



Лекция 6. Чем кормить генеративные модели?

Дмитрий Калашников, 19.03.2024



Почему мы здесь стоим: Заметки

✳️ Посоветуй, как заставить преподов курса выложить домашние задания?

Закрыть Esc

Я, конечно, не препод, но могу поделиться опытом. Как говорится, все начинается с мотивации. Если вы действительно хотите, чтобы преподаватели курса выложили домашние задания, то вам нужно проявить инициативу. В частности, предложить свою помощь в организации этого дела. В любом случае, это дело преподавателя. И только он может решить, будет ли он выкладывать материалы или нет.

Вставить в заметку

⟳ Переписать



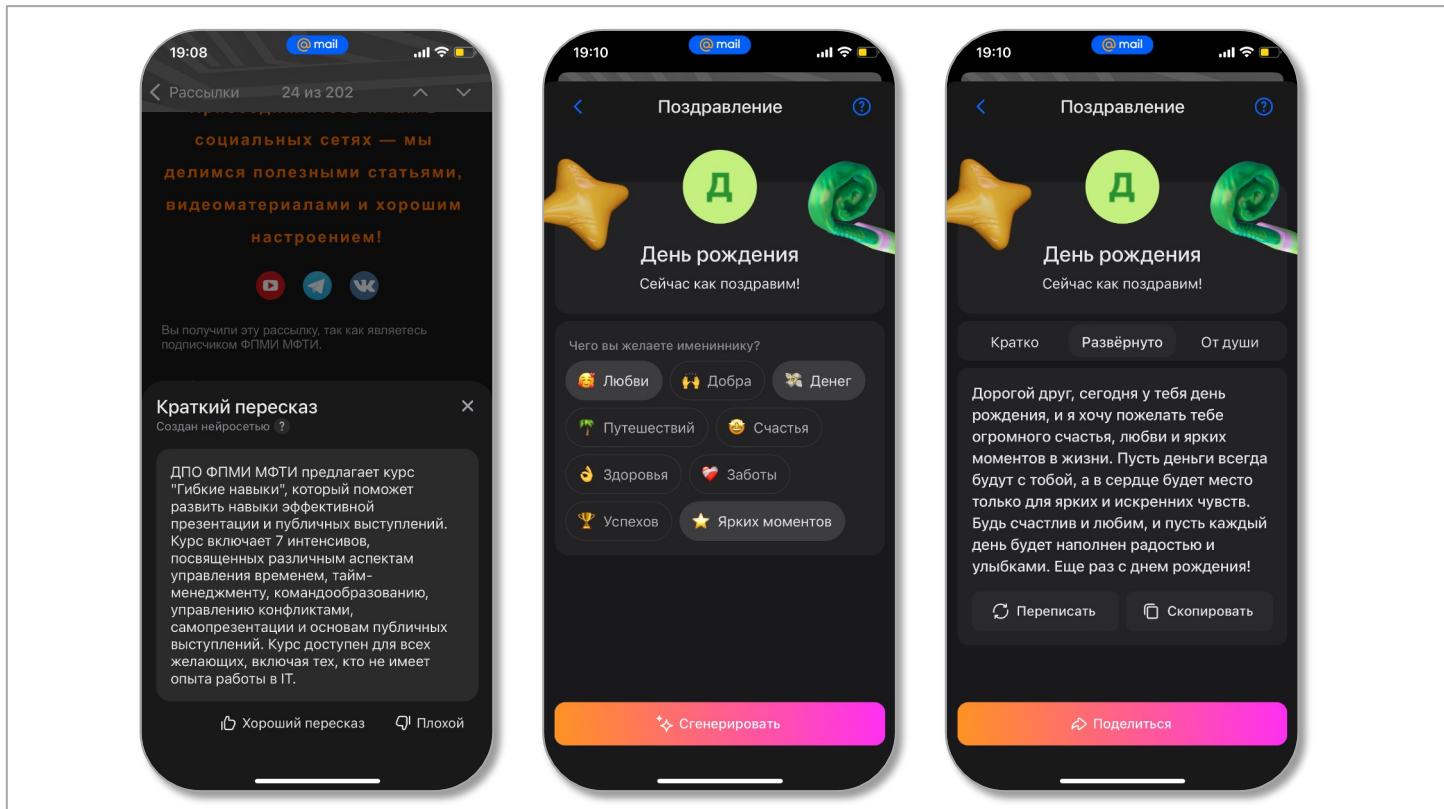
Почему мы здесь стоим: Облако

The screenshot shows a web-based document editor interface. At the top, there's a navigation bar with links to Mail.ru, Почта, Облако (highlighted), Одноклассники, ВКонтакте, Новости, Знакомства, Игры, and Все проекты. The main toolbar includes File, Edit, Insert, Format, Table, Tools, View, and Help. A status bar at the bottom indicates "Страница 1 из 1" and "Все изменения сохранены".

A sidebar on the right contains several sections:

- Скопировать** (Copy): A button with a clipboard icon.
- Новая домашка по LLM-ассистенту. Дедлайн через 2 недели**: A message bubble with text and a rocket emoji.
- Внимание, уважаемые пользователи!**: A message bubble with text and a star emoji.
- Мы рады представить вам нашу новую домашнюю работу по LLM-ассистенту!**: A message bubble with text and a pencil emoji.
- Забудьте о скучных учебниках и скучных лекциях - теперь у вас есть возможность погрузиться в мир LLM-ассистента и стать настоящим экспертом в этой области!**: A message bubble with text and a rocket emoji.
- Дедлайн через 2 недели, так что у вас есть достаточно времени, чтобы подготовиться к этому важному экзамену. Мы уверены, что вы сможете справиться с этой задачей, если будете настойчивыми и целеустремленными.**: A message bubble with text and a hand emoji.
- Написать пост для соцсетей**: A button with a pencil icon.
- Предложите тему**: A text input field with a star icon.

Почему мы здесь стоим: Почта и Календарь



Разминка



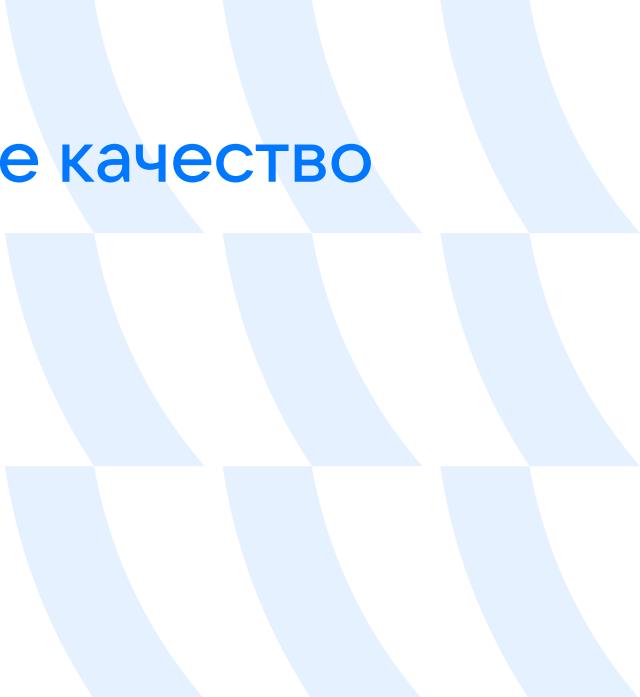


Сколько тайн, сколько загадок...

... и никаких ответов!

Подходящий домен & Подходящее качество

- Данные для обучения влияют на качество генерации / представлений, как минимум, в терминах домена
- Даже если данные взяты в нужном домене, они всё равно могут быть "плохими"
- Сегодня – про данные в терминах их качества



Alignment



Как мы уже умеем



СКОЛЬКО НУЖНО ШТОВ



Few-shot



One-shot



Zero-shot

И GPT нам мало

GPT-1

- Сама по себе хорошо решает одну задачу: "наиболее вероятно" продолжает текст
- Для решения конечных задач: fine-tuning (FT)

GPT-2

- Произвольным инструкциям может следовать, но не очень хорошо / стабильно
- Хороша в режиме few-shot, сомнительна в режиме zero-shot

GPT-3

- Произвольным инструкциям следует лучше, но всё ещё не очень хорошо / стабильно
- Хороша в режиме few-shot, сомнительна в режиме zero-shot

Llama

- Произвольным инструкциям следует неплохо, но всё ещё не очень хорошо / стабильно
- Неплоха в режиме few-shot, сомнительна в режиме zero-shot

Alignment

Что умеют данные модели – хорошо продолжают текст

В результате пре-трейна в них заложены определенные знания о мире и языке, которые можно было извлечь из обучающей выборки

Общая проблема – не умеют следовать пользовательским инструкциям (ожиданиям, намерениям – “user intent”)

Alignment (выравнивание) модели – обучение модели следовать пользовательским ожиданиям

Alignment

Для следования пользовательским ожиданиям недостаточно просто уметь генерировать следующий наиболее вероятный токен

GPT учились на задачу генерации наиболее правдоподобного текста (с точки зрения встречаемости в обучающей выборке)

Эта задача часто расходится с задачей получения ответа, который желал бы получить пользователь в ответ на свой произвольный запрос

Желаемые свойства ответа: полезный, честный и безвредный

Alignment vs GPT-3

GPT-3: Language Models are Few-Shot Learners

- Уже по названию видим акцент на few-shot;
- <https://arxiv.org/abs/2005.14165>

В результате выравнивания получим модель, которая:

- Хорошо понимает пользовательскую инструкцию, попадает в ожидание пользователя;
- Делает это без примеров в режиме zero-shot.

Alignment

- При обучении не вкладываем новых знаний "о мире" в модель;
- Только определяем формат ответа, либо то, как модель должна использовать заложенные в нее на пре-трейне знания в зависимости от контекста на входе;
- Осуществляется на этапе FT: дообучаем модель на задачу следования пользовательским ожиданиям.

Можем обновить схему



Этапы работы с генеративной моделью

Pretrain

- заложили все знания о мире в модель

FT (с целью alignment)

- Показали, как следовать пользовательским инструкциям
- Показали формат, в котором нужно отвечать пользователю
- Показали, как обращаться с заложенными в модель знаниями. В том числе, на что можно отвечать, на что нельзя

Виды FT

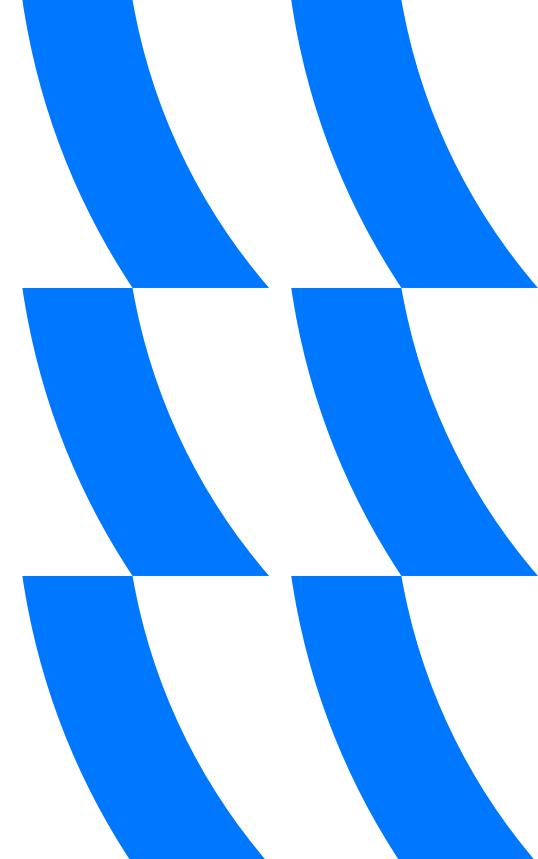
Supervised fine-tuning (SFT)

- Имеем входной запрос, имеем идеальный ответ на него
- Дообучаем модель на задачу языкового моделирования (seq-to-seq)

Обучение с подкреплением (RL)

- Имеем входной запрос, ответ и "награду" (reward) – оценку данного ответа
- Альтернативный вариант: запрос и несколько отранжированных по качеству ответов
- Обучаем reward-модель, оптимизируем политику для генерации ответа

Комбинация SFT и RL



Данные для предобучения



Всё дело в данных



Данные -- самая
важная часть LLM



Важно как количество,
так и качество



Касается как BERT-
подобных моделей, так
и GPT-подобных

Предобучение BERT-like

Только на Википедии учить LM не стоит – тексты “однообразные”
Новостные статьи – на практике хороший источник

GPT-1

- BookCorpus (16 GB)

BERT

- BookCorpus (16 GB)
- English Wikipedia (< 1 GB)

RoBERTa

- BookCorpus (16 GB),
- CC-News – прочистили CommonCrawl News (76 GB)
- OpenWebText – Reddit с ≥ 3 лайками (38 GB)
- Stories – тексты из CommonCrawl в формате историй (31GB)

C4

- Есть завершающая пунктуация на конце текста
- ≥ 5 слов, ≥ 3 предложений
- Удалили все тексты, где встречаются слова из "Списка грязных, озорных, непристойных или иных плохих слов"
- Удалили все тексты с вхождениями "Javascript", "lorem ipsum" или "{", с уведомлениями о политике использования, уведомлениями о куках
- Удалили все маркеры цитирования
- Дедубликация: если какие-то три предложения подряд из некоторого текста содержатся в каком-либо другом тексте, то оставляем только один из них
- Использовали классификатор языка, оставили только тексты с вероятностью 0.99 для английского
- 20 ТВ / месяц -> суммарно 750 GB

"Colossal Clean Crawled Corpus" – очищенный Common Crawl (обкачанный и переведенный в текст веб-архив) – для T5

Предобучение GPT-3

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Предобучение GPT-3

Как чистили данные

- фильтрация Common Crawl
- нечеткая дедубликация

Фильтрация

- Обучение классификатора качественности текста
- Логрег на текстовых фичах (токенизация и частотность)
- Позитивные примеры: тексты из заведомо качественных датасетов (WebText, Wiki и книги)
- Негативные примеры – неотфильтрованный CommonCrawl
- Оставляем текст если: $\text{np.random.pareto}(\alpha) > 1 - \text{document_score}$,
 $\alpha=0.9$

Дедубликация:

- MinHashLSH – наборы последовательностей токенов в тексте отображаем в число (хеш)
- В одном хеше лежат похожие тексты => нечеткие дубли

Предобучение Llama

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

Что нового?

English CommonCrawl (CCNet pipeline)

- клф-р fastText для детекта языка
- n-граммная модель для детекции текстов низкого качества
- клф-р: использовалась бы страница в качестве ссылки на Вики или нет

Github

- Только под разрешающей лицензией
- Фильтрация по длине файла и пропорции alphanumerical
- Удаление служебной разметки, четкая дедубликация

Вики

- Добавили статьи на 20 языках, удалили служебную разметку

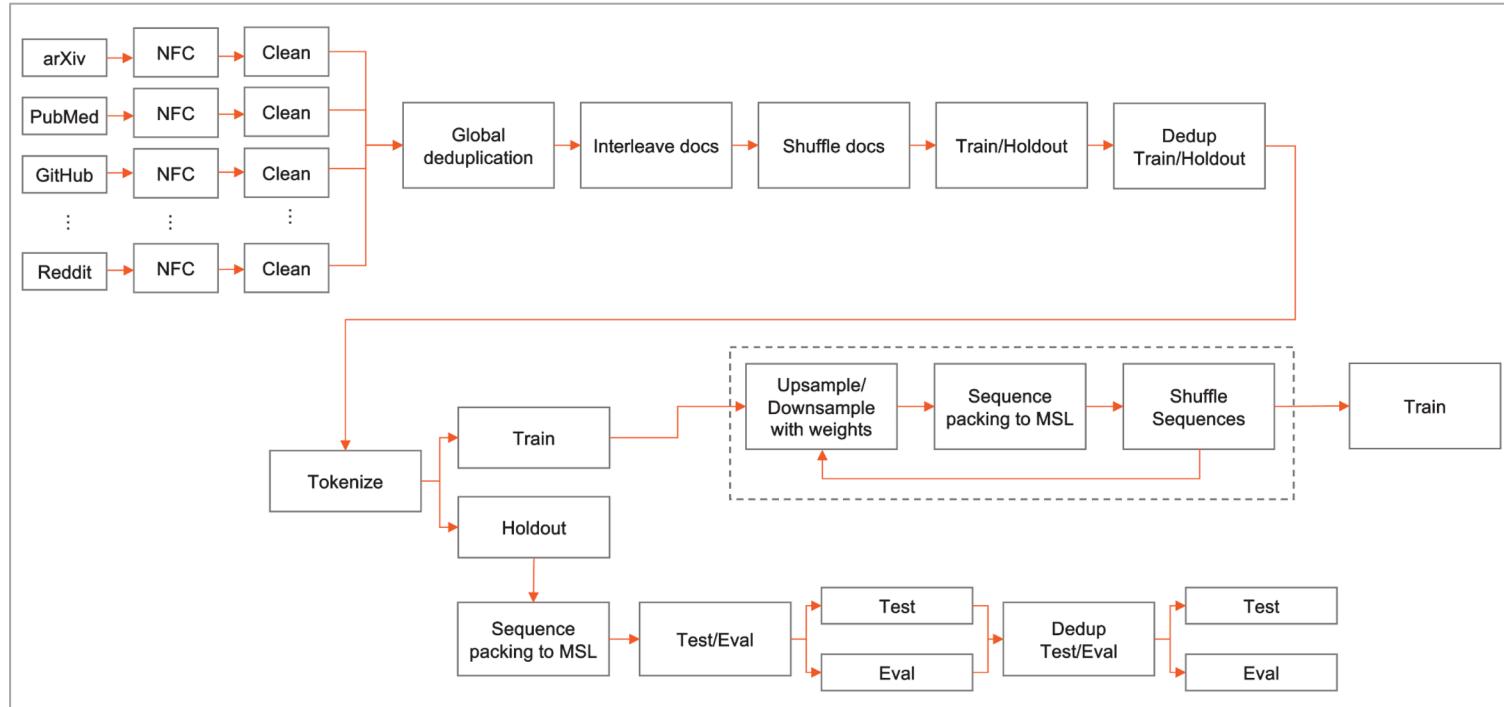
Arxiv

- Добавили научные статьи, удалили всё до первой секции, комменты, библиографию

Stack Exchange

- Вопросы и ответы по разнообразным доменам, сортировка по оценкам ответов

RedPajama и SlimPajama



https://github.com/Cerebras/modelzoo/tree/main/modelzoo/transformers/data_processing/slimpajama

SlimPajama

- NFC-нормализация: удаление неюникод-символов
- Минимальная длина текстов – 200 символов без пунктуации и пробельных символов.
- Дедубликация MinHashLSH с построением графа похожих (дубликат – мера жаккара ≥ 0.8)
- Стратификация семплирования из разных источников с весами
- Умное разделение на трейн / тест, дедубликация теста относительно трейна
- Много оптимизаций, чтобы всё работало быстро и параллельно
- Отличный гайд по очистке выборки с большим количеством доп. ссылок



Как готовить претрейн датасет для LLM

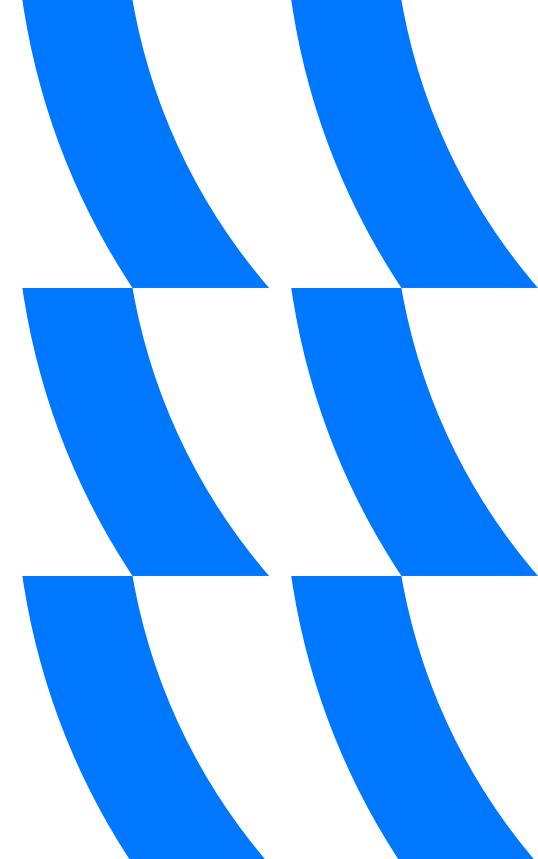
- Определиться с необходимыми языками и доменами, на которых хочется показывать хорошее качество;
- Найти большое количество данных по необходимым срезам, желательно легальных (иначе – анонимизация);
- Извлечение “чистых” текстов, удаление служебной разметки;
- Фильтрация некачественных документов: эвристики и ML-классификаторы;
- Дедубликация: четкая и нечеткая (LSH);
- Во время обучения: стратифицированное семплирование данных из каждого среза (какие-то срезы “показывать” модели чаще, какие-то реже).

Оценка качества LLM



GLUE-бенчмарк для энкодеров

- General Language Understanding Evaluation benchmark
- Цель: оценить качество текстового энкодера на разнообразном срезе NLP задач и выразить в виде метрики либо набора метрик
- Имея метрику (GLUE-score) и независимый разносторонний бенчмарк, можем честно сравнивать между собой разные текстовые энкодеры



GLUE-бенчмарк

Corpus	Train	Test	Task	Metrics	Domain
Single-Sentence Tasks					
CoLA	8.5k	1k	acceptability	Matthews corr.	misc.
SST-2	67k	1.8k	sentiment	acc.	movie reviews
Similarity and Paraphrase Tasks					
MRPC	3.7k	1.7k	paraphrase	acc./F1	news
STS-B	7k	1.4k	sentence similarity	Pearson/Spearman corr.	misc.
QQP	364k	391k	paraphrase	acc./F1	social QA questions
Inference Tasks					
MNLI	393k	20k	NLI	matched acc./mismatched acc.	misc.
QNLI	105k	5.4k	QA/NLI	acc.	Wikipedia
RTE	2.5k	3k	NLI	acc.	news, Wikipedia
WNLI	634	146	coreference/NLI	acc.	fiction books

Table 1: Task descriptions and statistics. All tasks are single sentence or sentence pair classification, except STS-B, which is a regression task. MNLI has three classes; all other classification tasks have two. Test sets shown in bold use labels that have never been made public in any form.

Короткое описание: <https://arxiv.org/pdf/1810.04805.pdf>

Статья: <https://openreview.net/pdf?id=rJ4km2R5t7>

Лидерборд: <https://gluebenchmark.com/leaderboard/>

Бенчмарк SQuAD

Stanford Question Answering
Dataset (SQuAD) –

есть параграф и вопрос.

Нужно ответить на вопрос,
выделив кусок текста с
ответом в параграфе.

A prime number (or a prime) is a natural number greater than 1 that has no positive divisors other than 1 and itself. A natural number greater than 1 that is not a prime number is called a composite number. For example, 5 is prime because 1 and 5 are its only positive integer factors, whereas 6 is composite because it has the divisors 2 and 3 in addition to 1 and 6. The fundamental theorem of arithmetic establishes the central role of primes in number theory: any integer greater than 1 can be expressed as a product of primes that is unique up to ordering. The uniqueness in this theorem requires excluding 1 as a prime because one can include arbitrarily many instances of 1 in any factorization, e.g., 3, $1 \cdot 3$, $1 \cdot 1 \cdot 3$, etc. are all valid factorizations of 3.

What is the only divisor besides 1 that a prime number can have?

Ground Truth Answers: itself itself itself itself itself

What are numbers greater than 1 that can be divided by 3 or more numbers called?

Ground Truth Answers: composite number composite number composite number primes

What theorem defines the main role of primes in number theory?

Ground Truth Answers: The fundamental theorem of arithmetic fundamental theorem of arithmetic fundamental theorem of arithmetic fundamental theorem of arithmetic fundamental theorem of arithmetic

Бенчмарк RACE

ReADING Comprehension from Examinations (RACE) – понимание прочитанного.
Есть длинный пассаж и 4 вопроса по нему, нужно выбрать верный

A.1 Example Question of Passage Summarization

Passage: Do you love holidays but hate gaining weight? You are not alone. Holidays are times for celebrating. Many people are worried about their weight. With proper planning, though, it is possible to keep normal weight during the holidays. The idea is to enjoy the holidays but not to eat too much. You don't have to turn away from the foods that you enjoy.

Here are some tips for preventing weight gain and maintaining physical fitness:

Don't skip meals. Before you leave home, have a small, low-fat meal or snack. This may help to avoid getting too excited before delicious foods.

Control the amount of food. Use a small plate that may encourage you to "load up". You should be most comfortable eating an amount of food about the size of your fist.

Begin with soup and fruit or vegetables. Fill up beforehand on water-based soup and raw fruit or vegetables, or drink a large glass of water before you eat to help you to feel full.

Avoid high-fat foods. Dishes that look oily or creamy may have large amount of fat. Choose lean meat. Fill your plate with salad and green vegetables. Use lemon juice instead of creamy food.

Stick to physical activity. Don't let exercise take a break during the holidays. A 20-minute walk helps to burn off extra calories.

Questions:

What is the best title of the passage?

Options:

- A. How to avoid holiday feasting
- B. Do's and don'ts for keeping slim and fit.
- C. How to avoid weight gain over holidays.
- D. Wonderful holidays, boring experiences.

Оценка генеративных моделей (GPT-2)

- Перплексия – неправдоподобность (сгенерированного) текста;
- Чем ниже, тем лучше;
- Замеряется на тренировочной выборке, на валидационной, а также можно использовать как оценку качества генерации на специально выделенных срезах;
- Языковую модель можно использовать и для классификации: вероятность какого токена больше: "да" или "нет"? С такой клф-ей можно также мерить качество.

Language Models are Unsupervised Multitask Learners

LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
------------------	------------------	-----------------	-----------------	--------------------	--------------	-----------------	----------------	----------------------	--------------

GPT-2 бенчмарки

- Children's Book Test (CBT) – качество на разных категориях (включая части речи) слов. Пропускаем слово, 10 вариантов ответа на вопрос про него.
- LAMBADA – понимание долгосрочных зависимостей в тексте: предсказываем последнее слово в тексте с длинным контекстом
- Winograd Schema Challenge (WSC) – способность к "здравому смыслу" путем разрешения двусмыслистостей в тексте
- Conversation Question Answering dataset (CoQA) – понимание прочитанного, 7 доменов, диалоги, учим модель отвечать на вопросы "на понимание" по типу "почему?"
- CNN and Daily Mail – суммаризация
- WMT-14 – перевод
- Natural Questions – Question Answering

Что полезно проверять

- Generalization vs Memorization: насколько модель обобщается на новые запросы, а насколько просто "запомнила" тексты из трейн-датасета и на инференсе их воспроизводит
- Данные в teste / бенчмарках не содержатся в каких-либо текстах из трейн-датасета
- Генерация ответа на вопрос из теста / бенчмарка не содержится в каких-либо текстах из трейн-датасета



Вернемся к
Alignment



Как хотим и как не хотим

Prompt *Explain the moon landing to a 6 year old in a few sentences.*

Completion GPT-3

Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.

InstructGPT

People went to the moon, and they took pictures of what they saw, and sent them back to the earth so we could all see them.

Alignment

“Выравниваем” модель – учим ее следовать пользовательским ожиданиям, генерировать текст в определенном формате, стиле и тп

SFT: делаем шаблонизированный вход

- Системный запрос (system prompt)
- Запрос / задание
- Контекст запроса (если нужен)



Шаблон промпта

{instruction} - задание, {input} - контекст

Below is an instruction
that describes a task,
paired with an input that
provides further context.
Write a response that appropriately
completes the request.

```
### Instruction:  
{instruction}
```

```
### Input:  
{input}
```

```
### Response:
```

Below is an instruction
that describes a task.
Write a response that appropriately
completes the request.

```
### Instruction:  
{instruction}
```

```
### Response:
```

Что видит пользователь

Пользователь задает запрос в свободном формате, система переводит его в инструкцию и контекст, вставляет в шаблон и передает в генеративную модель

Как студенту сдать спецкурс по LLM?

Вот несколько советов, которые могут помочь студенту успешно сдать спецкурс по Large Language Models (LLM):

1. Активно участвуйте в занятиях. Посещайте лекции, семинары и практические занятия, задавайте вопросы, участвуйте в дискуссиях. Это поможет вам лучше понять материал.
2. Тщательно изучайте рекомендованную литературу и ресурсы.

Ты - умный помощник, который помогает пользователю решать его задачи.

Ниже дана инструкция, описывающая задачу.
Напиши ответ, который в точности выполняет пользовательский запрос

@@@ Инструкция:
Как студенту сдать спецкурс по LLM?

@@@ Ответ:
Вот несколько советов, которые могут помочь студенту успешно сдать спецкурс по Large Language Models (LLM):
1. Активно участвуйте в занятиях. Посещайте лекции, семинары и практические занятия, задавайте вопросы, участвуйте в дискуссиях. Это поможет вам лучше понять материал. Тщательно изучайте рекомендованную литературу и ресурсы.

Как использовать

- Как правило, системный промпт для всех примеров одинаковый;
- Instruction, input (если есть) и response (если пример обучающий) – свои для каждого обучающего примера;
- Генеративная модель учится через SFT на response;
- Во время инференса используем тот же шаблон, оканчивающийся на "... response: ", и просим модель "продолжить" текст;
- "Правильного" шаблона нет, под каждую модель нужно подбирать свой в соответствии с решаемой задачей и доменом.



Данные. Сколько и откуда



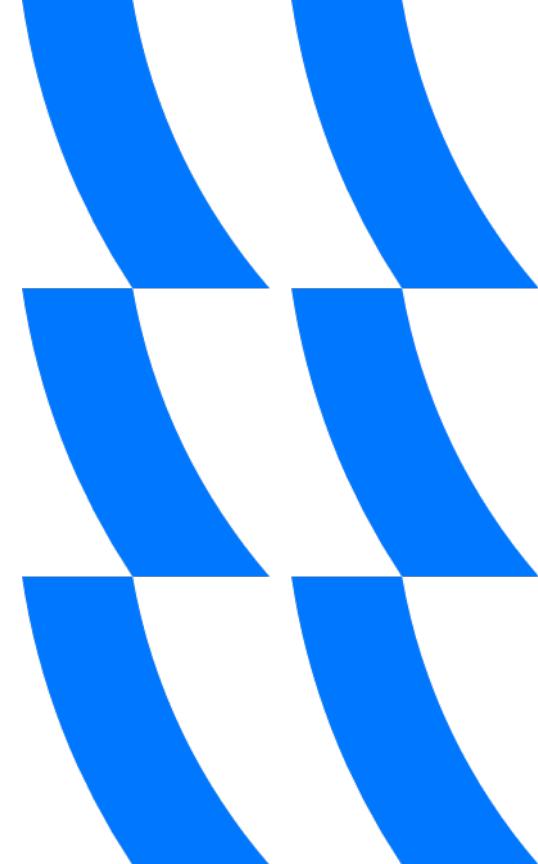
Инструктивный датасет

1. Выравниваем модель с помощью Supervised Fine-Tuning в нужном домене и нужном формате;
2. Для того, чтобы научить модель следовать инструкциям, нужно собрать инструктивный обучающий датасет;
3. Откуда брать данные для SFT:
 - Писать с нуля с помощью редакторов (дорого, но лучше всего)
 - Использовать доступные в Интернете множества запросов
 - Генерировать (такие данные называются синтетическими)



Self-instruct

- Полуавтоматический подход для генерации обучающих данных (инструкций, пользовательских запросов) с помощью LLM;
- С помощью него можно создать большой инструктивный датасет для SFT с малым количеством ручной разметки;
- С помощью инструктивного датасета создадим инструктивную модель;
- Instruction-tuned LM = fine-tuned to respond to instruction.



Self-instruct

Пример постановки: хотим научить модель вести диалог;

Дано: модель, которая хорошо продолжает текст и плохо ведет человекоподобный разговор;

Проблемы запросов и ответов, написанных людьми:

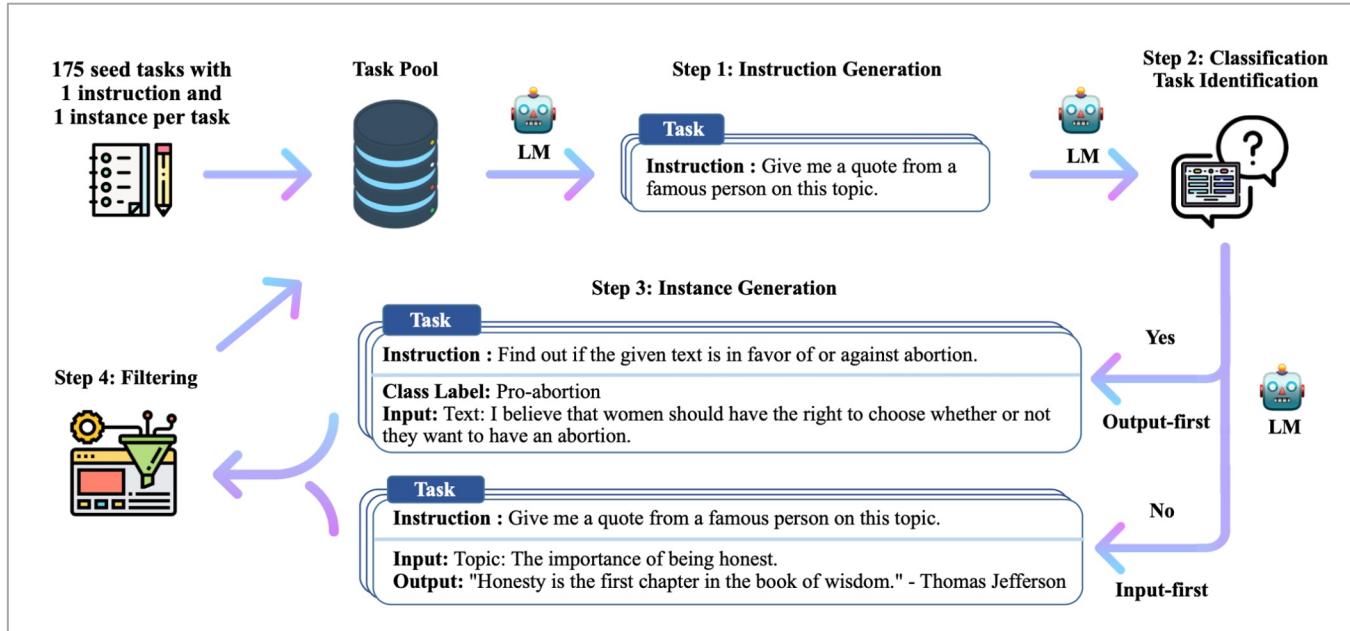
- низкая скорость написания => малое количество текстов
- ограниченное разнообразие текстов
- ограниченная креативность текстов (фантазия не бесконечна)
- дорого и часто нужна высокая экспертиза в домене

Эти проблемы могут привести к плохой генерализации модели

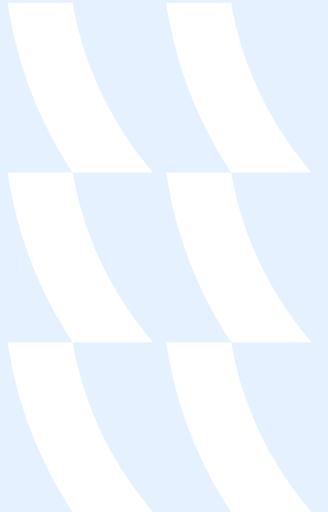
Решение: генерируем выборку с помощью инструктивной LLM

Нюанс: нужна хорошая инструктивная модель, в качестве которой можно взять InstructGPT и ее аналоги, например, от OpenAI

Self-instruct



Self-instruct – алгоритм



1

Вход: Формируем начальный пул задач (seed-set) – 175 качественных ручных примеров;

4

Фильтруем околодубликаты и некачественные инструкции, input-ы и output-ы;

2

С помощью LLM генерируем инструкции для нескольких новых задач;

5

Повторяем пока не получим нужное количество задач, input-ов и таргетов для них

3

Создаем по несколько input и output для этих задач;

6

Выход: большой датасет разнообразных качественных инструкций и таргетов

Self-instruct – детали

1. Семплируем 8 инструкций из пула задач: 6 ручных и 2 сгенерированных (для разнообразия);
2. Просим LLM для инструкции сгенерировать input и output:
 - Если запрос сводится к классификации (текст -> конечное число классов), то оптимальнее сначала генерировать по инструкции output (класс), а потом по инструкции и классу – input;
 - В противном случае оптимальнее instruction -> input -> output.
3. LLM часто генерирует околодубликаты => фильтрация:
 - По похожести: ROUGE-L $\geq 0.7 \Rightarrow$ не добавляем новую задачу в task pool;
 - По качеству: различные эвристики;
4. Как только сгенерировали нужное кол-во инструкций – используем датасет для SFT какой-нибудь LLM

Self-instruct – результат

1

Получено 52к
качественных
инструкций, а
также input-ы и
output-ы для них

2

Слабое
пересечение с
seed-set –
высокое
разнообразие

3

Дообученная на
полученных
инструкциях LM
лучше чем
исходная эта LM
на 33%

4

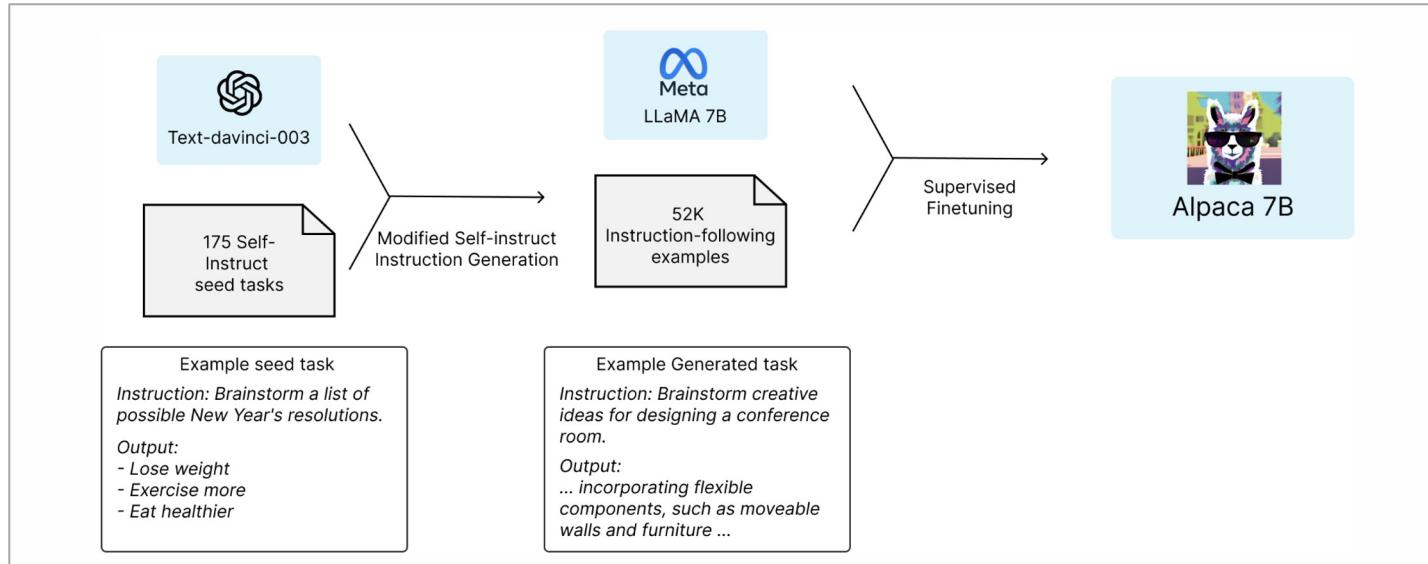
Качество близко к
InstructGPT

5

Ускорило
развитие LLM в
опенсорсе

Alpaca

- Стенфорд дообучил LLaMa 7B на 52к инструкций, сгенерированных с помощью self-instruct;
- Качество близко к инструктивной GPT-3.5, но при этом модель и подход доступны всем.



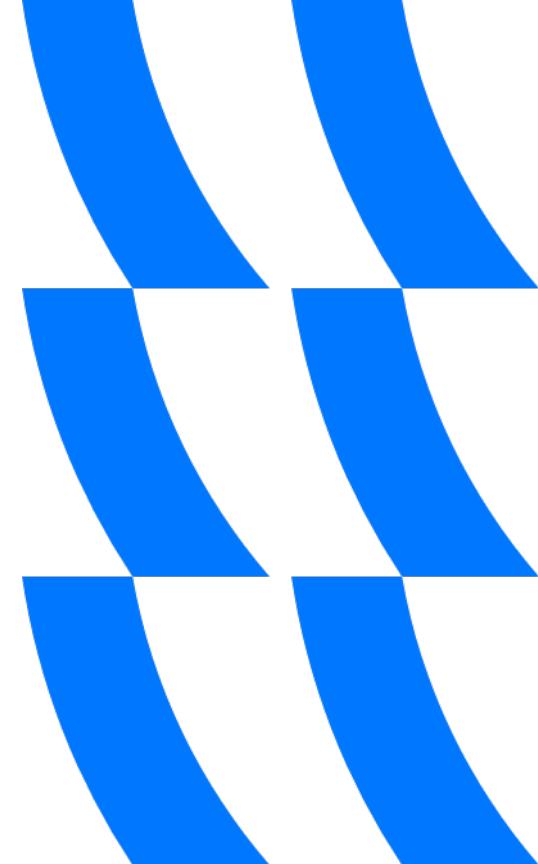
Важность хороших данных

1. Частые проблемы при составлении инструктивного датасета:

- плохое разнообразие и околодубли;
- синтетические данные часто выглядят слишком синтетически.

2. Если в датасете много похожих инструкций, то у модели будет на них bias, и она будет плохо обобщаться на новые инструкции;

3. Важные критерии датасета: качество, репрезентативность и разнообразие.



LIMA – СКОЛЬКО НУЖНО ИНСТРУКЦИЙ

- Superficial Alignment Hypothesis: все знания в LLM заложены на этапе пре-трейна, а во время alignment выучивается только стиль или формат взаимодействия с пользователем
- В таком случае, для выравнивания модели нужно небольшое количество качественных, разнообразных и репрезентативных точек и ответов на них в требуемом формате
- Аккуратно собрали 1000 точек, дообучили на них LLaMa 65B – модель близка по качеству к GPT-4, выровненной на огромных датасетах, и лучше Alpaca 65B, выровненной на 52k self-instruct инструкций
- Для LLaMa 7B достаточно ≥ 2000 точек

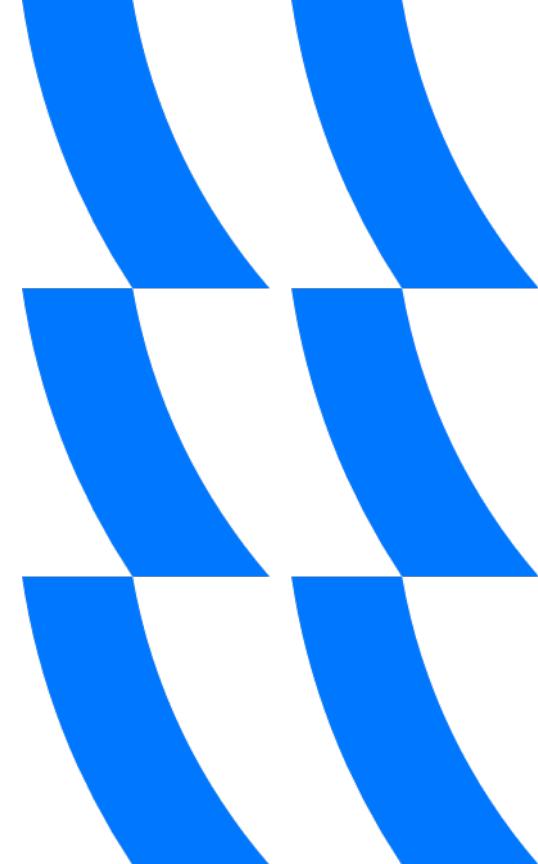
<https://arxiv.org/pdf/2305.11206.pdf>

Про репрезентативность

Откуда брали данные:

- Stack Exchange
- wikiHow
- The Pushshift Reddit Dataset
- SNI
- написанные руками, в том числе про этику и безопасность

1. Покрыли максимальное количество тематик: на каждом из срезов выбиралось по несколько точек из каждой подтемы;
2. Много эвристик, фильтрующих по качеству;
3. На тесте – измерение согласованности асессоров между собой.



“Умная” дедубликация



Дедубликация

Четкая

удаляем четкие совпадения, посимвольно либо после предобработки (например, удалить из текста всё, кроме букв, и привести к нижнему регистру);



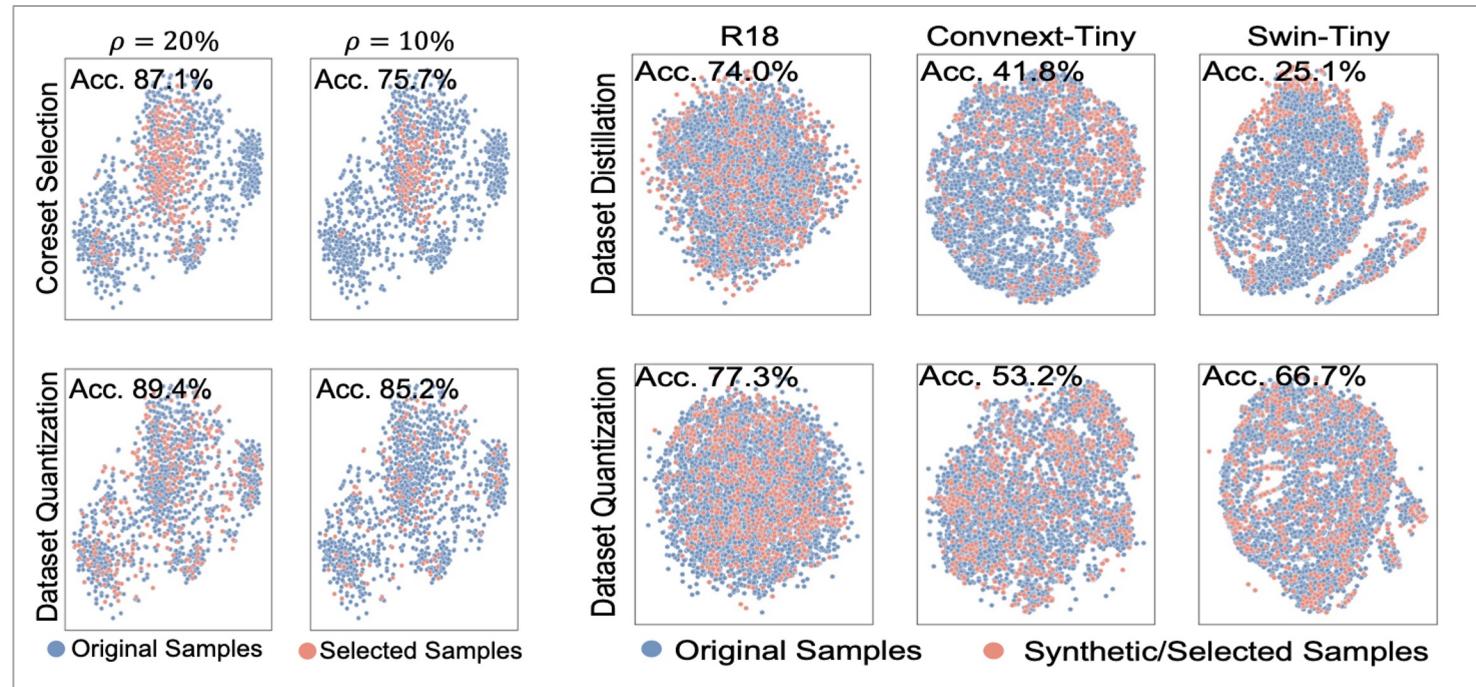
Нечеткая – по степени похожести

- на текстовом уровне: доля общих n-грамм, MinHashLSH и пр.
- на векторном уровне: переводим тексты в векторное пространство (BERT), считаем попарно косинусную близость / формируем на ее основе кластера, удаляем дубли по порогу / связности.

Умная дедубликация – зачем

1. Убрать дубли для улучшения качества
2. Уменьшить размер выборки без ухудшения ее качества
 - ускорение обучения
 - выравнивание размеров различных срезов (для избежания сдвига распределения на какой-то слишком частый срез)
3. Почему нельзя просто семплировать из выборки?
 - Возникает selection bias – с большей вероятностью выбираются точки в областях с высокой плотностью точек
 - Точки в областях с низкой плотностью часто сильно влияют на качество
 - Приводит к ухудшению разнообразия новой выборки относительно старой

Какая дедубликация визуально лучше?



Dataset Distillation



1

Есть датасет под какую-то задачу, есть модель, которую можно на нее обучить

2

Вместе отсева текущих точек синтезируем малое количество наиболее информативных точек

3

Как синтезируем:
минимизируем для
выбранной модели,
обученной на синтетике,
лосс на исходной
тренировочной выборке

4

Минус: если меняем архитектуру модели, то обучать на такой синтетике ее уже нельзя

Coreset Selection

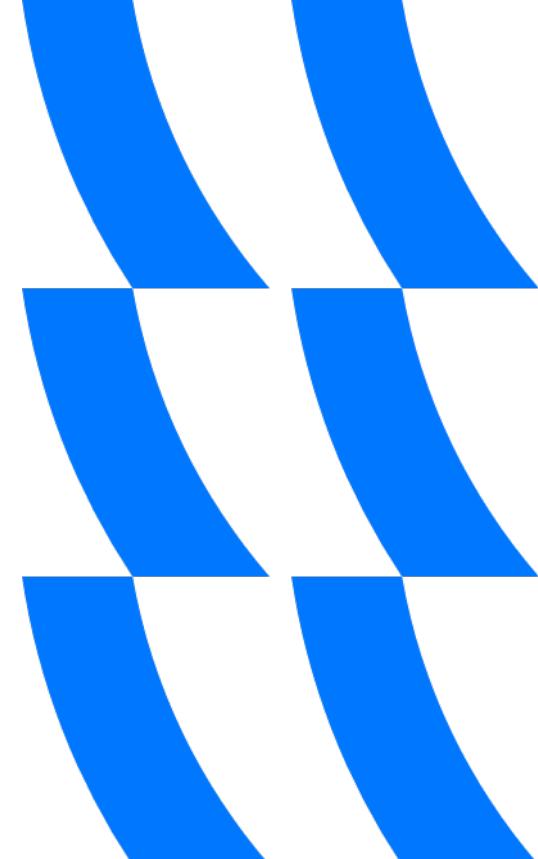
Идея:

итеративно отбираем точки из исходной выборки в
отдельную корзину

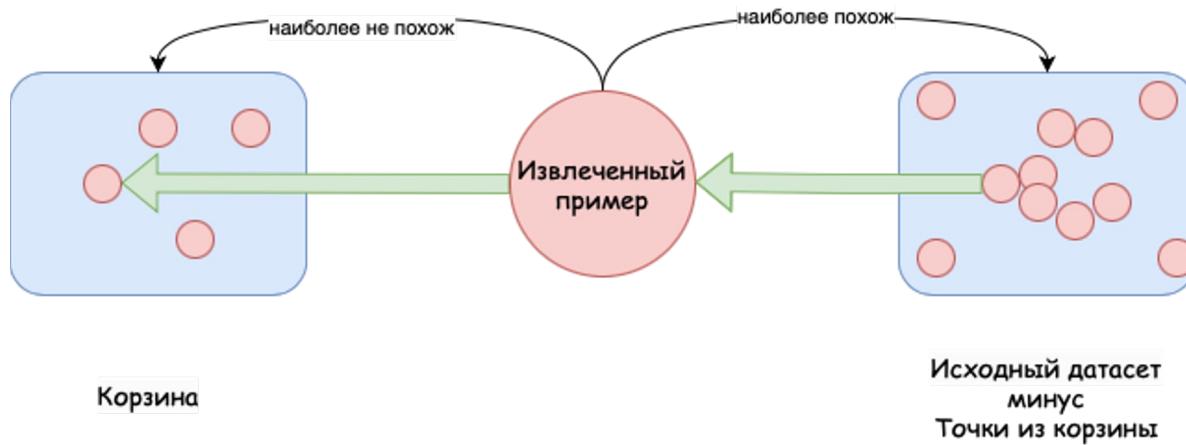
На каждом шаге для перемещения выбирается точка,
которая:

- максимально похожа на оставшуюся
исходную выборку (репрезентативность)
- максимально непохожа на точки из уже
собранной корзины (разнообразие)

В случае текстов: косинусная похожесть
семантических эмбеддингов



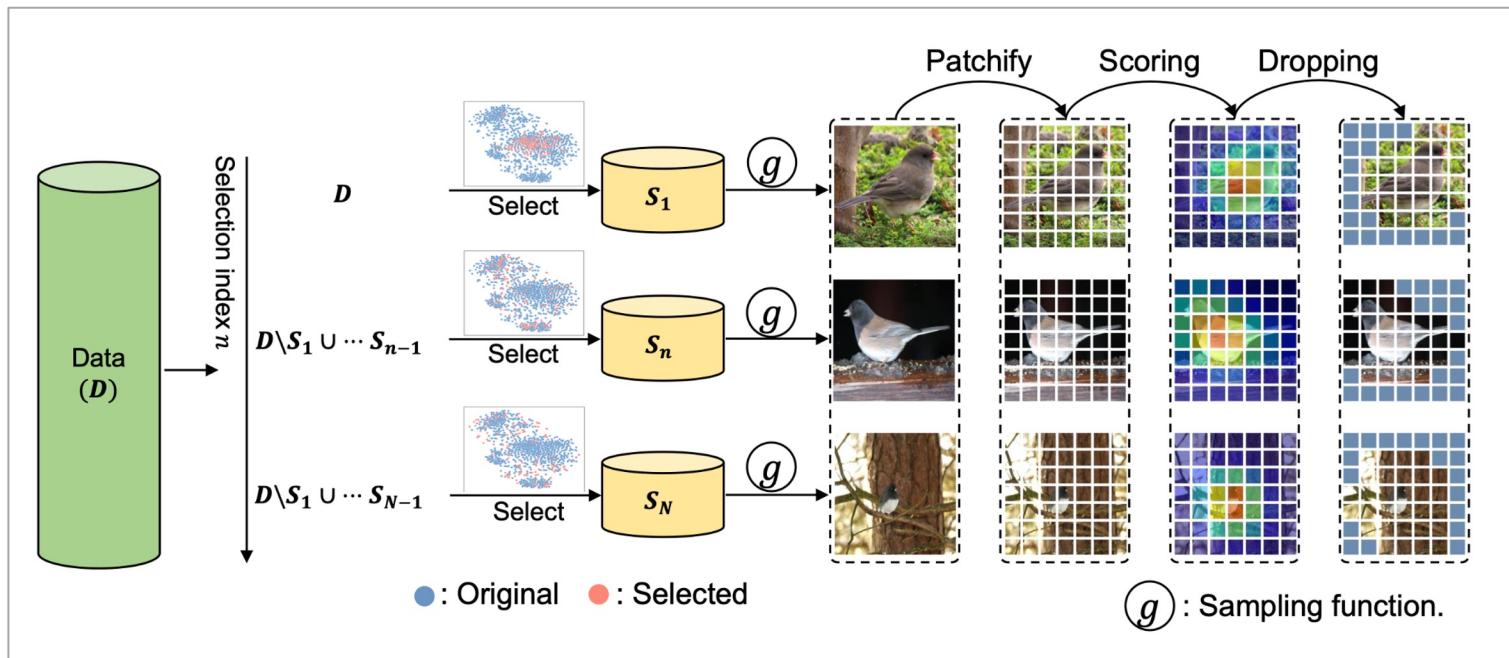
Coreset Selection



$$P(x_k) = \sum_{p \in \mathbf{S}_1^{k-1}} \underbrace{\|f(p) - f(x_k)\|_2^2}_{C_1(x_k)} - \sum_{p \in \mathbf{D} \setminus \mathbf{S}_1^{k-1}} \underbrace{\|f(p) - f(x_k)\|_2^2}_{C_2(x_k)},$$

Dataset Quantization

Selection bias будет и здесь => несколько раз применяем CS

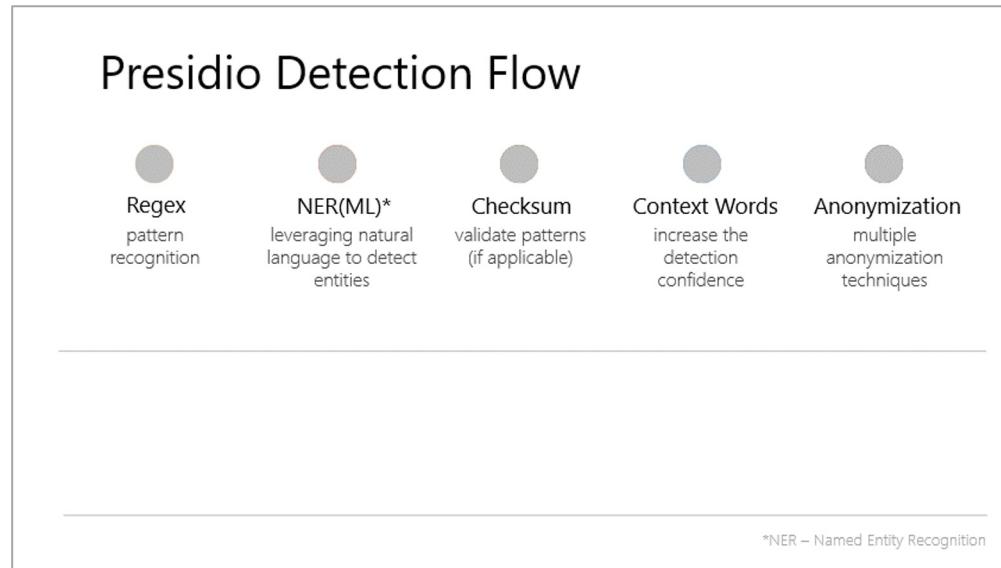


DQ vs CS vs DD

Method	Arch. generalized	Scalable	Time Efficient	Diverse	Data Efficient
DD	✗	✗	✗	✓	✓
Coreset	✓	✓	✓	✗	✗
DQ	✓	✓	✓	✓	✓

Анонимизация

Пользовательские данные даже из открытых источников (посты в социальных сетях) бывает важно анонимизировать. Почитать можно тут:



<https://microsoft.github.io/presidio/>

Качество генерации



Качество инструктивной модели



Измерение качества генерации текста – сложная задача



Перплексия – говорит о правдоподобности текста, но часто ничего не говорит о его качестве



В каждой прикладной задаче должны быть свои критерии

Оценка качества

Абсолютная:

- формулируем критерии
- собираем тестовый сет
- пишем качественную инструкцию с примерами
- размечаем сами / привлекаем асессоров / LLM API

Относительная:

- сравниваем модели между собой на лучше / хуже (side-by-side – SBS)
- оценка людьми либо LLM API (например, GPT-4)
- также нужны критерии сравнения и примеры

Общие критерии

Полезность

- насколько качественно LLM выполняем задачу, поставленную пользователем.
- помимо самой задачи, сюда можно включить также читаемость, стилистику, грамматику

Честность

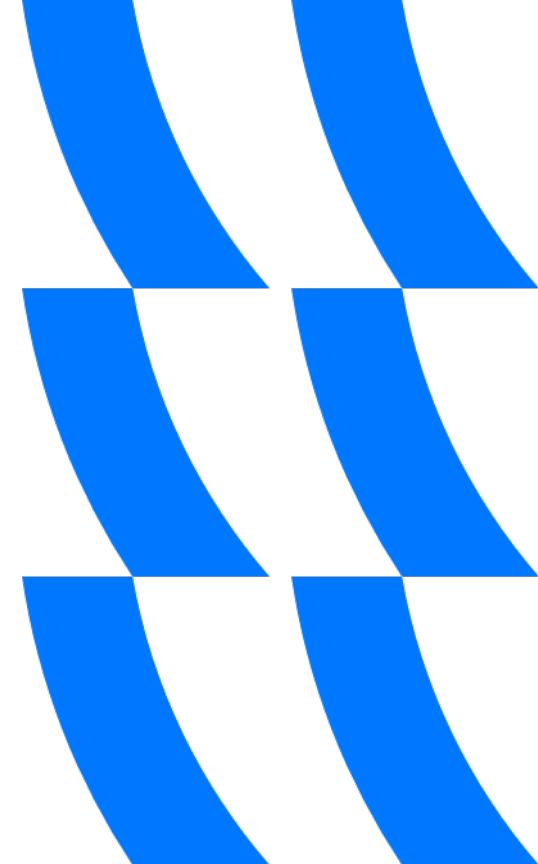
- насколько ответ соответствует действительности: отсутствие галлюцинаций, фактовых ошибок и тд

Безопасность

- насколько модель способна навредить пользователю и другим людям своим ответом
- например, модель не должна использовать ненормативную лексику и угрожать:)
- модель должна отвечать этично и безопасно даже если пользователь вынуждает модель делать по-другому, а также задаем неэтичные и небезопасные вопросы

Исправление ошибок

- Частая проблема – безопасность модели;
- Нужно уметь исправлять дыры в безопасности и прочие проблемы в ответах в целевых задачах;
- Простой способ – собираем фидбек / отсматриваем запросы глазами, пишем руками корректные ответы, докидываем в SFT пары "старый запрос - новый ответ";
- Более продвинутый – RL, chain of highlights, пары "плохой-хороший", ранжирование ответов и использование этого для обучения – хороший способ делать hotfix в модели. В следующих лекциях.



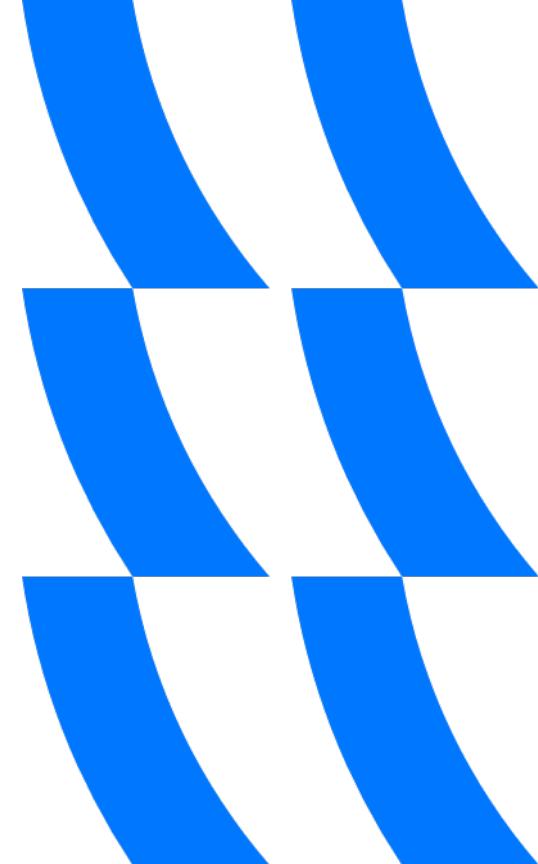
Как дешево улучшить качество генерации

Prompt-engineering

- подбираем промпт так, чтобы задача решалась лучше, добавляем критерии в запрос и тд;
- Также с помощью него можно улучшить качество генерируемых данных с помощью self-instruct;
- Обратная сторона – если в системный шаблон модели подавать слишком много контекста, критериев и ограничений, то модель может начать галлюцинировать, привлекая в ответе текст из системного шаблона.

Выводы

- Что такое LLM Alignment
- Как готовить данные для предобучения
- Оценка качества предобученной модели
- Self-instruct и Alpaca
- Сколько нужно точек для выравнивания
- Умная дедубликация
- Оценка качества инструктивной модели



Спасибо за внимание!

Дмитрий Калашников, разработчик-исследователь

