

## **Постановка задачи**

Требуется разработать чат-бота, который должен вести диалог как определенный персонаж сериала, имитируя стиль и манеру конкретного персонажа сериала.

**Github проекта:** [https://github.com/YurezSml/MIPT\\_NLPGener\\_HW2/](https://github.com/YurezSml/MIPT_NLPGener_HW2/)

**Обученная модель:** [https://huggingface.co/yurezsml/phi2\\_chan](https://huggingface.co/yurezsml/phi2_chan)

**Colab с обучением чат-бота:**

[https://colab.research.google.com/drive/1bIn2r\\_c95wnVgJnyrt0x5CwezHYp6cs2](https://colab.research.google.com/drive/1bIn2r_c95wnVgJnyrt0x5CwezHYp6cs2)

**Colab с запуском чат-бота:**

[https://colab.research.google.com/drive/1IaFwWL12LMLIdNgq9u1QCkxL87\\_a9GAx](https://colab.research.google.com/drive/1IaFwWL12LMLIdNgq9u1QCkxL87_a9GAx)

## **Описание и анализ данных данных**

В качестве рассматриваемого сериала был выбран сериал «Друзья»

Источник данных: <https://www.kaggle.com/datasets/gopinath15/friends-netflix-script-data>

Датасет состоит из 5 столбцов

Text – реплика персонажа и технический текст

Speaker – персонаж, говорящий реплику

Episode – номер и название эпизода

Season – номер сезона

Show – название шоу

Всего датасет содержит 69974 записи. Как будет видно далее, из анализа, не все они являются репликами персонажей, часть можно отнести к техническим записям, связанным с рекламными вставками, окончанием серии и т.д. Несколько записей исходного датасета приведены на рисунке 1.

Ввод [123]:

```

1 df = pd.read_csv('Friends.csv')
2 df

```

Out[123]:

	Text	Speaker	Episode	Season	Show
0	Originally written by Marta Kauffman and David...	NaN	Episode-01-The One Where Monica Gets a New Roo...	Season-01	Friends
1	Transcribed by guineapig.	NaN	Episode-01-The One Where Monica Gets a New Roo...	Season-01	Friends
2	CENTRAL PERK. (ALL PRESENT EXCEPT RACHEL AND ...	SCENE 1	Episode-01-The One Where Monica Gets a New Roo...	Season-01	Friends
3	There's nothing to tell! He's just some guy I...	MONICA	Episode-01-The One Where Monica Gets a New Roo...	Season-01	Friends
4	C'mon, you're going out with the guy! There's...	JOEY	Episode-01-The One Where Monica Gets a New Roo...	Season-01	Friends
...	...	...	...	...	...
69969	Then I'm happy too. (They're still hugging - ...	Ross	Episode-15-The One Where Estelle Dies	Season-10	Friends
69970	COMMERCIAL BREAK	NaN	Episode-15-The One Where Estelle Dies	Season-10	Friends
69971	Estelle's memorial service. Joey is giving a ...	[Scene	Episode-15-The One Where Estelle Dies	Season-10	Friends
69972	Thank you all for coming. We're here today to...	Joey	Episode-15-The One Where Estelle Dies	Season-10	Friends
69973	THE END	NaN	Episode-15-The One Where Estelle Dies	Season-10	Friends

69974 rows × 5 columns

Рисунок 1 – Исходный датасет, состояние до обработки

В ходе анализа и первичной обработки датасета все имена авторов реплик были приведены к одному виду. В датасете обнаружены пропущенные значения для поля с авторами реплик (6284), в ходе анализа было принято решение использовать часть из них в качестве возможных разделителей при разбиении датасета на несвязанные диалоги. Остальные записи с пропущенными авторами реплик были удалены, как разрывающие контекст внутри диалога. Кроме того, в качестве разделителей между диалогами были приняты специальные отметки в поле ‘Text’.

Результаты анализа и первичной обработки представлены в файле [https://github.com/YurezSml/MIPT\\_NLPGener\\_HW2/blob/main/Analysys.ipynb](https://github.com/YurezSml/MIPT_NLPGener_HW2/blob/main/Analysys.ipynb)

## Подготовка данных

В результате первичной обработки был подготовлен датасет, состоящий из двух полей:

Text – реплика персонажа и технический текст

Speaker – персонаж, говорящий реплику

Независимые или слабо зависимые по контексту диалоги отделены с помощью разделителя 'break' в поле ‘Speaker’ (Рисунок 2). Нельзя говорить о полной независимости диалогов друг от друга т.к. возможно продолжение общего контекста как через несколько диалогов, так и в разных сериях. Чтобы устранить это и сделать максимально связный контекстом датасет с репликами требуется значительное количество времени, ручного труда и погружение в сюжет.

	Text	Speaker
0	central perk.	break
1	there's nothing to tell! he's just some guy i ...	monica
2	c'mon, you're going out with the guy! there's ...	joe
3	so does he have a hump? a hump and a hairpiece?	chandler
4	wait, does he eat chalk?	phoebe
...	...	...
63484	yeah, yeah, oh!	ross
63485	oh! oh, i'm so happy.	rachel
63486	then i'm happy too.	ross
63487	estelle's memorial service. joe is giving a s...	break
63488	thank you all for coming. we're here today to ...	joe
63489 rows × 2 columns		

Рисунок 2 – Датасет после первичной обработки

Полученный датасет позволяет выбрать реплики любого из персонажей в качестве объекта для обучения. В своей работе я выбрал Чендлера (chandler), поскольку его реплики и шутки являются одними из самых ярких в сериале и выделяются на фоне остальных. Реплики персонажей преобразовывались в датасет для обучения следующим образом: сначала записывались все реплики, предшествующие реплике выбранного персонажа, без разделения на других персонажей, далее записывалась реплика персонажа. Процедура повторялась до тех пор, пока не была достигнута метка break или заданная глубина контекста CONT\_DEPTH. Если диалог начинается с реплики выбранного персонажа, она не добавлялась в итоговый датасет т.к. у нее нет предшествующей реплики. На рисунке 3 представлено распределение количества реплик в зависимости от глубины контекста диалога. Дополнительное исследование показало, что максимальный из диалогов имеет 84 предшествующих пары вопрос-ответ.

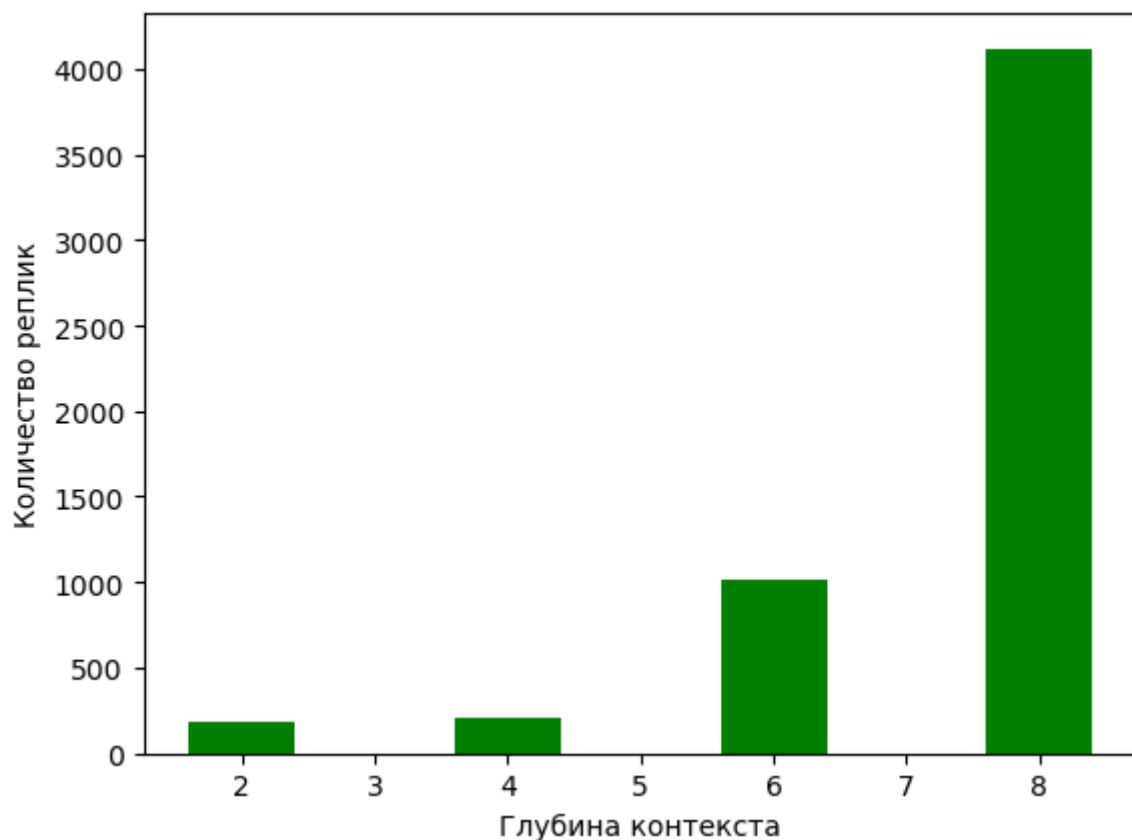


Рисунок 3 – Количество реплик от глубины контекста диалогов в итоговом датасете

### Архитектура и обучение модели

В качестве модели для дообучения я выбрал Phi-2 от Microsoft. Выбор обусловлен возможностью дообучения модели на бесплатных мощностях google colab. Архитектура обучения модели представлена на рисунке 4. При этом дообучение проводилось с использованием Peft, дообучение проводилось только для последних слоев модели.

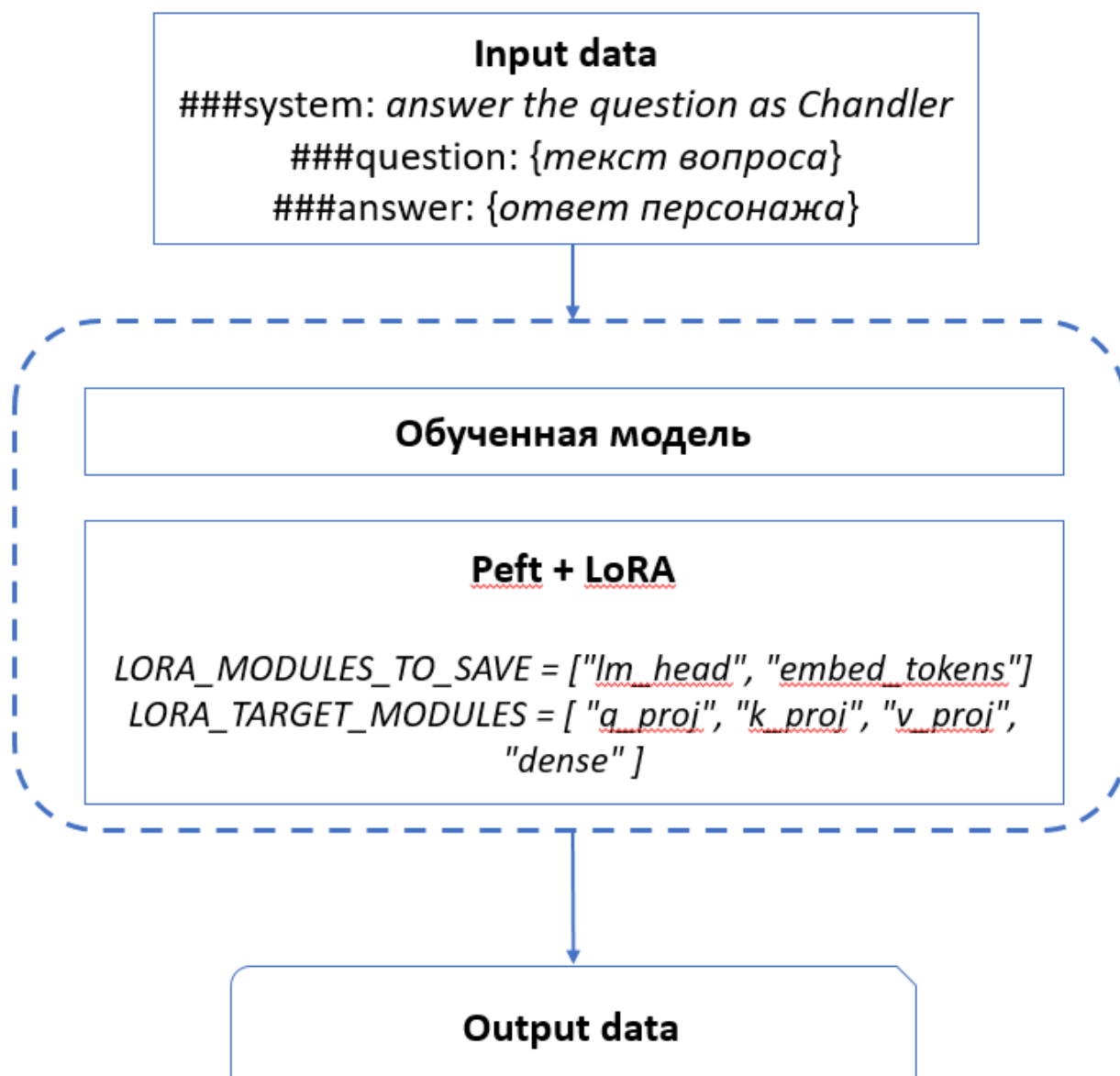


Рисунок 4 – Архитектура обучения модели

При подготовке датасета для обучения с помощью тегов `###question` и `###answer` отделялись реплики, соответствующие вопросам и ответам в контексте.

Модель обучалась одну эпоху, обучение заняло 1.5 часа. Это связано с ограничениями по времени предоставления бесплатных мощностей в google colab. Более длительное обучение скорее всего позволило бы улучшить качество ответов. Гиперпараметры для обучения подбирались исходя из нескольких критериев:

1. доступная в google colab память
2. желание сохранить адекватный размер батча
3. стремление уменьшить галюцинирование результатов.

Также, для уменьшения галюцинирования результата была использована регуляризация путем сокращения весов `weight_decay`. Дополнительно, в целях уменьшения галюцинирования, была введена подсказка-инструкция с тегом `###system`. Нарастить

исходные данные и расширить смысловой набор данных для модели не представляется возможным т.к. модель уже обучается на всех репликах и диалогах выбранного персонажа.

На рисунке 5 представлен график обучения модели для одной из попыток. Полный набор графиков обучения представлен в проекте по ссылке: [https://wandb.ai/yurezsml\\_mipt/NLPGener\\_HW2](https://wandb.ai/yurezsml_mipt/NLPGener_HW2). Коллаб с обучением модели находится по ссылке: [https://colab.research.google.com/drive/1bIn2r\\_c95wnVgJnyrt0x5CwezHYp6cs2](https://colab.research.google.com/drive/1bIn2r_c95wnVgJnyrt0x5CwezHYp6cs2).

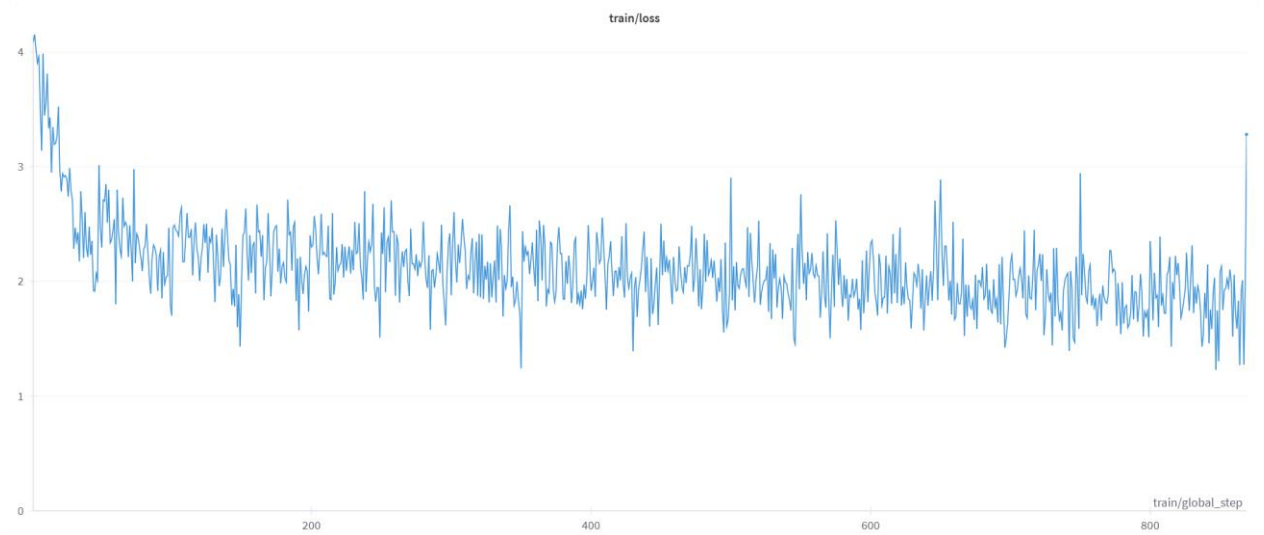


Рисунок 5 – График обучения модели

### Инференс и валидация модели

Инференс модели реализован в виде асинхронного вебсервиса на базе gradio и позволяет отображать диалог в виде реального чата. Общая архитектура чат-бота представлена на рисунке 6.

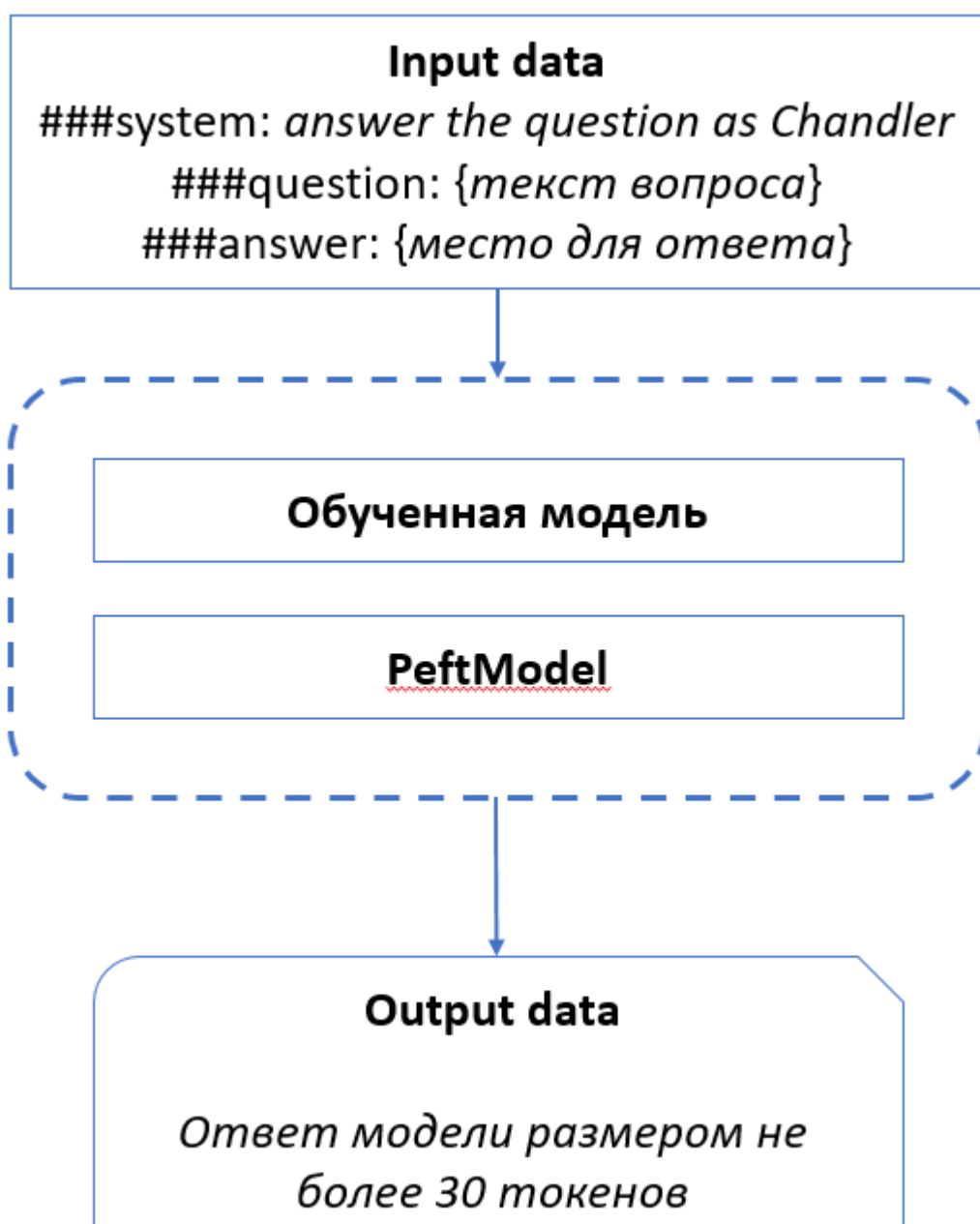


Рисунок 6 – Архитектура чат-бота

Для валидации результатов выбраны следующие тесты:

1. Простое приветствие ('hello, how are you?')

Проверка ответа бота на простой вопрос и оценка времени работы медленного алгоритма. Результат представлен на рисунке 8. Результат выглядит как продолжение какого-то другого диалога.

#### Асинхронный сервис для чат-бота по сериалу Друзья

Question

hello, how are you?

Clear Submit

Chat

you: hello, how are you?  
Chandler: i'm fine.

Flag

Рисунок 8 – Результат работы чат-бота на тестовой задаче № 1

#### 2. Диалог из нескольких реплик

Проверка работы бота на имитации реального диалога. В качестве реплик старался выбирать те, что спровоцируют отвечать в стиле выбранного персонажа. Результат представлен на рисунке 9. Диалог получился связным и соответствующим персонажу т.к. большую часть сериала они и правда зависали с друзьями в кофейне.

#### Асинхронный сервис для чат-бота по сериалу Друзья

Question

nice, where are you?

Clear Submit

Chat

you: hello  
Chandler: hey, it's me. i'm sorry i missed your call.  
you: what are you doing?  
Chandler: oh, just hanging out with some friends.  
you: nice, where are you?  
Chandler: at a coffee shop. why?

Flag

Рисунок 9 – Результат работы чат-бота на тестовой задаче № 2

### Выводы

Разработан чат-бот с помощью дообучения модели Phi-2. Проведена работа по улучшению фактологической связности ответов. Реализован асинхронный вебинтерфейс на базе gradio. Работоспособность бота продемонстрирована на тестовых примерах. Качество работы оценивалось исходя из адекватности ответов заданному вопросу.

### Инструкция

Запуск чат-бота осуществляется посредством запуска блокнота [https://colab.research.google.com/drive/1IaFwWL12LMLIdNgq9u1QCkxL87\\_a9GAx](https://colab.research.google.com/drive/1IaFwWL12LMLIdNgq9u1QCkxL87_a9GAx) на локальной машине или в google colab. После запуска в последней ячейке появится интерфейс на базе gradio. В левом окне вводятся вопросы, в правом появляется диалог в режиме чата с учетом контекста предыдущих ответов. Чтобы начать диалог заново, забыв контекст, нужно перезапустить ячейку с интерфейсом gradio.