

ВАЛЕНТИН ЮЛЬЕВИЧ АРЬКОВ

Цифровые технологии и искусственный интеллект

УЧЕБНОЕ ПОСОБИЕ



Валентин Юльевич Арьков

Цифровые технологии и искусственный интеллект

Учебное пособие

Шрифты предоставлены компанией «ПараТайп»

© Валентин Юльевич Арьков, 2024

Если к любому названию добавить слово «цифровые», получается что-то новое, заманчивое. Обещающее хорошую зарплату. А если к цифровизации добавить искусственный интеллект, то зарплата должна подняться до неприличных высот. Что скрывают от нас цифровизаторы и оптимизаторы? Как выглядит под капотом тот самый искусственный интеллект, которому почему-то «нет оснований не доверять»? Разбираемся и удивляемся.



Создано в интеллектуальной издательской системе Ridero

ОГЛАВЛЕНИЕ

[Цифровые технологии и искусственный интеллект](#)

[Введение](#)

[1. Конструирование запросов](#)

[Поисковый запрос](#)

[Оформляем отчёт](#)

[Запрос к чат боту.](#)

[Программируем без программирования](#)

[Опыты с запросами](#)

[Ссылки](#)

ВВЕДЕНИЕ

В рамках лабораторных работ мы будем знакомиться с возможностями интеллектуальных систем и сразу же будем рассматривать практические примеры применения этих возможностей. Область искусственного интеллекта охватывает самые разные методы и инструменты. На сегодняшний день чаще всего говорят о нейросетях. Вот эту тему мы и будем обсуждать на протяжении нескольких занятий.

Материалы по нашему курсу будут постепенно появляться на страничках проекта на GitHub и GitVerse.

1. КОНСТРУИРОВАНИЕ ЗАПРОСОВ

В этой работе мы рассматриваем основные правила построения запросов к нейронной сети. Контент, который будет сгенерирован в ответ на наш запрос, зависит от того, насколько подробно и тщательно мы сформулировали наше задание. Один из самых популярных интеллектуальных инструментов на сегодняшний день — это большие языковые модели — Large Language Models (LLM). Они действительно большие, в буквальном смысле слова, потому что внутри содержат миллиарды параметров (коэффициентов). При обучении таких моделей подбирают значение этих коэффициентов, а для этого используют огромное количество материалов, в основном, взятых из интернет.

Интеллектуальные системы, которые создают новые объекты (текст, изображение, звук, видео и так далее), в настоящее время обычно называют генеративным искусственным интеллектом (Generative Artificial Intelligence, Gen-AI). Технология составления заданий для таких систем называется конструирование запросов (Prompt Engineering).

Интеллектуальные системы — это не какая-то очередная заумная, абстрактная теория. Это практичные, работающие инструменты, причём иногда даже неплохо работающие инструменты. Так что в процессе знакомства с искусственным интеллектом мы будем не просто искать информацию в интернете, но и сразу же применять полученные знания.

Если где-то в интернете, на каком-то сайте что-то написано, наша задача будет проверить и убедиться в этом лично. Одно дело, когда нам советуют: «Запрос к нейросети надо писать вот так...» Мы

посмотрим на эту инструкцию и сразу же проверим, насколько это правда. Насколько это полезно и удобно.

Всё, что пишут в интернете, пишут люди. А в последнее время даже и не люди, а нейросети или боты. И нет здесь никакой ответственности — за правильность и за результаты.

Специалисты по информационной безопасности предупреждают и объясняют... Проблема в том, что интернет — это не то же самое, что водопровод. Если открыть водопроводный кран, из него потечет вода. И во многих наших городах её даже можно пить — прямо из крана. Есть города, где сначала надо бы её прокипятить. Так вот, из интернет «потечёт» всё, что угодно. Может потечь питьевая вода, а может потечь какая-нибудь отравка. И когда мы выходим в интернет, мы сталкиваемся с тем, что не всем и не всему можно верить. Наша задача научиться отличать — где правда и где неправда. И сразу же проверять. Как говорится, доверяй, но проверяй.

Кстати говоря, поскольку большие языковые модели обучают на материалах из интернета, они осваивают всё — в том числе вредное, неправильное и нежелательное. Это примерно как учить ребёнка читать по надписям на заборе. После такого «обучения» он будет удивлять своих родителей и радовать сверстников. Поэтому при обучении нейросетей есть дополнительный этап, на котором сгенерированные ответы проверяют специально подготовленные люди. Это нужно, чтобы обучить нейросеть «хорошим манерам».

В рамках этого занятия нас будет интересовать такой вопрос: как писать запросы к нейросети? Запросы по-английски называются промты или промпты — prompt.

ПОИСКОВЫЙ ЗАПРОС

Первое действие очень простое. Открываем поисковую машину, например, Яндекс. А потом Google. И задаем ей такой вопрос: «Как писать запросы к нейросети?» Естественно, на экране появится

множество ссылок. Мы просматриваем хотя бы первую страницу и пытаемся перейти по некоторым ссылкам, которые нам приглянулись. Наша цель — вначале понять основную идею, посмотреть на основные моменты — что нам советуют. Мы должны просмотреть результаты поиска и извлечь для себя основные, ключевые моменты. В интернете мы находим много всякого текста, но наша задача извлечь из этого что-то полезное.

Затем мы возвращаемся к началу. А зачем мы это спрашивали? Какая была конечная цель? И тогда можно будет наш запрос постепенно уточнять.

Задание. Используя поисковые машины Яндекс и Google, сформируйте список основных правил составления запросов к нейросети. Это будет начало нашего отчёта. Отчёт у нас будет в электронном виде.

ОФОРМЛЯЕМ ОТЧЁТ

Попутно мы с вами проходим весь цикл работы. От начала до оформления результата.

По каждой работе мы составляем неформальный отчёт. В этом отчёте мы описываем то, что мы сделали и что нам удалось узнать.

Результаты будем оформлять тоже в облаке. Как вы знаете, есть облачные офисные продукты. Есть документы Google. Есть аналогичные отечественные сервисы, например, документы Яндекс — в них и будем работать. Опять же, это не самый лучший и не самый худший вариант. По крайней мере, отечественный. И есть надежда, что он не заблокируется и не исчезнет в ближайшее время.

В отдельной вкладке открываем Яндекс. Документы. Для этого понадобится учётная запись Яндекса — такая же, как для почты Яндекса. Создаём новый документ. Интерфейс чем-то напоминает Word. Вот в этом облачном документе нам и предстоит начать писать отчет по лабораторной работе. Затем мы можем просто передать

ссылку на отчёт и не пересылать сам файл: Поделиться – Просмотр – Скопировать ссылку.

Вначале мы создаём новый документ. Как его назвать? Обычное решение: «лаба», или «отчет», или «мой отчет». Это хорошо, если такой документ у нас один-единственный. Теперь представим себе два-три таких файла с названием «Отчет» и «мой отчет»... Становится понятно, что в названии должен быть смысл. А ещё у нас могут быть разные дисциплины, а по ним могут быть разные виды занятий, так что «отчеты» могут быть разные.

Теперь подумайте, какие сведения в названии будут самые главные? То, что это отчет? Или то, что это «лаба»? И если вы этот отчет кому-то еще отправляете, как его отличить от остальных, от 20 – 30 таких же документов – тоже с названием «отчет»?

Напомним, что отчет – это такое произведение, у которого есть название и автор. Так что, на самом деле, разница будет в названии предмета и самого документа. Ещё у нас есть фамилия автора документа – конкретного студента. У нас редко бывает одинаковые фамилии в одной группе – тогда добавим инициалы. И, конечно, есть номер группы. Вот это ключевые моменты для названия документа. Теперь расположим их в порядке важности: фамилия, группа, название предмета. Название предмета можно сокращенно. Можно указать номер лабораторной работы. Получаем что-то вроде «Иванов-ЛР1-ЦТИИ-ИБТ123».

В нашем ВУЗе есть некоторые традиции – как оформляют отчеты, курсовые и прочие документы. Вначале идет особая, первая страничка. У книги бывает обложка, а у отчёта бывает титульный лист.

Итак, документ начинается с титульного листа. Есть такое понятие, титульный лист. Это первый лист документа, на котором говорится, что это за произведение такое такое, на какую тему это произведение и кто его автор. Эту традицию вы обнаружите во многих учреждениях.

На самом верху титульного листа пишется название министерства, к которому мы относимся. По какому министерству или ведомству мы с вами проходим. Попробуйте выяснить в интернете: Министерство чего? Кому подчиняется наш университет? Можно даже на сайте вуза попытаться это обнаружить. Наверное что-то, связанное с образованием. Вот эта фраза должна быть первой строчкой титульного листа.

Теперь, когда мы начинаем печатать, нужно сразу выбрать размер шрифта. Какой высоты будут наши буквы, чтобы потом это другой человек ещё и смог прочитать... Например 14 или 16 пунктов – кому как нравится. Слишком мелко не надо, слишком крупно тоже не надо.

Периодически название министерства меняется. Ваша цель проверить, правда ли, что оно теперь называется Минобрнауки. Находим сайт и убеждаемся, что это действующий сайт и что это действующее министерство. Находим, проверяем. То, что мы с вами делаем, – это мы проверяем информацию. В области нейросетей это действие называется «факт-чекинг» – fact checking. Check – это проверка. Мы проверяем факты – facts.

Вторая строчка – это название нашего учебного заведения. Как оно правильно называется? Обычно на главной веб-страничке указывают полное название вуза – и мы пишем полное, а не сокращённое.

Дальше у нас должно быть указано название кафедры, которая проводит занятия. Например, если вы проходите занятия на кафедре иностранных языков, а учитесь на кафедре АСУ, мы пишем кафедра иностранных языков. Если занятие проводит кафедра АСУ, тогда мы пишем кафедра АСУ.

И только после этого мы уже начинаем объяснять, а что это за документ. И документ этот – отчет по лабораторной работе. Не лабораторная номер один, а отчет по лабораторной работе №1. Потому что кроме отчета бывают и другие документы.

Дальше — тема нашего занятия. Слово «тема» писать не обязательно. А вот название лабораторной работы хорошо бы указать. Для нашей текущей работы темой будет «Конструирование запросов».

По-английски это «пром프트-инжиниринг» — prompt engineering.

Название предмета тоже надо куда-то вставить.

Видимо, после строчки «отчет по лабораторной работе» нужно будет сказать по какой дисциплине. Как у нас предмет называется? Длинное красивое название, мы указываем название нашего предмета. У кого-то лабораторные по физике, у кого-то по химии, а у нас по нашему предмету.

Затем мы указываем группу, фамилию и инициалы студента.

В самой нижней строчке — город и год.

Когда мы справились с оформлением титульного листа, можно переходить к работе по существу. Это внутреннее содержание отчета. Нас в этой работе что вы делали и что вам удалось выяснить. То есть какой запрос вы отправили и куда. Какие ключевые, полезные моменты вы для себя вынесли.

Действие первое было — поисковая машина Яндекс. Как вы сформулировали запрос к поисковой машине? Какие полезные советы вы для себя определили? Здесь полностью переписывать не нужно. Здесь ценность в том, что вы через себя это пропускаете. Какие полезные правила составления запросов вы для себя обнаружили? Нам нужно свои мысли отразить.

На всякий случай поясним. Зачем мне нужно эти бумажки писать, эти документы и отчёты составлять? В любой работе вам придется составлять какие-нибудь документы, какие-нибудь тексты. Пусть даже заявление на отпуск или о приеме на работу. Или резюме. А в рамках нашей профессии — информатика и вычислительная техника — приходится разные документы составлять.

Здесь художественный талант не требуется. Здесь нужно очень коротко написать, чего вы сделали и что вы получили. Кому

отправили запрос, как звучал запрос и что вы для себя из этого вынесли.

ЗАПРОС К ЧАТ БОТУ

Второе действие. Мы можем спросить и у самой нейросети то же самое, что мы пишем в поисковой машине.

Открываем на новой вкладке Яндекс GPT — просто в качестве примера. Мы не говорим, что этот сервис лучше всех или хуже всех. Это просто очередной инструмент. Причём отечественный. На главной странице сервиса читаем описание и смотрим просветительские ролики. Просвещаемся. И вот мы задаём ему вопрос — точно такую же фразу. Фиксируем в отчёте результаты.

Берём следующий инструмент: GigaChat от Сбера. Тоже просвещаемся по материалу главной страницы. Задаем ему тот же самый запрос: «Как писать запросы к нейросети?» Можно работать без регистрации. Можно войти с авторизацией для сохранения истории запросов. Чтобы войти в гигачат, потребуется учетная запись или SberID по номеру телефона. В отчете тоже нужно отразить, что нового узнали.

По каждому полезному совету подумайте и приведите примеры. Например, «контекст». В каком контексте нам нужен ответ? Придумайте пример, как объяснить это другому человеку. Это может быть описание проблемы, которую мы решаем: мы хотим научиться писать качественные запросы и для этого выясняем, какие есть советы и рекомендации, какие есть элементы/составные части в таких запросах и т. д.

По стилю и форме ответа тоже можно сформулировать требование: «Объясни простым языком, как для пятилетнего ребенка» — и она объяснит максимально простым языком. Одно дело — скопировать, скачать страничку текста и совсем другое дело — объяснить своими словами.

Можно встретить совет избегать отрицаний, отказаться от слов «не», «без», «кроме» и им подобных. Так что можно сказать: «Мужчина без волос». Или можно сказать: «Лысый». Как сказать другими словами, что у человека нет бороды? Каким-то позитивным утверждением выразить ту же самую идею.

Роль, поведение. Как звучит пример назначения роли? «Ты опытный программист» или «Действуй как редактор».

Итак, вот у нас уже есть контекст, у нас есть избегание отрицания, у нас есть роль. Мы на эти моменты посмотрели.

Теперь вернемся к нашему запросу: «Как писать запрос к нейросети». Уточним наш запрос. Он должен быть более конкретным. Сегодня мы в основном будем работать с вами с текстом, в ответах всплывают полезные советы про картинки. Поэтому в следующих запросах мы уже будем говорить по поводу работы с текстом, а не про запросы вообще. С учетом того, что мы узнали, попробуйте сформулировать более подробный, более детальный запрос.

Переходим к следующему инструменту, который мне очень даже нравится. Называется Perplexity. Вводим наш запрос и изучаем ответ. В процессе работы мы выясняем, что в технологии написания запросов есть расплывчатые полезные советы, а есть конкретный список элементов, из которых этот запрос будет состоять, если мы хотим генерировать текст. Какие элементы должны быть в запросе к нейросети? Сделайте более подробный запрос, чтобы получить более осмысленный список. Мы можем очень чётко сформулировать, что именно мы хотим. Из каких элементов должен состоять запрос? Или: какие полезные советы могут пригодиться при составлении запроса? Если мы говорим: «Составь список полезных советов», — мы получим именно список советов.

Мы должны объяснить нейросети, чего мы хотим. Для чего мы это хотим. Нас интересует генерация текста с помощью чат-ботов. В некоторых случаях мы получаем текст программы. А мы хотели

не текст программы, мы хотели текст запроса. Как писать запрос к нейросети, если мы генерируем текст с помощью чат-бота.

Так что указываем, что именно мы хотим из чат-бота вытащить. Это слишком общие слова, слишком расплывчатая формулировка: «как писать запрос». Конкретное задание звучит так: «Составь список их двадцати полезных советов» или «топ 10 сайтов...»

Далее появляется ещё один элемент запроса: формат вывода, формат ответа. Мы можем отдельно попросить нейросеть объяснить этот конкретный пункт нашего списка, привести примеры того, как он звучит в запросах. И мы можем даже использовать пример формата вывода в тексте нашего запроса.

И вот мы подбираемся к ещё одной формулировке запроса. В промпте должны быть роль, контекст и формат. Теперь мы можем напечатать этот список и сказать: «Дополни этот список. Объясни, какие еще элементы должны быть в запросе к нейросети». Таким образом, мы постепенно выясняем для себя что-то и дополняем наш список и уточняем наш запрос.

Далее, доступ к Гигачату можно получить не только через браузер, но и через Телеграм. А этот мессенджер доступен и как мобильное приложение, и как десктопное приложение, и как сервис в браузере. Проверьте, поэкспериментируйте с этим сервисом. Каждый раз, после каждого эксперимента не забывайте отразить результаты в отчёте.

Итак, в конечном счете нас интересует список из 5, или 10, или 20 пунктов, которые действительно должны быть в нашем запросе к нейросети. И вот с этими вещами мы дальше будем экспериментировать.

Есть ещё один полезный совет, который удалось вытащить из нейросети. Задание должно звучать как задание, как команда. Мы говорим, даём команду, что надо сделать: предложи, напиши, сформулируй, обоснуй. Команда, задание — это действие. Мы не просто говорим «список», а мы говорим «составь список». Мы не говорим «программа», мы говорим «напиши программу». Бывают

ситуации, когда мы говорим: «вот программа, исправь ошибку». Конкретная команда содержит в себе действие, и это глагол. Возможно, даже когда-то в далёком детстве, в школе что-то такое было. Теперь начинает приходить.

Следующий элемент запроса — ограничение. Мы должны указать, какой объем слов, или строк, или абзацев, или страниц хотим получить. Или: Напиши так, чтобы среднестатистический человек мог прочитать это за 5 минут. Можно даже потребовать нужное количество букв: Напиши ответ, в котором 20 букв.

Ещё один элемент запроса — это вариативность, или температура. Попробуйте уточнить у самой же нейросети: «Как нам задавать вариативность или температуру, когда мы пишем запрос к нейросети?» Как её указывать в запросе? Пускай сама нейросеть приведет пример. Мы так и говорим: Приведи 5 примеров того, как...

Что еще в запросе может быть? Оказывается, стиль. Разговорный стиль. Формальный, неформальный. Художественный. Как в такой-то книге — в качестве примера. Стиль какого-то знаменитого человека. Мы можем указать, каким языком, в каком стиле, на кого похоже.

ПРОГРАММИРУЕМ БЕЗ ПРОГРАММИРОВАНИЯ

Переходим к инструменту под названием COZE, самому необычному на сегодняшний день. Здесь потребуется зарегистрироваться. Бесплатно. Например, с учетной записью Google. Нажимаем кнопку Get started справа сверху — то есть начать работу. Здесь мы сможем создать своего собственного бота, с которым будем дальше общаться.

Нажимаем кнопку Create bot слева сверху — создать бота.

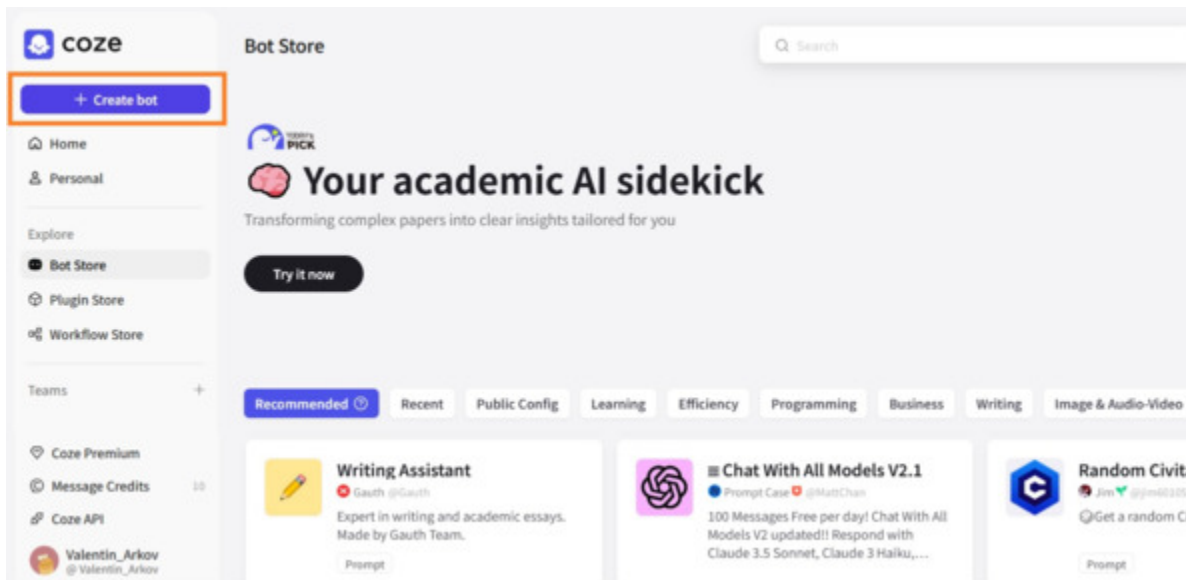


Рис. Создаём бота

Далее в диалоговом окне Create bot – Workspace – Personal. Мы придумываем ему свое собственное название Bot name. Описываем его работу Bot function description. Генерируем ему иконку – картиночку. Нажимаем кнопку Confirm – подтвердить.

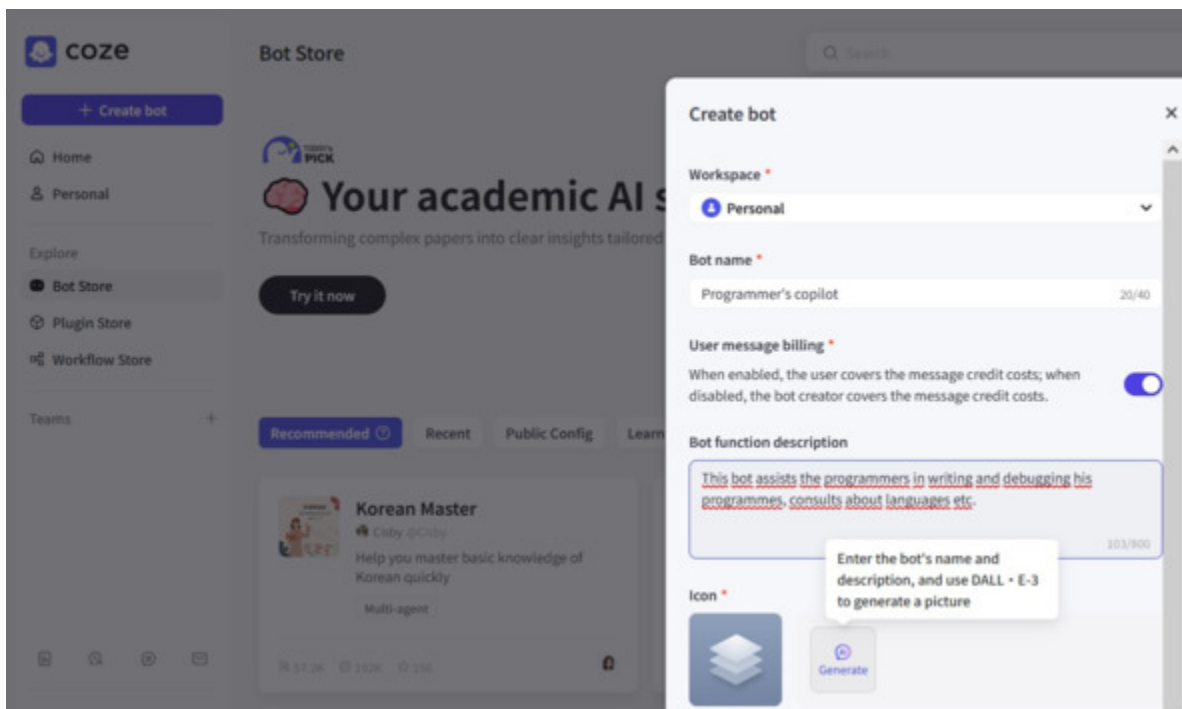


Рис. Открывающий диалог

Теперь сразу же выбираем языковую модель попроще, чтобы не потратить лимиты на запросы. На сегодняшний день этот сервис бесплатно обслуживает ограниченное число запросов. Например, для модели GPT-4o даётся только один запрос в день.

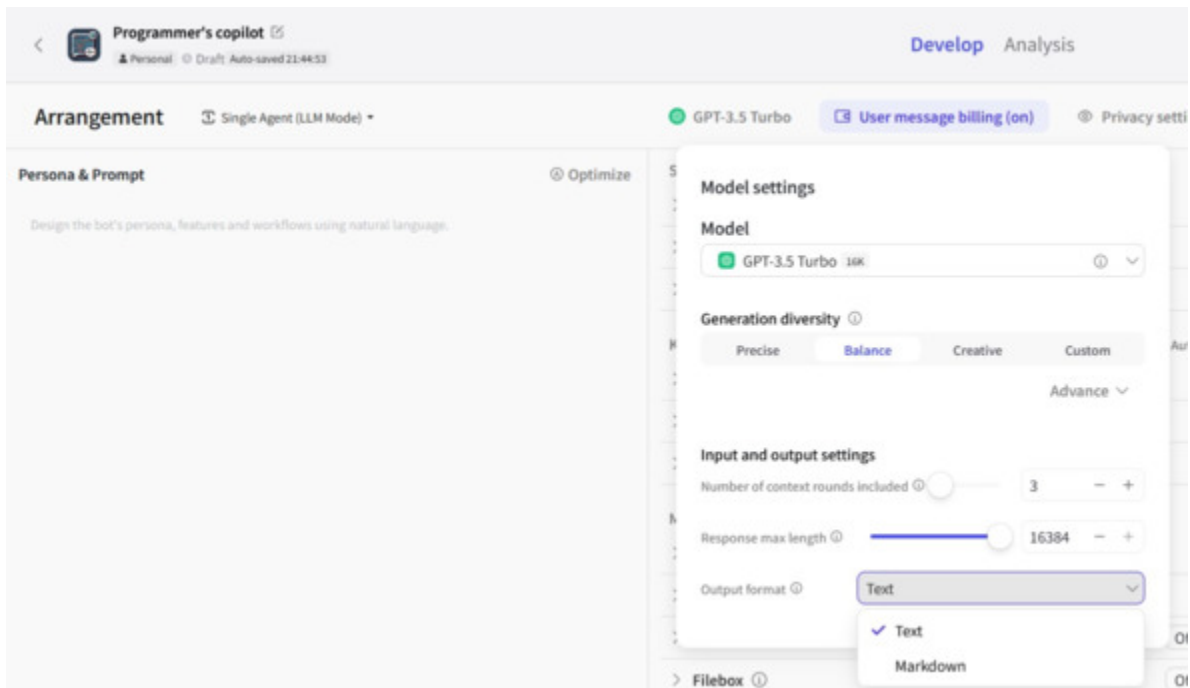


Рис. Выбор языковой модели

Далее, в разделе Skills — возможности — подключаем plugins — плагины — дополнения. Например, Google Web Search — поиск с помощью Google. Теперь у нашего бота есть выход в интернет для поиска свежей информации.

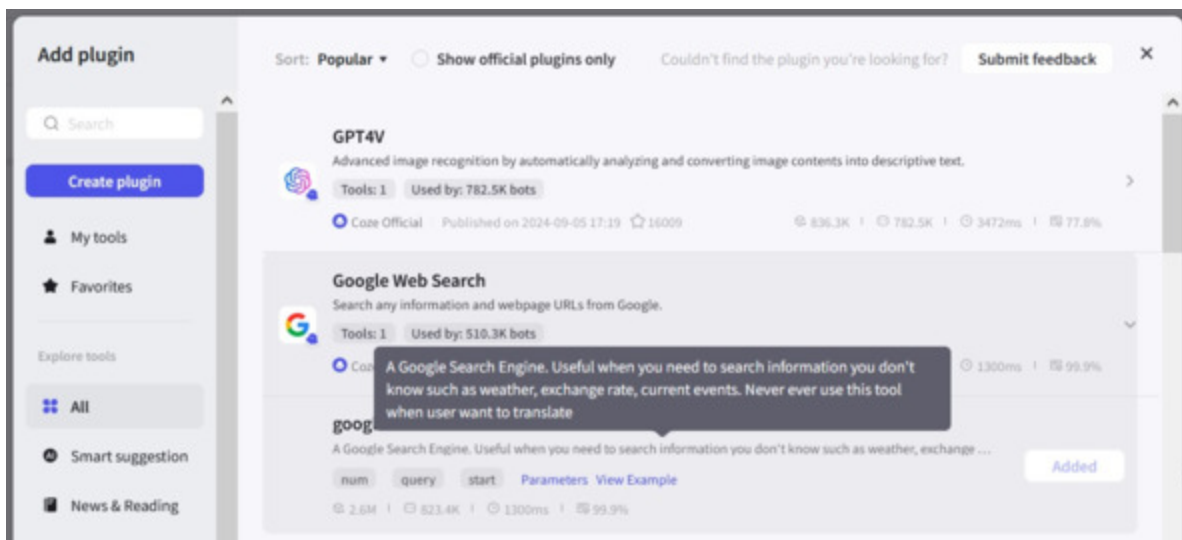


Рис. Возможности поиска

В разделе Persona & Prompt мы описываем общий запрос. И это будет System prompt. Есть запрос пользователя, а есть системный запрос. Уточните в любом чат-боте, что такое System prompt и как этим пользоваться. В двух словах, этот промпт будет всегда незаметно добавляться к запросу пользователя. И здесь мы можем описать все общие элементы наших запросов. Или можем задать любые ограничения, например: «в своих ответах говори намёками, чтобы пользователю пришлось подумать самому».

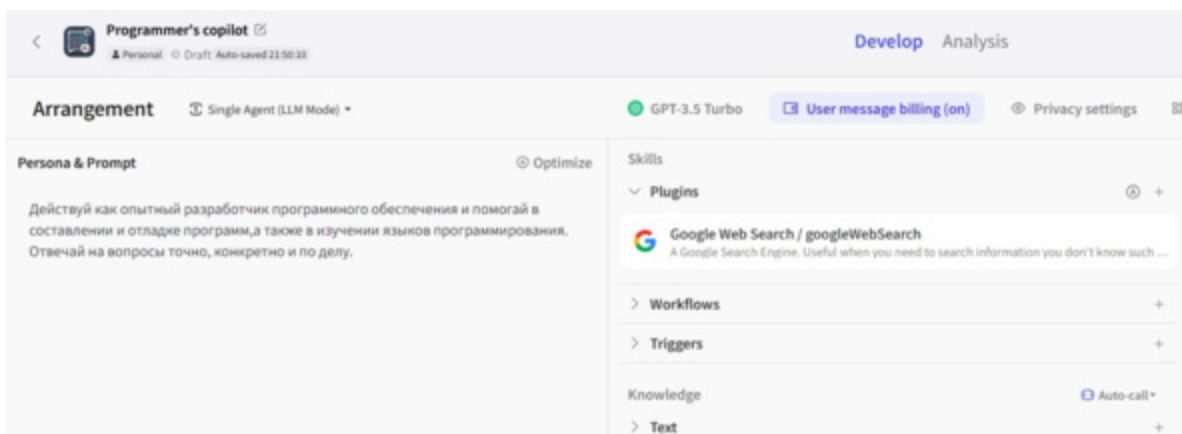


Рис. Системный промпт

В разделе Preview & Debug – предварительный просмотр и отладка – мы можем поработать с нашим ботом. Если нужно, сразу же исправляем его настройки.

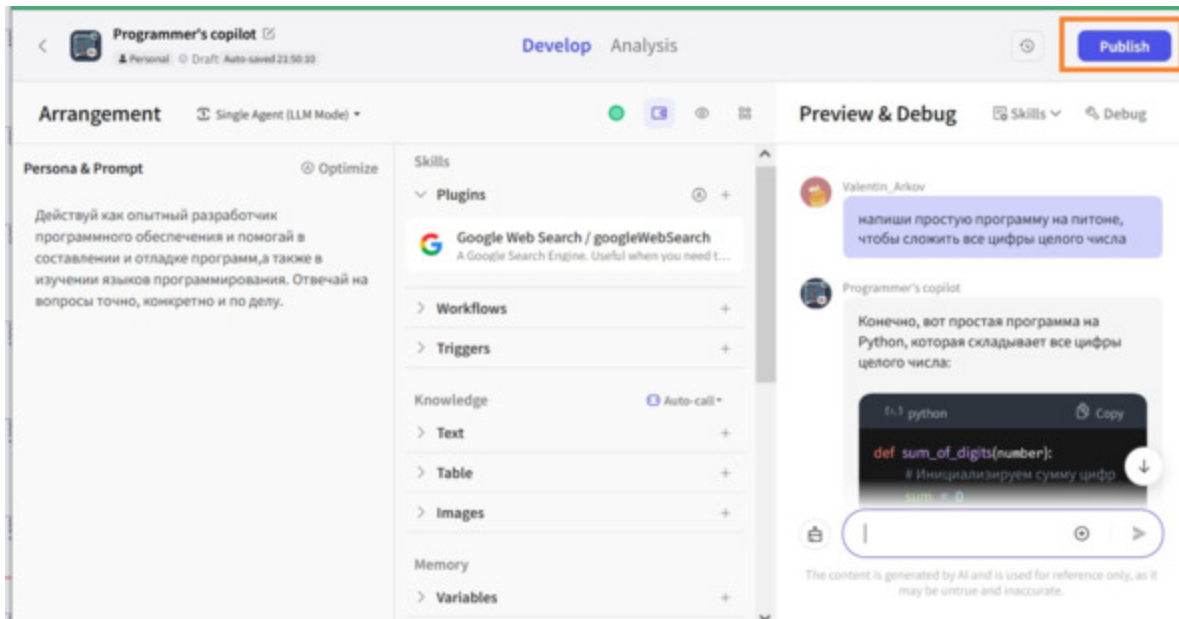


Рис. Предпросмотр

Дальше на сервисе COZE мы можем сделать бота, с которым будем общаться через Телеграм. Для этого переходим к публикации. Справа сверху нажимаем кнопку Publish. В диалоговом окне настроим Conversation opener – начало диалога и рекомендуемые вопросы (их можно удалить). Нажимаем Confirm – подтвердить.

Complete a few more steps to make your bot stand out! ✕

Add a conversation opener ⓘ

A conversation opener gives users a quick understanding of your bot's functions, increasing user conversion.

Opening text

Ⓐ ▾ B I Ⓢ | ☰ ☷ 66 | 🔗 📎 {} 👤

Привет от второго пилота! Я помогу в составлении и отладке программ.

↶ ↷

Preset opening questions ⓘ

⋮ Enter the opening question 🗑

Delete question

Skip and publish

Confirm

Рис. Настройка приветствия

В диалоге Publish to указываем платформу для публикации бота снимаем галочку Coze Bot Store — официальный магазин ботов. Видим, что в разделе Telegram у нас галочка не активна. Зато справа есть кнопочка Configure — настроить.

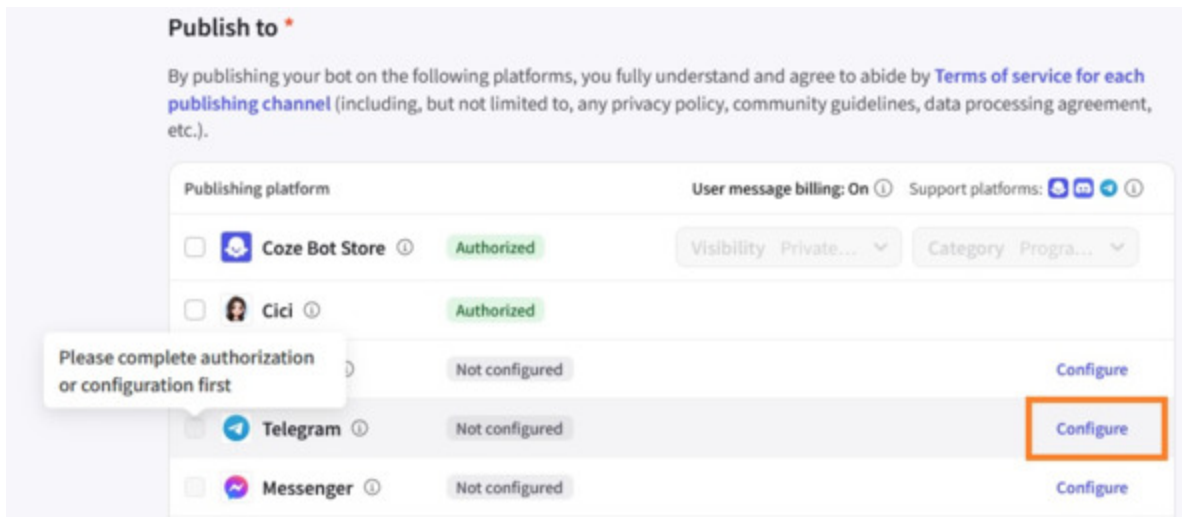


Рис. Публикуем в Телеграме

Здесь нужно ввести токен, который делается с помощью крёстного отца всех ботов – BotFather.

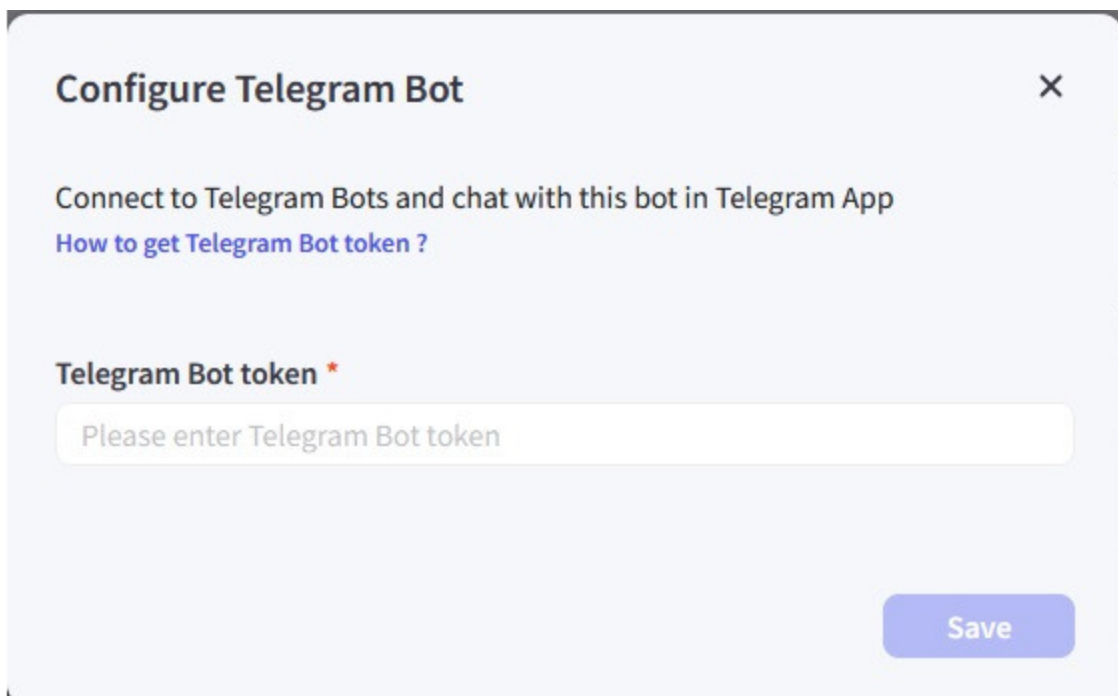


Рис. Запрос токена

В телеграмме мы находим бота под названием @BotFather. Будьте внимательны: есть много подделок с похожими названиями. Нажимаем Меню и выбираем команду /newbot. Создаём нового бота. Придумываем ему название и официальное имя для Телеграма.

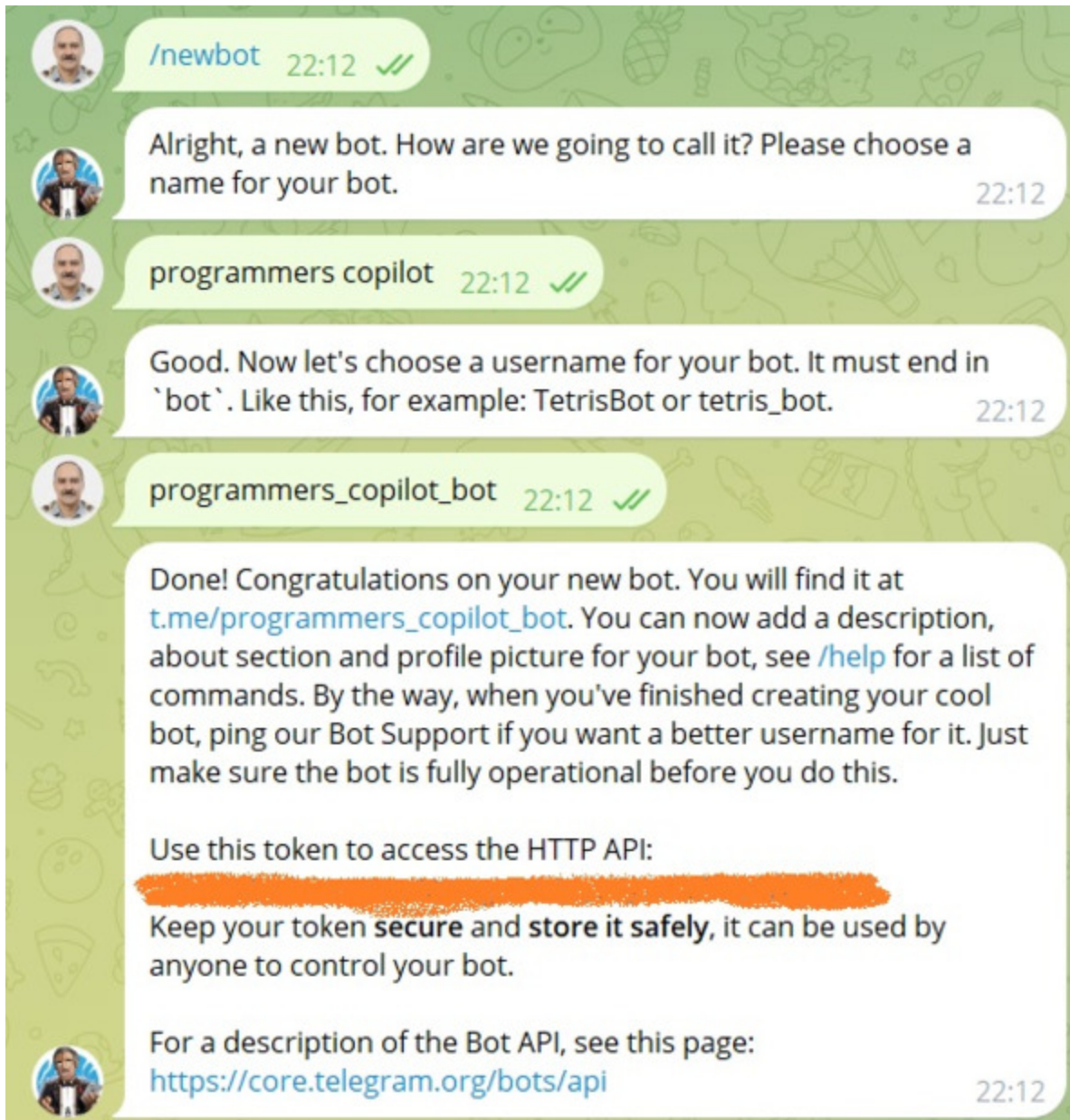


Рис. Получение токена

После создания бота нам сообщают токен. Это длинная строка символов. Копируем эту строчку в буфер обмена и вставляем в COZE. Нажимаем кнопку Save – Сохранить.

Теперь ставим галочку напротив Telegram и нажимаем кнопку Publish – Опубликовать.

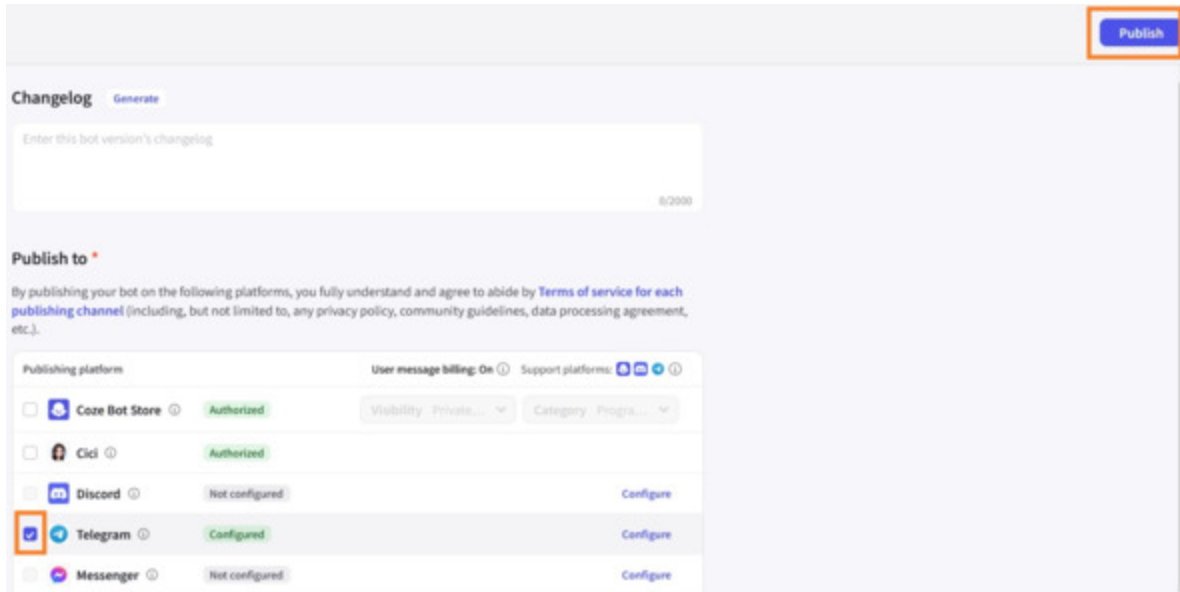


Рис. Публикация в Телеграме

Нам сообщают: Publication submitted! – Бот опубликован.

Можем перейти по ссылке Chat now – начать общаться с ботом. Либо можем просто скопировать ссылку на бота в буфер обмена.

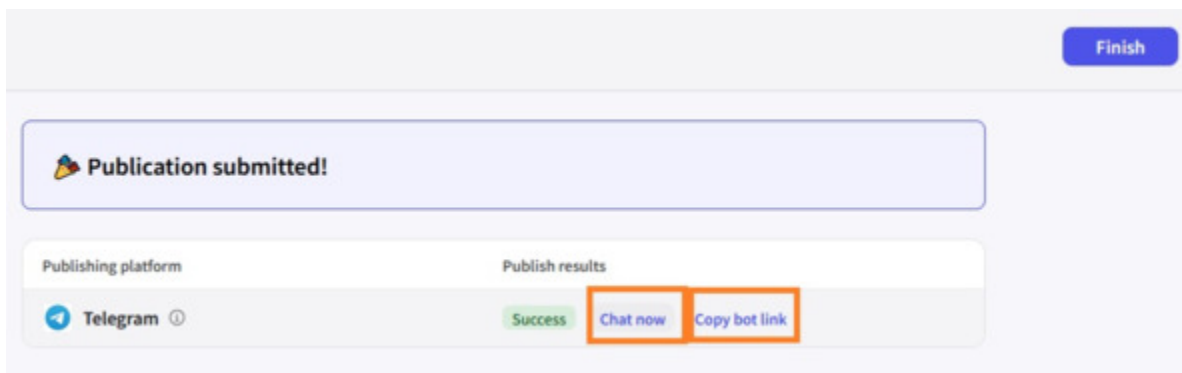


Рис. Переход к беседе

Бот поможет нам написать программу, провести анализ её вычислительной сложности и оптимизировать код — если только его об этом попросить.

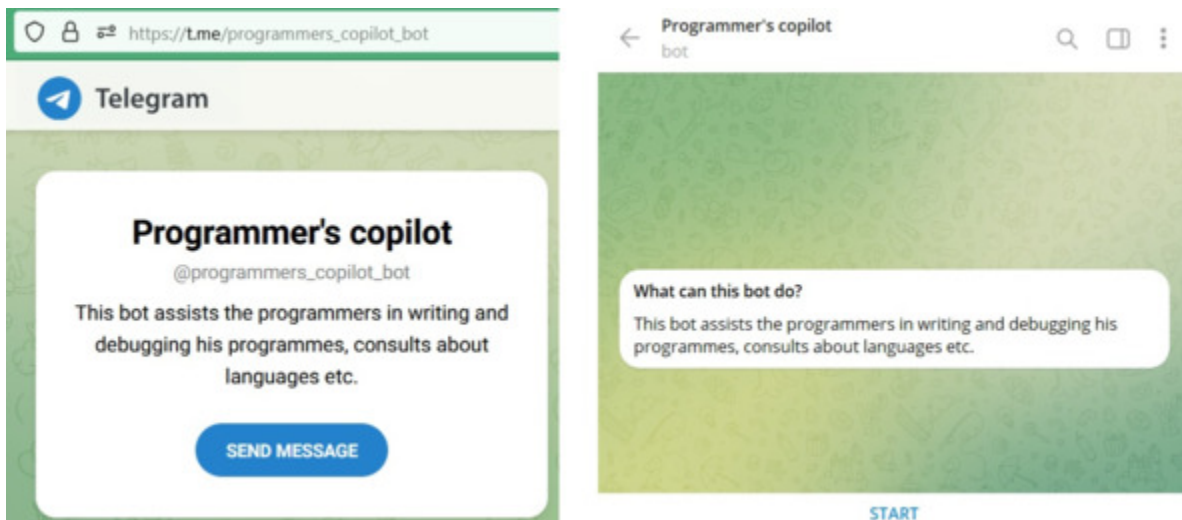


Рис. Запуск бота

Наш девиз — это классический афоризм: «Компьютер делает не то, что вы хотели, а то, что вы просили!»

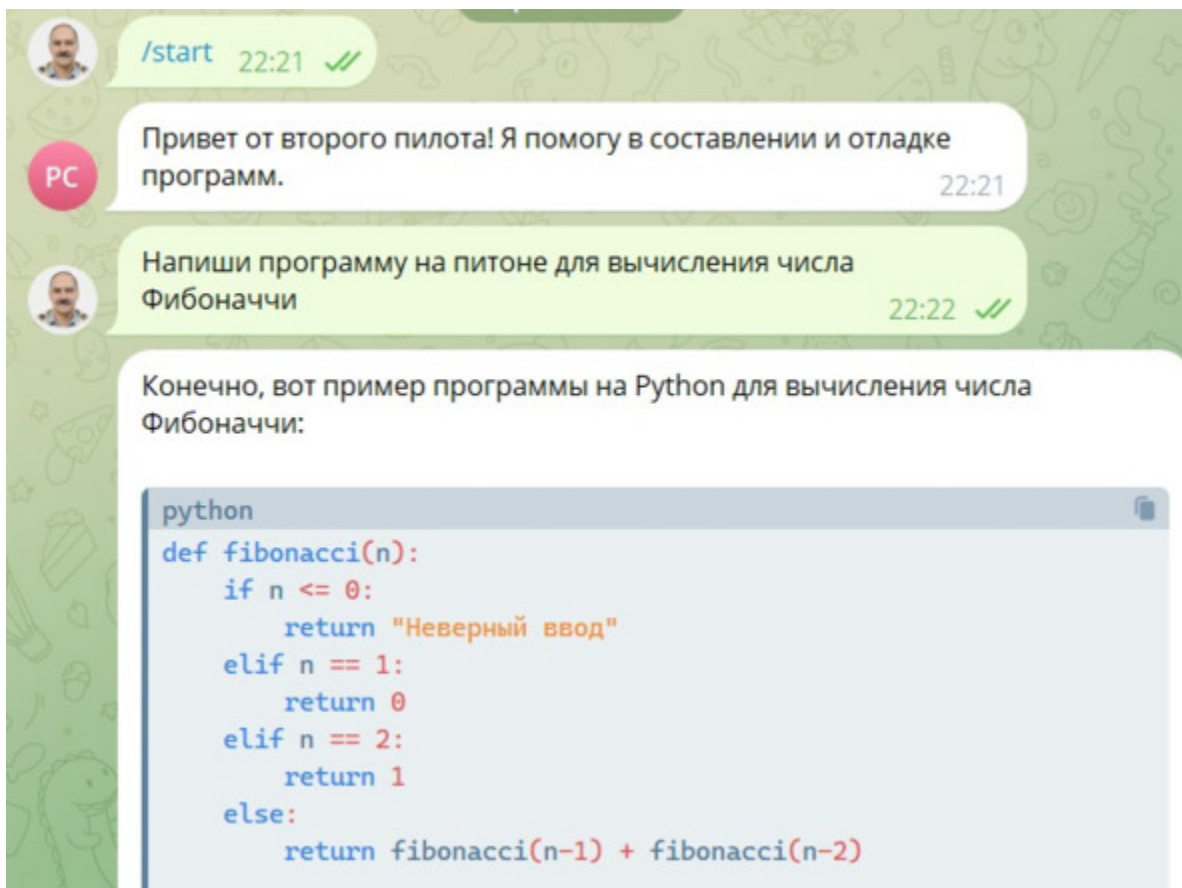


Рис. Генерируем программу

Это был пример того, как можно создавать бота без программирования. Называется такой подход No-code или Zero-code.

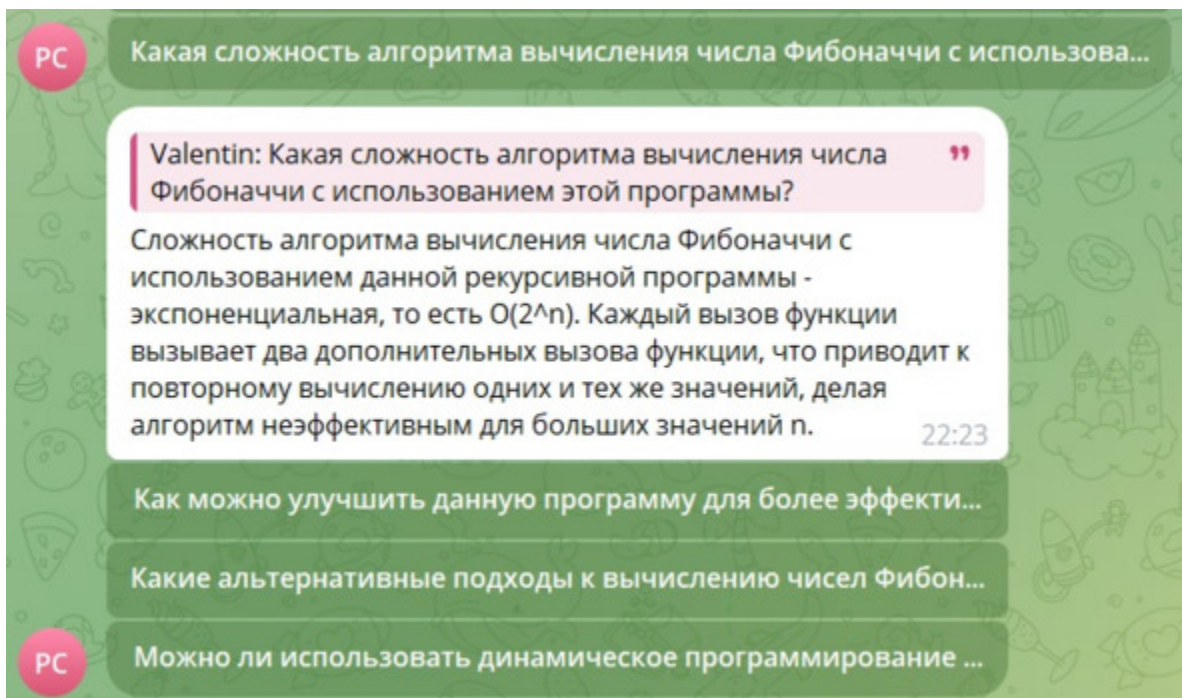


Рис. Оценка вычислительной сложности

Теперь пришло время поделиться своим ботом с соседом. Для этого мы с вами используем сервис сокращения ссылок, например, Яндекс. Кликер. Здесь адрес любой длины можно сократить до 5–6 символов. Можно также получить QR-код, чтобы другие могли отсканировать его телефоном и сразу перейти по нужному адресу.

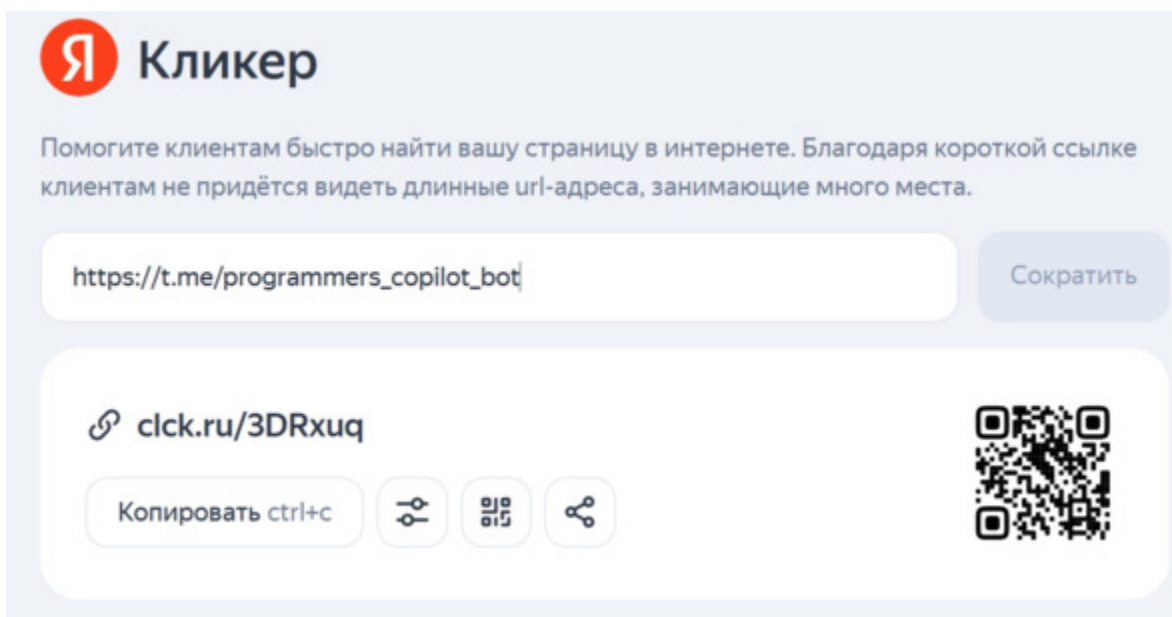


Рис. Сокращаем ссылку

ОПЫТЫ С ЗАПРОСАМИ

Следующий наш этап — посмотреть на собранные рецепты и полезные советы и потренироваться, исследовать их в работе. Действительно применить их и посмотреть, чем это кончится.

Мы испытываем назначение роли, контекста, формата, размера, тональности и так далее. Включая температуру. Сформулируйте очень подробный запрос про то, как писать запросы к нейросетям. Используйте те самые полезные советы, которые удалось собрать. При необходимости просите его о помощи.

Добейтесь, чтобы наш интеллектуальный помощник отвечал именно именно то, что мы хотим. При этом мы фиксируем в отчете всё, что происходит.

Берем самый длинный и подробный промпт и начинаем с ним работать. Вначале мы должны были назначить роль. Попробуйте назначить другую роль. Всё остальное точно так же. Меняем роль. Посмотрите, чем это кончится, как изменится ответ. Как влияет

назначение роли на ответ нейросети? А запрос по-прежнему тот же: как писать запросы к нейросети. Основная часть та же самая, просто роли разные должны быть.

Естественно, для этого опыта мы основной текст запроса держим в файле отчета. Оттуда его копируем, чтобы не приходилось каждый раз его полностью печатать. Отмечаем результаты в отчете. Какая роль, какой ответ.

Посмотрите на температуру. Как влияет температура на результаты? Запускайте запрос несколько раз. Каждый раз обнуляем – удаляем беседу Thread и создаем новую.

Отчет по лабораторной работе отправляем в виде ссылки. Форма для загрузки расположена на странице нашего курса на гитхабе. Переходим по ссылке, вводим фамилию и ссылку на отчет. Нам достаточно использовать доступ на чтение.

Поскольку это облачный документ, мы можем открыть доступ на чтение и на запись. Нас интересует только чтение.

ССЫЛКИ

Учебные материалы на GitHub

<https://github.com/Valentin-Arkov/Digital-Tech-AI>

Учебные материалы на GitVerse

<https://gitverse.ru/Valentin-Arkov/Digital-Tech-AI>

YandexGPT

<https://ya.ru/ai/gpt-3>

YandexGPT API

<https://yandex.cloud/ru/services/yandexgpt>

Документы Google

<https://docs.google.com>

Яндекс. Документы – бесплатный онлайн-редактор

<https://docs.yandex.ru>

Чат с YandexGPT

<https://console.yandex.cloud> – foundation-models – chat

Квоты и лимиты в Yandex Foundation Models

<https://yandex.cloud/ru/docs/foundation-models/concepts/limits>

YandexGPT API overview

<https://yandex.cloud/en-ru/docs/foundation-models/concepts/yandexgpt/>

GigaChat – бесплатная нейросеть без vpn на русском языке, которая общается как человек

<https://giga.chat/>

Perplexity. Where knowledge begins

<https://www.perplexity.ai/>

GigaChat. Нейросетевая модель от Сбера

https://t.me/gigachat_bot

GigaChat. Главный канал про AI на русском языке – про настоящее и будущее

https://t.me/official_gigachat

Coze. Next-Gen AI Chatbot Developing Platform

<https://www.coze.com/>

BotFather. The one bot to rule them all.

<https://telegram.me/BotFather>

Яндекс Кликер. Помогите клиентам быстро найти вашу страницу в интернете.

<https://clck.ru/>