

Описание функциональных характеристик программного обеспечения

Наименование: NafanyaClassifierAI — Классификатор строительных ресурсов

Версия: 1.0 (production)

Правообладатель: Снегирёв Николай Валерьевич, гражданин Российской Федерации

Основание исключительного права: собственная разработка (ст. 1257 ГК РФ)

Дата документа: 02.07.2026

1. Назначение

Программное обеспечение (далее — ПО) предназначено для автоматической классификации строительных ресурсов по текстовому описанию. Система сопоставляет пользовательский запрос с элементами загруженной базы ресурсов (классификатор строительных ресурсов, КСР) с использованием методов машинного обучения на основе нейросетевых языковых моделей (BERT) и векторного поиска по косинусной близости.

ПО применяется в сфере строительства, сметного дела и нормативно-справочной информации для:

- быстрого подбора группы или номенклатуры ресурса по свободному текстовому описанию;
- пакетной обработки списков ресурсов из файлов Excel/CSV;
- поиска ресурсов по кодам;
- администрирования базы классификатора.

2. Область применения

- организации, ведущие сметные расчёты и нормативную базу строительных ресурсов;
- внутренние и корпоративные информационные системы в строительной отрасли;
- задачи импортозамещения зарубежных решений классификации и семантического поиска по отраслевым справочникам.

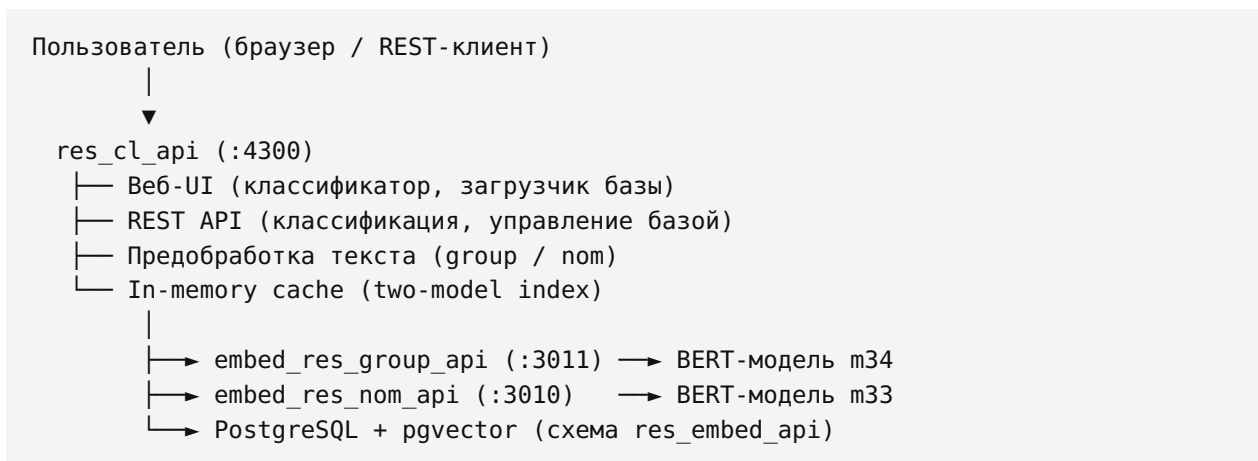
3. Состав программного комплекса

ПО реализовано как распределённый программный комплекс из трёх взаимосвязанных сервисов:

Компонент	Назначение	Порт по умолчанию
res_cl_api	Основной сервис: веб-интерфейс, REST API, загрузка базы, классификация	4300
embed_res_nom_api	Сервис инференса нейросетевой модели для номенклатур (модель m33)	3010
embed_res_group_api	Сервис инференса нейросетевой модели для групп (модель m34)	3011

Дополнительно используется СУБД **PostgreSQL** с расширением **pgvector** для хранения метаданных базы и векторных представлений.

4. Архитектура



4.1. Алгоритм классификации

Режим «Искать группу»:

1. Предобработка текста запроса в режиме `group`.
2. Получение векторного представления (эмбединга) через `embed_res_group_api`.
3. Сравнение с эмбедингами всех строк базы по косинусной близости.
4. Для каждой группы выбирается максимальная близость среди строк группы.
5. Возвращается список top-K групп с оценкой релевантности (score).

Режим «Искать номенклатуру» (двухшаговый поиск):

1. Предобработка запроса в режимах `group` и `nom`.
2. Получение двух эмбедингов (`group` и `nom`).
3. По `group`-эмбеддингу выбираются top-K групп.
4. Внутри выбранных групп по `nom`-эмбеддингу ранжируются строки базы.

5. Возвращается список top-K номенклатур с метаданными (код, наименование, единица измерения, группа, score).

4.2. Хранение и поиск

- База ресурсов хранится в PostgreSQL (таблица `resource_base_rows`).
- Для каждой строки сохраняются метаданные КСР, подготовленные тексты и векторы.
- Основной поиск выполняется через in-memory кэш нормализованных векторов (`float32`, `numpy`).
- В БД создаются HNSW-индексы по векторным полям для резервного и перспективного использования.

5. Функциональные возможности

5.1. Веб-интерфейс

ПО предоставляет два раздела с русскоязычным интерфейсом:

URL	Раздел	Функции
GET /	Классификатор	Ввод запроса, выбор режима, параметры top-K, отображение результатов, статус базы
GET / base	Загрузчик базы	Загрузка базы из файла по умолчанию или пользовательского файла, очистка базы, мониторинг фоновой обработки

Классификация доступна только при статусе базы `ready`.

5.2. REST API

Проверка работоспособности

Метод	Путь	Описание
GET	/health	Состояние сервиса

Управление базой

Метод	Путь	Описание
GET	/api/base/status	Статус текущей базы
POST	/api/base/load-default	Загрузка базы из файла по умолчанию
POST	/api/base/upload	Загрузка пользовательского файла (xlsx, xls, csv, json)

Метод	Путь	Описание
POST	/api/base/clear	Полная очистка базы

Классификация

Метод	Путь	Описание
POST	/api/classify	Одиночная классификация
POST	/api/classify/batch	Пакетная классификация (Excel/CSV, до 10 000 строк)
GET	/api/classify/template	Шаблон файла для пакетной обработки
GET	/api/classify/ksr	Скачивание файла классификатора КСР
POST	/api/classify/export-single	Экспорт результата одиночной классификации в Excel

Ресурсы

Метод	Путь	Описание
POST	/api/resources/by-codes	Поиск ресурсов по кодам

5.4. Загрузка и обработка базы

- Поддерживаемые форматы: `xlsx`, `xls`, `csv`, `json`.
- Обязательные колонки: `book_code`, `book_name`, `chapter_code`, `chapter_name`, `section_code`, `section_name`, `group_id`, `group_code`, `group_name`, `nom_id`, `nom_code`, `nom_name`, `res_id`, `res_code`, `res_name`, `unit`.
- Загрузка выполняется в фоновом режиме батчами (размер батча по умолчанию: 1024 строки).
- При загрузке новой базы предыдущая очищается автоматически.
- Строки, не прошедшие валидацию на стороне embed-сервисов (`problematic_indices`), исключаются и логируются.

5.5. Предобработка текста

Для одного исходного наименования ресурса формируются два варианта текста:

- `group` — для поиска группы (словарь замен `changes_group.xlsx`);
- `nom` — для поиска номенклатуры (словарь замен `changes_nom.xlsx`).

Предобработка унифицирует текст перед подачей в нейросетевые модели.

6. Технические характеристики

Параметр	Значение
Язык реализации	Python 3.12
Веб-фреймворк	FastAPI
СУБД	PostgreSQL 14+ с расширением pgvector
Размерность эмбединга	768
Максимальная длина запроса	10 000 символов
Максимальный размер пакетной обработки	10 000 строк
Рекомендуемое число worker-процессов	1 (in-memory cache)
Язык интерфейса	Русский

6.1. Системные требования (минимальные)

Компонент	Требование
ОС	Linux (Debian, Ubuntu, Astra Linux, РЕД ОС и аналоги)
CPU	4 ядра
RAM	16 ГБ (32 ГБ рекомендуется при GPU-инференсе)
GPU	Опционально (NVIDIA CUDA для ускорения embed-сервисов)
Диск	20 ГБ свободного места (модели BERT, база, логи)
PostgreSQL	14+ с расширением vector
Сеть	Локальная сеть между сервисами (порты 3010, 3011, 4300)

7. Ограничения

- Для работы классификации необходима предварительная успешная загрузка базы.
- Embed-сервисы должны быть доступны на момент загрузки базы и классификации.

8. Признак отнесения к сфере искусственного интеллекта

ПО использует:

- дообученные трансформерные языковые модели семейства BERT;
- векторные представления текстов (эмбединги);

- семантический поиск по косинусной близости.

9. Безопасность

- ПО не содержит принудительного обновления с зарубежных серверов.
- Управление обновлениями осуществляется правообладателем на серверах в РФ.
- Лицензионные ключи для активации не используются.
- ПО не предназначено для защиты конфиденциальной информации и не требует сертификации ФСТЭК.