ИНСТРУКЦИЯ К КОНКУРСНОМУ ПРОЕКТУ
«Разработка метода интеллектуального анализа рисков информационной безопасности в ИТ-организации»

Образовательная организация: филиал ФГБОУ ВО «НИУ «МЭИ» в г. Смоленске
Автор: Дроздов Никита Владимирович
Куратор: Булыгина Ольга Валентиновна (доктор экономических наук, доцент кафедры Информационных технологий в экономике и управлении)

СОДЕРЖАНИЕ
1 Работа в Google Colab.....1
2 Работа локально в Jupyter или VS Code.....3
3 Работа через веб-приложение.....3
Возможные проблемы и их решение:.....5

1 Работа в Google Colab

Данный способ является самым простым и не требует установки дополнительного программного обеспечения на компьютер. Все необходимые файлы и модули (src/data_loader.py, src/damage_estimator.py, src/risk_model.py, src/risk_aggregator.py, src/recommendations.py) создаются автоматически прямо в облачном окружении.

Перейдите по ссылке на блокнот Google Colab (https://colab.research.google.com/drive/1r-q4aZSkAop_up1bRu3qsvwV0b0xZPsN?usp=sharing). Для доступа может запросить авторизацию в Google-аккаунт.

Ознакомьтесь с описанием и инструкцией в начале блокнота.

Последовательно запускайте ячейки в блокноте. Для этого нажимайте кнопку воспроизведения слева от каждой блока или используйте комбинацию клавиш Shift + Enter. Следующие ячейки можно запускать до того, как завершатся предыдущие, так как их выполнение выстраивается в очередь. Их расположение представлено следующим образом:

- Ячейка 1 (Установка библиотек) – запустите и дождитесь завершения установки всех зависимостей (pandas, numpy, scikit-learn, plotly, ipywidgets, fuzzytree и др.). Это занимает несколько минут и выполняется один раз.
- Ячейка 2 (Структура каталогов) – создает папки «src/», «data/», «models/», «notebooks/».
- Ячейки 3-7 (Создание модулей) – записывают исходный код Python-модулей (data_loader.py, damage_estimator.py, risk_model.py, risk_aggregator.py, recommendations.py), предназначенных для работы проекта, в папку «src/».
- Ячейка 8 (Консольный скрипт) – создает «run_pipeline.py» для запуска анализа в консоли без интерфейса.

- Ячейка 9 (Интерфейс и README) – создает файл «interface.ipynb» для локального запуска проекта через Jupyter или VS Code, README.txt с описанием и requirements.txt для установления зависимостей.
- Ячейка 10 (Архив) – упаковывает все созданные файлы в «project_files.zip», после чего происходит его скачивание.
- Ячейка 11 (Тестовый запуск) – запускает pipeline и создает пример отчета «risk_report.csv».
- Ячейка 12 (Запуск интерфейса) – запускает интерактивный веб-интерфейс прямо в Google Colab.

Обратите внимание, что ячейки необходимо запускать строго по порядку, сверху-вниз, не пропуская ни одну из них (за исключением ячеек с упаковкой в архив и запуском pipeline, которые можно пропустить).

После запуска 12-й ячейки ниже появится интерактивная панель:

Вкладка «Данные» появляется в начале работы проекта. Выберите режим для работы с данными: «Сгенерировать» (позволяет определить сценарий (низкие риски, сбалансированный или высокие риски) и количество записей датасета, которые будут созданы случайным образом для дальнейшего анализа) или «Загрузить файл» (позволяет выбрать и загрузить собственный датасет в форматах «.csv» или «.xlsx»). Доступен предпросмотр записей датасета. Далее необходимо нажать на кнопку загрузки.

Следующим этапом выступает вкладка «Анализ». Настройте количество нечетких деревьев в ансамбле (от 10 до 100) и выберите режим экспертной оценки ущерба («Числовой» с ползунками весов или «Лингвистический» с выпадающими списками). Ущерб разделен по категориям (финансовый, репутационный, операционный, юридический, потеря данных), которые позволяют определить, какие риски важнее всего учитывать. В конце нажмите кнопку «Запустить анализ». Ожидайте конца анализа, в это время над полосой прогресса будут сменяться сообщения о статусе (например, «Обучение деревьев (15/30)...» или «Экспертная оценка ущерба...»). Дождитесь, пока шкала не заполнится на 100% и не появится сообщение «Анализ завершен!». Не закрывайте вкладку и не обновляйте страницу, чтобы не прервать процесс.

После завершения анализа автоматически произойдет переход на страницу результатов. На ней представлены метрики модели (Accuracy, F1, AUC), таблица результатов с рекомендациями по обработке рисков и кнопки для скачивания отчета в форматах CSV, HTML или PDF.

Для того, чтобы перейти к визуализации результатов нажмите на кнопку «Показать визуализацию». На странице будут представлены 11 интерактивных графиков с различной информацией и суррогатное дерево, которое представляет собой обычное дерево решений, обученное на данных для анализа, чтобы продемонстрировать принцип работы модели. Доступна фильтрация (по отделам, типу событий и критичности риска) и скачивание всех графиков в ZIP-архиве.

Для перехода между страницами используйте кнопки навигации или переключение между вкладками. Скачать отчеты и датасеты в различных форматах можно при помощи соответствующих кнопок.

2 Работа локально в Jupyter или VS Code

Данный способ подходит для работы без интернета и для глубокого изучения кода. Он требует установки необходимых компонентов и зависимостей.

Скачайте архив «project_files.zip», созданный в ячейке 10 блокнота Colab (см. п. 1 инструкции), и распакуйте его в любой папке на вашем компьютере.

Убедитесь, что у вас установлен Python (версии 3.10 или новее), одна из двух сред разработки: VS Code с установленным расширением «Jupyter» или Jupyter Notebook. Откройте терминал (командную строку) или PowerShell, для этого воспользуйтесь поисковой строкой или комбинацией Win + R и напишите в появившемся окне «cmd» или «powershell» (при использовании ОС Windows).

Перейдите в папку проекта через команду «cd <путь к проекту>\ project_files». Рекомендуется создать виртуальное окружение для проекта, написав в терминале команду: «python -m venv venv». Далее необходимо активировать его:

- для Windows – «venv\Scripts\activate».
- для macOS/Linux – «source venv/bin/activate».

Установите все необходимые зависимости, для этого введите в терминале команду: «pip install -r requirements.txt». Дождитесь конца установки.

Примечание: для экспорта PDF-отчетов библиотеке «WeasyPrint» на Windows могут потребоваться системные библиотеки «GTK3». Если данные библиотеки отсутствуют и PDF не формируется, воспользуйтесь выгрузкой в HTML или CSV.

Запустите проект в одном из двух вариантов:

- С интерфейсом – откройте файл «interface.ipynb» в Jupyter Notebook или VS Code. Выполните ячейку (Run All). Появится тот же интерактивный интерфейс, что и в Colab (см. в п. 1 результат запуска 12-й ячейки).
- Через консоль – если вам нужен только текстовый результат без графического интерфейса, выполните в терминале команду: «python run_pipeline.py». Результаты (отчет «risk_report.csv») сохранятся в папку data/.

Обратите внимание, что все файлы сохраняются в папку «data/».

3 Работа через веб-приложение

Веб-приложение было разработано по инициативе автора и куратора от образовательной организации в дополнение к конечным результатам конкурсного проекта, которые были указаны в заявке. Оно призвано обеспечить

простоту использования, наглядность и доступность разработанного решения. Это полноценное клиент-серверное приложение, разделенное на бэкенд (Python, FastAPI) и фронтенд (React, Redux) с использованием необходимых для них библиотек. Оно обеспечивает максимальную производительность и удобство работы.

Для локального запуска проекта скачайте его с GitHub по ссылке (https://github.com/KlonChannel/ISrisk-analyzer) убедитесь, что на компьютере установлены Python 3.10 или новее и Node.js LTS (18.x или 20.x). При первом запуске потребуется Интернет-соединение, чтобы установить Python- и Node-зависимости. В будущем приложение будет работать локально.

Для запуска в «один клик» откройте папку с проектом и запустите файл «start.bat». Откроется командное окно, которое при первом запуске автоматически создаст виртуальное окружение, установит все Python- и Node-зависимости (займет 5-10 минут). После этого и во все следующие запуски будут запускаться два окна для запуска фронтенда и бэкенда. Далее автоматически откроется браузер по адресу «http://localhost:3000».

Если «start.bat» не работает, то запуск необходимо произвести вручную. Отройте два терминальных окна (см. п. 2 инструкции), и последовательно введите следующие команды:

Первое окно (Бэкенд):

- «cd backend».
- «venv\Scripts\activate».
- « uvicorn main:app --reload --port 8000 ».

Второе окно (Фронтенд):

- «cd frontend»;
- «npm start».

Если после успешного запуска страница в браузере не открылась автоматически, то перейдите по адресу: «http://localhost:3000».

Для остановки приложения закройте все терминальные окна (или нажмите Ctrl + C в каждом из них).

В случае запуска через ресурс, предоставленный автором, перейдите по ссылке «https://isrisk.dypster.ru». После чего откроется приложение.

Интерфейс веб-приложения разделен на 4 логические вкладки, переключение между которыми возможно с помощью меню навигации вверху:

«Данные» – текущая сессия создается автоматически. Вы можете сгенерировать синтетический датасет (выбрав сценарий и количество записей) или загрузить свой файл (CSV/XLSX). Здесь же доступен предпросмотр таблицы с пагинацией и скачивание датасета.

«Анализ» – задайте количество нечетких деревьев в ансамбле. Для экспертного задания ущерба в режиме «Числовой» настройте веса ущерба по категориям ползунками, в «Лингвистическом» – выберите оценки ущерба по категориям из списков. Нажмите «Запустить анализ» и следите за прогресс-баром.

«Результаты» – отображаются метрики качества (Accuracy, F1, AUC), веса ущерба и таблица рекомендаций по обработке рисков с пагинацией. Внизу доступны кнопки скачивания отчета (CSV, HTML, PDF).

«Визуализация» – отображает 11 аналитических графиков и суррогатное дерево. С помощью фильтров можно отбирать данные по отделам, типам событий и уровням риска. Есть кнопка скачивания всех графиков в ZIP-архиве.

Выполнение этапов представлено с помощью прогресс-бара под меню навигации. После выполнения этапа внизу страницы появляется сообщение о возможности перехода на следующую страницу.

Для сброса результатов анализа используйте кнопку «Сбросить» на вкладке анализа или кнопку со схожим названием на вкладке данных. Также это действие можно произвести при помощи кнопки «Новая сессия», расположенный в правом верхнем углу страницы. К тому же данная кнопка генерирует новый идентификатор текущей сессии.

Файлы отчетов, загруженных и выгруженных датасетов сохраняются на сервере в папки «uploads/» и «reports/».

Обратите внимание: хранилище сессий находится в оперативной памяти, поэтому при перезапуске сервера все текущие результаты будут сброшены.

Примечание: для экспорта PDF-отчетов библиотеке «WeasyPrint» на Windows могут потребоваться системные библиотеки «GTK3». Если данные библиотеки отсутствуют и PDF не формируется, воспользуйтесь выгрузкой в HTML или CSV.

Возможные проблемы и их решение:

- Ошибка «python не является внутренней или внешней командой» – при установке Python не была поставлена галочка «Add Python to PATH». Переустановите Python, отметив эту опцию.
- Ошибка «npm не является внутренней или внешней командой» – установите Node.js и перезагрузите компьютер.
- Ошибка allowedHosts[0] should be a non-empty string при запуске фронтенда без интернета – создайте в корне папки «frontend» файл «.env» и добавьте в него строку «DANGEROUSLY_DISABLE_HOST_CHECK=true», затем перезапустите «npm start».
- Ошибка при загрузке датасета – убедитесь, что в вашей таблице (CSV или XLSX) присутствуют следующие обязательные столбцы: «timestamp»,

«department», «event_type», «cvss_score», «asset_criticality», «hour_of_day», «anomaly_frequency», «access_attempts», «network_load_mbps», «prev_incidents_week», «incident_flag».