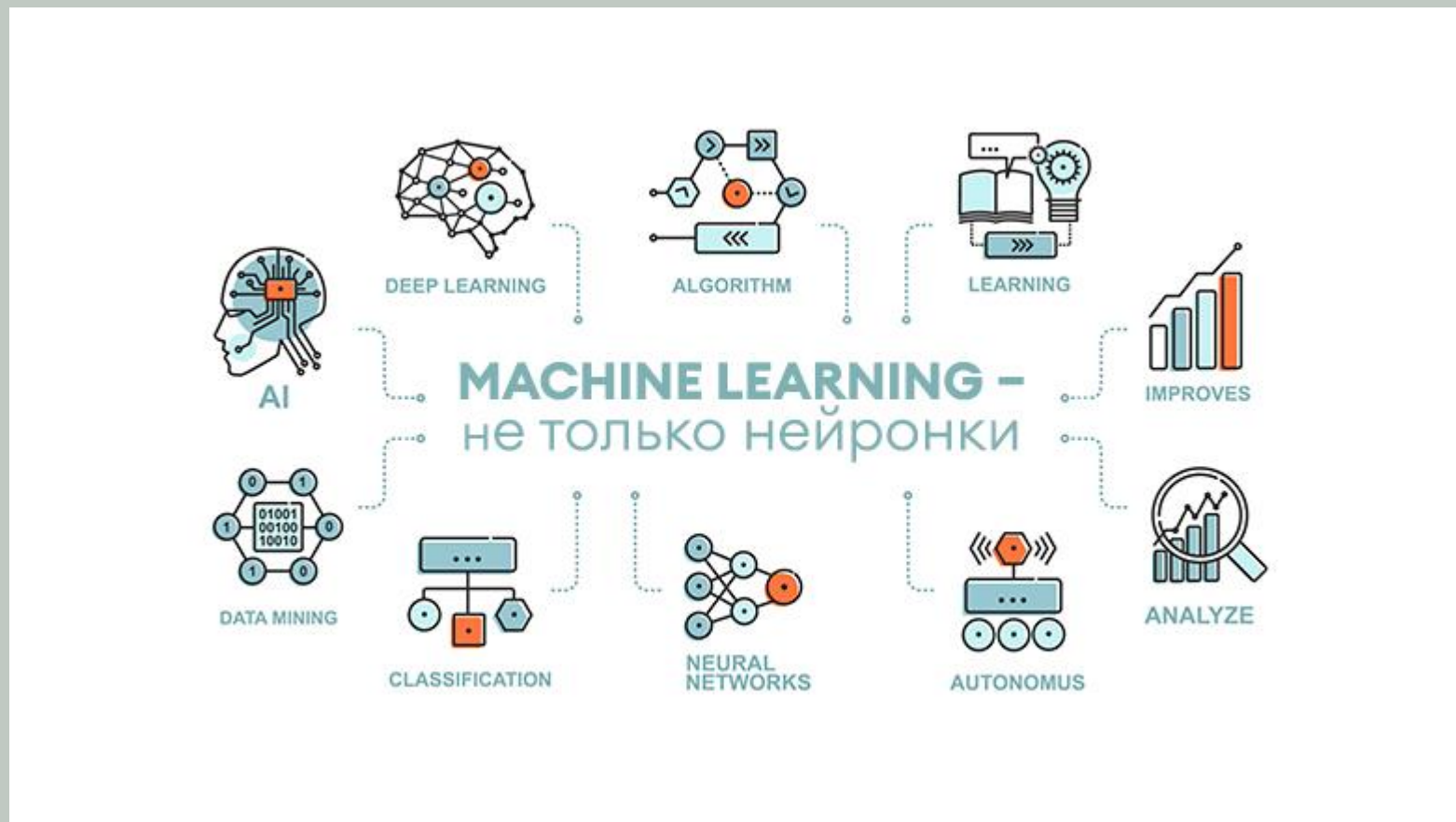
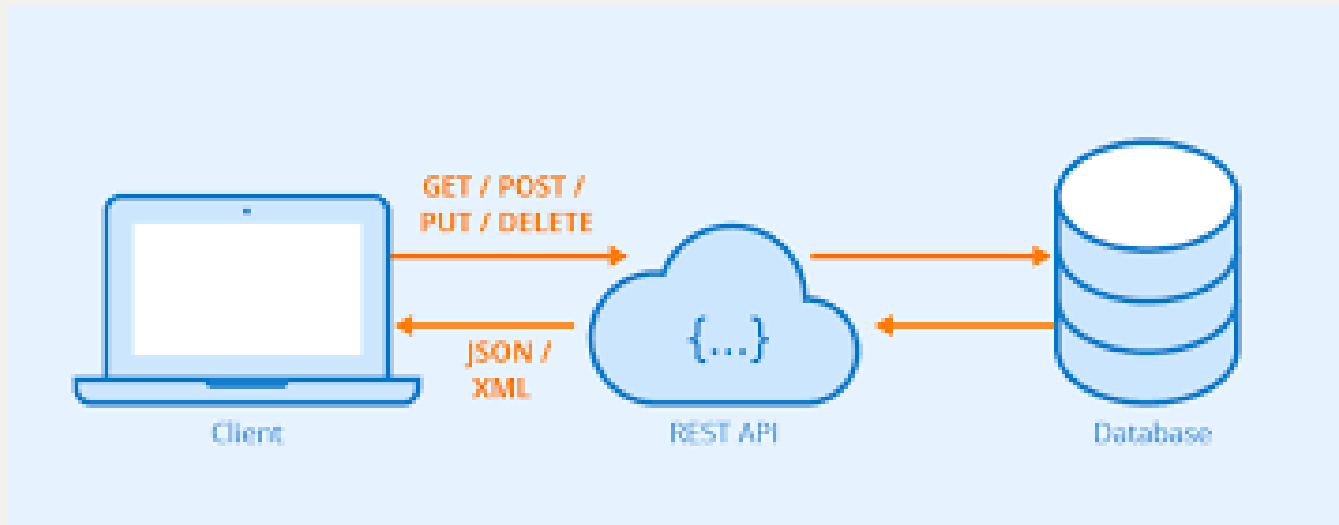


Разработка приложений ML



API (Application Programming Interface)

API (Application Programming Interface) — это набор правил, протоколов и инструментов, который позволяет разным программам взаимодействовать друг с другом.



FastAPI



FastAPI — это современный, высокопроизводительный Python-фреймворк для создания API (веб-серверов) с минимальным кодом.

Он идеально подходит для:

- ML-моделей в продакшене.
- Микросервисов.
- Бэкенда для мобильных/веб-приложений.

<https://fastapi.tiangolo.com/>


01 Установка



```
(.venv) PS C:\Users\rezed\PycharmProjects\MLProject22> pip install fastapi
```


```
(.venv) PS C:\Users\rezed\PycharmProjects\MLProject22> pip install uvicorn
```


✓  **MLProject22** C:\Users\rezed\PycharmProjects\MLProject22


✓  **.venv** library root


>  etc

>  Lib

>  Scripts


>  share

 .gitignore


 pyvenv.cfg


 app_streamlit.py

 main.py

 model_knn1.pkl

 vectorizer.pkl

>  External Libraries

 Scratches and Consoles

Структура проекта.

Библиотеки

```
from fastapi import FastAPI
import pickle
from pydantic import BaseModel
import pandas as pd
import numpy as np
import string
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import pymorphy3
```

Загрузка моделей

```
app = FastAPI()

with open('model_knn1.pkl', 'rb') as file:
    model = pickle.load(file)

with open('vectorizer.pkl', 'rb') as file:
    vectorizer = pickle.load(file)
```

Методы для предобработки текста

```
def fun_punctuation_text(text): 1 usage
    text = text.lower()
    text = ''.join([ch for ch in text if ch not in string.punctuation])
    text = ''.join([i if not i.isdigit() else ' ' for i in text])
    text = ''.join([i if i.isalpha() else ' ' for i in text])
    text = re.sub(pattern: r'\s+', repl: ' ', text, flags=re.I)
    text = re.sub(pattern: '[a-z]', repl: '', text, flags=re.I)
    st = ')\xa0'
    text = ''.join([ch if ch not in st else ' ' for ch in text])
    return text
```

```
def fun_lemmatizing_text(text): 1 usage
    tokens = word_tokenize(text)
    res = list()
    for word in tokens:
        p = pymorphy3.MorphAnalyzer(lang='ru').parse(word)[0]
        res.append(p.normal_form)
    text = " ".join(res)
    return text
```

Методы для предобработки текста

```
def fun_tokenize(text):  
    1 usage  
    nltk.download('stopwords')  
    russian_stopwords = nltk.corpus.stopwords.words('russian')  
    russian_stopwords.extend(['т.д.', 'т', 'д', 'это', 'который', 'свой', 'своём', 'всем', 'всё', 'её', 'оба', 'ещё', '  
    t = word_tokenize(text)  
    tokens = [token for token in t if token not in russian_stopwords]  
    text = " ".join(tokens)  
    return text
```

```
def fun_pred_text(text):  
    text = fun_punctuation_text(text)  
    text = fun_lemmatizing_text(text)  
    text = fun_tokenize(text)  
    return text
```


Функция предсказания кластера

```
def predict_cluster(text): 1 usage
    text_vectorized = vectorizer.transform([fun_pred_text(text)])
    prediction = model.predict(text_vectorized)
    probabilities = model.predict_proba(text_vectorized)
    rez1 = f"Класс: {prediction[0]}"
    rez2 = f"Вероятности: {probabilities}"
    mapping = {
        0: ' 0 - кластер',
        1: ' 1 - кластер',
        2: ' 2 - кластер',
        3: ' 3 - кластер',
        4: ' 4 - кластер'
    }
    selected_cluster = mapping[prediction[0]]
    return selected_cluster, rez2
```

HTTP-метод post для отправки данных

```
# Модель для входных данных
class Item(BaseModel):
    text: str

# метод для подключения к интерфейсу
@app.post("/predict")
def post_pred_text(item: Item):
    return {'cluster': predict_cluster(item.text)}
```

Запуск сервера

```
uvicorn main:app --reload
```

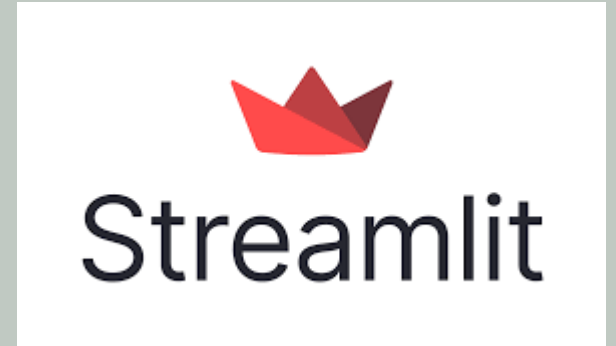
```
uvicorn main:app --reload
```

- `main` — имя файла (например, `main.py`).
- `app` — переменная с `FastAPI()`.
- `--reload` — автоматическая перезагрузка при изменении кода (только для разработки).

Сервер будет доступен по адресу:

- <http://127.0.0.1:8000>
- Документация Swagger: <http://127.0.0.1:8000/docs>
- Альтернативная документация: <http://127.0.0.1:8000/redoc>

02 Интерфейс приложения



Streamlit — это фреймворк на Python для создания веб-приложений с минимальным кодом. Он идеально подходит для:

- Быстрого прототипирования ML-моделей.
- Визуализации данных.
- Создания интерактивных дашбордов.

Установка Streamlit

```
(.venv) PS C:\Users\rezed\PycharmProjects\MLProject22> pip install streamlit
```

[Streamlit documentation](#)

app_streamlit.py

```
import streamlit as st
import requests
import os

# Обнуление статусов прокси для корректного подключения
os.environ['HTTP_PROXY'] = ''
os.environ['HTTPS_PROXY'] = ''
```

```
st.title("Предсказание кластера фильма")
# Поле ввода
input_text = st.text_area("Введите описание фильма", height=200)

if st.button("Предсказать"):
    if input_text == "":
        st.write(f"Введите текст")
    else:
        data = {
            "text": input_text
        }

        url = "http://127.0.0.1:8000/predict"
        response = requests.post(url, json=data)
        result = response.json()
        clust = result.get("cluster")

        # Вывод предсказанного кластера
        st.markdown(f"""
            ### Предсказанный кластер
            """)
        st.write(f"Кластер: {clust[0]}")
        st.write(f"{clust[1]}")
```

Запуск приложения

```
(.venv) PS C:\Users\rezed\PycharmProjects\MLProject22> streamlit run app_streamlit.py
```

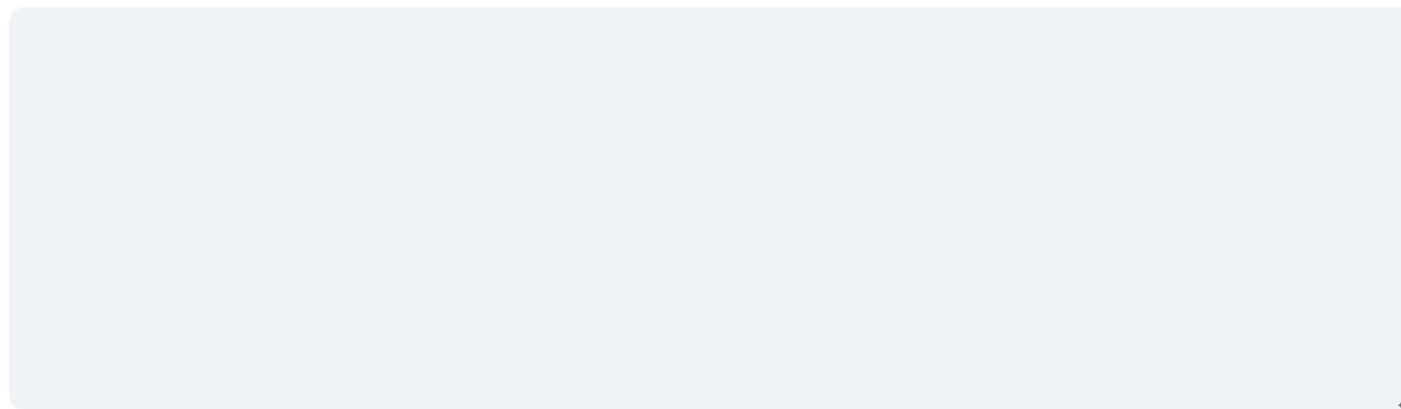
Запуск:

1. FastAPI: `uvicorn api:app --reload`
2. Streamlit: `streamlit run app.py`

03 Приложение

Предсказание кластера фильма

Введите описание фильма



Предсказать

03 Приложение

Предсказание кластера фильма

Введите описание фильма

Бухгалтер Энди Дюфрейн обвинён в убийстве собственной жены и её любовника. Оказавшись в тюрьме под названием Шоушенк; он сталкивается с жестокостью и беззаконием; царящими по обе стороны решётки. Каждый, кто попадает в эти стены; становится их рабом до конца жизни. Но Энди; обладающий живым умом и доброй душой; находит подход как к заключённым; так и к охранникам; добиваясь их особого к себе расположения.

Предсказать

Предсказанный кластер

Кластер: 0 - кластер

Вероятности: [[0.6 0.2 0. 0.2 0.]]

О3 Приложение

Предсказание кластера фильма

Введите описание фильма

21 июня 1941 года. Молодой лейтенант Коля Плужников, получив назначение на постоянное место службы, приезжает в Брест. Переполненные залы ожидания вокзала и толпа увешанных багажом людей не настораживают охваченного радостными надеждами юношу. Коля спешит к месту расположения своей части — в Брестскую крепость... Солдата не успевают зачислить в личный состав военнослужащих, а в четыре утра раздаются артиллерийские разрывы — началась война. Он не был зачислен в ряды военнослужащих Брестской крепости, но всё равно принял участие в бою.

Предсказать

Предсказанный кластер

Кластер: 2 – кластер

Вероятности: $[[0.0 \ 0.2 \ 0.4 \ 0.4 \ 0.]]$

О3 Приложение

Предсказание кластера фильма

Введите описание фильма

Российский ракетный подводный крейсер специального назначения бесследно исчезает во время секретной миссии в Гренландском море. На его поиски отправляется экипаж под командованием Виктора Воронина, чей старший брат командовал пропавшей субмариной. Перед моряками стоит задача любыми силами найти подлодку и не допустить, чтобы новое секретное оружие попало в руки врага. В это же время из-за геологических исследований на полярной станции в северных водах пробуждается Кракен — гигантское чудовище, способное сливаться с подводной тьмой, обладающее высоким интеллектом. Именно с ним предстоит столкнуться морякам, разыскивающим пропавшую подводную лодку.

Предсказать

Предсказанный кластер

Кластер: 3 – кластер

Вероятности: [[0. 0. 0.2 0.4 0.4]]