Бенчмарки, на которых тестировались различные LLM

1. https://huggingface.co/datasets/gsm8k – grade school math – набор простых задачек на логику и арифметику. Пример:

```
Natalia sold clips to 48 of her friends in April, and
then she sold half as many clips in May. How many
clips did Natalia sell altogether in April and May?
```

2. https://huggingface.co/datasets/openai_humaneval – HumanEval – 164 задачи на программирование на Python, сформулированные на естественном языке, с тестами. Заданы в виде шаблонов функций. Пример:

```
def incr_list(l: list):
    """Return list with elements incremented by 1.
    >>> incr_list([1, 2, 3])
    [2, 3, 4]
    >>> incr_list([5, 3, 5, 2, 3, 3, 9, 0, 123])
    [6, 4, 6, 3, 4, 4, 10, 1, 124]
    """
    return [i + 1 for i in l]
```

```
def solution(lst):
    """Given a non-empty list of integers, return the sum of all of the odd elements
    that are in even positions.

    Examples
    solution([5, 8, 7, 1]) ==>12
    solution([3, 3, 3, 3, 3]) ==>9
    solution([30, 13, 24, 321]) ==>0
    """
    return sum(lst[i] for i in range(0,len(lst)) if i % 2 == 0 and lst[i] % 2 == 1)
```

   Желтое – ответ LLM.

3. https://huggingface.co/datasets/mbpp?row=20 – Mostly Basic Python Problems – датасет из задач на базовый уровень программирования на Python, на работу с модулями. Пример:

```
Write a function
to increment the
numeric values in
the given strings
by k.
```

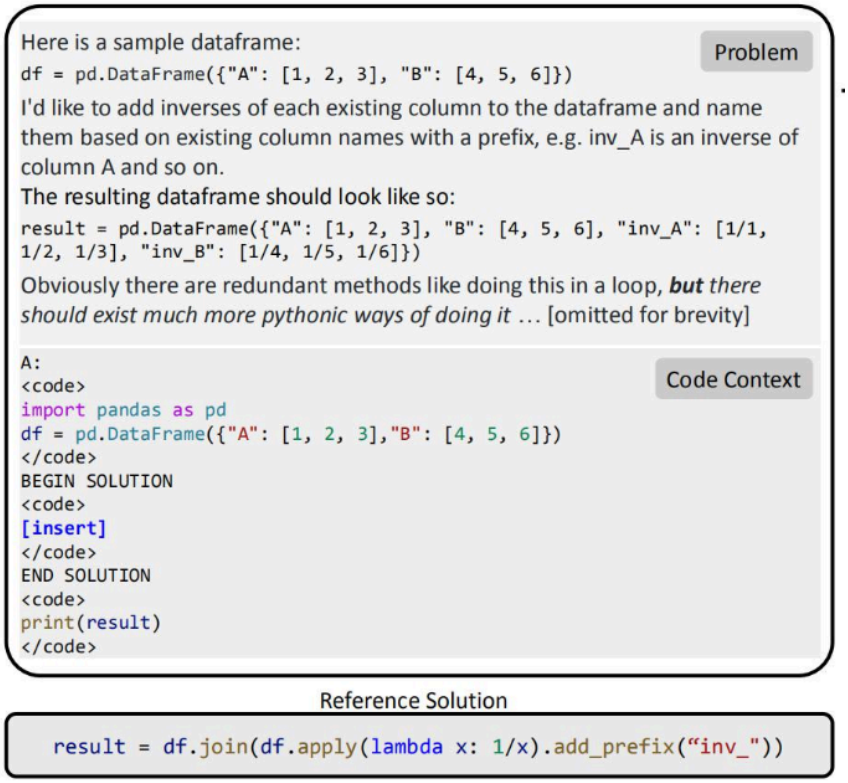
4. https://huggingface.co/datasets/nuprl/MultiPL-E – Multi-Programming Language Evaluation – датасет из задач HumanEval и MBPP, переведенных на 18 языков программирования.

5. https://huggingface.co/datasets/codeparrot/apps – 10000 задач по программированию, написанных на естественном языке. Варьируются по уровню сложности. Пример:

```
You are playing a very popular game called
Cubecraft. Initially, you have one stick and
want to craft $k$ torches. One torch can be
crafted using one stick and one coal.
Hopefully, you've met a very handsome
wandering trader who has two trade offers:
exchange $1$ stick for $x$ sticks (you lose
$1$ stick and gain $x$ sticks). exchange $y$
sticks for $1$ coal (you lose $y$ sticks and
gain $1$ coal). During one trade, you can
use only one of these two trade offers. You
can use each trade offer any number of times
you want to, in any order. Your task is to
find the minimum number of trades you need
to craft at least $k$ torches. The answer
always exists under the given constraints.
You have to answer $t$ independent test
cases. -----Input----- The first line of the
input contains one integer $t$ ($1 \le t \le
2 \cdot 10^4$) – the number of test cases.
Then $t$ test cases follow. The only line of
the test case contains three integers $x$,
$y$ and $k$ ($2 \le x \le 10^9$; $1 \le y, k
\le 10^9$) – the number of sticks you can
buy with one stick, the number of sticks
required to buy one coal and the number of
torches you need, respectively. -----Output-
---- For each test case, print the answer:
the minimum number of trades you need to
craft at least $k$ torches. The answer
always exists under the given constraints. -
----Example----- Input 5 2 1 5 42 13 24 12
11 12 1000000000 1000000000 1000000000 2
1000000000 1000000000 Output 14 33 25
2000000003 1000000001999999999
```

6. https://github.com/hendrycks/math – более сложные математические задачи, чем в gsm8k.

7. https://ds1000-code-gen.github.io/ – датасет с прикладными задачами по DataScience. На нем тестировался DeepSeek Coder. Пример:



Взаимодействие на русском языке

Пойду по списку, который присылал ранее:

1. CodeLlama. Базовая CodeLlama не предназначена для генерации кода по запросу на естественном языке, однако ее вариация instruct отлично справляется с этим и понимает русский язык. С простыми задачами хорошо сработала версия на 7 миллиардов параметров.

2. Phind-CodeLlama. Она есть только с 34 миллиардами параметров, и бесплатный colab не дал мне возможности развернуть ее. Но могу предположить, что так как она “надстраивалась” над базовой CodeLlama плюс обучалась на диалог с пользователем, то с русским языком тоже будет все хорошо.

3. WizardCoder. Так как это надстройка над StarCoder, а тот в свою очередь понимает русский (о чем написано ниже), то здесь ситуация как с Phind-CodeLlama.

4. DeepSeek-Coder. У них есть отличная api https://chat.deepseek.com/i в формате чата, где можно потестировать модель. Русский язык хорошо понимает, не ошибается (в отличие от copilot).

5. StarCoder. У них также есть простая api https://huggingface.co/HuggingFaceH4/starchat-beta, где можно убедиться, что есть поддержка русского языка.