AI Product / Hackathon

Yandex Cloud

NAPOLEON IT

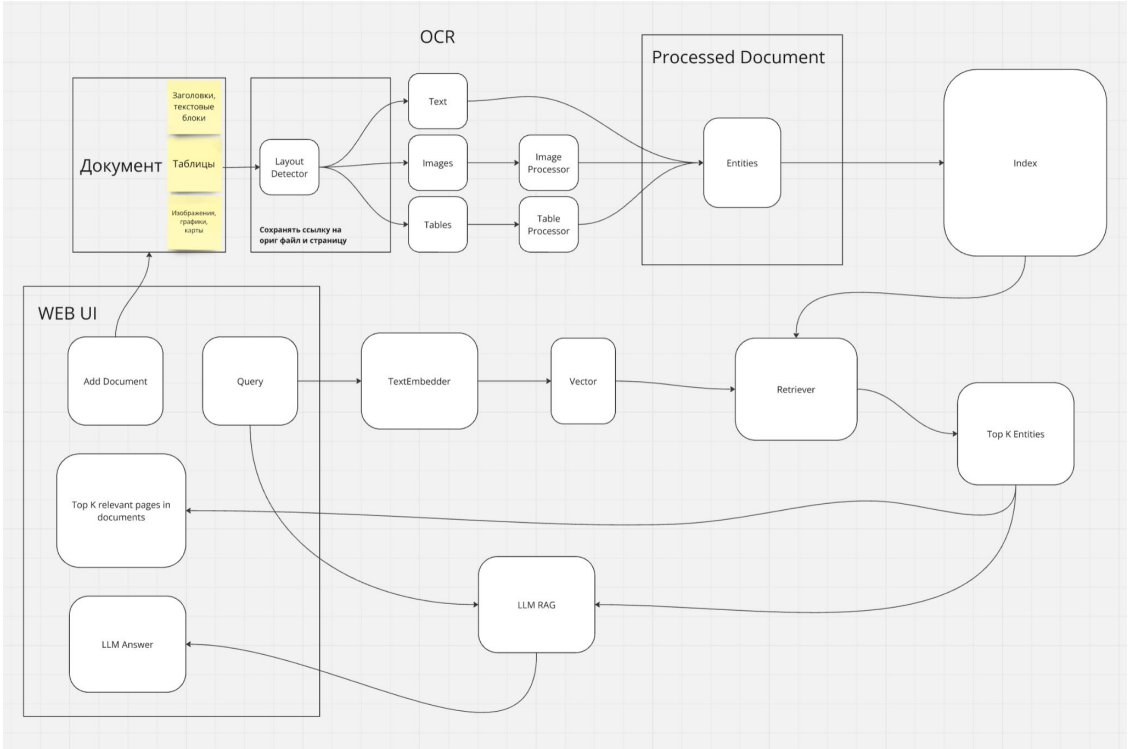
AI Talent Hub

document-search

stay fine-tuned

Кейс №12. Автоматизация поиска и анализа нормативной документации по строительству объектов

4 сентября


OCR
Processed Document
Документ
Заголовки, текстовые блоки
Таблицы
Изображения, графики, карты
Layout Detector
Сохранять ссылку на ориг файл и страницу
Text
Images
Tables
Image Processor
Table Processor
Entities
Index
WEB UI
Add Document
Query
TextEmbedder
Vector
Retriever
Top K Entities
Top K relevant pages in documents
LLM RAG
LLM Answer


OCR
Document reader
class DocumentReader:
...
def process_pdf(self, document_path: str) -> ProcessedDocument:
...
def process_docx(self, document_path: str) -> ProcessedDocument:
...
...
Document entities
class ProcessedDocument:
name: str
id_: str
num_pages: int
original_format: Literal['pdf', 'docx']
entities: list[DocEntity]
class EntityPosition:
parent_document_id: str
page: int
position: float # подумать как лучше
class DocEntity(Protocol):
id_: int
position: EntityPosition
class TextDocEntity(DocEntity):
id_: int
position: EntityPosition
text: str
class TableDocEntity(DocEntity):
id_: int
position: EntityPosition
table: str
class ImageDocEntity(DocEntity):
id_: int
position: EntityPosition
image: ?
Document entities + vector
class TextEntityEmbedder(Protocol):
def vectorize(self, item: TextDocEntity) -> VectorizedDocEntity: ...
Document entities + vector
class VectorizedDocEntity(DocEntity):
vector: np.ndarray
class VectorizedDocument:
name: str
id_: str
entities: list[VectorizedDocEntity]
Storage
class DocumentStorage(Protocol):
def _add_entity(self, entity: DocEntity) -> None: ...
def add_document(self, document: VectorizedDocument) -> None: ...
def get_relevant_indexes(self, document: VectorizedDocument, k: int) -> list[DocEntity]: ...
Retriever
class Retriever(Protocol):
def __init__(self, storage: DocumentStorage): ...
def get_relevant_entities(self, query: str, k: int) -> list[DocEntity]: ...
5 сентября

6 сентября

Артём - Document entities (OCR), Web UI
Игорь - CI, readme и интерфейсы, Storage, Backend
Стас - Retriever, Embedder, RAG
Настя - презентация к чекпоинту, проработка идей и гипотез

Репозиторий:

Сложности

- OCR и извлечение сущностей
- Сложная структура документов
- Индексирование разных сущностей (таблицы, графики)

Планы

- Стабильный baseline
- Проработка разных вариантов работы с таблицами и изображениями
- Добавление RAG

AI Product / Hackathon

Команда

Игорь Павлов
Капитан команды
@bom_bo0m

Артем Иванов
Разработчик
@incncvble

Анастасия Остапчук
ПМ
@anastasiia_ost_v

Станислав Стафиевский
Разработчик
@stafstas