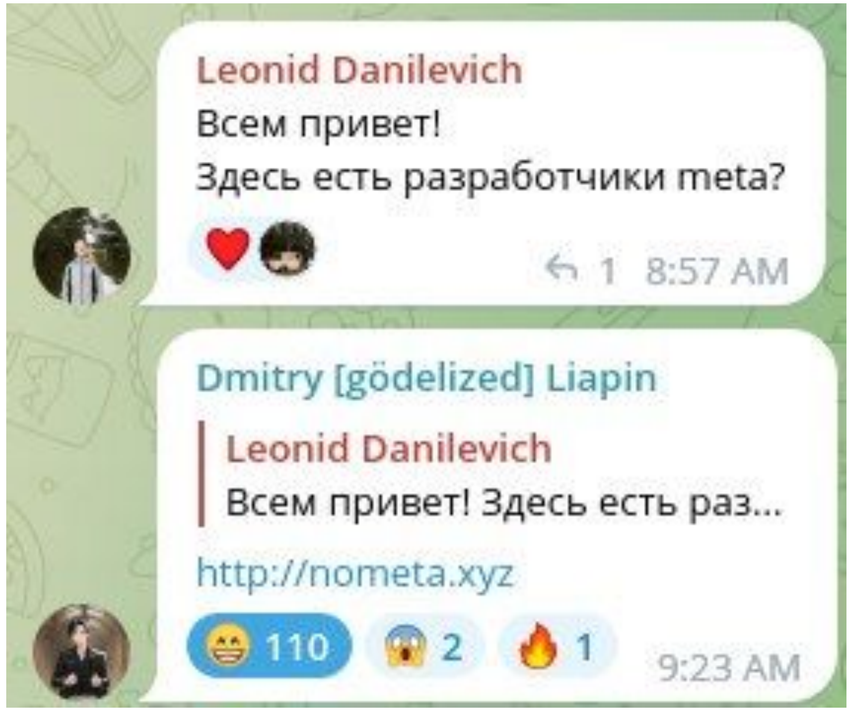
Nometa

Aleksandr Shefer, Aleksandr Kariakin, Artem Karamyshev

Nometa.xyz

- Meta-questions are questions that reflect on the process of questioning, rather than seeking specific information or a direct answer.



Related Work

- Bot that deletes messages that chat users have indicated that it is meta: https://github.com/telegram-ru/ru-bot/tree/master. However, this bot doesn’t use any ML-solutions.

- https://github.com/leshark/nometa?tab=readme-ov-file - chat analytics.

Aim of the project

Investigate how modern classification models perform on the problem of Meta-question detection.

Goals:

- Collect a Dataset (test Dataset consisting of human questions and train consisting of generated questions)
- Choose modern classification models (like DistillBERT)
- For each model estimate result after querying with no prompt or after applying zero-shot learning
- Analyze the results and model reasoning

Dataset. Train. Synthetic part

- Generated by GPT-4
  - “Meta-question is a question that implies other questions like:
    May I ask a question?
    Is there anyone experienced in N?
    ...
    <Examples in expected format>
    ”
- Examples of generated data:
  - Can someone help with JavaScript frameworks?
  - Is anyone available to provide feedback on UI/UX designs?
  - Do you have tips on how to handle real-time data processing?
- No-meta questions for the training dataset are also synthetic!

Dataset. Train. Real examples part

- How to collect real nometa.xyz questions?
- Manually - very time-consuming…

- Method: Design and execute an SQL query on a dataset of over 50 billion Telegram messages.
- Benefits: While still time-consuming, this method is automated and highly effective.
- Outcomes: 350 high-quality **real** meta-questions

Dataset. Test.

- Meta questions parsed from telegram it-chats (manually).
  - “Guys, is there anyone working at MTS?”
  - “Are there any Meta developers here?”
- Non-meta questions filtered from “Software Engineering Interview Questions” dataset.
  - “What is lazy loading in web development?”
  - “How do you optimize a SQL query for better performance?”

Implementation

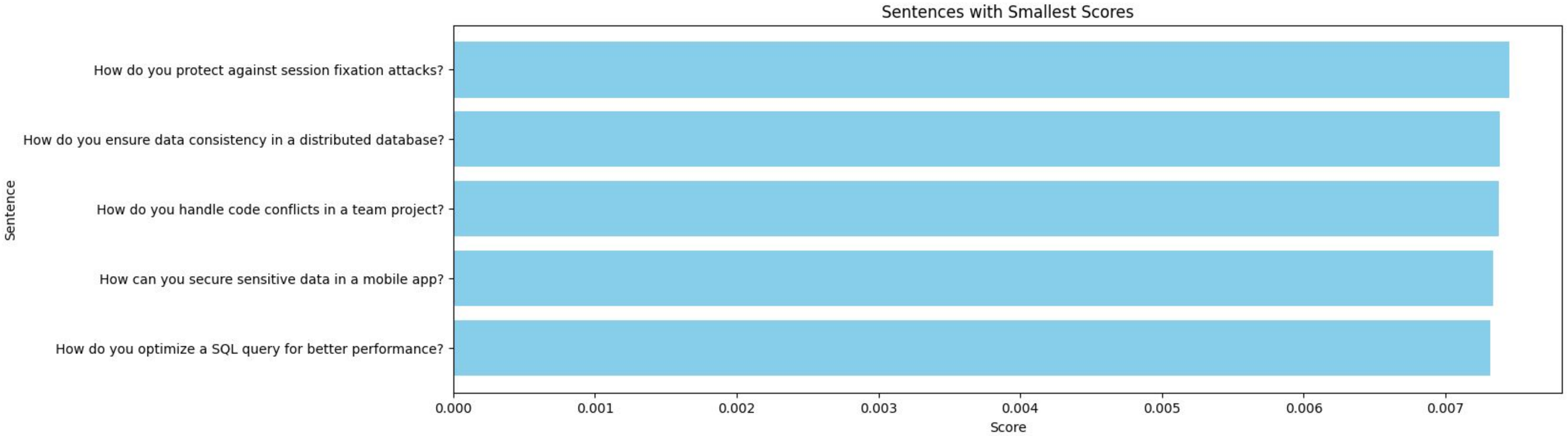
- Distil-BERT model
- Fine Tuning pipeline via Hugging Face
- Translation to English language using OpenAI API

Results

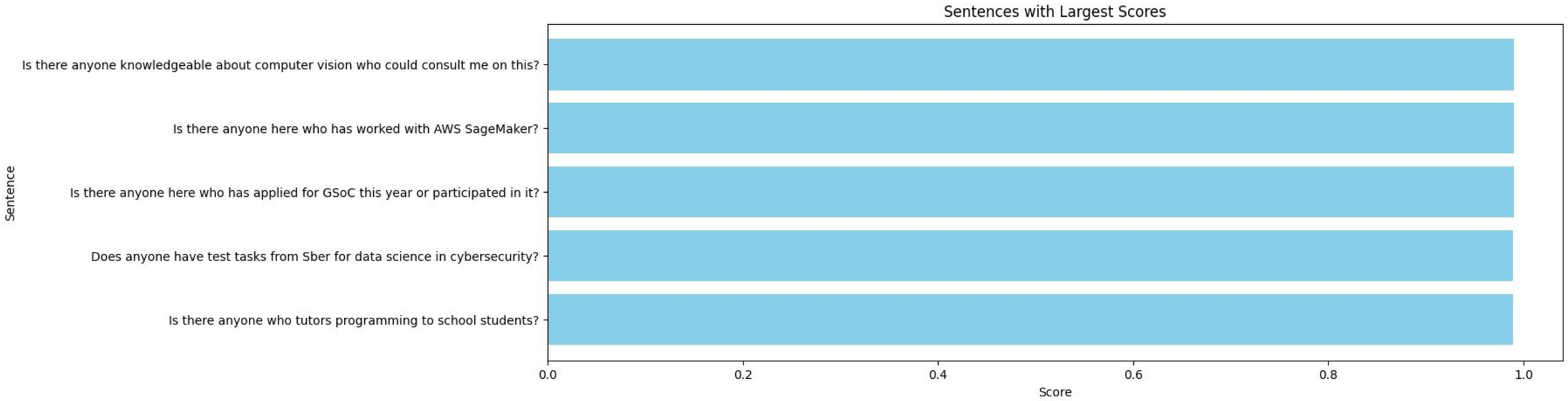
Prompt format	Accuracy	F1 Score	Train Dataset version
No prompt	0.85	0.91	Synthetic
Zero-shot	0.976	0.976	Synthetic
No prompt	0.96	0.96	Synthetic + Real
Zero-shot	**0.982**	**0.982**	**Synthetic + Real**

Distil-BERT model performance in various settings

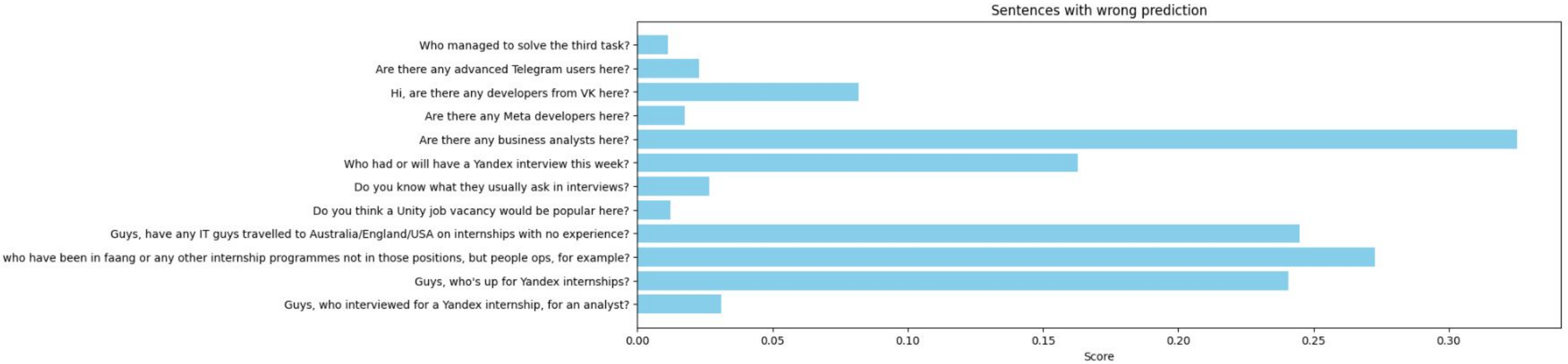
Evaluation. Smallest meta score.



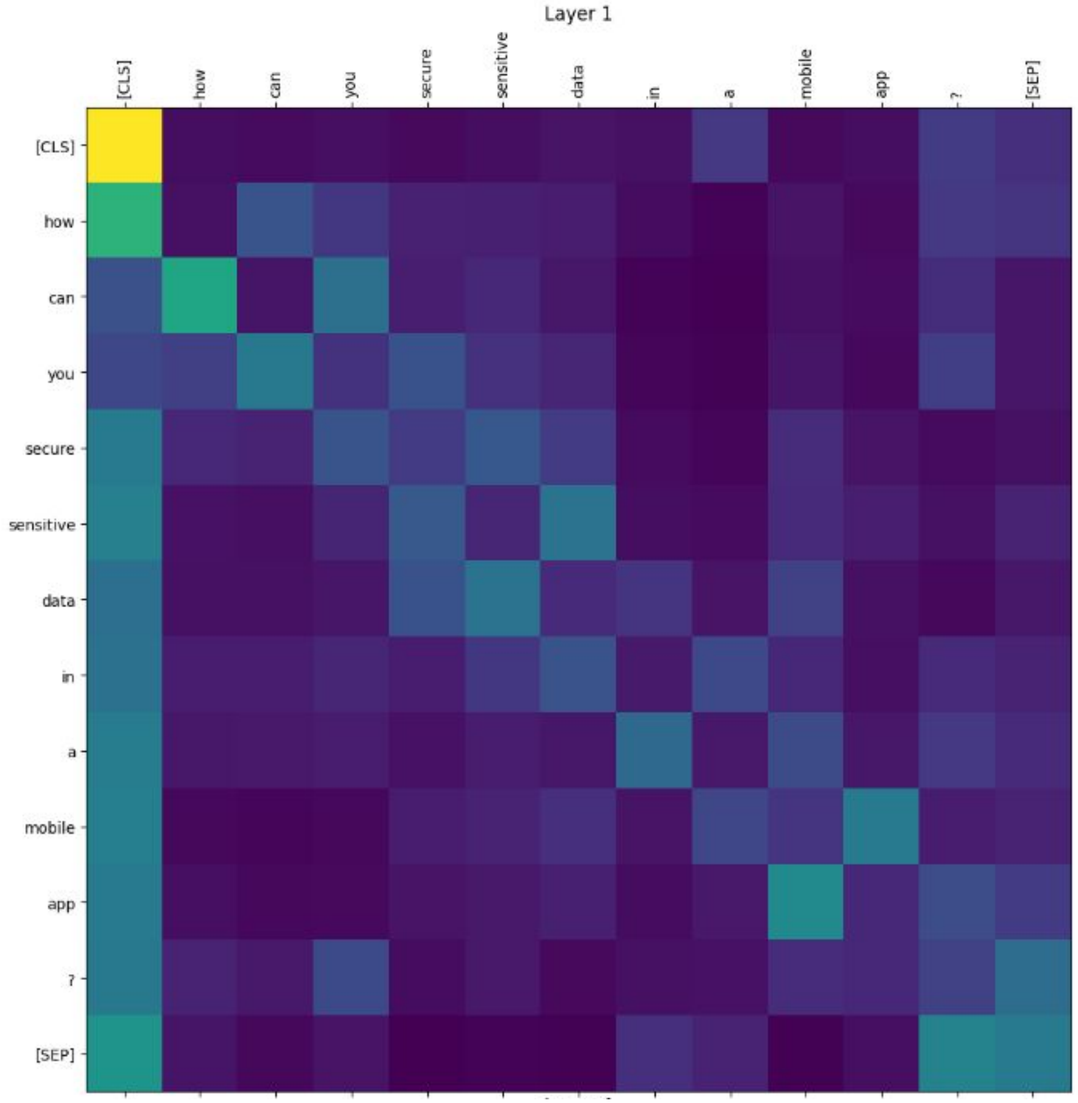
Evaluation. Largest meta score.

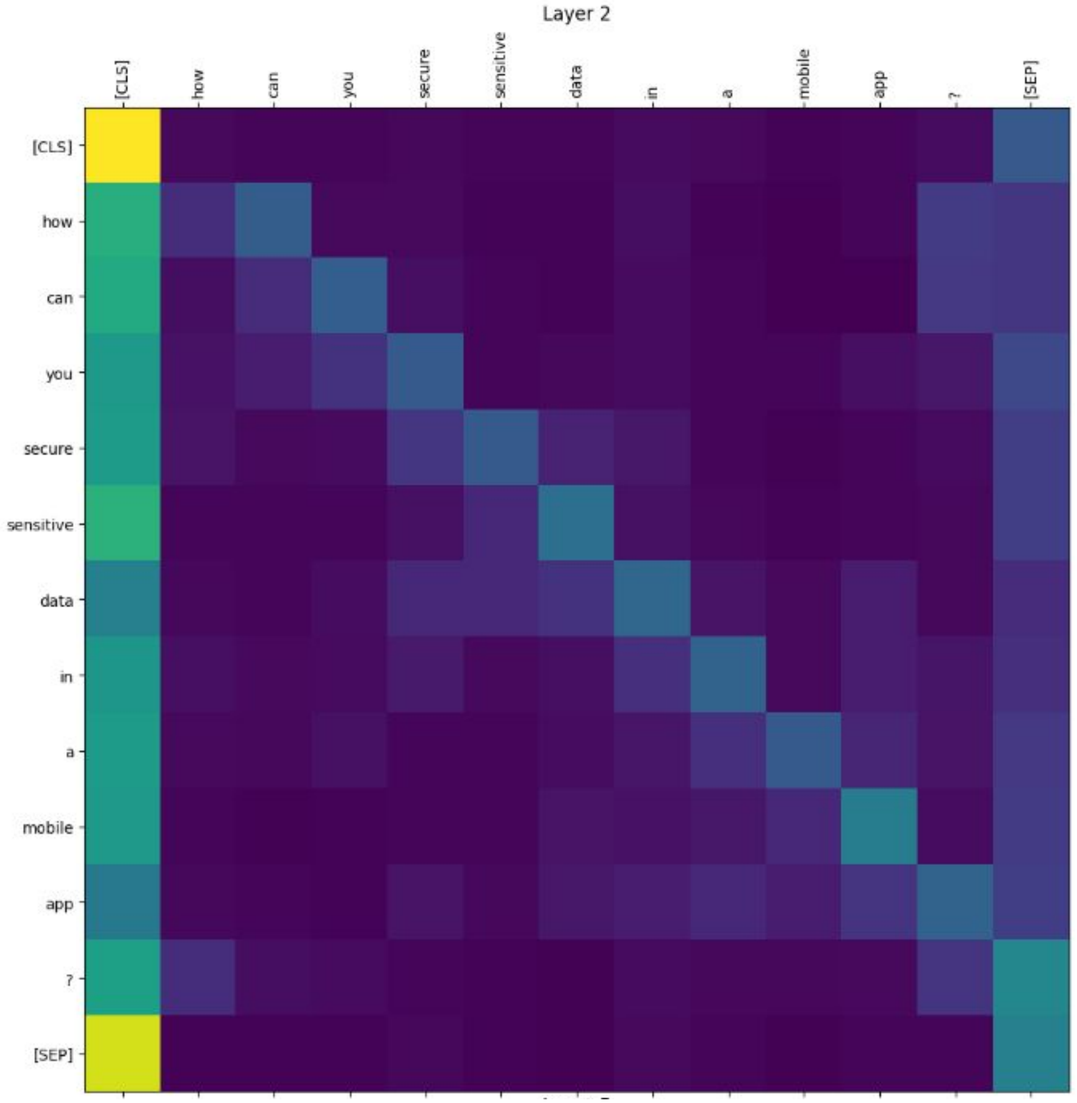


Evaluation. Mistakes.

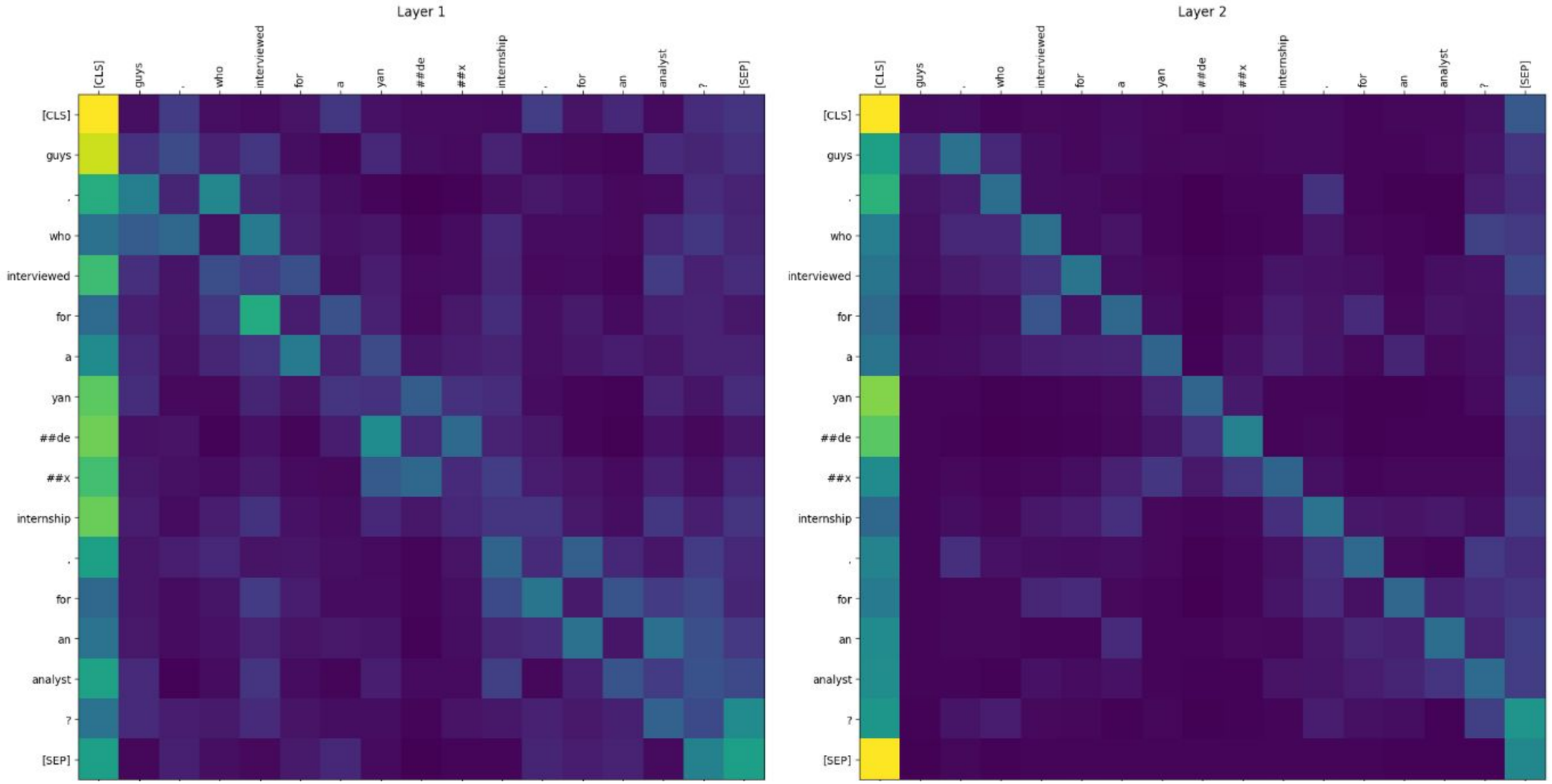


Attention matrix. Meta.





Attention matrix. Non-meta.



Additional tests.

- Concerns after review of the results
  - Unbalanced question words: “How” for non-meta, “Is” and “Does” for meta.
  - Chat-specific keywords for meta questions: “Yandex”, “internship”, etc.
- Hand-crafted tests
  - Presence / absence of specific words like “Yandex”, “internship”, “guys”
  - Question format changes (“How do you do” → “Is there a way” etc.)
  - Result: predictions didn’t change

Conclusion

- Dataset with meta-questions and no-meta questions was successfully created
- The obtained outcomes were analyzed for different settings
- The model reasoning was observed and analyzed

Future work

- Extend the existing dataset with more real meta-questions
- Deploy the existing solution
- Create a Telegram bot as a chat assistant