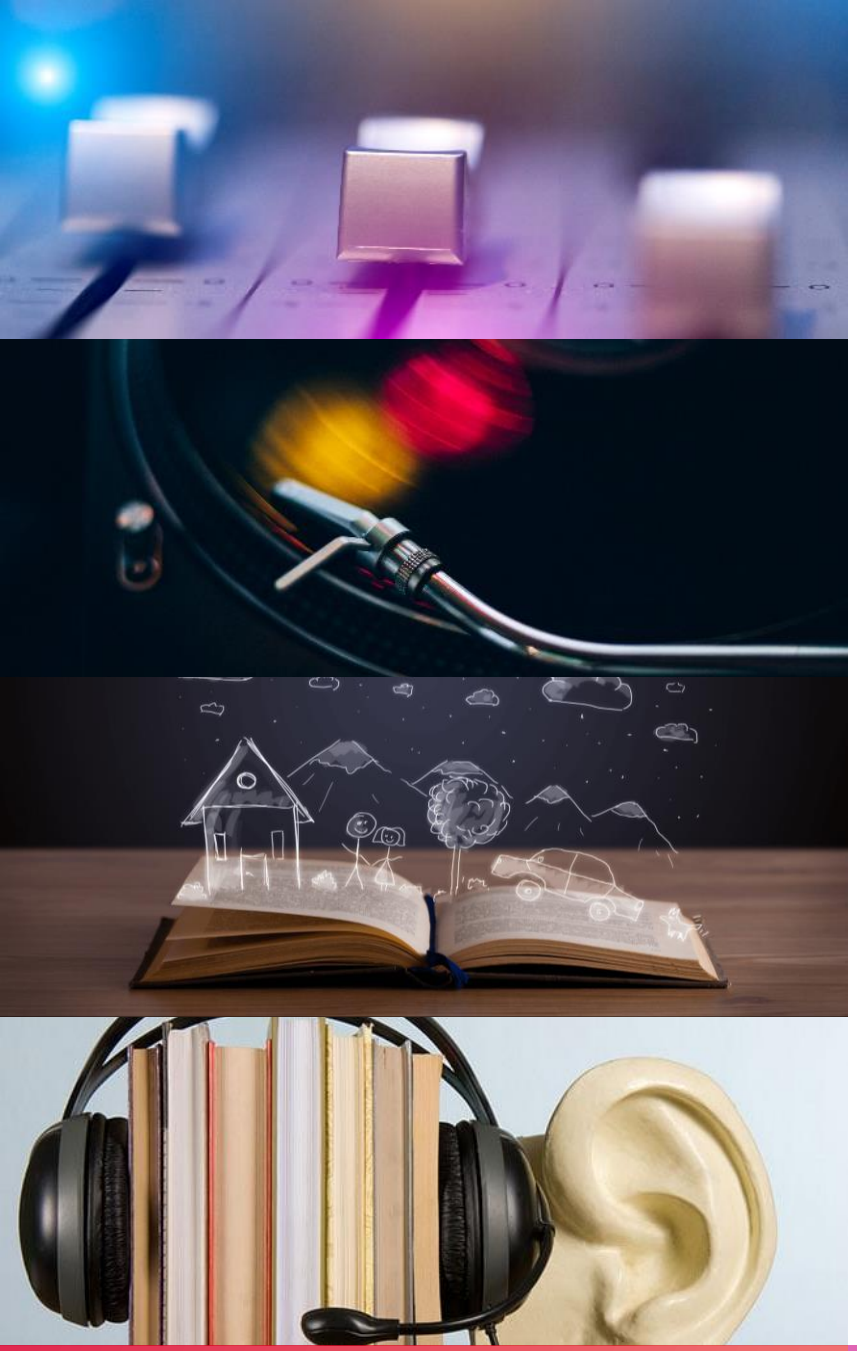

УДАРЕНИЕ-
АНАЛИЗАТОР
ТЕКСТА
ОТ КОМАНДЫ
«ОтрубИ»



ЦЕЛЬ ПРОЕКТА

- Разработка алгоритма контекстной расстановки ударений
- Реализация пользовательского интерфейса
- Алгоритм планируется использовать для анализа текстов при подготовке книги для озвучивания.
- Код проекта и детальные описания к проекту в репозитории https://github.com/KirillStarodubovVR/CutEggOff/tree/main

AGENDA

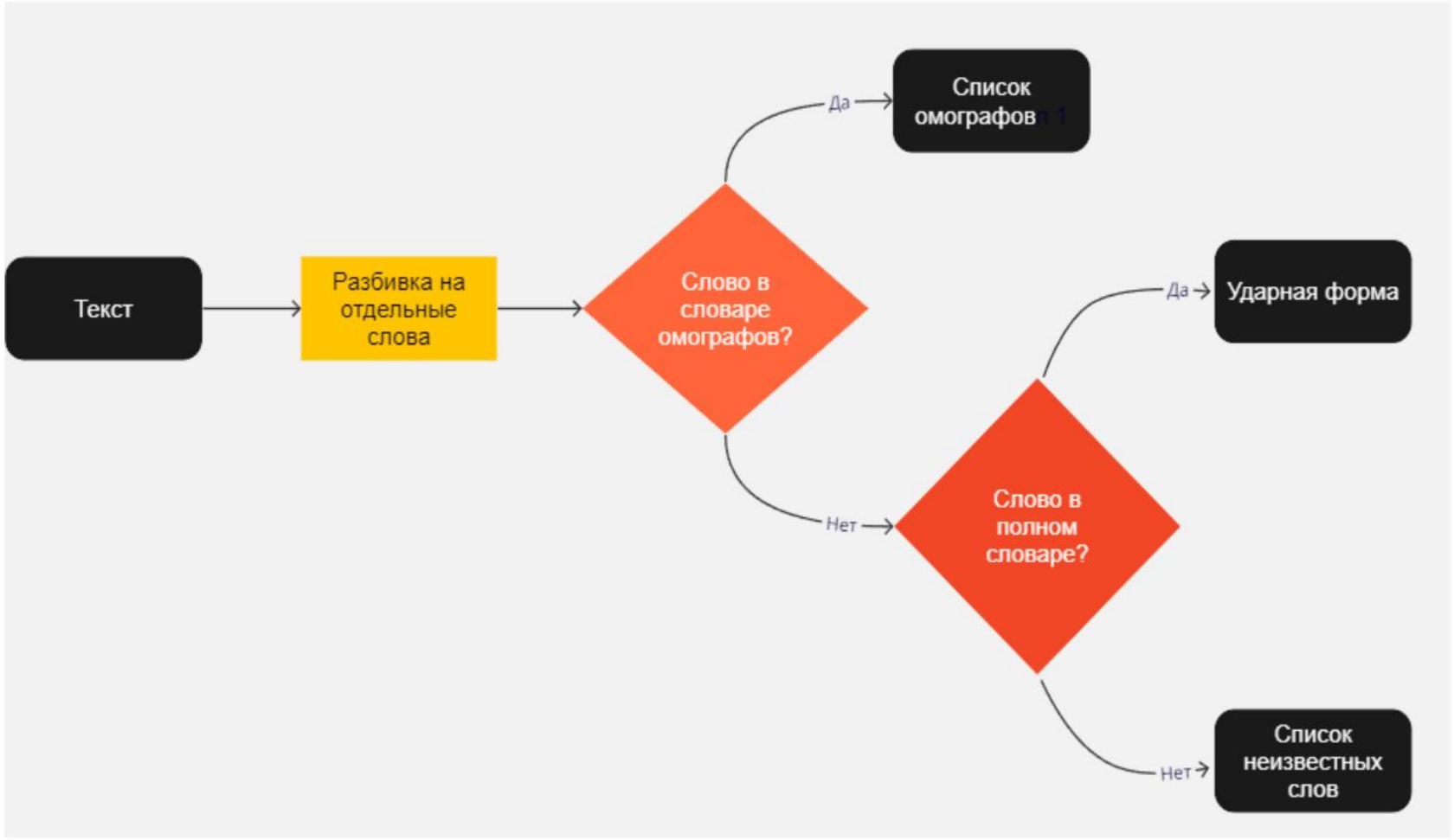
Алгоритм

Словари

Эксперименты

Использование

АЛГОРИТМ



- В качестве исходных данных алгоритм получает текст на русском языке и возвращает 3 файла:
- текст с ударениями,
- список омографов,
- список ненайденных в словаре слов

СЛОВАРИ

- В основе словарей - полная акцентуированная парадигма по Зализняку А.А., дополненная переработанным корпусом с wiktionary.
- С точки зрения пользователя Викисловарь представляет собой сайт, объединяющий сотни тысяч или миллионы страниц, связанных друг с другом гиперссылками.
- Словари, собранные с wiktionary доступны по ссылке.

ДЕТАЛИ

Возможности

- Анализ контекста предложения.
- Сравнение слова с предоставленным словарем.
- Расстановка ударений по тексту книги.
- Выделение многозначных случаев расстановки ударений.
- Выделение слов, для которых невозможно найти ударение в словарях.
- Возможность загрузки дополнительных словарей.

Ограничения

- Алгоритм работает только с русским языком
- Возможность расстановки ударения ограничено размером словаря
- Не проведена ё-фикация словарей и алгоритма
- Алгоритм работает только с прописными
- Словарь омографов, используемый в проекте, включает не только омографы как таковые, но и слова, имеющие разную ударную форму или разные ударения по версиям использованных словарей.

ЭКСПЕРИМЕНТЫ

- 🛠 Объединенный словарь омографов и словоформ с отдельным словарем лемм; итеративный поиск ударения на основе выделенной леммы
- 🛠 Использование отдельных словарей омографом и словоформ; прямой поиск слов в словарях.

Алгоритм	Точность	Комментарии
Объединенный словарь + словарь лемм с wiktionary	66%	С учетом омографов Словарь - 600 тыс. словоформ.
Отдельные словари НКРЯ	94%	Без учета омографов, выборка из тех же данных, что и словарь. Словарь - 3 млн. словоформ
☆ Отдельные словари Зализняк А.А. + wiktionary	90%	Без учета омографов. Словарь - 1,2 млн. словоформ

2 выборки по 5 тыс. текстов с ударениями, собранных с НКРЯ.

Метрика - **accuracy**, как % exact match

ИСПОЛЬЗОВАНИЕ МОДЕЛИ

ИСПОЛЬЗОВАНИЕ В ИНТЕРПРЕТАТОРЕ

```
|   README.md
|   ruaccent.py - основной файл алгоритма
|   text_split.py - разбивка на слова
|   web_interface.py - для запуска UI с flask
|   pc_app.py - для запуска UI с customtkinter
|
├──templates - оформление веб-интерфейса
|       index.html
|       result.html
|
├──dictionaries
|       .gitattributes
|       accents.json - словарь словоформ
|       omographs.json - словарь омографов
|
├──test_data
|       test_1_5000.csv - тестовая выборка на основе НКРЯ
|       test_2_5000.csv - тестовая выборка на основе НКРЯ
|       metrics_review.ipynb - блокнот с расчетом метрики
```

Для начала работы нужно:

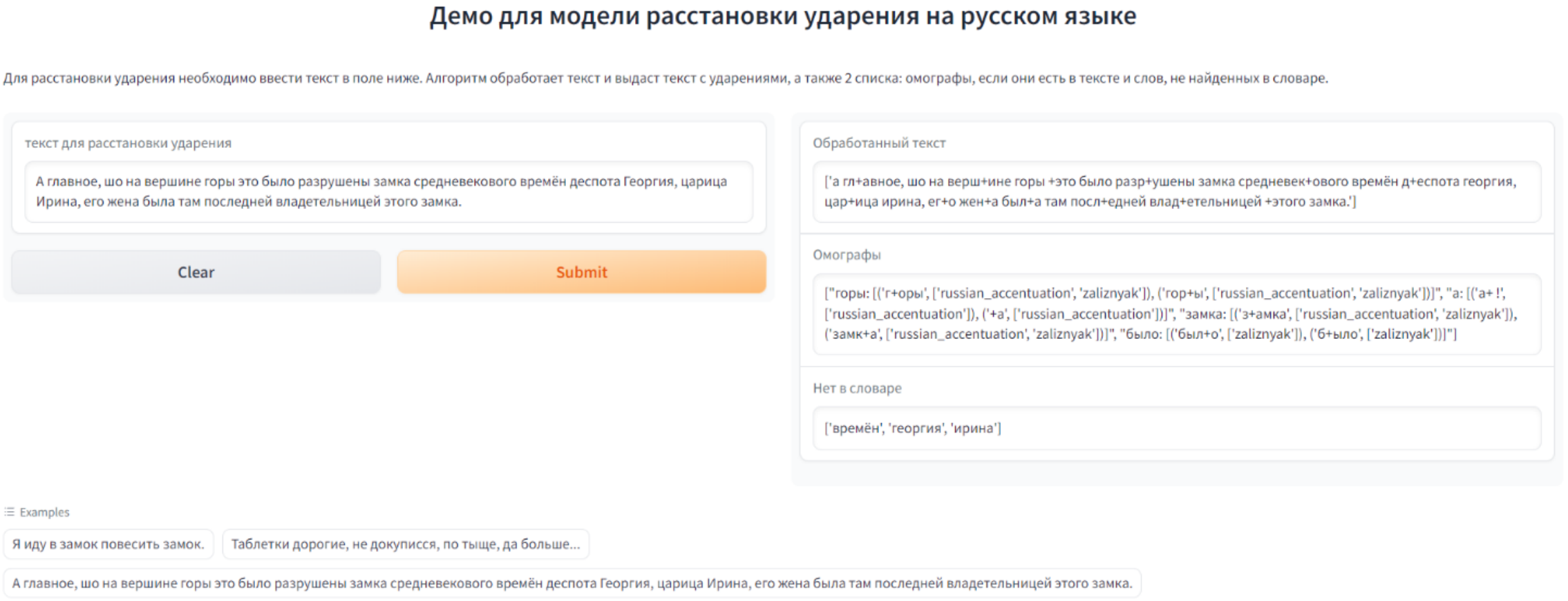
- клонировать репозиторий
- скачать словари и добавить их в папку **dictionaries**
- создать объект класс **RUAccent** и загрузить в него словари

```
ru_accent = RUAccent()
ru_accent.load()
```

```
text_to_process = "В этом замке совершенно нет ни одного замка."
processed_text = ru_accent.process_all(text_to_process)
print(processed_text)
```

РАЗВЕРТЫВАНИЕ НА БАЗЕ gradio

Используется интерфейс gradio на платформе hugging face. Попробовать версию интерфейса можно здесь.



РАЗВЕРТЫВАНИЕ НА БАЗЕ

Используется интерфейс flask на платформе hugging face.
Можно ввести предложения для расстановки ударения или загрузить файл, программа подготовит три файла.
Попробовать версию интерфейса можно [здесь.]

Russian Language Accent

Input Text:

Process

Upload a Text File (txt in utf-8 only!!!)

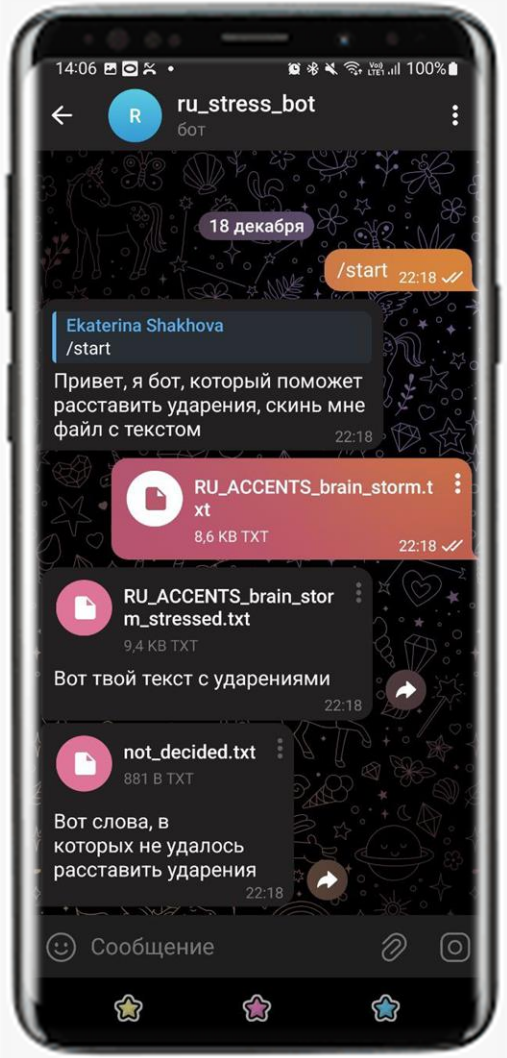
Choose File No file chosen Upload

Processing Complete

Download the processed files:

- Text with accents
- Omographs
- Unknown words

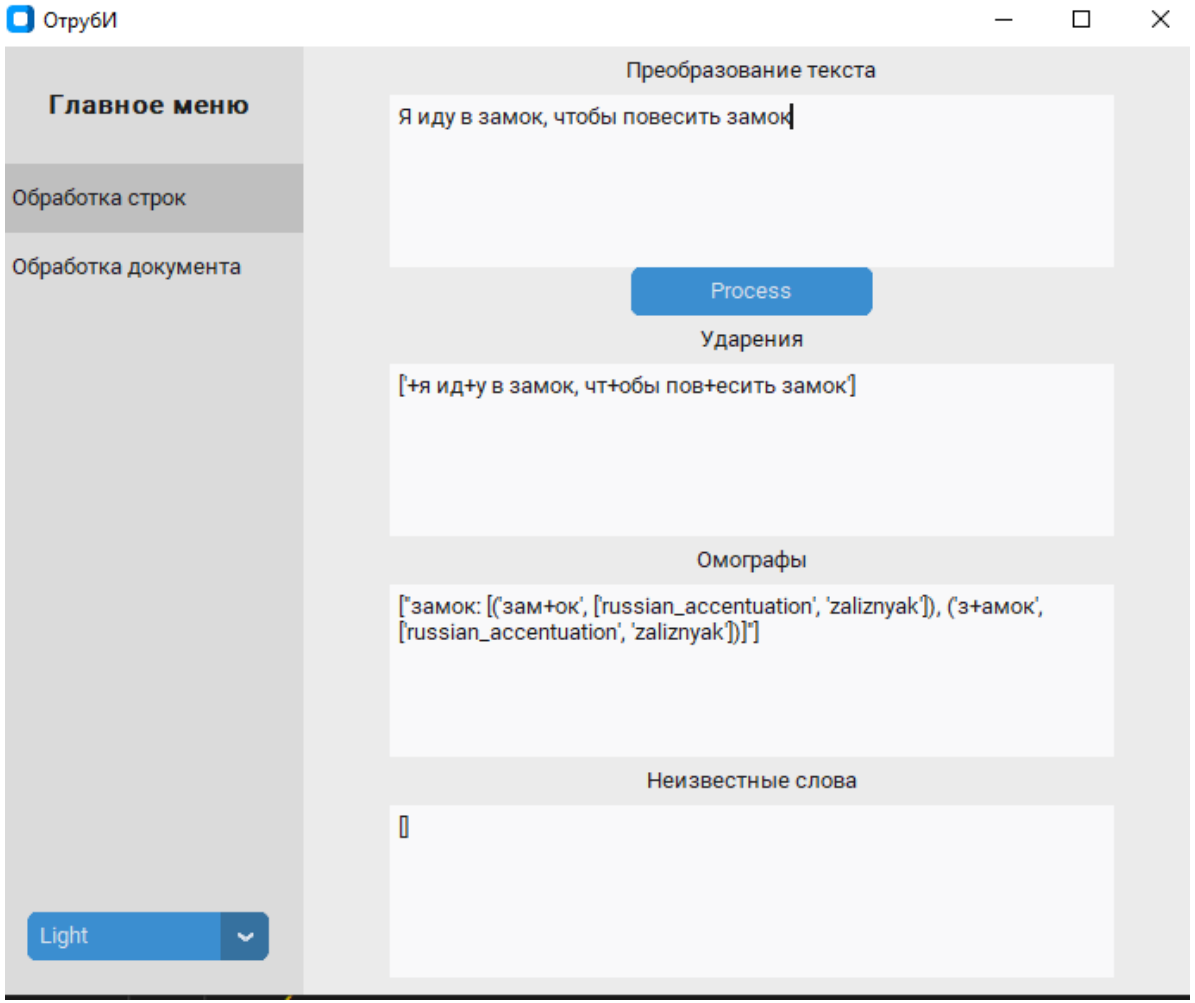
ИСПОЛЬЗОВАНИЕ ТЕЛЕГРАМ-БОТА



В качестве одного из первых вариантов был использован чат-бот **@ru_stress_bot** для обработки текста по итеративному принципу.

> ⚠ **Warning**
>
> В основе чат-бота не финальный алгоритм! Чат-бот запущен на платном сервисе и после окончания хакатона будет отключен!

РАЗВЕРТЫВАНИЕ НА БАЗЕ Tkinter



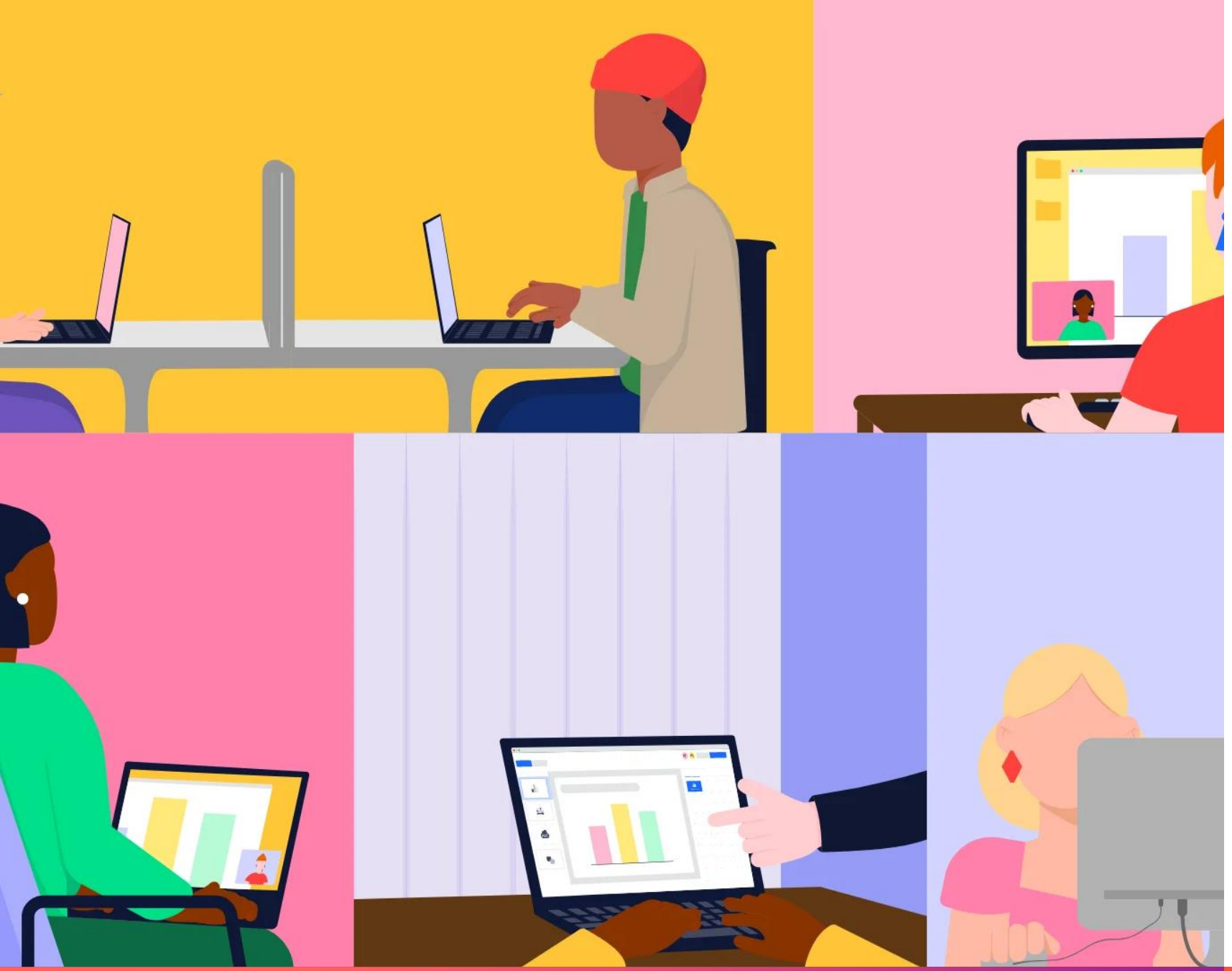
Для начала работы нужно:

- клонировать репозиторий
- скачать словари и добавить их в папку **dictionaries**
- Запустить файл с кодом Tkinter

Интерфейс запускается только локально и файл должен быть в той же папке, что и основной код!

ДАЛЬНЕЙШЕЕ РАЗВИТИЕ ПРОЕКТА

- Подсвечивание омографов и неизвестных слов.
- Добавление в список не только слов, но и контекста.
- Добавить синтез речи
- Классифицировать неизвестные слова имена собственные, топонимы и т.д.
- Добавить возможность вносить отдельные слова в словари
- Добавить нейронную сеть для ударения в неизвестных словах
- Возможность исправления словарей через интерфейс, выводить из книг примеры использования и т.д.



TEAM

Кирилл Стародубов, Анна Зверева - код анализатора текста

Александр Левакин, Андрей Илюхин, Екатерина Шахова, Дмитрий Петров - работа со словарями и веб-интерфейсами

THANK YOU