Анализ публикуемых новостей

создание ETL-процесса формирования витрин данных
для анализа публикаций новостей

Общее описание проекта

Общая задача: создать ETL-процесс формирования витрин данных для анализа публикаций новостей с трех информационных источников:

- https://lenta.ru/rss/
- https://www.vedomosti.ru/rss/news
- https://tass.ru/rss/v2.xml

Подробное описание задачи:

1. Разработать скрипты загрузки данных в 2-х режимах:
    - Инициализирующий – загрузка полного слепка данных источника
    - Инкрементальный – загрузка дельты данных за прошедшие сутки
2. Организовать правильную структуру хранения данных
    - Сырой слой данных
    - Промежуточный слой
    - Слой витрин

Цели проекта и требуемый результат

В качестве результата работы программного продукта необходимо написать скрипт, который формирует витрину данных следующего содержания:

- Суррогатный ключ категории
- Название категории
- Общее количество новостей из всех источников по данной категории за все время
- Количество новостей данной категории для каждого из источников за все время
- Общее количество новостей из всех источников по данной категории за последние сутки
- Количество новостей данной категории для каждого из источников за последние сутки
- Среднее количество публикаций по данной категории в сутки
- День, в который было сделано максимальное количество публикаций по данной категории
- Количество публикаций новостей данной категории по дням недели

Дополнение: необходимо привести названия категорий в разных источниках к единому виду.

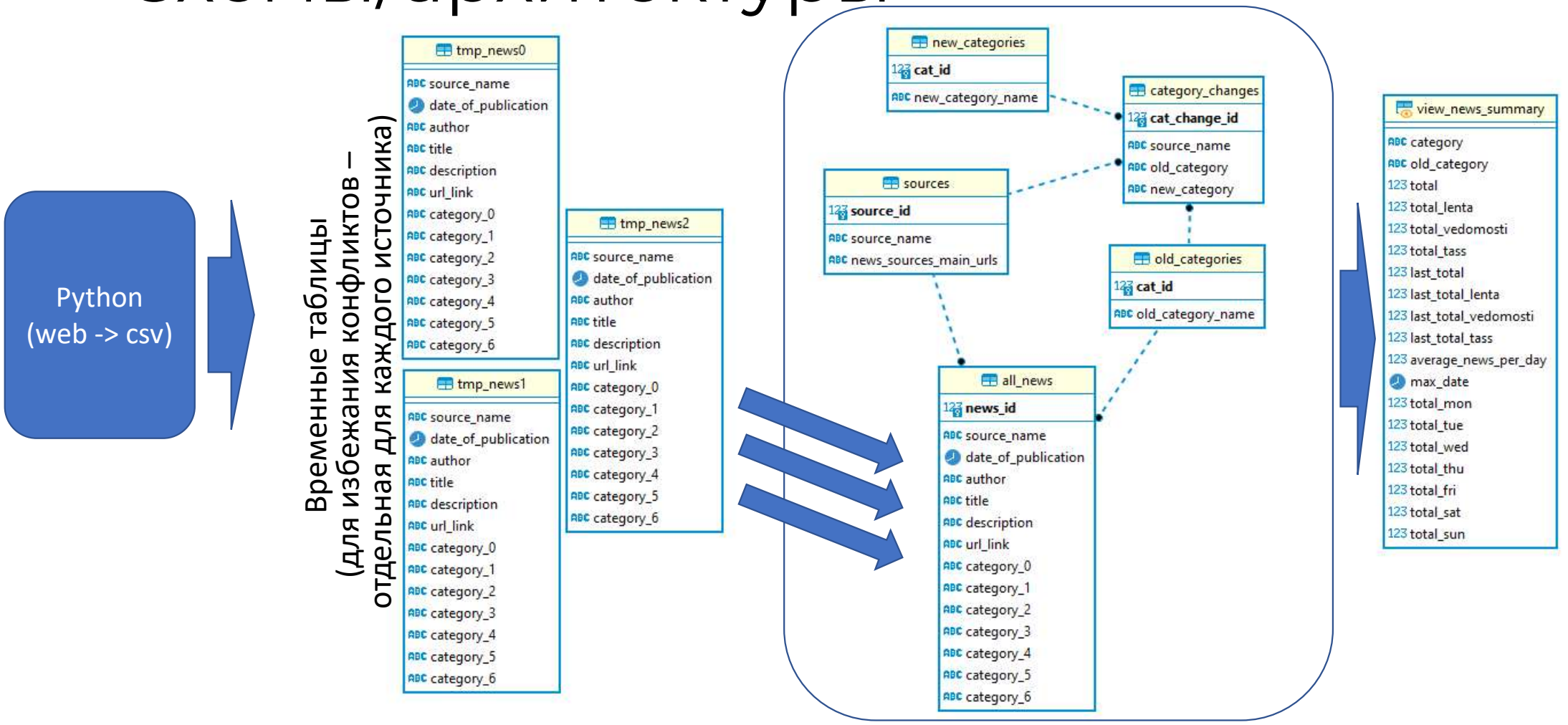
План реализации

Используемые технологии

- Docker

- Python
  - datetime, pytz – для работы с временными данными
  - pandas – обработка данных
  - sqlalchemy – работа с SQL
  - requests, bs4 – загрузка web страниц
  - os – получение списка загруженных файлов и времени последней новости из их названий
  - re – выражения для дополнительного разбора страниц и анализа списка файлов.
- PostgreSQL
  - Materialized view – для витрины данных
- Airflow
  - DAG – для их создания
  - Variable – для хранения названия подключения
  - PythonOperator – запуск сбора и обработки данных
  - PostgresHook – подключение к PostgreSQL внутри кода Python
  - PostgresOperator – выполнение обособленных действий с PostgreSQL

Схемы/архитектуры



Результаты разработки и выводы

На базе докер-контейнера airflow разработан программный продукт, позволяющий осуществлять сбор данных из трех новостных источников, сохранять их в виде .csv.gz файлов, загружать в PostgreSQL и формировать по расписанию витрину данных в виде материализованного представления.

В тестовом режиме разработана процедура предсказания категории (на базе kNN). В докере пока не удалось реализовать, т.к. по неизвестной причине sklearn не хочет подгружаться в докер (в отличие от других пакетов в requirements.txt)