**Документация, содержащая описание функциональных характеристик модуля программного обеспечения**

**Модуль «Тегизация»**

**г. Москва 2025**

**Содержание**

[1. Введение 3](#_heading=h.uxwsgqfuxmj)

[2. Назначение модуля 3](#_heading=h.5iufcs7m0ltw)

[3. Функциональные возможности 3](#_heading=h.l3gkj4f5h3f1)

[4. Архитектура и принцип работы 4](#_heading=h.5tu2rn18qtwc)

[6. Используемые языки программирования и фреймворки 5](#_heading=h.xc60nrr3rypl)

## **1. Введение**

Модуль "Тегизация" предназначен для автоматического формирования тематик контента на основе интересов пользователей. Он использует алгоритмы машинного обучения для анализа действий пользователей, обработки контента и формирования персонализированных тегов.

Данный модуль является частью системы персонализации контента и предназначен для автоматизации процесса категоризации видеоматериалов, что позволяет минимизировать ручной труд редакторов и повысить удовлетворенность пользователей.

## **2. Назначение модуля**

Модуль "Тегизация" выполняет следующие ключевые задачи:

* Автоматическая классификация контента по тематическим тегам.
* Формирование персонализированных рекомендаций на основе интересов пользователей.
* Интеграция с рекомендательной системой для отображения релевантного контента.
* Обеспечение механизма обновления тегов при изменении пользовательских предпочтений.

## **3. Функциональные возможности**

Основные функции

* Получение и обработка данных о пользователях и контенте.
* Анализ пользовательских предпочтений на основе истории взаимодействий.
* Генерация тегов и тематических кластеров.
* Формирование персонализированных подборок контента.
* API для интеграции с внешними сервисами и рекомендательными системами.
* Поддержка механизма "холодного старта" для новых пользователей.

Работа с тегами

* Генерация тегов на основе семантического анализа контента.
* Кластеризация тегов для упрощения поиска и навигации.
* Определение релевантных тегов для каждого пользователя.
* Динамическое обновление тегов с учетом изменений предпочтений пользователей.

Рекомендательный алгоритм

* Определение близости пользователя к тематическим кластерам.
* Анализ интересов пользователя по различным категориям контента.
* Генерация подборок с учетом истории взаимодействий и предпочтений

## **4. Архитектура и принцип работы**

Общая архитектура

Модуль "Тегизация" построен на микросервисной архитектуре и состоит из нескольких ключевых компонентов:

Сервис обработки данных – получает и обрабатывает данные о пользователях и контенте.

Модуль машинного обучения – выполняет анализ данных и формирует тематические кластеры.

API-сервис – предоставляет доступ к тегам и персонализированным рекомендациям.

Система хранения данных – включает базы данных PostgreSQL и ClickHouse для хранения тегов и истории взаимодействий.

Принцип работы

1. Получение данных из внешних источников (каталоги контента, данные пользователей).
2. Анализ контента и создание тегов на основе модели машинного обучения.
3. Определение пользовательских предпочтений и формирование персонализированных кластеров.
4. Передача данных в рекомендательную систему и генерация персонализированных подборок.
5. Обновление тегов с учетом изменений в поведении пользователей.

Потоки данных

Система получает данные от пользователей и из внешних источников.

Анализируются текстовые описания, метаданные и взаимодействия пользователей.

Генерируются тематические теги и кластеры.

Теги передаются в рекомендательную систему и используются для персонализации контента.

**5. Роли и полномочия**

Основные роли в системе

1. **Администратор** – управляет настройками модуля, редактирует данные, следит за корректностью работы системы.
2. **Пользователь** – получает персонализированные подборки контента на основе тегов.
3. **Редактор контента** – может вносить корректировки в тематические теги.

Полномочия ролей

**Администратор:** доступ ко всем функциям, настройке алгоритмов и мониторингу работы модуля.

**Редактор контента:** просмотр и правка тегов для улучшения качества категоризации.

**Пользователь:** просмотр тематических подборок без возможности редактирования тегов.

## **6. Используемые языки программирования и фреймворки**

Backend: Python (Flask, FastAPI), Java (Spring Boot)

Machine Learning: PyTorch, CatBoost, Scikit-learn

Data Processing: Apache Spark, Polars, PyArrow, DeltaLake

Database: PostgreSQL, ClickHouse

Infrastructure: Yandex Cloud, Kubernetes, Docker, Terraform