Національний технічний університет України «Київський політехнічний інститут ім. І. Сікорського» Інститут Прикладного Системного Аналізу

Лабораторна робота 3
з дисципліни “Моделювання складних систем”

Виконала студентка групи КА-22

Гаврилюк Богдана

Київ 2025

Лабораторна 3. Transformers (NLP)

Ідея лабораторної: студенти працюють з **готовими трансформерними моделями** (Hugging Face Transformers) для задач NLP, обирають **один датасет** згідно з варіантом і **обов'язково порівнюють щонайменше дві архітектури** (наприклад, BERT vs DistilBERT, RoBERTa vs DeBERTa, BART vs T5 тощо).

Варіант (датасет + завдання)

Рекомендація: ID = (останні 2 цифри залікової) % N + 1, де N — кількість варіантів (нижче 20 варіантів) ID = 26 % 13 + 1 = 1.

SQuAD v2.0 (unanswerable questions).

Завдання: QA з можливістю «немає відповіді».

Порівняти: roberta-base vs deberta-v3-base.

Додатково: проаналізувати, які типи не-відповідних питань найважчі.

Мета роботи

1. Освоїти базовий пайплайн fine-tuning трансформера під конкретну задачу.
2. Навчитися працювати з бібліотеками **Hugging Face Transformers** і **Datasets**.
3. Порівняти **різні архітектури** за якістю, швидкістю, розміром моделі.
4. Навчитися будувати прості, але коректні експерименти (однакові спліти, гіперпараметри, фіксація seed).

Метою роботи було дослідження та порівняння ефективності сучасних трансформерних архітектур у вирішенні задачі Question Answering (QA). Для експерименту було обрано датасет SQuAD v2.0. Ключовою особливістю цієї версії датасету, на відміну від v1.1, є наявність "unanswerable questions" — питань, на які неможливо дати відповідь, спираючись лише на наданий контекст.

Модель повинна навчитися не лише знаходити відповідь (початковий та кінцевий токен) у тексті, але й класифікувати питання як таке, що не має

відповіді (повертати пустий рядок або вказувати на токен [CLS]). Це значно ускладнює задачу та вимагає від моделі глибшого розуміння контексту.

Для порівняльного аналізу було обрано дві моделі архітектури Encoder-only, які є стандартом для екстрактивних задач NLP:

1. RoBERTa-base (roberta-base): це оптимізована версія BERT, розроблена Facebook AI. Вона використовує динамічне маскування, тренувалася на значно більшому обсязі даних, більшими батчами та довше, ніж оригінальний BERT. Також у ній відсутня задача передбачення наступного речення (NSP).
2. DeBERTa-v3-base (microsoft/deberta-v3-base): модель від Microsoft, яка вдосконалює BERT та RoBERTa за допомогою двох технік: розплутаної уваги, де контент і позиція кодуються окремо, та покращеного декодера масок. Версія v3 додатково використовує метод навчання ELECTRA (Replaced Token Detection), що теоретично має забезпечувати вищу якість.

Для забезпечення коректності порівняння обидві моделі навчалися в ідентичних умовах на апаратному забезпеченні NVIDIA Tesla T4 (Google Colab).

- Датасет: підмножина SQuAD v2.0 (2000 прикладів для навчання, 500 для валідації). Обмеження введене для економії часу в рамках лабораторної роботи.
- Гіперпараметри:
    - Кількість епох: 2
    - Batch size: 16
    - Learning Rate: 2e-5
    - Max sequence length: 384
    - Optimizer: AdamW
- Метрика оцінки: Validation Loss (функція втрат на валідаційній вибірці), оскільки на малій вибірці та без складного пост-процесингу вона є надійним індикатором збіжності моделі.

На основі отриманих логів навчання було сформовано таблицю порівняння ключових характеристик моделей.

	Модель	Параметри (M)	Час навчання (сек)	Validation Loss	Розмір ваг (MB)
0	roberta-base	124.06	136.45	1.6766	496.23
1	microsoft/deberta-v3-base	183.83	169.11	2.1602	735.33

DeBERTa має на ~48% більше параметрів через специфіку архітектури та ембедінгів. Більша кількість параметрів прямо впливає на займане місце на диску. RoBERTa навчалася на 33 секунди швидше завдяки меншій архітектурній складності. Інференс та навчання RoBERTa швидші. RoBERTa показала меншу помилку на валідації.

Всупереч теоретичним очікуванням, де DeBERTa v3 зазвичай перевершує RoBERTa, у даному експерименті на малій вибірці (2000 прикладів) RoBERTa-base показала кращий результат. Її Validation Loss склав 1.67 проти 2.16 у DeBERTa.

Це можна пояснити тим, що DeBERTa v3 є складнішою моделлю, яка потребує більше даних для стабілізації градієнтів та "розігріву" (warmup), тоді як RoBERTa виявилася більш стійкою в умовах обмежених даних (few-shot scenario). Також RoBERTa виявилася на ~20% швидшою у навчанні, що робить її ефективнішим вибором при обмежених обчислювальних ресурсах.

Для перевірки роботи моделей було використано текст про історію трансформерів.

Питання 1 (з відповіддю): "Who introduced BERT?"

- RoBERTa: "Google introduced BERT in 2018" (score: 0.1476) — Відповідь повна і правильна.
- DeBERTa: " Google" (score: 0.4560) — Відповідь правильна, але коротка. Модель більш впевнена (вищий score).

Питання 2 (без відповіді): "When was ChatGPT released?" (в тексті немає цієї інформації).

- RoBERTa: пустий рядок, score: 0.0006 — Правильно. Модель ідентифікувала відсутність відповіді.

- DeBERTa: " BERT in 2018." score: 0.0373 — Помилка (Галюцинація). Модель "притягнула" дату з іншого контексту, не розпізнавши, що питання стосується ChatGPT, а не BERT.

Цей тест підтверджує метрики Loss: RoBERTa краще впоралася з логікою SQuAD v2.0, коректно відфільтрувавши питання без відповіді, тоді як DeBERTa намагалася знайти відповідь будь-якою ціною.

Висновки

У ході лабораторної роботи було реалізовано пайплайн для донавчання трансформерів на задачі Question Answering. На обмеженому наборі даних (2000 семплів) модель RoBERTa-base виявилася кращою за DeBERTa-v3-base як за якістю (нижчий Loss, коректна робота з unanswerable questions), так і за швидкістю навчання. DeBERTa-v3 займає майже в 1.5 рази більше пам'яті та довше навчається, що на малих датасетах не виправдовується приростом якості. Експеримент показав важливість правильного вибору архітектури під обсяг даних. Хоча DeBERTa є більш сучасною (SOTA) архітектурою, старіші та простіші моделі (як RoBERTa) можуть демонструвати кращу стабільність при fine-tuning на малих вибірках.