі: $Z = \{(x_i, y_i)\}_{i=1,\dots,N}$, де обидва x_i та y_i – послідовності токенів. Після етапу FT, описаного формулою 1.3, модель має параметри Φ_0. Ми оновлюємо модель $\Phi_0 + \Delta\Phi$ шляхом слідування градієнту з метою максимізації функції правдоподібності:

$$\max_{\Phi} \sum_{(x,y)\in Z} \sum_{t=1}^{|y|} \log P_\Phi(y_t|x, y_{<t})$$

Формула 1.4 – Цільова функція для FFT мовної моделі

Хоча FFT, як правило, приносить найкращий результат і дає найбільшу гнучкість під час адаптації моделі під певне завдання або предметну область, з ним пов'язана проблема катастрофічного забування. Катастрофічне забування, феномен вперше ідентифікований в **[44]**, залишається фундаментальною проблемою в тренуванні моделей глибинного навчання. Суть цієї проблеми полягає в тому, що під час тренування моделі для багатьох задач по порядку, вони схильні забувати попередньо отримані знання, що веде до значного падіння в точності на попередньо навчених завданнях **[21]**.

Окрім того, FFT є ресурсоємним процесом, оскільки вимагає оновлення кожного параметра моделі, а його використання з невеликими наборами даних

збільшує ймовірність перенавчання – однієї з головних проблем машинного навчання.

Для подолання цих проблем запропоновано ряд рішень, наприклад адаптивно обирати швидкість навчання по-різному для кожного з шарів **[52]**, а також ряд інших технік, розглянутих в підрозділах нижче.

1.5.2. Часткове точне налаштування

PFT полягає в тренуванні не всіх параметрів моделі, але лише частини з них. Під час такого навчання деякі шари моделі заморожуються. PFT вирішує ряд проблем, пов'язаних з FFT: зменшується ризик катастрофічного забування і перенавчання та необхідна менша кількість обчислювальних ресурсів.

Питання вибору шарів для фіксації або співвідношення параметрів, які необхідно тренувати, є широко розкритим в науковій літературі. Так, згідно з **[59]**, для багатьох завдань, лише останні декілька шарів змінюються найбільше для моделі BERT **[19]**. Лі та інші в **[36]**, досліджуючи BERT та RoBERTa **[60]**, встановили, що лише четверта частина з останніх шарів має бути точно налаштована, щоб досягти 90% оригінальної якості.

Автори **[13]** показали, що, заморожуючи 25% або навіть 50% відсотків нижніх шарів таких трансформерних моделей, як Gemma, Phi-2 та MiniCPM, можна досягти продуктивності такої ж або навіть кращої ніж FFT. В той же час, PFT на 30-50% зменшило витрату пам'яті і пришвидшило швидкість тренування на 20-30%.

Для подолання проблеми катастрофічного забування, в **[23]** представлений один з варіантів часткового налаштування, за якого половина параметрів не оновлюється. Це дозволило відновити оригінальні знання Llama 2 під час FT.

1.5.3. Налаштування адаптерів

Налаштування адаптерів – метод FT, який полягає додаванні адаптерів між шарами LLM **[51]**. Зазвичай, це невеликі нейронні мережі. Тренування відбувається

саме на цих адаптерах, тоді як оригінальні ваги моделі залишаються фіксованими. Цей метод, як і PFT, є більш ефективним. Окрім цього, він дозволяє легко заміняти адаптери і отримувати повністю нову адаптовану модель.

Математично це можна представити так. Нехай є мовна модель P_{Φ}, а параметри моделі $\Phi = (w, v)$, де w – зафіксовані параметри базової моделі, а v – параметри адаптерів, які ми навчаємо. На початку v ініціалізується так, що модель з адаптерами поводиться так само, як і базова $P_{w,v} \approx P_w$. Тоді метод налаштування адаптерів передбачає лише тренування параметрів v.

Проте цей метод має недолік – додаткові адаптери збільшують час генерації через додаткові обчислення.

1.5.4. Низькорангове адаптування

LoRA – це метод FT, який передбачає повну фіксацію попередньо тренованих ваг моделі і впровадження навчальних матриць з низьким рангом в шари архітектури LLM.

Такий метод вирішує три головні проблеми FFT, згадані раніше: ризик катастрофічного забування, перенавчання та потреба у великій кількості обчислювальних ресурсів. Наприклад, для GPT-3, яка має 175 мільярдів параметрів, з LoRA кількість тренованих параметрів може зменшитись в 10000 разів, що зменшує вимоги до графічних процесорів в три рази **[42]**.

Так само як і налаштування адаптерів, LoRA дозволяє заміняти матриці, отримані шляхом рангової декомпозиції, і отримувати нову адаптовану модель. Також метод LoRA вирішує проблему налаштування адаптерів пов'язану з збільшеним часом генерації, оскільки після тренування навчальні матриці можна поєднати з оригінальними шарами моделі. Нарешті, FT з LoRA продемонструвала однакову або навіть кращу якість аніж FFT на багатьох LLM **[42]**.

1.5.5. Налаштування підказок

Налаштування підказок – один з видів FT, за якого всі параметри моделі залишаються незмінними, а під час етапу зворотного поширення помилки тренуються спеціальні токени, які додаються на початок текстової підказки. Ці токени налаштовуються для представлення інформації з набору анотованих прикладів. Таке налаштування є набагато продуктивнішим ніж звичайні техніки конструювання підказок, такі як підказка з декількома прикладами **[37]**.

1.5.6. Виділення ознак з подальшим точним налаштуванням

Метод виділення ознак передбачає використання LLM для отримання вбудованих векторів, які потім можна використати як ознаки для моделей машинного навчання. Цей метод і FT є видами передавального навчання **[54]**, оскільки обидва методи передбачають використання попередньо тренованої моделі для нових завдань. Зазвичай метод виділення ознак використовується для задач класифікації.

Для першого етапу виділення ознак вхідний текст необхідно перетворити у векторні представлення. Зазвичай, це досягається за допомогою моделей лише з кодувальником, таких як BERT або RoBERTa. Після отримання числових векторів для другого етапу можна використати класичні моделі машинного навчання, такі як логістична регресія, випадковий ліс або опорно-векторні машини. Також можна використати глибинну мережу для розподілу текстів у відповідні класи **[65]**.

Згідно з **[14]**, використання базової моделі BERT (оригінальні шари якого заморожені під час навчання) для виділення ознак для класифікації текстів може показати кращі результати, ніж точно налаштований аналог. Проте інші роботи вказують на те, що FT є демонструє кращу продуктивність **[27]**.

Однак ці два методи не є взаємовиключними. Для більшої якості їх можна поєднати в гібридному підході, коли спочатку модель використовується як фіксований кодувальник, а далі розморожується для FT разом з класифікатором.

Висновки до розділу 1

У першому розділі було здійснено комплексний огляд розвитку LLM – від SLM до сучасних трансформерних архітектур. Окрім цього, розглянуто парадигму PT+FT, що дозволяє адаптувати LLM до конкретних задач і предметних областей. Описано основні підходи до FT, включаючи повне, часткове, налаштування адаптерів, підказок та методи LoRA і виділення ознак. Таким чином, цей розділ створює фундамент для аналізу архітектури сучасних LLM та вибору найоптимальнішого методу FT, що розглядатиметься в наступному розділі.

РОЗДІЛ 2
ТЕОРЕТИЧНІ ОСНОВИ МЕТОДУ НИЗЬКОРАНГОВОГО АДАПТУВАННЯ ТА АРХІТЕКТУРИ LLM

2.1. Теорія низькорангового адаптування як методу точного налаштування

З огляду на перелічені в минулому розділі переваги методу LoRA – значне зменшення кількості тренованих параметрів, зниження ризику катастрофічного забування та перенавчання, скорочення обчислювальних витрат, збереження оригінального часу генерації та якості тренування однакової або кращої за FFT – цей метод був обраний як основний підхід до FT в рамках цієї роботи.

Метод LoRA будується на гіпотезі, що зміна ваг під час адаптації моделі має низький внутрішній ранг **[42]**. Це означає, що під час навчання під нове завдання або дані, вагові матриці змінюються не довільно, а у певному напрямку, який не займає багато вимірів. Тому такі зміни можна апроксимувати за допомогою більш меншої структури. У випадку LoRA цими структурами є матриці з низьким рангом.

Це значить, що для попередньо натренованої вагової матриці $W_0 \in \mathbb{R}_{d\times k}$ ми обмежуємо її оновлення, натомість тренуючи нові матриці $B \in \mathbb{R}_{d\times r}$ і $A \in \mathbb{R}_{r\times k}$, де ранг $r \ll min(d,k)$. Тоді зміна ваг – $\Delta W = BA$. На рисунку 2.1 схематично зображено адаптацію ваг у методі LoRA через додавання добутку BA. При цьому W_0 залишається замороженим під час FT.

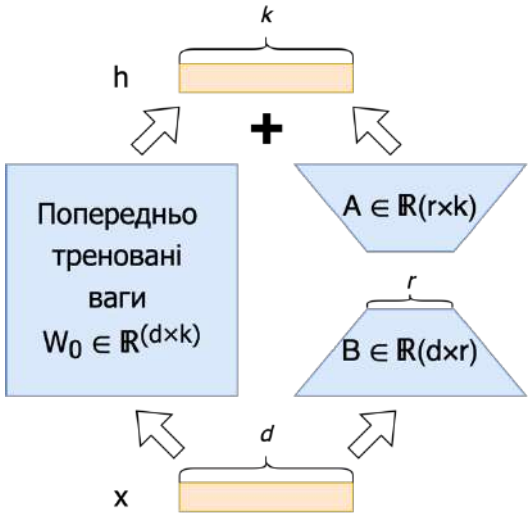

Рисунок 2.1 – Адаптація ваг у методі LoRA

У LoRA використовується випадкова Гаусова ініціалізація для A і нульова для B, тому $\Delta W = 0$ на початку FT. Також метод LoRA передбачає множення ΔW на $\frac{\alpha}{r}$, де α – це спеціальна константа, за допомогою якої ми можемо масштабувати наші зміни ΔW. Під час прямого поширення обидва W_0 і BA множаться на вхід, їх відповідні вихідні вектори підсумуються поелеметно **[42]**:

$$h = W_0 x + \Delta W x = W_0 x + \frac{\alpha}{r} BAx$$

Формула 2.1 – Модифіковане пряме поширення з LoRA

Тоді задача FT за допомогою LoRA до нового завдання зводиться до оптимізації параметрів $\mathcal{A}$ і $\mathcal{B}$, які складаються відповідно з усіх матриць A і B, тоді як основні ваги Ф залишаються фіксованими. Це можна записати у вигляді формули максимізації функції правдоподібності, аналогічної до формули 1.4 для FFT, але з оновлюваними тільки параметрами $\mathcal{A}$ і $\mathcal{B}$:

$$\max_{\mathcal{B},\mathcal{A}} \sum_{(x,y)\in Z} \sum_{t=1}^{|y|} \log(P_{\Phi,\mathcal{B},\mathcal{A}}(y_t|x, y_{<t}))$$

Формула 2.2 – Цільова функція для FT методом LoRA

де $P_{\Phi,\mathcal{B},\mathcal{A}}$ – це модель, яка використовує параметри Ф (заморожені) з низькоранговими доповненнями $\mathcal{B},\mathcal{A}$, які оновлюються; $Z = \{(x_i, y_i)\}_{i=1,\dots,N}$ – набір тренувальних прикладів, в якому обидва x_i та y_i – послідовності токенів.

2.2. Вибір LLM для низькорангової адаптування

Протягом останніх трьох років з'явилося багато відкритих LLM. Відкриті LLM – це моделі з публічно доступними для використання вагами. Наявність таких моделей сприяє ширшій участі в дослідженнях, стимулює інновації через співпрацю технічної спільноти і прискорює розвиток штучного інтелекту для NLP **[56]**. Проте через велике різноманіття доступних моделей, перед науковцями постає питання вибору найкращої моделі для їхніх досліджень.

Таким чином, необхідно сформувати чіткі критерії для вибору LLM для FT за допомогою LoRA. Насамперед, цими критеріями є:

- підтримка LoRA (використання трансформерної архітектури);
- розмір моделі;
- висока якість на загальних тестах продуктивності (стандартизовані тести для вимірювання та порівняння LLM);
- сумісність з існуючими фреймворками.

Розмір є ключовим фактором для вибору моделі, оскільки від нього залежить, який обсяг відеопам'яті потрібен для проведення FT. Обсяг доступних обчислювальних ресурсів зазвичай значно обмежує вибір моделей для дослідження. Також високі показники на тестах продуктивності демонструють високий початковий рівень генерації тексту, що може дати можливість досягти найвищих результатів після проведення FT. А сумісність з існуючими фреймворками, таких як Transformers **[68]**, Transfomer Reinforcement Library (TRL) **[69]** або Parameter-Efficient Fine-Tuning (PEFT) **[53]** для Python, спрощує та пришвидшує реалізацію FT.

Враховуючи необхідність в ефективному FT з обмеженими наявними обчислювальними ресурсами, був проведений аналіз відкритих трансформерних

LLM в діапазоні 1-2 мільярдів параметрів з підтримкою вищезгаданих бібліотек. Такі моделі з легкістю можна точно налаштовувати з LoRA використовуючи ефективний розмір міні-пакетів у межах від 4 до 8. До таких моделей належать Qwen 2.5 1.5B, Llama 3.2 1B та Gemma 2 2B.

Qwen 2.5 1.5B – компактна LLM від компанії Alibaba, представлена у вересні 2024 року. Модель містить 1.54 мільярди параметрів та була натренована на 18 трильйонах токенів, з фокусом на математику та кодування. Модель підтримує контекстне вікно розміром 32 000 токенів **[56]**.

Llama 3.2 1B – LLM від Meta, випущена також у вересні 2024 року. Вона має 1.24 мільярди параметрів і була навчена на 9 трильйонах токенів. Llama 3.2 1B ідеально підходить для завдань підсумовування тексту та перефразування тексту. Обсяг контекстного вікно становить 128 000 **[39]**.

Gemma 2 2B – одна з моделей сімейства відкритих моделей Gemma, випущених 27 червня 2024 року. Вони були розроблені компанією Google використовуючи технології для розробки їхньої більшої LLM під назвою Gemini, з якою моделі сімейства Gemma мають однакові архітектурні компоненти **[4]**. Сама серія складається з трьох моделей, кожна з яких має різну кількість параметрів, серед яких Gemma 2 2B є найменшою – з 2.61 мільярдами. Моделі Gemma 2 були натреновані на 2 трильйонах токенів з використанням техніки дистиляції знань. Ці токени були зібрані з різноманітних джерел, таких як вебдокументи, код, наукові статті **[16]**. Контекстне вікно Gemma 2 2B має довжину в 8192 токен.

На таблиці 2.1 розглянуті результати найпопулярніших тестів продуктивності загальних завдань ARC-C **[67]**, Winogrande **[72]**, HellaSwag **[22]**, IFEval **[29]** вищезгаданих моделей. Результати були взяті з відповідних технічних звітів моделей **[56, 39, 16]**

Таблиця 2.1 – Результати тестів продуктивності для LLM-кандидатів

Назва моделі	ARC-C (25-shot), %	Winogrande (5-shot), %	HellaSwag (10-shot), %	IFEval, %
Qwen 2.5 1.5B	54.7	65	67.9	42.5
Llama 3.2 1B	32.8	Немає даних	Немає даних	59.5
Gemma 2 2B	55.7	71.5	74.6	61.9

Таблиця показує, що Gemma 2 2B демонструє найвищі результати на низці тестах продуктивності для оцінювання загальних завдань серед моделей з однаковим розміром. Завдяки цьому Gemma 2 2B забезпечує вищу початкову якість генерації, що потенційно може дозволити досягти найкращих результатів після FT. Також, згідно з **[16]**, в деяких випадках вона виявилась конкуретноспроможною серед моделей, більших в два рази.

Окрім цього, замість покращення за рахунок більшої кількості тренувальних даних, яке масштабується логарифмічно (наприклад, одна з моделей від Meta AI Llama потребувала до 15 трильйонів токенів для покращення точності на 1-2% **[41]**), для тренування Gemma 2 2B використовувалась техніка під назвою дистиляція знань **[24]**.

Для проведення FT в кваліфікаційній роботі розмір контекстного вікна не мав суттєве значення, оскільки більшість прикладів для навчання містять невеликі текстові фрагменти.

Враховуючи сумісність Gemma 2 2B з LoRA і існуючими фреймворками та перевагу над моделями в однаковому діапазоні розміру, для виконання практичної частини кваліфікаційної роботи була обрана саме вона.

2.2. Архітектура LLM на прикладі Gemma 2 2B

На рисунку 2.2 схематично зображена архітектура Gemma 2 2B, яка поділена на головні компоненти: вхідний шар вбудовування, стек з 26 трансформерних блоків (з чергуванням механізму локальної та глобальної уваги), позиційне кодування, та фінальне декодування.

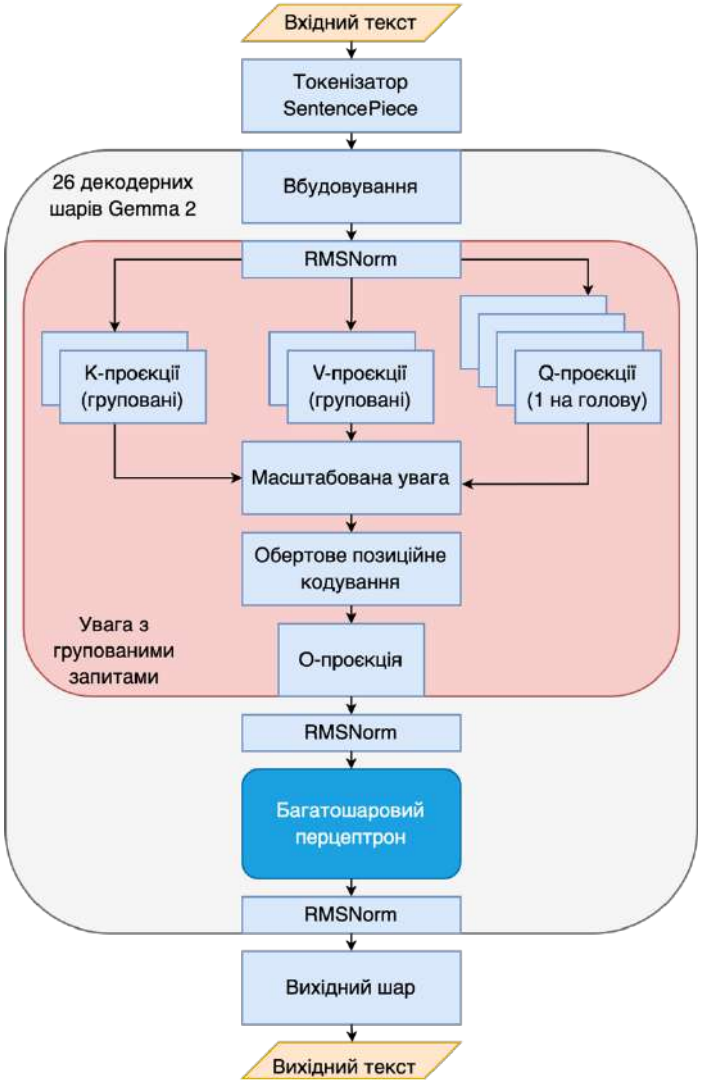

Рисунок 2.2 – Архітектура LLM Gemma 2 2B

Gemma 2 2B на вхід приймає до 8192 токенів, які на першому шарі вбудовування перетворюються у вектори. Для Gemma 2 2B використовується токенізатор SentencePiece з розділенням цифр, збереженням пробілів та байтовим кодуванням **[33]**.

Словник моделі налічує 256000 окремих токенів, тому кожен із них можна закодувати за допомогою одиничного вектора у 256000-вимірному просторі. Результатом цього процесу є вхідна матриця $I_E \in \mathbb{R}^{8192 \times 256000}$. Проте такі вектори не відображають семантичні властивості слів, тому необхідно використати матрицю вбудовування $W_E \in \mathbb{R}^{256000 \times 2304}$, щоб отримати нові вектори, які б передавали це значення – $X_{\text{вбудовані токени}} \in \mathbb{R}^{8192 \times 2304}$.

Далі модель використовує X у маскованому шарі багатоголової самоуваги. Для розуміння принципу роботи цього компонента необхідно почати з концепції самоуваги. На цьому етапі кожен токен на початку представлений вектором, у якому закодовано його загальне семантичне значення. Проте загального значення слів недостатньо, щоб далі генерувати змістовний текст. Завданням цього шару є збагачення векторів-слів інформацією про інші слова, які їх оточують. Окрім того, самоувага надає певний пріоритет кожному з токенів у тексті, вказуючи, наскільки вони впливають на інші токени.

Самоувага в Gemma 2 2B реалізується за допомогою уваги з групованими запитами.

У моделі використовується чергування локальної та глобальної уваги. В кожному другому шарі використовується локальна увага з ковзним вікном з розміром 4096 токенів, а в решті використовується глобальна увага, яка бере до уваги весь контекст з розміром 8192 токенів. Це дозволяє більш ефективно моделювати залежності різного розміру.

Самоувага розпочинається з створення трьох проекційних матриць $W_q \in \mathbb{R}_{2304\times2408}$, $W_k \in \mathbb{R}_{2304\times1024}$ та $W_v \in \mathbb{R}_{2304\times1024}$. Ці матриці потім множаться з вхідною матрицею X для отримання трьох різних матриць Q, K та V, які відповідно представляють запити (query), ключі (key) та значення (value). Усього є 8 «запитних» Q-голів, але лише 4 пари «ключ-значення» (KV-голови). Кожна KV-голова обслуговує дві Q-голови. Розмір однієї голови – 256, а тому Q-тензор має $8 \times 256 = 2048$ вимірів, а K і V – $4 \times 256 = 1024$. Інтуїтивно, можна пояснити так: Q визначає, яку інформацію потрібно знайти (що шукає токен), K визначає, що токен може запропонувати іншим, а V містить саму інформацію, яку передає токен.

Функцію уваги можна описати як відображення запиту та набору пар «ключ-значення» у вихідний результат. Вона описується за допомогою формули:

$$Y = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Формула 2.3 – Функція уваги в LLM

де d_k є розмірністю одного ключа на голову **[2]**. $d_k = 256$ для Gemma 2 2B.

Результат після де об'єднання голів передається через перетворення $O = Y \cdot W_0$, де $W_0 \in \mathbb{R}_{2048\times2304}$.

Наступним етапом є позиційне кодування. Ціллю цього процесу є кодування у вбудованих векторах не тільки семантичних властивостей слів, які вони позначають, а і їхніх позицій у вхідному тексті. У Gemma 2 2B для цього використовується обертове позиційне кодування **[61].** Цей метод дозволяє ефективно працювати з довгими послідовностями. Для збереження в вбудованих векторах позиційного кодування, вектори просто додаються.

Разом з маскованим шаром багатоголової самоуваги додатково використовуються нейронні мережі прямого поширення. Вони служать як механізм нелінійного перетворення, який покращує виражальні можливості моделі. Функція активації – наближений варіант GeGLU **[64]**. Це означає, що замість точного обчислення GeGLU в архітектури Gemma 2 2B використовується спрощена формула: $GeGLU(x) \approx GELU(x) \cdot \tanh(x)$. Тоді повний блок обчислюється наступним чином: $MLP(x) = (x \cdot W_u) \cdot GeGLU\left(x \cdot W_g\right) \cdot W_d$, де $W_g \in \mathbb{R}_{2304\times9216}$, $W_u \in \mathbb{R}_{2304\times9216}$, $W_d \in \mathbb{R}_{9216\times2304}$.

Модулі додавання і нормалізації забезпечують пропуск зв'язків **[10]**, що допомагає зменшити проблему зникання градієнту, та нормалізації шарів **[3]**, що сприяє уникненню перенавчання. У Gemma 2 2B використовується RMSNorm **[74]**.

Ще однією архітектурною особливістю є логітне soft-capping. У кожному шарі уваги та фінальному шарі значення логітів обмежується за допомогою функції: $logits \leftarrow soft\_cap \cdot \tanh\left(\frac{logits}{soft\_cap}\right)$, де soft_cap = 50 для шарів самоуваги і 30 для фінального шару.

Після всіх 26 трансформерних шарів вхідна матриця X перетворюється на матрицю $Z \in \mathbb{R}_{8192\times2304}$, де кожен токен збагачений контекстною інформацією. Нарешті, модель обчислює $Softmax(Z \cdot W)$, де $W \in \mathbb{R}^{2304\times256000}$ (256000 – розмір словника). Із фінального вектора модель визначає ймовірність появи кожного наступного токена зі словника для даного попереднього тексту.

2.3. Покращення точного налаштування за допомогою графіків зниження темпу навчання

Одним з найважливіших параметрів для налаштування під час тренування будь-якої моделі глибокого навчання є коефіцієнт швидкості навчання. Якщо значення занадто велике, ваги моделі почнуть коливатися і матимуть значні зміни, не даючи моделі підлаштовуватися під зміни функції втрат, а якщо швидкість навчання занадто мала, навчання буде занадто повільним, що призведе до втрати часу і ймовірності застрягання моделі в локальному мінімумі.

Для того, щоб покращити збіжність моделі, необхідно використовувати графіки зниження темпу навчання (learning rate decay schedules). Таке покращення можна застосовувати під час FT. Існує декілька видів графіків зниження коефіцієнту швидкості навчання: ступінчастий, експоненціальний, інверсійний часовий та поліноміальний спади **[15, 31]**.

2.4.1. Поліноміальний спад

У даній роботі було обрано використання поліноміального спаду з декількох причин. По-перше, на відміну від ступінчастого або експоненціального спаду, поліноміальний гарантує досягнення кінцевого визначеного темпу навчання.

Математична формула поліноміального спаду виглядає наступним чином:

$$lr = lr_{initial} * \left(1 - \frac{epoch}{max\ epoch}\right)^{power}$$

Формула 2.4 – Поліноміальний спад темпу навчання

де lr – learning rate (коефіцієнт швидкості навчання), $lr_{initial}$ – початковий коефіцієнт швидкості навчання, $epoch$ – епоха, $max\ epoch$ – максимальна визначена епоха, $power$ – ступінь.

Для розрахунку коефіцієнту швидкості навчання було обрано ступінь 1, який передбачає, що швидкість навчання зменшується з постійною швидкістю (лінійно, бо power = 1), допоки не досягне кінцевої точки на відповідному кроці.

2.4.2. Косинусний спад

Авторами **[43]** було запропоновано незвичну евристику, яка ґрунтується на спостереженні, що ми, можливо, не хочемо занадто різко зменшувати швидкість навчання на початку і, більше того, ми, можливо, хочемо «вдосконалити» рішення в кінці, використовуючи дуже малу швидкість навчання. Це призводить до косинусного графіка (cosine scheduler) з наступною функціональною формою для темпу навчання в діапазоні t ∈ [0, T].

$$\eta_t = \eta_T + \frac{\eta_0 - \eta_T}{2}\left(1 + \cos\left(\frac{\pi t}{T}\right)\right)$$

Формула 2.5 – Косинусний спад темпу навчання

де η_0 – початкова швидкість навчання, η_T – швидкість, до якої прагне модель в момент часу T.

Імплементація методу, що запропонована бібліотекою **[48]** визначається як:

$$\eta_t = \frac{\eta_0}{2}\left(1 + \cos\left(\pi C \frac{t}{T}\right)\right)$$

Формула 2.6 – Косинусний спад з параметром масштабування

З параметром C, який було встановлено 0.5, аргумент косинуса змінюється з 0 до π, коли t прямує до T, створюючи одну спадну напівхвилю. Таким чином, швидкість навчання монотонно падає до нуля на останньому кроці. Якби нами було обрано значення C = 1, то цей повний цикл опускає швидкість навчання до 0 на півдорозі, а потім знову піднімає його. Таке «повторне прогрівання» небажане для більшості сценаріїв FT мовних моделей.

Косинусний спад визнається багатьма вченими за свої переваги у більш плавному зменшенні швидкості навчання до нуля та мінімальній кількості параметрів для налаштування. Саме за цих причин даний метод було обрано для тренування другої мовної моделі.

Висновки до розділу 2

У другому розділі було розглянуто теоретичні основи методу LoRA як одного з найефективніших способів FT. Було обґрунтовано доцільність використання цього методу з огляду на його численні переваги і описано принцип його роботи. Проведено аналіз відкритих LLM для вибору моделі сумісної з LoRA та придатної для FT за умови обмежених обчислювальних ресурсів. Враховуючи тести продуктивності, було обрано LLM Gemma 2 2B, на прикладі якої було детально описано архітектуру LLM. Також розглянуто методи покращення процесу FT через використання графіків зниження темпу навчання, зокрема поліноміального та косинусного спадів. Усі ці компоненти другого розділу створюють теоретичну основу для практичного використання LoRA та графіків зниження коефіцієнту швидкості навчання під час FT для конкретних предметних областей.

РОЗДІЛ 3
ВИБІР ПРЕДМЕТНИХ ОБЛАСТЕЙ ДЛЯ ТОЧНОГО НАЛАШТУВАННЯ ТА ОПИС ДАНИХ

3.1. Вибір предметних областей для точного налаштування LLM

Для кваліфікаційної роботи було обрано три предметні області для FT: науково-теоретична медицина, клінічна медицина, а також фінансовий аналіз. Кожна з цих галузей характеризується складною термінологією, потребою у глибоких предметних знаннях і високим потенціалом для практичного використання. При цьому такий вибір дозволяє показати здатність точно налаштованих LLM працювати з різними типами професійного тексту.

3.1.1. Науково-теоретична медицина

Науково-теоретична медицина охоплює дисципліни, що лежать в основі медичної освіти: прикладну медицину, анатомію, біохімію, мікробіологію, фармакологію, педіатрію тощо. Ці знання є критично важливими для професійної підготовки фахівців сфери охорони здоров'я.

Метою FT для цієї предметної області є створення високоточної LLM, яка має великий обсяг знань з медичних дисциплін і вміє відповідати на типові запитання студентів в зручній формі. Таку модель можна використовувати для навчання та підготовки до іспитів. Застосування моделі є виключно освітнє, а не клінічне, тобто вона не призначена для медичного консультування.

3.1.2. Клінічна медицина

Клінічна медицина – це прикладна сфера, орієнтована на діагностику, лікування, профілактику та медичну комунікацію. В цій предметній області необхідно не лише знати медичну теорію, а застосовувати її для надання клінічних порад.

Модель, яка адаптована до клінічної практики, має бути здатною інтерпретувати симптоми, формувати первинні поради, закликати до консультації з лікарем, і проявляти емпатію в тексті. При цьому модель не має бути засобом надання медичної допомоги та не може в жодному разі заміняти лікаря. Її використання повинно обмежуватися виключно допоміжними функціями для інформованого прийняття рішень користувачами.

3.1.3. Фінансова аналітика

Третьою предметною областю стала фінансова аналітика, зокрема завдання автоматизованої тональної класифікації речень у фінансових новинах.

У фінансовій аналітиці одним з ключових завдань є розуміння того, як новини впливають на поведінку ринку. Інформаційний потік з новин часто спричиняє коливання цін акцій, тому здатність швидко аналізувати тональність інформації може стати великою перевагою для інвестиційних рішень. Мовна модель, орієнтована на фінансову аналітику, повинна вміти визначати настрій тексту (позитивний, негативний або нейтральний), що дозволяє вчасно реагувати на зміни в ринку та потенційно збільшити прибуток.

3.2. Опис використаних наборів даних

Тренування LLM – це багатоступеневий процес, який передбачає опрацювання декількох наборів даних для всебічного ознайомлення моделі з певною термінологією та форматами завдань.

FT для науково-теоретичної медицини включало в себе 1) базове тренування моделі на наборі даних Medical Flashcards; 2) розширене FT моделі з додаванням MedQA та MedMCQA; 3) тестування моделі на еталонному наборі даних MMLU та MedQA-USMLE.

FT для клінічної медицини передбачало тренування та тестування моделі на наборі даних Health Care Magic, який містить реальні клінічні запитання та відповіді лікарів.

FT для фінансового аналізу здійснювався на основі набору даних Financial Phrasebank, який дозволяє налаштувати модель для аналізу тональності фінансових текстів і класифікувати їх за трьома категоріями: позитива, негативна або нейтральна.

Для ширшого розуміння використаних наборів даних наступні секції даного підрозділу охоплюватимуть їх розгорнутий опис.

3.2.1. Набори даних для адаптації LLM до науково-теоретичної медицини

Medical Meadow – відкритий набір навчальних даних, який був представлений у роботі **[46]**. Він фокусується на різних аспектах медичних знань і практики, забезпечуючи комплексну базу для навчання моделі. Для даної роботи було використано одну частину даної колекції, а саме набір Medical Flashcards Used by Medical Students.

Медицина охоплює широкий спектр предметів, які включають в себе глибоке розуміння основних медичних наук, прикладних знань та навичок. Флеш-картки

створюються студентами медиками, та охоплюють такі дисципліни, як фармакологія, фізіологія, патологія, анатомія та багато іншого. Такі флеш-картки зазвичай містять в собі стислу мнемоніку та короткий підсумок, допомагаючи у вивченні та запам'ятовуванні важливих медичних понять. Автори даного набору даних використали флеш-картки як джерело для створення пар питання-відповідь для FT моделі. Для формування зв'язних та контекстуально релевантних пар, авторами було використано GPT-3.5-Turbo від OpenAI для реструктуризації флеш-карток. Загалом, питання та відповіді є стислими та цілеспрямованими, оскільки зазвичай флеш-картки мають обмежений простір для розміщення великої кількості інформації на них.

Описаний набір даних містить 32047 пар питання-відповідь.

Таблиця 3.1 представляє собою приклади пар питання-відповідь, які наявні у вищезазначеному наборі даних.

Таблиця 3.1 – Репрезентативні запитання з набору даних Medical Flashcards

Запитання	Відповідь
What does low REM sleep latency and experiencing hallucinations/sleep paralysis suggest?	Low REM sleep latency and experiencing hallucinations/sleep paralysis suggests narcolepsy.
What are some possible causes of low PTH and high calcium levels?	PTH-independent hypercalcemia, which can be caused by cancer, granulomatous disease, or vitamin D intoxication.

Другим набором даних для FT моделі під науково-теоретичну медичну сферу став набір даних MedQA (Medical Question Answering). У 2020 році, автори статті **[70]** представили набір даних MedQA, який створено для вирішення медичних проблем, що представляють собою складний реальний сценарій. Запитання в наборі даних були зібрані з іспитів медичних комісій США, Китаю та Тайваню, де лікарів оцінювали за їх професійними знаннями та можливістю приймати клінічні

рішення. У рамках даної роботи, модель навчалась лише на основі набору даних, який містить інформацію англійською мовою.

Питання на цих іспитах різноманітні і торкаються великої кількості областей медицини, тому, як правило, вимагають глибокого розуміння відповідних концепцій, які зазвичай подані медичними посібниками. Набір даних MedQA містить запитання, які у більшості випадків представлені довгим абзацом, включаючи опис стану пацієнта, а також відповідь. На кожне запитання пропонується кілька варіантів відповідей, з яких слід обрати лише одну, яка є найбільш підходящою. Датасет налічує 10178 прикладів.

Таблиця 3.2 представляє собою приклад запитання, які містить в собі даних набір даних.

Таблиця 3.2 – Репрезентативне запитання з набору даних MedQA

Запитання та варіанти відповідей	Правильна відповідь
Q: A 35-year-old male presents to his primary care physician with complaints of seasonal allergies. He has been using intranasal vasoconstrictors several times per day for several weeks. What is a likely sequela of the chronic use of topical nasal decongestants?A: {'A': 'Epistaxis', 'B': 'Hypertension', 'C': 'Permanent loss of smell', 'D': 'Persistent nasal crusting', 'E': 'Persistent congestion'}	E: Persistent congestion

Третім набором даних було обрано MedMCQA (Medical Multiple-Choice Question Answering). Набір даних MedMCQA вперше був представлений авторами **[49]** та являє собою великомасштабний набір з відповідями на запитання з множинним вибором, розроблений для вирішення реальних питань іспитів у медичних університетах. Набір даних складається з 194 тис. високоякісних питань з медичної тематики, що охоплюють 2,4 тис. тем з охорони здоров'я та 21 медичний предмет. Окрім запитання, правильної відповіді (відповідей) та інших варіантів

відповідей, він складається з різних допоміжних даних, основними з яких є детальне пояснення рішення. Питання взяті зі збірника для іспитів AIIMS та NEET PG, де оцінюються професійні знання студентів-випускників медичних факультетів.

На рисунку 3.1 представлено розподіл медичної тематики для набору даних. Тематика охоплює широкий спектр медичних дисциплін: від прикладних напрямів (ендокринологія, інфекції, гематологія, респіраторні захворювання тощо) і хірургії (загальна хірургія, ендокринологія, хірургія молочної залози, судинна хірургія тощо) до таких галузей, як радіологія та біохімія.

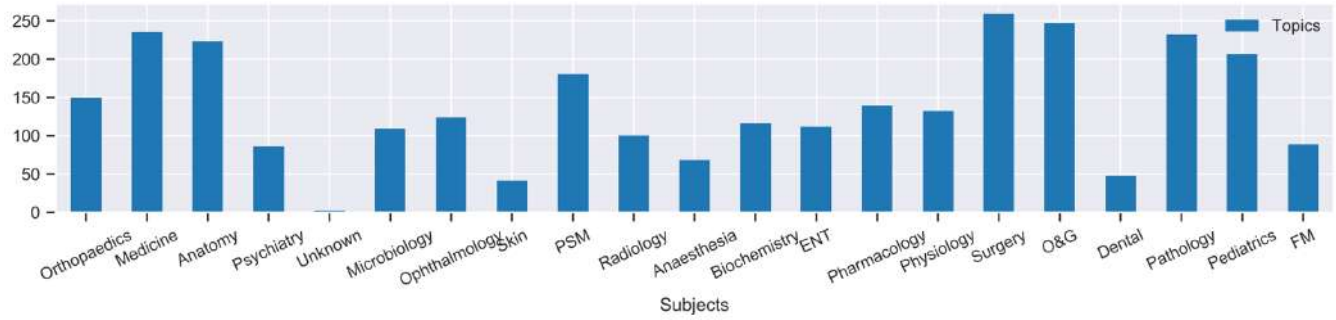

*Рисунок 3.1 – Розподіл тем за предметами в наборі даних **[49]**.*

Приклад структури даних MedMCQA наведено у таблиці 3.3 Окрім наведених колонок у таблиці, набір даних додатково містить пояснення правильної відповіді, назву предметної області та тему даної предметної області.

Таблиця 3.3 – Репрезентативні запитання з набору даних MedMCQA

Запитання	Варіант A	Варіант B	Варіант C	Варіант D	Правильна відповідь
The pharmacokinetic change occurring in geriatric patient is due to	Gastric absorption	Liver metabolism	Renal clearance	Hypersensitivity	C
True regarding lag phase is?	Time taken to adapt in the new environment	Growth occurs exponentially	The plateau in lag phase is due to cell death	It is the 2nd phase in bacterial growth curve	A

3.2.2. Набори даних для оцінки продуктивності адаптованої до науково-теоретичної медицини LLM

Для всебічного тестування точно-налаштованої моделі під науково-теоретичну медичну область, використовувалися різні набори даних. Одним з них є MMLU (Масштабне Багатозадачне Розуміння Мови), який був представлений авторами даної статті **[45]** і наразі визнається одним з еталонних наборів даних і є одним з найпопулярніших тестів продуктивності. Він складається з питань з декількома варіантами відповідей з різних галузей знань, охоплюючи гуманітарні, суспільні, точні та інші науки.

Питання для набору даних було зібрано вручну з вільно доступних джерел в Інтернеті, їх кількість складає 15908 і 57 предметних областей. Є приклади завдань, які охоплюють той самий предмет, але на певному рівні складності. Наприклад, завдання по психології «Професійна психологія» спирається на запитання з вільнодоступних практичних завдань для іспиту для професійної практики в психології, тоді як завдання «Психологія середньої школи» містить запитання, подібні до тих, що містяться в іспитах з психології для випускників навчальних закладів.

MMLU охоплює широкий спектр різноманітних тем. Наприклад, є завдання, пов'язані з гуманітарними науками. Інші питання стосуються суспільних наук. Також є завдання, що перевіряють знання з біології, фізики, хімії, інформатики та з багатьох інших дисциплін, що відносяться до категорії наук, технологій, інженерії та математики (STEM). Окремі теми включають питання-відповіді, які вимагають професійної підготовки, наприклад завдання «Професійна медицина».

Структура даних для кожного виду категорій однакова, і містить в собі: запитання, предмет запитання, варіанти відповідей, правильна відповідь.

У таблиці 3.4 наведено приклад запитання з набору даних MMLU.

Таблиця 3.4 – Приклад запитання з набору даних MMLU

Запитання	Область	Варіанти відповідей	Правильна відповідь
Which of the following conditions does not show multifactorial inheritance?	Medical Genetics	{'A': 'Pyloric stenosis', 'B': 'Schizophrenia', 'C': 'Spina bifida (neural tube defects)', 'D': 'Marfan syndrome'}	D

Можемо побачити, що набір даних включає в себе велику кількість областей, проте так як завдання даної роботи полягає в побудові моделі, яка базується на медичних даних, то і тестовий набір даних має відповідати поданій тематиці.

Виходячи з цього, з 57 областей мною було обрано 13 областей для тестування продуктивності моделі, точно налаштованої для науково-теоретичної медицини, що включають: anatomy (анатомія), clinical knowledge (клінічні знання), college biology (біологія рівня коледжу), college chemistry (хімія рівня коледжу), college medicine (медицина рівня коледжу), high school biology (біологія рівня старшої школи), high school chemistry (хімія рівня старшої школи), high school psychology (психологія рівня старшої школи), medical genetics (медична генетика), professional medicine (професійна медицина), professional psychology (професійна психологія), virology (вірусологія), nutrition (харчування).

Також для того, щоб різносторонньо оцінити продуктивність моделі, було прийнято рішення обрати додатковий тестовий розподіл даних – MedQA-USMLE.

MedQA-USMLE – набір даних, що є частиною MedQA, який був описаний вище у цьому розділі. USMLE відповідає за позначення місця проведення іспиту, з якого було взято питання та пов'язані з ним відповіді. У даному випадку, USMLE денотується як запитання з іспиту в США.

Тестовий набір містить в собі 1.27 тис. рядків і складається з: опису проблеми, за яким слідує запитання, 4 варіантів відповідей та правильного варіанту відповіді. Приклад запитання з цього набору даних наведено в таблиці 3.5.

Таблиця 3.5 – Репрезентативне запитання з набору даних MedQA-USMLE

Запитання	Варіант №1	Варіант №2	Варіант №3	Варіант №4	Відповідь
A 35-year-old male presents to his primary care physician with complaints of seasonal allergies. He has been using intranasal vasoconstrictors several times per day for several weeks. What is a likely sequela of the chronic use of topical nasal decongestants?	Epistaxis	Permanent loss of smell	Persistent nasal crusting	Persistent congestion	D

3.2.3. Набір даних для адаптації та тестування LLM у клінічній медицині

Для точного налаштування LLM під клінчну медичну предметну область був використаний Health Care Magic. Це набір даних, який складається з реальних діалогів між лікарем і пацієнтом, зібраних з HealthCareMagic.com, платформи медичних консультацій в Інтернеті. Початково набір даних складався з 200 тис. необроблених діалогів, проте після ретельного очищення, він почав містити 100 тис. пар запитання-відповідь. Набір даних дуже широко використовується для FT LLM для відповідей на медичні питання, зокрема, для роботи системи ChatDoctor **[8]**.

Приклад пари запитання-відповідь подано в таблиці 3.6.

Таблиця 3.6 – Репрезентативне запитання з набору даних Health Care Magic

Запитання	Відповідь
Hi, my name is Wendy. For the last couple of months I've been experiencing severe lower back pain, extending around the front into my groin area and radiating down my thigh. I am currently a cancer patient, and nobody seems able to tell me what it might be. The pain is so intense that it brings tears to my eyes. Could you possibly advise what this could be?	Thank you for your query. You did not specify the type of cancer you are being treated for, which is important information. Pain originating in the lower back and radiating down the thigh may be caused by metastatic spread to the spine or to nearby lymph nodes. I recommend consulting your treating oncologist and arranging a CT scan of the abdomen and pelvis, as well as a bone scan. Further treatment will depend on the results of these examinations and the definitive diagnosis.

3.2.4. Набір даних для точного налаштування та тестування LLM у сфері фінансового аналізу

Для демонстрування можливості FT для класифікації послідовностей і більшої різноманітності обраних предметних областей, було обрано набір даних, який містить речення з фінансових новин під назвою Financial Phrasebank.

Дані були вперше представлені авторами статті **[20]** у 2014 році. Вони складаються з англомовних новин про всі компанії, що котируються на біржі OMX Helsinki. Новини було завантажено з бази даних LexisNexis. Авторами було відібрано випадкову підмножину з 10 000 статей, щоб отримати хороше охоплення малих і великих компаній, компаній з різних галузей, а також різних джерел новин. Окрім цього, ними було вилучено речення, що не містять жодної лексичної одиниці, зменшивши таким чином вибірку до 53 тис. речень. У кінцевому випадку, близько 5 тис. речень було обрано випадково для представлення всього набору даних.

Авторами було виконано анотування на рівні фраз, мета цього – віднести кожне речення до позитивної, нейтральної чи негативної категорії, спираючись лише на інформацію, яка явно міститься в реченні. Так як дослідження було зосереджене лише на фінансовій та економічній сферах, анотаторами було запропоновано розглядати речення з точки зору інвестора, тобто чи подана інформація може мати позитивний, негативний чи нейтральний ефект вплив на ціну акцій. Отже, набір даних містить речення та анотацію експертів.

У таблиці 3.7 наведено приклад представлення даних.

Таблиця 3.7 – Репрезентативні речення з набору даних Financial Phrasebank

Речення	Анотація
Pharmaceuticals group Orion Corp reported a fall in its third-quarter earnings that were hit by larger expenditures on R&D and marketing.	Negative
According to the company's updated strategy for the years 2009–2012, Basware targets a long-term net sales growth in the range of 20%–40% with an operating profit margin of 10%–20% of net sales.	Positive

Висновки до розділу 3

У третьому розділі було обґрунтовано вибір предметних областей для FT та надано детальний опис використаних наборів даних. Для демонстрації можливостей адаптації моделі до різних даних, було обрано три відмінні сфери: науково-теоретичну медицину, клінічну медицину та фінансову аналітику. Для науково-теоретичної медицини було використано комбінацію освітніх наборів, таких як Medical Flashcards, MedQA і MedMCQA, а для тестування – MMLU та тестова вибірка MedQA. Клінічна медицина потребувала наборів діалогів з реальних консультацій (Health Care Magic), а для фінансової аналітики було застосовано Financial Phrasebank, який містить анотовані за тональністю фінансові речення. Результатом розділу є розуміння обраних предметних областей і даних, що є важливим для врахування потреб кінцевих користувачів, проведення FT і оцінювання продуктивності.

РОЗДІЛ 4
ПРАКТИЧНА РЕАЛІЗАЦІЯ ТОЧНОГО НАЛАШТУВАННЯ

4.1. Попередня обробка даних

Дані, що були детально описані в минулому розділі, перед тим, як використовувати для FT, необхідно попередньо обробити, оскільки для FT ключовим є форматування під шаблон діалогу та токенізація. Також попередня обробка може включати чистку даних, розділення на навчальний, валідаційний і тестовий набори.

Набір даних Medical Meadow: Medical Flashcards вже представляв собою набор пар питання-відповідь, а тому передбачав лише один етап попередньої обробки даних – розділення на навчальний та валідаційний набори. Для валідації навчання було випадково вибрано 10% набору, що становить 3354 прикладів.

Датасет MedQA був відформатований, щоб отримати пари питання-відповідь. Цей набір даних вже був розділений на навчальну та тестову вибірки авторами.

Набір даних MedMCQA також був відформатований у вигляді пар питання-відповідь. Окрім того, були відфільтровані рядки з пропущеними значеннями. З метою зменшення ризику упередженості LLM після FT, зокрема схильності частіше обирати певний варіант відповіді, було здійснено балансування набору даних за допомогою методу зменшення вибірки. Таким чином, було отримано рівномірний розподіл прикладів – по 25% для кожного з чотирьох варіантів. MedMCQA вже був поділений авторами на тренувальну та валідаційну вибірки і після попередньої обробки набір даних налічував 155852 прикладів для тренування та 3300 для валідації.

З метою тренування для області науково-теоретичної медицини вищезгадані дані були об'єднані. Після всієї попередньої обробки частки окремих джерел у фінальному датасеті розподілилися наступним чином: MedMCQA – 80,27%, Medical Meadow: Medical Flashcards – 15,55%, а MedQA – 4,18%.

Для адаптації моделі до клінічної медицини використовувався набір даних Health Care Magic, який потребував ретельної перевірки, очищення та покращення. Основними проблемами цього датасету були численні граматичні помилки у відповідях лікарів, дублікати, велика кількість урізаних та неповних відповідей (ймовірно, внаслідок некоректного вебскрапінгу з ресурсу HealthCareMagic.com авторами датасету), а також присутність особистої інформації, а саме імен лікарів. Такі дефекти суттєво знижують якість FT. Їхня присутність створює ризик того, що модель запам'ятає граматично неправильні конструкції, відтворюватиме особисті імена лікарів або генеруватиме неповні відповіді, що є неприйнятним у клінічному застосуванні.

Для вирішення цих проблем було прийнято рішення використання моделі GPT-4o-mini для 1) виправлення будь-яких граматичних помилок з збереженням оригінального фразування та лексики відповідей 2) маркування неповних відповідей для їх подальшого видалення з набору 3) анонімізації даних. Зокрема, використання LLM для анонімізації медичних даних є відомою практикою. Згідно з **[1]**, LLM можуть слугувати автоматизованим інструментом для видалення особистої ідентифікаційної інформації з медичних даних і досягати успішності на рівні 98,05%. Також з датасету були прибрані дублікати.

Після попередньої обробки розмір набору даних зменшився з 112165 прикладів до 103428.

Набір даних Health Care Magic був розділений на навчальний, тестовий та валідаційний набори. Навчальна вибірка мала розмір 92569 (89.5% від усього обсягу оригінального датасету), валідаційна – 10342 (10%), а тестова – 517 (0.5%).

Усі вищезгадані датасети були відформатовані для узгодження зі структурою діалогового шаблону моделі Gemma 2 2B. Форматування виглядає таким чином:

<start_of_turn>user
Запитання<end_of_turn>
<start_of_turn>model
Відповідь<end_of_turn>

Для FT LLM з метою класифікації речень з фінансових новин було використано датасет Financial Phrasebank. Набір даних доступний в чотирьох різних конфігураціях в залежності від відсотку згоди анотаторів щодо тональності речень: >50%, >66%, >75% та 100% згоди. Це означає, що якщо відсоток згоди для певного речення у відповідній конфігурації є меншим за значений, то воно було виключене з набору. Для FT було обрано конфігурації >75% як компроміс між якістю анотацій та розміром.

Попередня обробка включала лише розділення набору даних на навчальну, валідаційну та тестову вибірки. До навчальної вибірки було віднесено 95% оригінального набору, в той час як до тестової та валідаційної – по 5% кожна.

Останнім етапом попередньої обробки була токенізація, яка була застосована до всіх наборів даних перед проведенням FT.

4.2. Вибір гіперпараметрів та проведення низькорангового адаптування

Під час проведення FT важливим є вибір гіперпараметрів. Гіперпараметри – це величини, які задаються до початку тренування і не оновлюються під час нього. До проведення FT розробник може змінювати такі гіперпараметри, як кількість епох, розмір міні-пакетів, кількість кроків накопичення градієнтів (визначає, скільки кроків потрібно виконати перед оновленням ваг моделі, накопичуючи градієнти з кількох міні-пакетів), коефіцієнт швидкості навчання та багато інших. У практичній частині цієї роботи було використано метод LoRA, який вимагає додаткового вибору двох інших гіперпараметрів: значення рангу та коефіцієнт масштабування α.

Вибір гіперпараметрів передбачає балансування між якістю результатів FT та доступністю обчислювальних ресурсів. Усі рішення, пов'язані з процесом FT моделі Gemma 2 2B, були прийняті, враховуючи обмеження апаратного забезпечення, доступного для виконання практичної частини, зокрема використання однієї відеокарти NVIDIA A100 з 40 ГБ відеопам'яті та 89.6 ГБ оперативної пам'яті.

Для завантаження моделі Gemma 2 2B та її токенізатора використовувалася бібліотека Transformers, зокрема класи AutoModelForCausalLM та AutoTokenizer відповідно. Була використана 16-бітна точність з підтримкою BF16 для оптимізації використання пам'яті. FT моделі здійснювалося з використанням фреймворку TRL і його класу SFTTrainer у форматі питання-відповідь з дотримання інструкційного стилю, щоб не порушити вже наявну діалогову структуру моделі.

Для завантаження моделі готової для адаптації до класифікації послідовностей використовувався клас AutoModelForSequenceClassification, а саме FT здійснювалося за допомогою класу Trainer з бібліотеки Transformers.

Все FT виконувалося з застосування техніки LoRA, а для конфігурації адаптерів було використано клас LoraConfig бібліотеки PEFT.

4.2.1. Точне налаштування для науково-теоретичної медичної області

Низькорангова адаптація моделі до задач науково-теоретичної медицини відбувалася у два етапи. Перший етап включав базове тренування моделі на попередньо обробленому наборі даних Medical Flashcards. Це дозволило моделі ознайомитися з основною термінологією, притаманною цій галузі, що забезпечило кращі стартові параметри для другого етапу. На другому етапі модель донавчалася на датасетах MedQA та MedMCQA. Це дало змогу моделі глибше засвоїти профільні знання і сформувати шаблон поведінки для медичних запитань: якщо у вхідному тексті надано варіанти відповідей, модель генерує лише літеру, що відповідає правильному варіанту, а якщо ж варіанти відсутні – модель надає повну відповідь на відкриті запитання.

При цьому під час другого етапу до тренувальних даних знову був включений набір Medical Flashcards, щоб модель не втратила отримані під час першого етапу знання. Функція втрат відстежувалась на валідаційній підмножині MedMCQA та Medical Flashcards.

Для обох етапів адаптації було застосовано значення рангу та коефіцієнта масштабування, що дорівнюють 512. Низькорангове адаптування здійснювалося для всіх проєкційних шарів Gemma 2 2B: шар проєкції запитів, ключів, значень,

виходу (результату уваги), шари прямого поширення, включно з розширенням та зменшенням розмірності. Обрані гіперпараметри для методу LoRA забезпечили якість співставну з FFT.

Для першого етапу було проведено FT моделі з поліноміальним спадом, починаючи з темпом навчання зі значенням 1e-4 та поступово досягаючи 1e-5. Саме тренування виконувалося з одною епохою, розміром міні-пакета 4 та акумуляцією градієнтів за 2 кроки, що дорівнює ефективному розміру пакета 8. Окрім того, застосовувалося 100 кроків розігріву, щоб уникнути різких коливань під час оновлення ваг на початку навчання. Навчання на цьому етапі тривало 39 хвилин.

Для другого етапу темп навчання був сталим і мав значення 1e-5. Воно проводилося протягом двох епох для якісного засвоєння складних залежностей. Міні-пакет мав розмір 2, а акумуляція градієнтів виконувалась протягом 4 кроків, що також дорівнює ефективному розміру пакета 8. Така конфігурація збільшує тривалість навчання, проте була необхідною через обмежені обчислювальні ресурси. Час другого етапу FT становив 15 годин 9 хвилини.

Модель Gemma 2 2B містить 2614341888 параметрів, в той час як застосування методу LoRA з значенням рангу 512 для вищезгаданих модулів дозволило зменшити кількість тренованих параметрів до 664535040, що становить лише 25,42% від загальної кількості.

Точно налаштована модель отримала назву MedGemmaScience. Вона була збережена і доступна для завантаження за посиланням на платформі HuggingFace.

В таблиці 4.1 наведено приклади запитань та згенерованих моделлю відповідей.

Таблиця 4.1 – Приклади запитань і відповідей, згенерованих MedGemmaScience

Запитання	Відповідь
Question: Does the enzyme pyruvate carboxylase function in the glycolytic or gluconeogenic pathway, and what cofactor does it require?	Pyruvate carboxylase is an enzyme that functions in the gluconeogenic pathway, and it requires biotin as a cofactor.
Question: What is the first-line treatment for anaphylaxis? Options: A) Intravenous corticosteroids B) Oral antihistamines C) Intramuscular epinephrine D) Inhaled bronchodilators	C

4.2.2. Точне налаштування для сфери клінічної медицини

Адаптація Gemma 2 2B до області клінічної медицини вимагала іншої стратегії через неспівмірність обсягу набору даних Health Care Magic з масштабом самої області клінічної медицини. Незважаючи на відносно велику кількість прикладів в наборі, вони не можуть охопити всю можливу кількість клінічних запитань. Тому у цьому випадку не ставилося за мету передати моделі предметні знання з датасету. Натомість, метод LoRA був використаний для активації та організації вже наявних в LLM знань, подолання обмежень на відповіді медичного характеру та перенесення стилю мовлення, притаманного для лікарів. Це можна досягти використовуючи невелике значення рангу.

Таким чином для FT за допомогою методу LoRA було використане значення рангу 8 та коефіцієнта масштабування 4. Це означає що адаптери додаються до базової моделі з коефіцієнтом 0.5, що дозволяє більш помірну зміну оригінальних ваг моделі. Тренування тривало одну епоху. Також важливим є вибір проєкційних шарів. Для FT були змінені лише шари багатошарового перцептрону. Ці шари були обрані, тому що вони відповідають за нелінійне перетворення представлень, сформованих механізмом уваги, і беруть участь в побудові кінцевої структури відповіді. Це дозволяє краще організувати медичний зміст у відповідях і активувати наявні знання в LLM.

Для проведення FT використовувався графік зниження темпу навчання косинусний спад, з початковим значення швидкості 1e-5. Ефективний розмір пакета дорівнював 64, що дозволило знизити вплив окремих прикладів і навчатися загальному стилю. Тривалість FT склала 2 години 37 хвилин.

Використання матриць з рангом 8 для лише вищезгаданих шарів зменшило кількість тренованих параметрів до 7188480, що становить лише приблизно 0,275% від загальної кількості.

Адаптована до клінічної предметної області модель, яка отримала назву MedGemmaClinic, доступна для використання за посиланням.

4.2.3. Точне налаштування для фінансового аналізу

Хоча LLM з лише декодером, такі як Gemma 2 2B, зазвичай призначені для генерації тексту, їх можна використовувати для задач класифікації послідовностей. Для цього необхідна лише мінімальна зміна в архітектурі цих моделей – додати окремий лінійний класифікаційний шар W_{cls}, який перетворює вихідне представлення на вектор з розмірністю, що відповідає кількості класів.

Оскільки моделі є декодерними, щоб отримати представлення всієї послідовності, в бібліотеці Transformers для класифікації використовується прихований вектор останнього токену **[17]**. Через шари самоуваги цей вектор слугує представленням про інформацію всієї послідовності.

Для перетворення в ймовірності застосовується функція Softmax на виведених логітах. Після чого для отримання прогнозованого класу обирається той, в якому модель найбільш впевнена.

Для FT з метою класифікації у сфері фінансового аналізу було обрано конфігурацію, яка поєднує високе значення рангу (512) для всіх проєкційних шарів з повним тренуванням лише одного додаткового шару W_{cls}. Така конфігурація співставна з повним тренуванням, але було треновано лише 664541952 параметрів, що становить 25,42% від усіх параметрів моделі Gemma 2 2B. Навчання здійснювалося протягом двох епох з стабільним темпом навчання 1e-5 і з

використанням ранньої зупинки, якщо F-міра не покращується протягом однієї епохи.

Після FT модель отримала назву FinGemma та завантажена для вільного доступу на платформі HuggingFace (посилання).

В таблиці 4.2 надано приклади класифікації речень, взятих з фінансових новин, моделлю FinGemma.

Таблиця 4.2 – Приклади класифікації речень за допомогою FinGemma

Речення	Прогнозований клас
Xpeng's shares in Hong Kong surged over 10% Thursday following upbeat earnings and stronger-than-expected revenue forecast for the second quarter.	Позитивний
We can't rule out that the U.S. economy will fall into stagflation as the country faces huge risks from geopolitics, deficits and price pressures	Негативний

4.3. Результати тестування точно налаштованих моделей

4.3.1. Тестування MedGemmaScience

Для тестування моделі GemmaMedScience використовувалась логарифмічна ймовірність. Логарифмічна ймовірність – метрика, яку часто використовують при оцінці мовних моделей **[30]**. Логарифмічна ймовірність виражається як $\log P(E)$, де $P(E)$ – ймовірність появи токена у послідовності, обчислена за допомогою функції Softmax, застосованої до логітів моделі. Ймовірність токенів на логарифмічній шкалі варіюється в діапазоні $[-\infty;\ 0]$, де значення, які є ближче до нуля означають більшу впевненість моделі у виборі відповідного токену **[26]**.

Знаючи всі можливі варіанти відповіді, згенеровані LLM (літери A, B, C, D, які відповідають правильному варіанту), необхідно розрахувати логарифмічну ймовірність для кожної з цих чотирьох відповідей і обрати ту, в якій модель впевнена найбільше (значення найближче до нуля).

Такий метод був обраний в рамках цієї роботи через те, що він є швидшим за звичайну генерацію відповіді моделлю, оскільки він передбачає лише прямий прохід через модель для отримання логітів та обчислення самої логарифмічної ймовірності. Натомість, при використанні авторегресивної генерації, навіть враховуючи той факт, що GemmaMedScience при заданих варіантах відповідей генерує лише одну літеру, ми витрачаємо час на декодування.

В таблиці 4.3 продемонстровані результати тесту продуктивності MMLU для двох моделей Gemma 2 2B: базової і точно налаштованої під галузь науково-теоретичної медицини моделі MedGemmaScience.

Таблиця 4.3 – Результати тесту продуктивності MMLU для MedGemmaScience

Область	Точність базової моделі, %	Точність після FT, %	Покращення Δ, %
Анатомія	52.59	57.78	+5.19
Біологія рівня коледжу	56.25	63.19	+6.94
Біологія рівня старшої школи	64.19	68.39	+4.20
Вірусологія	42.77	43.37	+0.60
Клінічні знання	57.36	56.98	-0.38
Медична генетика	61.00	65.00	+4.00
Медицина рівня коледжу	53.18	55.49	+2.31
Професійна медицина	42.28	51.47	+9.19
Професійна психологія	48.37	50.16	+1.79
Психологія рівня старшої школи	73.94	75.96	+2.02
Нутриціологія	55.23	61.44	+6.21
Хімія рівня коледжу	37.00	40.00	+3.00
Хімія рівня старшої школи	35.96	39.90	+3.94

Найвище покращення простежується в області професійної медицини (+9,19%). Це пояснюється тим, що сам набір MedMCQA, який входив в тренувальні дані, складається з питань взятих з вступних іспитів (тобто професійний рівень

медичних запитань), тож покриття тем MedMCQA загалом збігаються з цією областю MMLU. Також високе покращення простежується в області біології рівня коледжу (+6,94%), концепти якої широко охоплені предметами мікробіологія та біохімія в MedMCQA. Область нутриціології теж позначилась великим приростом (+6,21%), оскільки знання про метаболічні цикли та вітаміни присутні в предметах фізіологія та педіатрія в MedMCQA. Високе покращення в області анатомії було цілком очікуваним, тому що в MedMCQA ці знання представлені окремим предметом анатомія.

Середній приріст спостерігається в областях хімії, медичної генетики та психології. Помірне покращення присутнє через те, що Gemma 2 2B вже має загальнонаукову інформацію, а FT на суміжних дисциплінах підсилює наявні концепти.

Присутнє мінімальне погіршення в області клінічних знань (-0,38%), але оскільки такий тип запитань не був представлений в тестовому наборі даних, ця мінімальна деградація є цілком очікуваною. Це може свідчити про обмеження через невеликий розмір моделі, в якій при додаванні нової інформації, відбувається незначна втрата вже наявних знань.

В середньому було досягнуто +3,77% покращення, що є успішним результатом для еталонного та популярного тесту продуктивності MMLU.

Також тестування проводилось на тестовому розподілі набору MedQA. Базова модель Gemma 2 2B продемонструвала точність на рівні 32,91%, тоді як точно налаштована MedGemmaScience коректно дала відповідь на 42,66% запитань, що на 9,75% перевищує результати базової моделі.

4.3.2. Тестування MedGemmaClinic

Тестування моделі MedGemmaClinic, адаптованої для клінічної предметної області, проводилося на основі тестової вибірки набору даних Health Care Magic. Суть тестування полягала в тому, щоб порівняти текст-еталон, написаний реальним лікарем, з текстом, згенерованим моделлю. Для цього було використано

BERTScore – метод, який дозволяє виміряти семантичну подібність між текстом-кандидатом та еталонним текстом **[6]**.

BERTScore ґрунтується на використанні контекстних вбудованих векторів отриманих із попередньо натренованої моделі RoBERTA. Подібність між двома реченнями в межах цього методу розраховується як сума косинусних подібностей між їхнім токенами у векторному просторі. Такий підхід дає змогу ідентифікувати використання синонімів.

За допомогою BERTScore можна обчислити влучність (precision), повноту (recall) та F1-міру. Для кожного токену з еталанного тексту $x = \{x_1, x_2, \dots, x_m\}$ та кожного токену з тексту-кандидату $\hat{x} = \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_m\}$ з їхніми відповідними векторами вбудовування x_i та $\widehat{\mathrm{x}_j}$:

$$Recall_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathrm{x}_i \widehat{\mathrm{x}_j}$$

Формула 4.1– BERTScore Recall

$$Precision_{BERT} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathrm{x}_i \widehat{\mathrm{x}_j}$$

Формула 4.2– BERTScore Precision

$$F1_{BERT} = 2 \cdot \frac{Precision_{BERT} \cdot Recall_{BERT}}{Precision_{BERT} + Recall_{BERT}}$$

Формула 4.3– BERTScore F1-score

В таблиці 4.4 показані результати тестування за допомогою описаних вище метрик для двох моделей Gemma 2 2B: базової і адаптованої до клінічної медицини моделі MedGemmaClinic.

Таблиця 4.4 – Приклади класифікації речень за допомогою FinGemma

Модель	Влучність	Повнота	F1-міра
Базова Gemma 2 2B	0.7670	0.8302	0.7972
Gemma 2 2B після FT	0.8614	0.8459	0.8535

Результати свідчать, що влучність зросла на 9,44% і модель після FT стала краще генерувати релевантний текст, який є більш близьким до відповідей реальних лікарів. Повнота теж покращилась на 1,57%, що показує те, що модель охоплює більше інформації з еталонного тексту.

Крім кількісних показників, аналіз показав, що відповіді MedGemmaClinic є точнішими з медичної точки зору: модель не уникає відповіді на клінічні запитання, визначає причину симптомів, пропонує наступні кроки, наприклад, які обстеження необхідно зробити або до якого спеціаліста слід звернутися.

Таблиця 4.5 містить порівняння згенерованих відповідей MedGemmaClinic та базової Gemma 2 2B на одне запитання.

Таблиця 4.5 – Порівняння відповідей MedGemmaClinic та Gemma 2 2B

Запитання	I am a 40-year-old woman. I've been feeling cold all the time, my skin is dry and flaky, and I've gained weight even though I eat very little. My periods are also getting really light. What could this be and how can I treat it?
Відповідь Gemma 2 2B	It sounds like you are experiencing a cluster of potential issues that warrant a visit to a medical professional. Your symptoms, like feeling cold, dry skin, weight fluctuations, and lighter periods, could be connected to factors like: * Hormone changes as you age. * Underlying medical conditions. * Nutrient deficiencies requiring medical advice. To address potential issues, schedule an appointment with your doctor. They can perform tests, rule out conditions, and provide guidance on the best course of action tailored to your specific circumstances.
Відповідь MedGemmaClinic	Hello, thank you for your question. You are experiencing some common symptoms of hypothyroidism. You can get a TSH level test done, and if it is high, then you should start a treatment of levothyroxine. The treatment should be continued for life, as you are already experiencing the symptoms of hypothyroidism. The symptoms of hypothyroidism are fatigue, weight gain, cold intolerance, dry skin, hair loss, and depression. Hope this helps. Thank you.

Як бачимо, відповідь MedGemmaClinic є значно точнішою і містить конкретну діагностичну інформацію. Модель називає причину симптомів, а саме гіпотиреоз, що дійсно узгоджується з зазначеною інформацією в запитанні. Відповідь включає правильну пораду зробити аналіз на тиреотропний гормон для підтвердження діагнозу, а також згадує коректний препарат левотироксин для лікування. В той же час, відповідь базової моделі є занадто загальною. Хоча вона визнає серйозність симптомів, наданий перелік можливих причин занадто широкий, неінформативний та може призвести до затримки в постановці діагнозу. Вказівка звернутися до лікаря є корисною, але не вказується до якого фахівця треба звернутись.

4.3.2. Тестування FinGemma

Для тестування FinGemma використовувалась тестова вибірка набору даних Financial Phrasebank. Для тестування було виміряно традиційні для задач класифікації метрики, а саме точність (accuracy), влучність (precision), повнота (recall) та F1-міра, які обраховуються за формулою:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Формула 4.4 – Точність

$$Precision = \frac{TP}{TP + FP}$$

Формула 4.5 – Влучність

$$Recall = \frac{TP}{TP + FN}$$

Формула 4.6 – Повнота

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Формула 4.7 – F1-міра

У результаті точно налаштована модель продемонструвала високу якість і досягла точності анотування 95,38% речень у тестовій вибірці. Ключові метрики класифікації підтверджують ефективність FT: влучність – 0,9339, повнота – 0,9647 та F1-міра – 0,9485.

4.4. Реалізація демонстраційних вебзастосунків

Для наочного показу можливостей точно налаштованих моделей в рамках цієї роботи, було створено два демонстраційні вебзастосунки відповідно для моделей MedGemmaClinic та FinGemma. Розроблені вебзастосунки були розгорнуті на платформі Hugging Face Spaces – хмарному середовищі, що надає зручний спосіб публікації демонстраційних застосунків машинного навчання. Саме розгортання було зроблено за допомогою Docker, що дозволило створити ізольоване середовище виконання з усіма необхідними залежностями. Демонстраційні застосунки доступні за такими посиланнями: MedGemmaClinic Space та FinGemma Space. Обидва застосунки покликані продемонструвати можливості моделей. В той же час, моделі мають ширший потенціал використання, який можна реалізувати під конкретні потреби користувачів або інтегрувати в більш масштабні рішення.

Для обох застосунків був використаний backend-фреймворк FastAPI для побудови REST API, а також бібліотеки Transformers, PyTorch, Uvicorn. Для створення frontend були використані HTML, TailwindCSS, FontAwesome та JavaScript.

MedGemmaClinic Space – застосунок, який призначений надавати клінічні поради, консультацію на основі симптомів або відповідати на медичні запитання. За допомогою класу TextIteratorStreamer бібліотеки Transformers відповідь відображається токен за токеном.

На рисунку 4.1 зображений знімок головного екрану MedGemmaClinic Space.

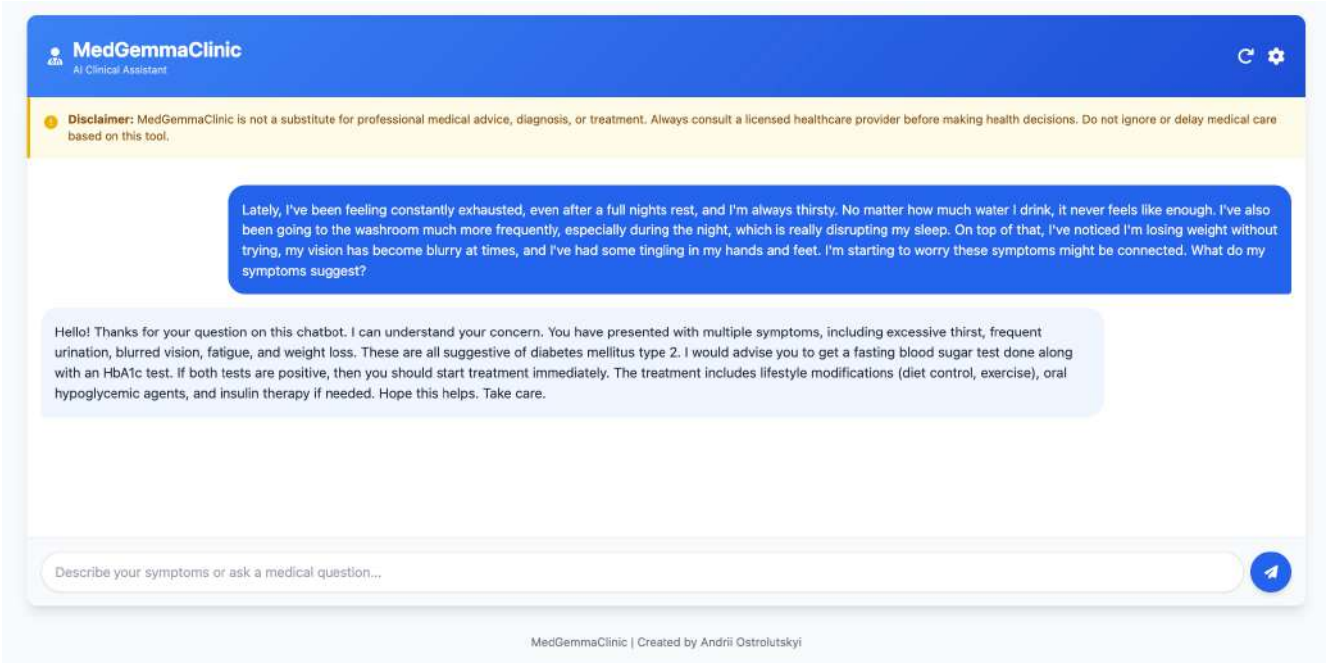

Рисунок 4.1 – Головна сторінка MedGemmaClinic Space

Також застосунок для демонстрації дозволяє користувачам експериментувати з параметрами генерації: вибиранням, температурою, top-p, штрафом за повторення. Сторінка з налаштуваннями зображена на рисунку 4.2.

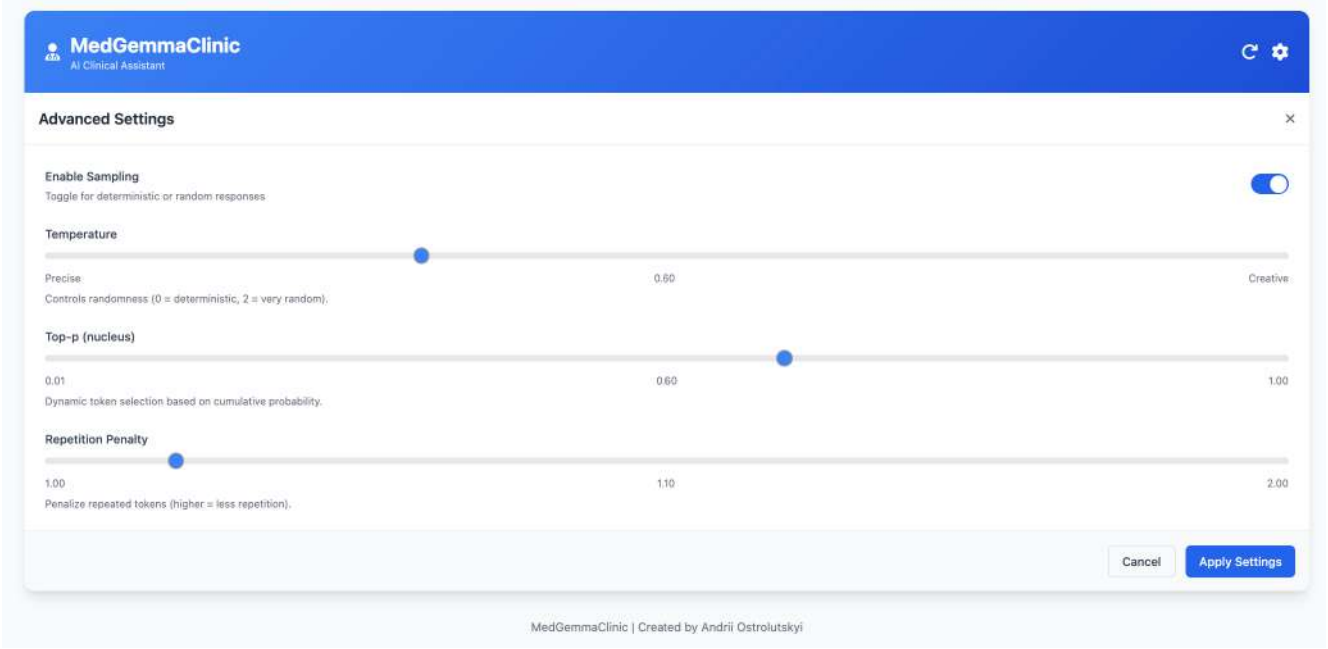

Рисунок 4.2 – Сторінка з параметрами генерації MedGemmaClinic Space

FinGemma Space – застосунок, який слугує аналітиком фінансових новин, який дозволяє користувачу надати посилання на статтю, автоматично завантажує її вміст, розбиває текст на речення та визначає для кожного з них тональність (позитивна, негативна, нейтральна) та рівень впевненості. Вебзастосунок обчислює середню тональність статті на основі її всіх речень і показує кількість речень кожного типу, візуальний індикатор настрою, класифікацію («Bullish», «Bearish» і т.д.). Для користувача також доступна фільтрація за типом настрою. Збір новин зроблений за допомогою бібліотеки newspaper3k, а бібліотеки spaCy була використана для розбиття тексту на речення. На рисунку 4.3 зображений результат аналізу фінансової статті в FinGemma Space.

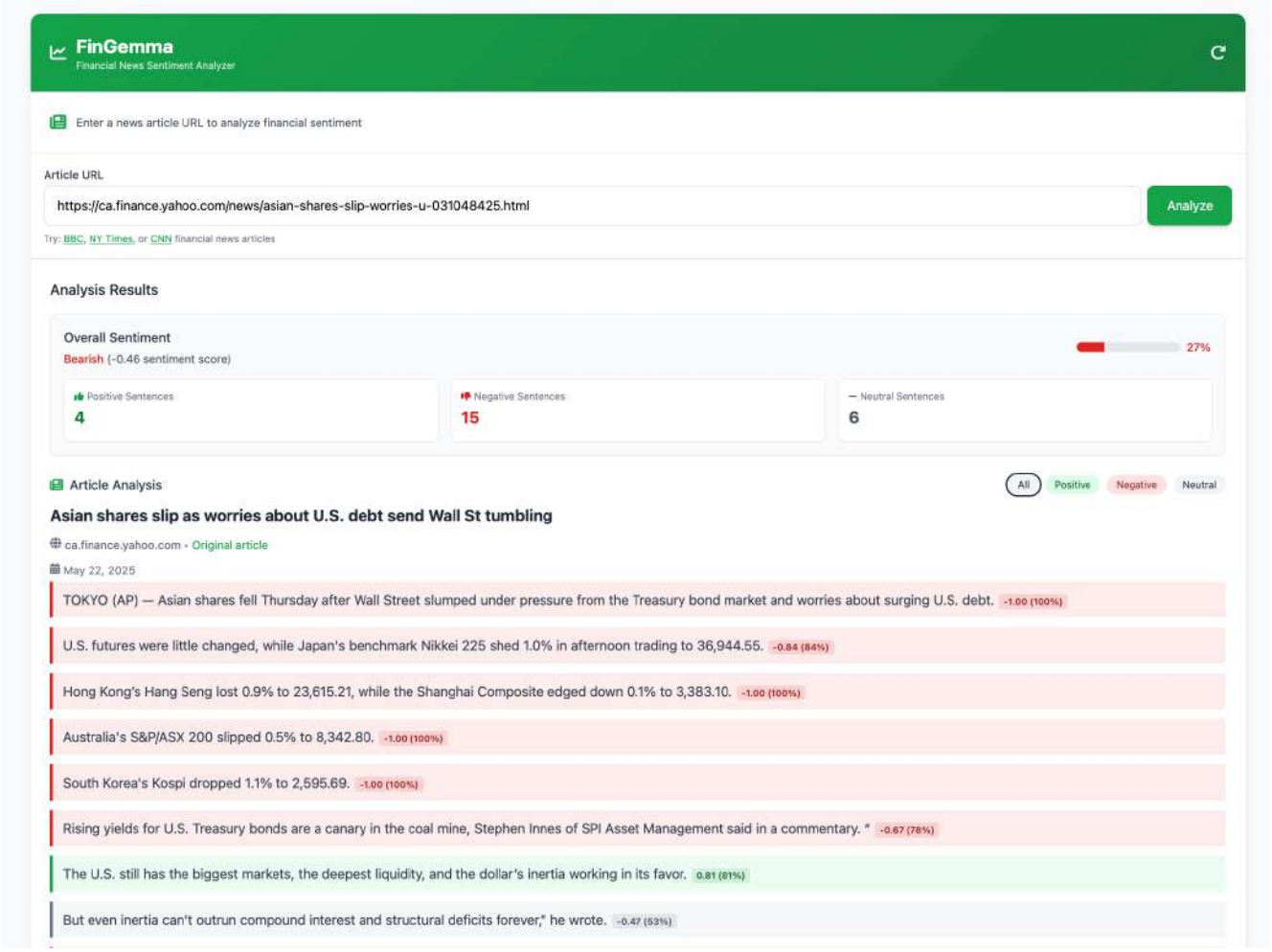

Рисунок 4.3 – Результат аналізу статті на FinGemma Space

Висновки до розділу 4

Четвертий розділ присвячений практичній реалізації FT для обраних предметних областей. Кожна з предметних областей вимагала різної стратегії для попередньої обробки даних, вибору гіперпараметрів для FT та методу LoRA. В цьому розділі описаний новий підхід до низькорангової адаптації, здійснений для області науково-теоретичної медицини, який забезпечив значне покращення результатів на MMLU (+3,77%) та MedQA (+9,75%). Для клінічної медицини описане проведення FT, що забезпечило зростання BERTScore F1-міри з 0,7972 до 0,8535. Нарешті, був розглянутий процес FT для фінансового аналізу, який дав високу точність класифікації на рівні 95,38%. Завершальним етапом стала розробка демонстраційних вебзастосунків, які дають уявлення, як точно налаштовані моделі можна використовувати. Таким чином, результати четвертого розділу підтверджують ефективність використання методу LoRA для FT LLM.

ЗАГАЛЬНІ ВИСНОВКИ

Кваліфікаційна робота мала на меті експериментальну реалізацію FT LLM для обраних предметних областей: науково-теоретичної медицини, клінічної медицини та фінансового аналізу.

У ході виконання дослідницької роботи було виконано аналіз сучасних методів до адаптації LLM і обрано найефективніший з них – низькорангове адаптування. Окрім цього було детально досліджено архітектуру LLM Gemma 2 2B, обраної для FT, та описано необхідні набори даних, які відповідають вищезгаданим предметним областям. Для проведення FT ці дані були ретельно оброблені.

Для науково-теоретичної медицини був запропонований двоетапний спосіб FT, який включає початкове тренування на медичному наборі даних для загального ознайомлення моделі з цією галуззю та подальше тренування з включенням основних наборів з метою поглиблення знань та навчанні правильному формату відповіді. Використання поліноміального спаду дало можливість контролювати темп навчання, а високий ранг для методу LoRA забезпечив якість донавчання, порівянну з FFT.

FT для клінічної медицини вимагало іншої стратегії, яка полягала в використанні низькогорангових адаптерних матриць виключно для шарів багатошарового перцептрону, що зумовило зміну в стилі відповідей та активацію наявних клінічних знань в моделі. Використання косинусного спаду для навчання забезпечило більш плавну зміну темпу навчання та кращу збіжність.

Останнє FT стосувалося сфери фінансового аналізу та потребувало зміни в архітектури LLM – додавання нового шару класифікації, який тренувався повністю. А застосування LoRA-адаптерів з високим рангом для решти проєкційних шарів підвищило якість FT.

Отримані після FT моделі були проаналізовані та протестовані. Зокрема, модель, адаптована до науково-теоретичної медицини, показала приріст точності

на 9,19% в категорії професійної медицини тесту продуктивності MMLU в порівнянні з базовою моделлю. Окрім цього, на тестові запитання з набору даних MedQA ця модель змогла дати правильні відповіді на 9,75% більше запитів, ніж базова Gemma 2 2B. Такий приріст показує глибше розуміння предметної області точно налаштованої моделі. FT моделі для клінічної медицини забезпечило зростання метрики BERTScore Precision (влучність) на 9,44%, що свідчить про значне покращення здатності генерувати релеватний, більш близький до відповіді реальних лікарів текст. А FT Gemma 2 2B для класифікації речень, взятих з фінансових статей, дало змогу отримати модель, яка показує 95,38% збігу з анотаціями експертів в фінансовій області.

Таким чином, виконане дослідження досягло поставленої мети розробки та експериментальної реалізації точного налаштування LLM для обраних предметних областей. Теоретичний аналіз обґрунтував вибір методу LoRA для FT в умовах обмежених обчислювальних ресурсів, а практичні результати підтвердили ефективність методу низькорангового адаптування. Використання двоетапного FT продемонструвало високий потенціал поєднання різних етапів налаштування для більш глибокого засвоєння фактичних знань. Нарешті, створені демонстраційні застосунки дають уявлення про те, як точно налаштовані моделі можуть бути використаними у клінічній та фінансових сферах, надаючи інформаційну підтримку користувачам.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Anonymizing medical documents with local, privacy preserving large language models: The LLM-Anonymizer / I. C. Wiest et al. *medRxiv*. 2024. P. 2024—06. URL: https://www.medrxiv.org/content/10.1101/2024.06.11.24308355v1.
2. Attention is all you need / A. Vaswani et al. *Advances in neural information processing systems*. 2017. URL: https://arxiv.org/pdf/1706.03762.
3. Ba J. L., Kiros J. R., Hinton G. E. Layer normalization. *ArXiv preprint arxiv:1607.06450*. 2016. URL: https://arxiv.org/abs/1607.06450.
4. Banks J. Gemma: introducing new state-of-the-art open models. *Google*. URL: https://blog.google/technology/developers/gemma-open-models/.
5. Bengio Y., Senecal J. S. Adaptive importance sampling to accelerate training of a neural probabilistic language model. *IEEE transactions on neural networks*. 2008. Vol. 19, no. 4. P. 713–722. URL: https://doi.org/10.1109/tnn.2007.912312.
6. BERTScore: Evaluating text generation with bert / T. Zhang et al. arXiv preprint arXiv:1904.09675. 2019. URL: https://arxiv.org/abs/1904.09675.
7. Carbon emissions and large neural network training / D. Patterson et al. *ArXiv preprint arxiv:2104.10350*. 2021. URL: https://arxiv.org/abs/2104.10350.
8. ChatDoctor: A Medical Chat Model Fine-Tuned on a Large Language Model Meta-AI (LLaMA) Using Medical Domain Knowledge / Y. Li et al. *Cureus*. 2023. URL: https://doi.org/10.7759/cureus.40895.
9. Deep contextualized word representations / M. Peters et al. *Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics: human language technologies, volume 1 (long papers)*, New Orleans, Louisiana. Stroudsburg, PA, USA, 2018. URL: https://doi.org/10.18653/v1/n18-1202.
10. Deep residual learning for image recognition / K. He et al. *2016 IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June 2016. 2016. URL: https://doi.org/10.1109/cvpr.2016.90 (date of access: 07.05.2025).
11. Distributed representations of words and phrases and their compositionality / T. Mikolov et al. *Advances in neural information processing systems*. 2013. No. 26.
URL: https://proceedings.neurips.cc/paper_files/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.

12. Efficient estimation of word representations in vector space / T. Mikolov et al. *ArXiv preprint arxiv:1301.3781*. 2013. URL: https://arxiv.org/abs/1301.3781.
13. Exploring selective layer freezing strategies in transformer fine-tuning: NLI classifiers with sub-3b parameter models / T. Hwang et al. URL: https://openreview.net/forum?id=kvBuxFxSLR.
14. Galal O., Abdel-Gawad A. H., Farouk M. Rethinking of BERT sentence embedding for text classification. *Neural computing and applications*. 2024. URL: https://doi.org/10.1007/s00521-024-10212-3 (date of access: 07.05.2025).
15. GeeksforGeeks. Learning Rate Decay - GeeksforGeeks. *GeeksforGeeks*. URL: https://www.geeksforgeeks.org/learning-rate-decay/.
16. Gemma Team. Gemma 2: improving open language models at a practical size. *ArXiv preprint arxiv:2408.00118*. 2024. URL: https://arxiv.org/abs/2408.00118.
17. Gemma2. *Hugging Face – The AI community building the future.* URL: https://huggingface.co/docs/transformers/en/model_doc/gemma2.
18. Generating wikipedia by summarizing long sequences / P. J. Liu et al. *ArXiv preprint arxiv:1801.10198*. 2018. URL: https://arxiv.org/abs/1801.10198.
19. GitHub - google-research/bert: TensorFlow code and pre-trained models for BERT. *GitHub*. URL: https://github.com/google-research/bert.
20. Good debt or bad debt: Detecting semantic orientations in economic texts / P. Malo et al. *Journal of the Association for Information Science and Technology*. 2013. Vol. 65, no. 4. P. 782–796. URL: https://doi.org/10.1002/asi.23062.
21. Haque N. Catastrophic forgetting in LLMs: a comparative analysis across language tasks. *ArXiv preprint arxiv:2504.01241*. 2025. URL: https://arxiv.org/pdf/2504.01241.
22. HellaSwag: can a machine really finish your sentence? / R. Zellers et al. *ArXiv preprint arxiv:1905.07830*. 2019. URL: https://arxiv.org/abs/1905.07830.
23. HFT: half fine-tuning for large language models / T. Hui et al. *ArXiv preprint arxiv:2404.18466*. URL: https://arxiv.org/abs/2404.18466.
24. Hinton G., Vinyals O., Dean J. Distilling the knowledge in a neural network. *ArXiv preprint arxiv:1503.02531*. 2015. URL: https://arxiv.org/abs/1503.02531.
25. History, development, and principles of large language models: an introductory survey / Z. Wang et. al. *AI and ethics*. 2024. URL: https://doi.org/10.1007/s43681-024-00583-7.
26. How to use Logprobs. *Vellum AI*. URL: https://www.vellum.ai/llm-parameters/logprobs.
27. Howard J., Ruder S. Universal language model fine-tuning for text classification. *Proceedings of the 56th annual meeting of the association for*

computational linguistics (volume 1: long papers), Melbourne, Australia. Stroudsburg, PA, USA, 2018. URL: https://doi.org/10.18653/v1/p18-1031.
28. Improving language understanding by generative pre-training / A. Radford et al. URL: https://www.mikecaptain.com/resources/pdf/GPT-1.pdf.
29. Instruction-following evaluation for large language models / J. Zhou et al. *ArXiv preprint arxiv:2311.07911*. 2023. URL: https://arxiv.org/abs/2311.07911.
30. Kauf C. et al. Log Probabilities Are a Reliable Estimate of Semantic Plausibility in Base and Instruction-Tuned Language Models //arXiv preprint arXiv:2403.14859. – 2024.
31. Keras learning rate schedules and decay - PyImageSearch. *PyImageSearch*. URL: https://pyimagesearch.com/2019/07/22/keras-learning-rate-schedules-and-decay/.
32. Kitaev N., Klein D. Constituency Parsing with a Self-Attentive Encoder. *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: long papers)*, Melbourne, Australia. Stroudsburg, PA, USA, 2018. URL: https://doi.org/10.18653/v1/p18-1249.
33. Kudo T., Richardson J. Sentencepiece: a simple and language independent subword tokenizer and detokenizer for neural text processing. *ArXiv preprint arxiv:1808.06226*. 2018. URL: https://arxiv.org/abs/1808.06226.
34. Language models are few-shot learners / T. Brown et al. *Advances in neural information processing systems*. 2020. Vol. 1, no. 3. P. 1877—1901. URL: https://arxiv.org/abs/2005.14165.
35. Learning entity representation for entity disambiguation / Z. He et al. *Proceedings of the 51st annual meeting of the association for computational linguistics (volume 2: short papers)*. 2013. P. 30–34. URL: https://aclanthology.org/P13-2006.pdf.
36. Lee J., Tang R., Lin J. What would Elsa do? Freezing layers during transformer fine-tuning. *ArXiv preprint arxiv:1911.03090*. 2019. URL: https://arxiv.org/pdf/1911.03090.
37. Lester B., Al-Rfou R., Constant N. The power of scale for parameter-efficient prompt tuning. *ArXiv preprint arxiv:2104.08691*. 2021. URL: https://arxiv.org/abs/2104.08691.
38. Li C. OpenAI's GPT-3 language model: a technical overview. *Lambda | GPU Compute for AI*. URL: https://lambda.ai/blog/demystifying-gpt-3.
39. Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. *AI at Meta*. URL: https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/.

40. LLaMA: open and efficient foundation language models / H. Touvron et al. *ArXiv preprint arxiv:2302.13971*. 2023. URL: https://arxiv.org/pdf/2302.13971.
41. Llama3/MODEL_CARD.md at main · meta-llama/llama3. *GitHub*. URL: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.
42. LoRA: low-rank adaptation of large language models / E. J. Hu et al. *Iclr*. 2022. Vol. 1, no. 2. URL: https://arxiv.org/pdf/2106.09685v1/1000.
43. Loshchilov I., Hutter F. SGDR: stochastic gradient descent with warm restarts. *ArXiv preprint arxiv:1608.03983*. 2016. URL: https://arxiv.org/abs/1608.03983.
44. McCloskey M., Cohen N. J. Catastrophic interference in connectionist networks: the sequential learning problem. *Psychology of learning and motivation*. 1989. P. 109–165. URL: https://doi.org/10.1016/s0079-7421(08)60536-8.
45. Measuring massive multitask language understanding / D. Hendrycks et al. *arXiv preprint arXiv:2009.03300*. 2020. URL: https://arxiv.org/abs/2009.03300.
46. MedAlpaca--an open-source collection of medical conversational AI models and training data / T. Han et al. *ArXiv preprint arxiv:2304.08247*. 2023. URL: https://arxiv.org/abs/2304.08247.
47. Open Ai. GPT-4 technical report. *ArXiv preprint arxiv:2303.08774*. 2023. URL: https://arxiv.org/pdf/2303.08774.
48. Optimization. *Hugging Face – The AI community building the future.* URL: https://huggingface.co/docs/transformers/v4.39.1/main_classes/optimizer_schedules.
49. Pal A., Umapathi L. K., Sankarasubbu M. Medmcqa: a large-scale multi-subject multi-choice dataset for medical domain question answering. Conference on health, inference, and learning. 2022. P. 248—260. URL: https://proceedings.mlr.press/v174/pal22a.html.
50. PaLM: scaling language modeling with pathways / A. Chowdhery et al. *Journal of machine learning research*. 2023. Vol. 24, no. 240. P. 1–113. URL: https://www.jmlr.org/papers/v24/22-1144.html.
51. Parameter-Efficient transfer learning for NLP / N. Houlsby et al. *International conference on machine learning*. 2019. P. 2790—2799. URL: https://proceedings.mlr.press/v97/houlsby19a.html.
52. Parra R. F. Evaluating adaptive layer freezing through hyperparameter optimization for enhanced fine-tuning performance of language models : Doctoral Thesis. 2024.

URL: https://dspace.mit.edu/bitstream/handle/1721.1/157169/figueroa-reyfp-meng-eecs-2024-thesis.pdf.

53. PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods / S. Mangrulkar et al. *GitHub repository*. 2022. URL: https://github.com/huggingface/peft.
54. Pijanowski K. Feature extraction with large language models, hugging face and minio. *MinIO Blog*. URL: https://blog.min.io/feature-extraction-with-large-language-models-hugging-face-and-minio/?utm_source=reddit&utm_medium=organic-social+&utm_campaign=feature_extraction.
55. Ponte J. M., Croft W. B. A language modeling approach to information retrieval. *ACM SIGIR Forum*. 2017. Vol. 51, no. 2. P. 202–208. URL: https://doi.org/10.1145/3130348.3130368.
56. Qwen2. 5 technical report / A. Yang et al. *ArXiv preprint arxiv:2412.15115*. 2024. URL: https://arxiv.org/abs/2412.15115.
57. Ramachandran P., Liu P. J., Le Q. V. Unsupervised pretraining for sequence to sequence learning. *ArXiv preprint arxiv:1611.02683*. 2016. URL: https://arxiv.org/abs/1611.02683.
58. Rei M. Semi-supervised multitask learning for sequence labeling. *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: long papers)*, Vancouver, Canada. Stroudsburg, PA, USA, 2017. URL: https://doi.org/10.18653/v1/p17-1194.
59. Revealing the dark secrets of BERT / O. Kovaleva et al. *ArXiv preprint arxiv:1908.08593*. 2019. URL: https://arxiv.org/pdf/1908.08593.
60. RoBERTa: A robustly optimized bert pretraining approach / Y. Liu et al. *arXiv preprint arXiv:1907.11692*. 2019. URL: https://arxiv.org/abs/1907.11692.
61. RoFormer: Enhanced transformer with Rotary Position Embedding / J. Su et al. *Neurocomputing*. 2023. P. 127063. URL: https://doi.org/10.1016/j.neucom.2023.127063.
62. Rosenfeld R. Two decades of statistical language modeling: where do we go from here?. *Proceedings of the IEEE*. 2000. Vol. 88, no. 8. P. 1270–1278. URL: https://doi.org/10.1109/5.880083.
63. Semi-supervised sequence tagging with bidirectional language models / M. Peters et al. *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: long papers)*, Vancouver, Canada. Stroudsburg, PA, USA, 2017. URL: https://doi.org/10.18653/v1/p17-1161.
64. Shazeer N. GLU variants improve transformer. *ArXiv preprint arxiv:2002.05202*. 2020. URL: https://arxiv.org/abs/2002.05202.

65. Text classification problems via BERT embedding method and graph convolutional neural network / L. Tran et al. *2021 international conference on advanced technologies for communications (ATC)*, Ho Chi Minh City, Vietnam, 14–16 October 2021. 2021. URL: https://doi.org/10.1109/atc52653.2021.9598337.
66. The ultimate guide to fine-tuning llms from basics to breakthroughs: an exhaustive review of technologies, research, best practices, applied research challenges and opportunities / V. B. Parthasarathy et al. *ArXiv preprint arxiv:2408.13296*. 2024. URL: https://arxiv.org/abs/2408.13296.
67. Think you have solved question answering? Try ARC, the AI2 reasoning challenge / P. Clark et al. *ArXiv preprint arxiv:1803.05457*. 2018. URL: https://arxiv.org/abs/1803.05457.
68. Transformers: state-of-the-art natural language processing / T. Wolf et al. *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*. 2020. P. 38—45. URL: https://aclanthology.org/2020.emnlp-demos.6.pdf.
69. TRL: Transformer Reinforcement Learning / L. von Werra et al. *GitHub repository*. 2020. URL: https://github.com/huggingface/trl.
70. What disease does this patient have? A large-scale open domain question answering dataset from medical exams / D. Jin et al. *Applied sciences*. 2021. Vol. 11, no. 14. P. 6421. URL: https://doi.org/10.3390/app11146421.
71. What do you mean by pretraining, finetuning and transfer learning?. *AIML.com*. URL: https://aiml.com/what-do-you-mean-by-pretraining-finetuning-and-transfer-learning-in-the-context-of-machine-learning-or-language-modeling/.
72. WinoGrande: an adversarial winograd schema challenge at scale / K. Sakaguchi et al. *Proceedings of the AAAI conference on artificial intelligence*. 2020. Vol. 34, no. 05. P. 8732–8740. URL: https://doi.org/10.1609/aaai.v34i05.6399.
73. Yu D., Deng L., Dahl G. Roles of pre-training and fine-tuning in context-dependent DBN-HMMs for real-world speech recognition. *Proc. NIPS workshop on deep learning and unsupervised feature learning*. 2010. URL: https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/dbn4asr-nips2010.pdf.
74. Zhang B., Sennrich R. Root Mean Square Layer Normalization. Advances in Neural Information Processing Systems. 2019. Vol. 32.
75. Zhang R., Isola P., Efros A. A. Split-Brain autoencoders: unsupervised learning by cross-channel prediction. *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, HI, 21–26 July 2017. 2017. URL: https://doi.org/10.1109/cvpr.2017.76.