Граф новостей

Сайт: http://5.178.2.42:5050/

Github: https://github.com/andreybabynin/semantic_news_graph

Проект по курсу ODS ML System Design

Участники

Андрей

Москва

Алексей

Москва

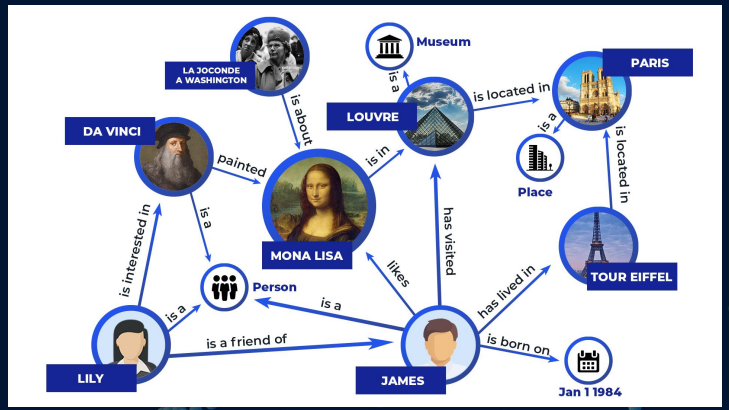
Илья

Туапсе

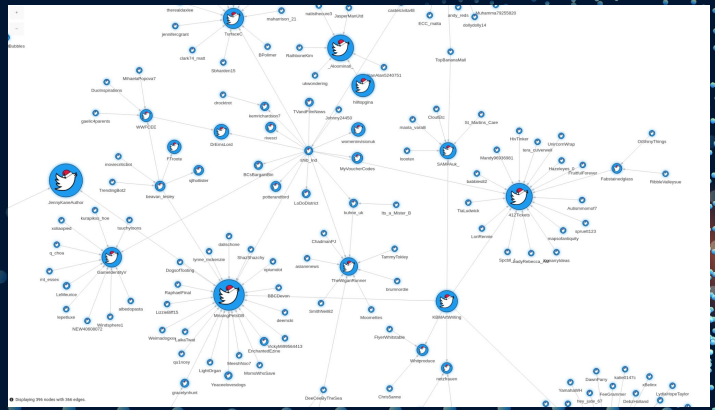
От идеи к воплощению: Что поменялось

- Вершины - люди, места, организации
- Ребра - новости

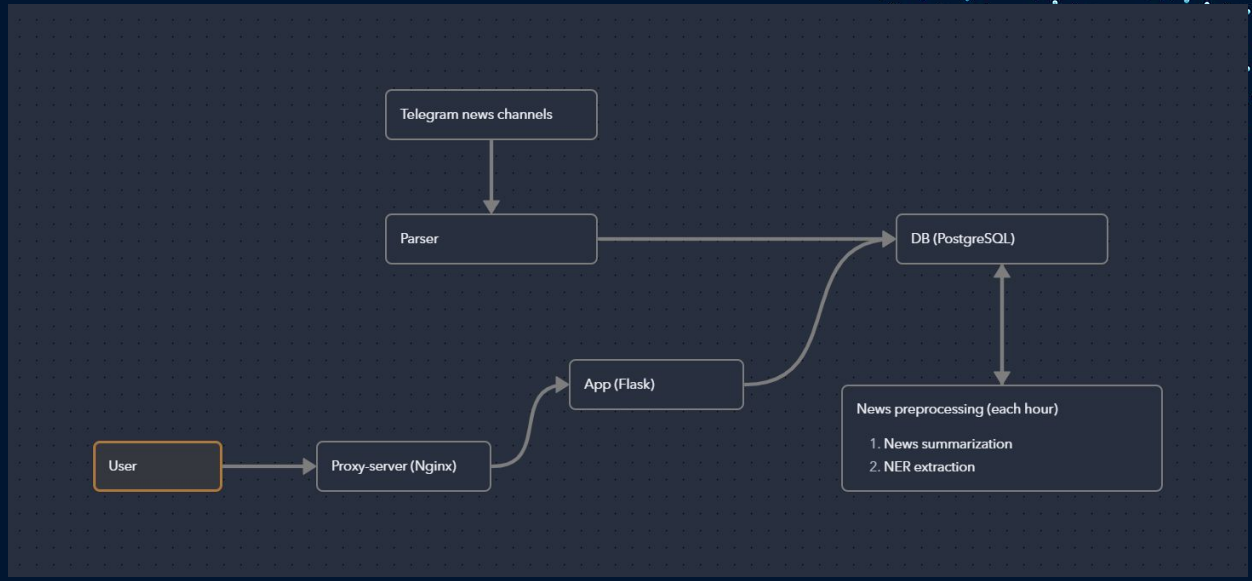
Граф знаний



Граф зависимостей



Структура проекта



Парсер

Telethon library

DB

PostgreSQL

Preprocessing pipeline

Natasha, stanza, hugging face

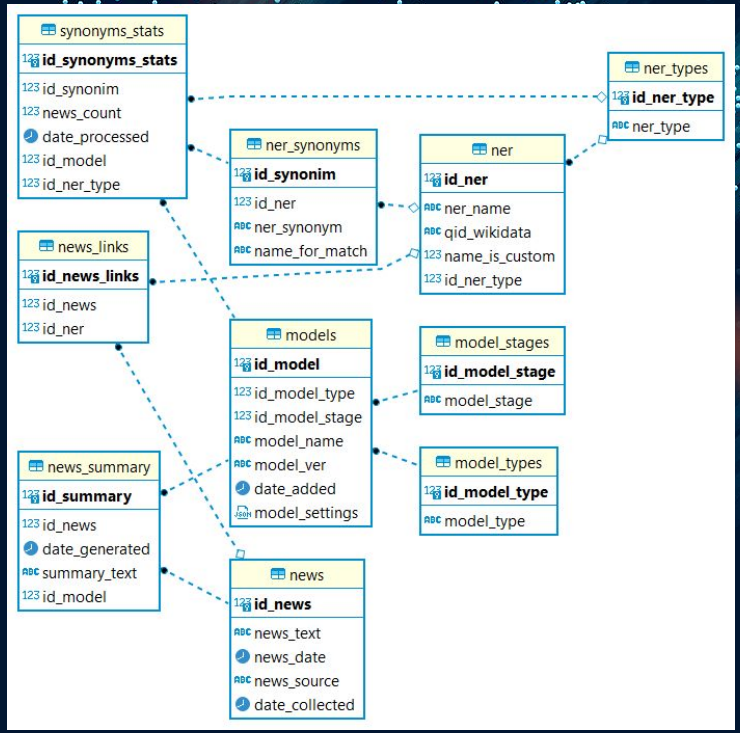
Frontend

Flask, d3.js

DB

Причины выборы SQL DB:

- Небольшой объем хранимых данных (200-300 записей в день)
- Необходимость в OLAP запросах при построении графов
- Необходимость в поддержании связей один ко многим (например, NER и его синонимы) и многие ко многим (новости и NER)



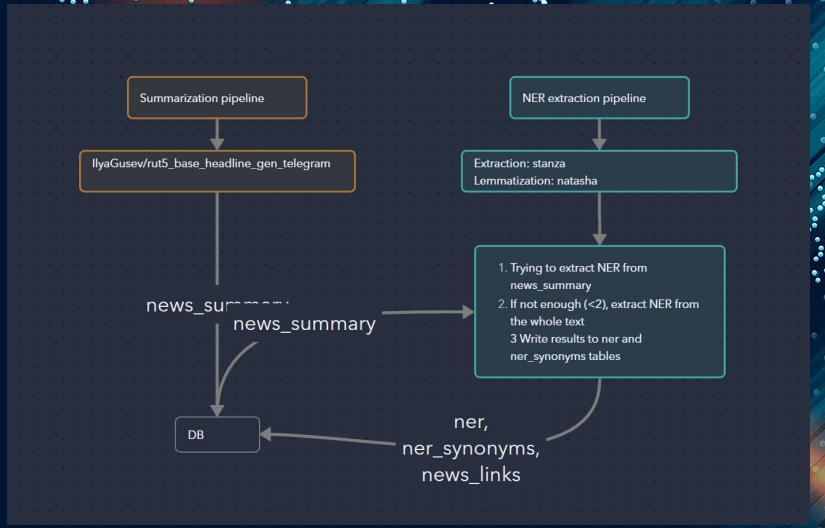
Препроцессинг

Основная задача: выделить NEs из текста:

- **В тексте может быть много NEs** -> строим предварительно summary, смотрим какие NEs остались, считаем их основными
- **В тексте могут встречаться аббревиатуры или сокращения** -> проверяем по словарю синонимов и делаем запрос по wikidata api

Для выделения NEs -> stanza (лучшее качество судя по нашим экспериментам)

Для нормализации (лемматизации) -> natasha



Многообразие синонимов и Entity Linking

Стремление к лаконичности в письменной речи журналистов, а также тот факт, что люди понимают контекст в отличие от машин создал для нас бесконечный поток проблем, связанных с установлением связей между синонимами.

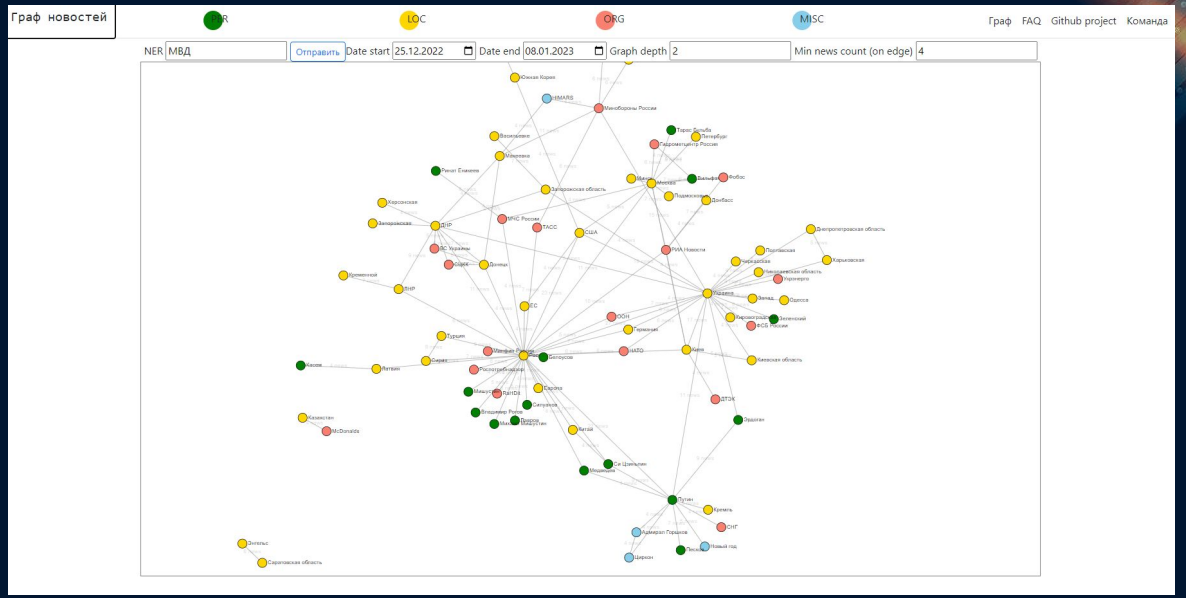
Примеры:

- Центробанк России - ЦБ РФ - Центробанк РФ - ЦБ
- Совбез ООН - Совет безопасности - Совбез - СБ ООН
- РЖД - Российские железные дороги - ОАО "РЖД"
- Петр I - Петр Великий - Петр Первый
- Михаил Мишустин - Мишустин

Frontend (Flask, d3.js)

Причины выборы d3.js:

- Большая библиотека готовых примеров
- Возможность глубокой кастомизации графиков
- Интерактивность



Что можно улучшить

- Сделать граф более информативным добавив ключевой статистики по упоминаемости и “важности” NEs. Добавить возможность “расширения”графа при клике по конкретной вершине.
- Предусмотреть шардирование базы данных для поддержания скорости OLAP запросов: хранить свежие записи в “быстром” хранилище (например, при росте количества источников даже до 20, выходим за ≈1 млн записей в год)
- Улучшить алгоритм распознавания NEs, обучив, например, трансформер на нашем корпусе новостей
- Добавить механизм очередей для обработки новостей “на лету”

Примерное распределение времени

15%

Поднятие docker, настройка БД

50%

Работа над выделением и сопоставлением NEs

35%

Настройка d3.js

Q&A