



Национальный  
исследовательский  
Томский  
государственный  
университет



# MVP Интеллектуальной системы распознавания товаров на полках

Христофорова Полина  
Студентка программы

# Проблематика

- **Ручной аудит** планogramмного соответствия занимает **12-16 часов на 1000 м<sup>2</sup>** торговой площади
- Еженедельные проверки **500+ магазинов** сети обходятся в **2.4 млн руб/мес** только **на** трудозатраты **мерчендайзера**
- **Ошибка** человеческого фактора достигает **32% при подсчете SKU** и OSA (Out-of-Stock Availability), что приводит к **потерям выручки до 7.8%** от плановых показателей.
- Современные полки содержат **45-68 товаров/погонный метр с перекрытием до 27%**, из-за чего классические подходы CV не решают задачу
- Высокая плотность (**50+ SKU**) и постоянные изменения ассортимента (**1800+ SKU/магазин**), требуют принципиально нового подхода на базе современных нейросетей и оптимизированных пайплайнов

---

<sup>1</sup>NielsenIQ Retail Execution Report 2025

<sup>2</sup>RetailTech Analytics: Computer Vision ROI Analysis, Q4 2025



# Цель и критерии успеха

Цель проекта: разработать гибкую систему распознавания товаров, способную адаптироваться к новым SKU и условиям освещения/расположения через механизмы дообучения.



## Обязательные критерии успеха MVP

- Целевая метрика  $F1 \geq 0,85$ ;
- Latency end-to-end (от request до response)  $\leq 30$  сек;
- JSON API валиден по установленной схеме

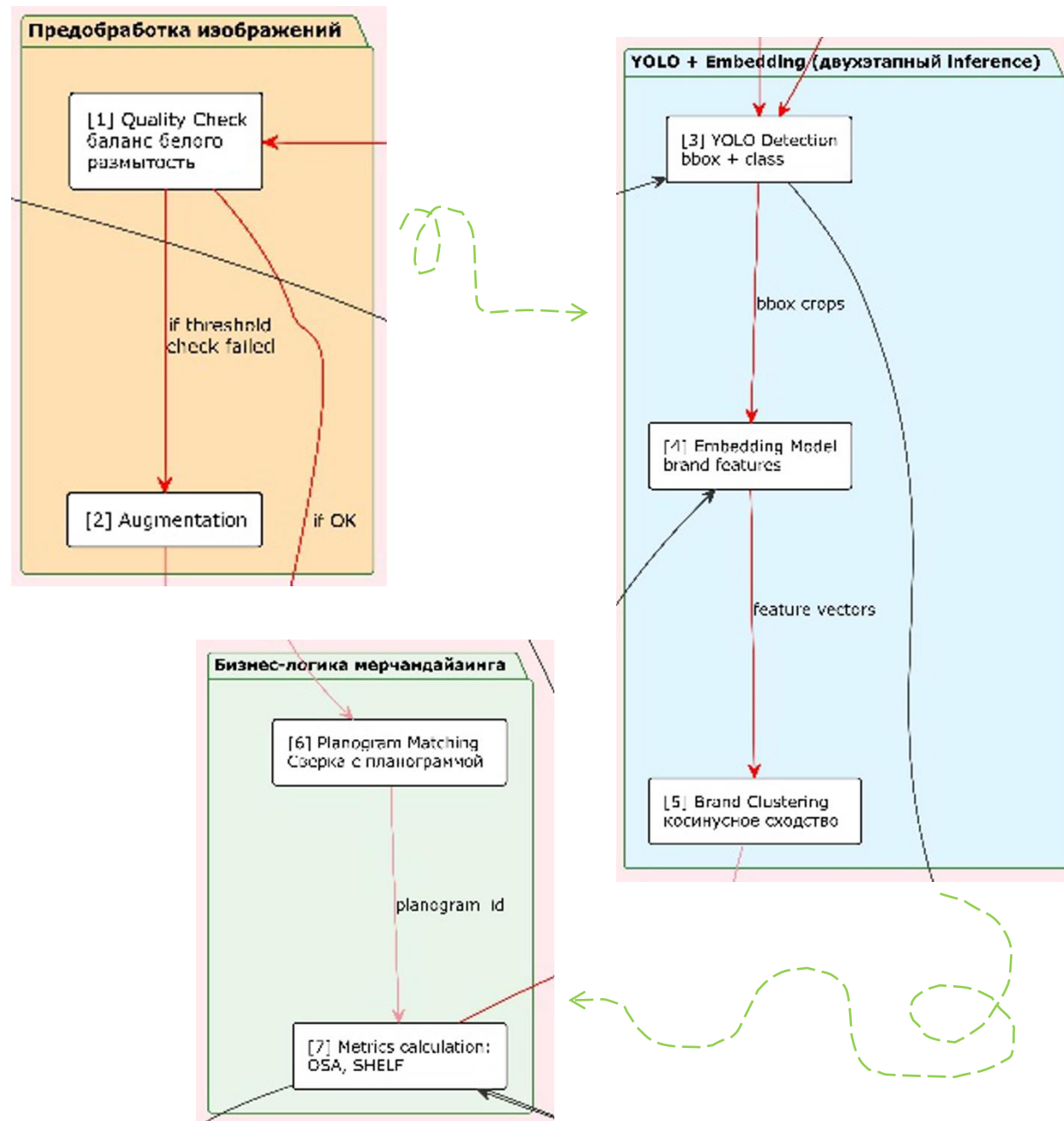


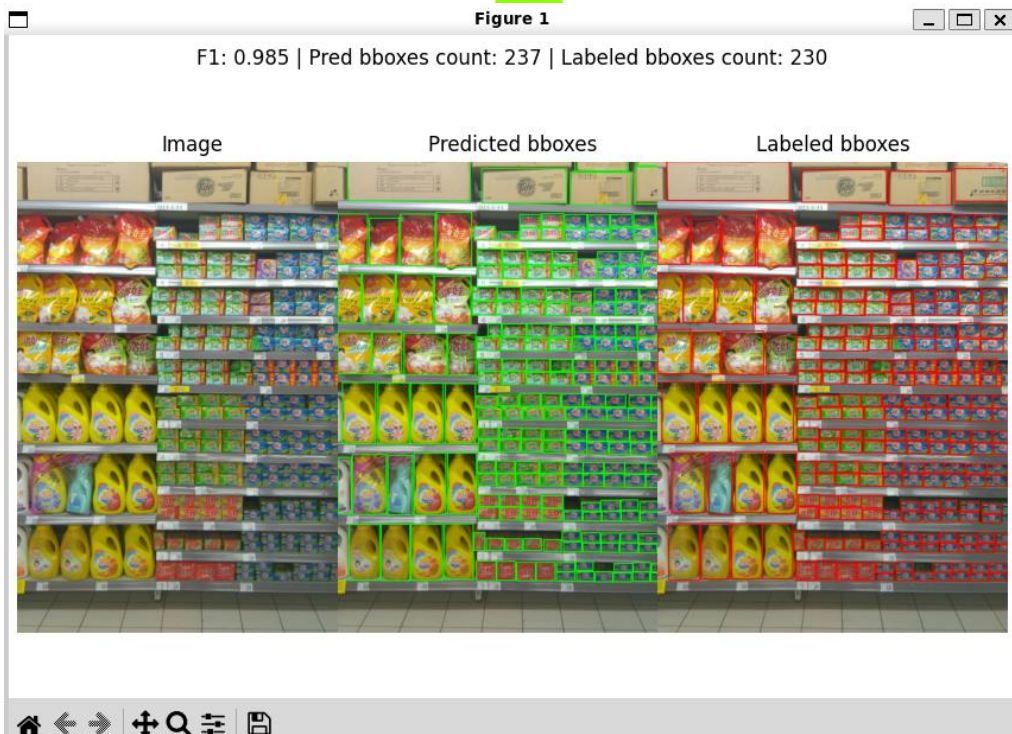
## Желательные критерии MVP

- Реализация пайплайна классификации задетекченных bbox'ов
- Гибкая реализация с возможностью дообучения

# Архитектура

- Архитектура всего сервиса разработана в формате микросервисной архитектуры: Клиенты → API → ML Pipeline → MLOps → хранилища данных
- Ее основное ядро – центральный слой ML-Pipeline. Именно он по большей части реализован в MVP





# Подходы к реализации

01

## Формирование датасета

Датасет на базе SKU110K:  
5000 предобработанных  
(orig/aug/dirty) с  
разбиением train/test/val  
3500/750/750

CPU Intel Xeon @ 2.00 GHz 31-32 GB

Kaggle

02

## Дообучение под детекцию

YOLOv8 дообучение на 16  
батчах и 20 эпохах с  
финальной валидацией на  
test выборке

GPU NVIDIA Tesla P100 PCIe 16GB

04

## Формирование ответа

По ожидаемой структуре  
содержит расчёт метрик  
IoU, precision, recall и f1,  
количество кластеров,  
информацию об SKU, а  
также визуализацию и логи

AMD Ryzen 7 32 GB

Local

03

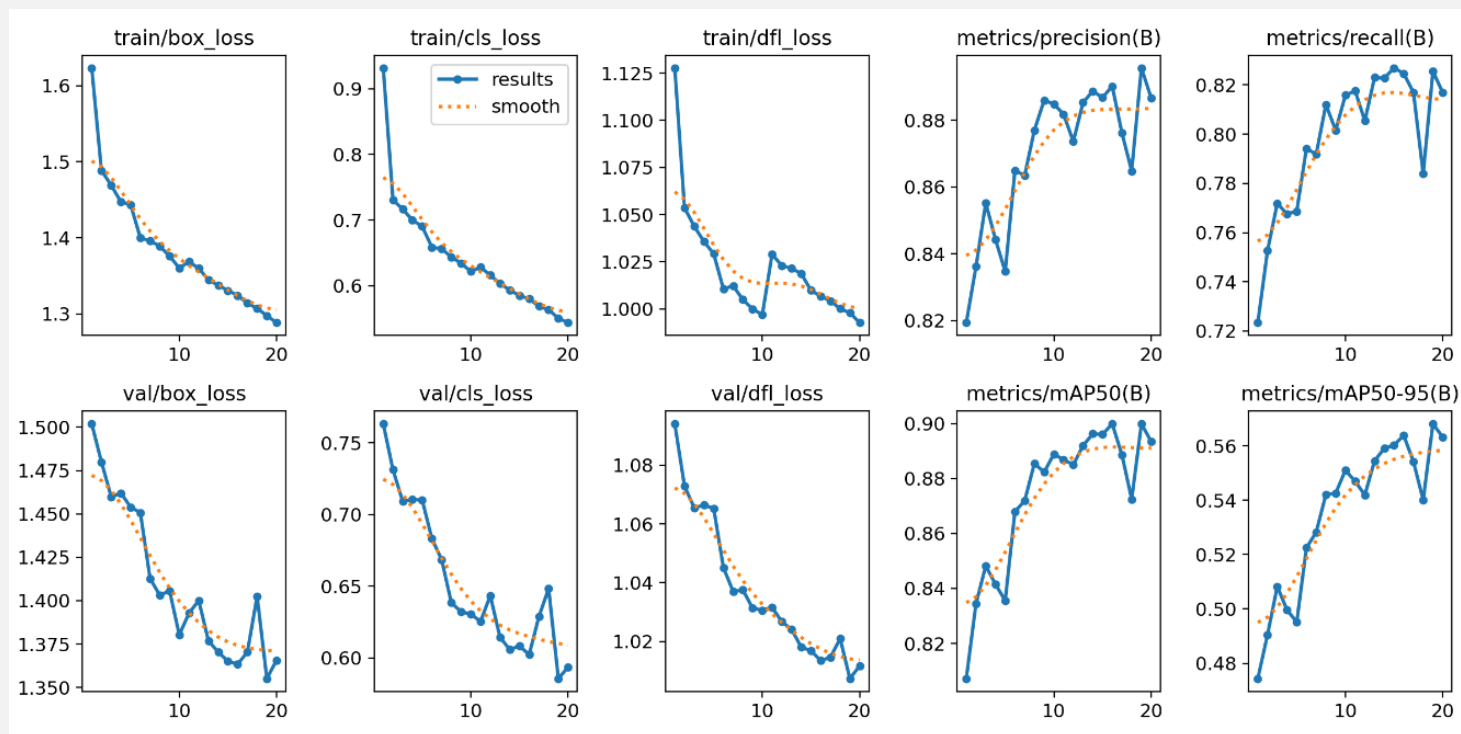
## Пайплайн кластеризации

HDBSCAN на эмбединги  
CLIP (версии ViT-B-326  
pretrain aion2b\_s34b\_b79k)  
с разбиением на  
определенные кластеры

NVIDIA GeForce RTX 5060 Laptop GPU 8 GB

```
***> Define analyzer
***> Start analyzing:
--> Get img and predict bboxes
--> Calculate metrics
--> Grouping in clusters
100% | 237/237 [00:02<00:00, 118.40it/s]
Found 4 clusters (+ True noise points)
***** Analyze completed *****
--> Exec time: 1.40s | 1403.11mls
--> Image path: data/examples/test_77.jpg
--> Results path: data/tmp_results/test_77_20260209023050
--> Response filepath: data/tmp_results/test_77_20260209023050/response.json
--> Metrics: {
  "true_positive": 230,
  "false_positives": 7,
  "false_negative": 0,
  "precision": 0.9704641350210971,
  "recall": 1.0,
  "f1": 0.9850107066381156,
  "detected_clusters": 4
}
RESPONSE 200: Success
```

# Ключевые результаты



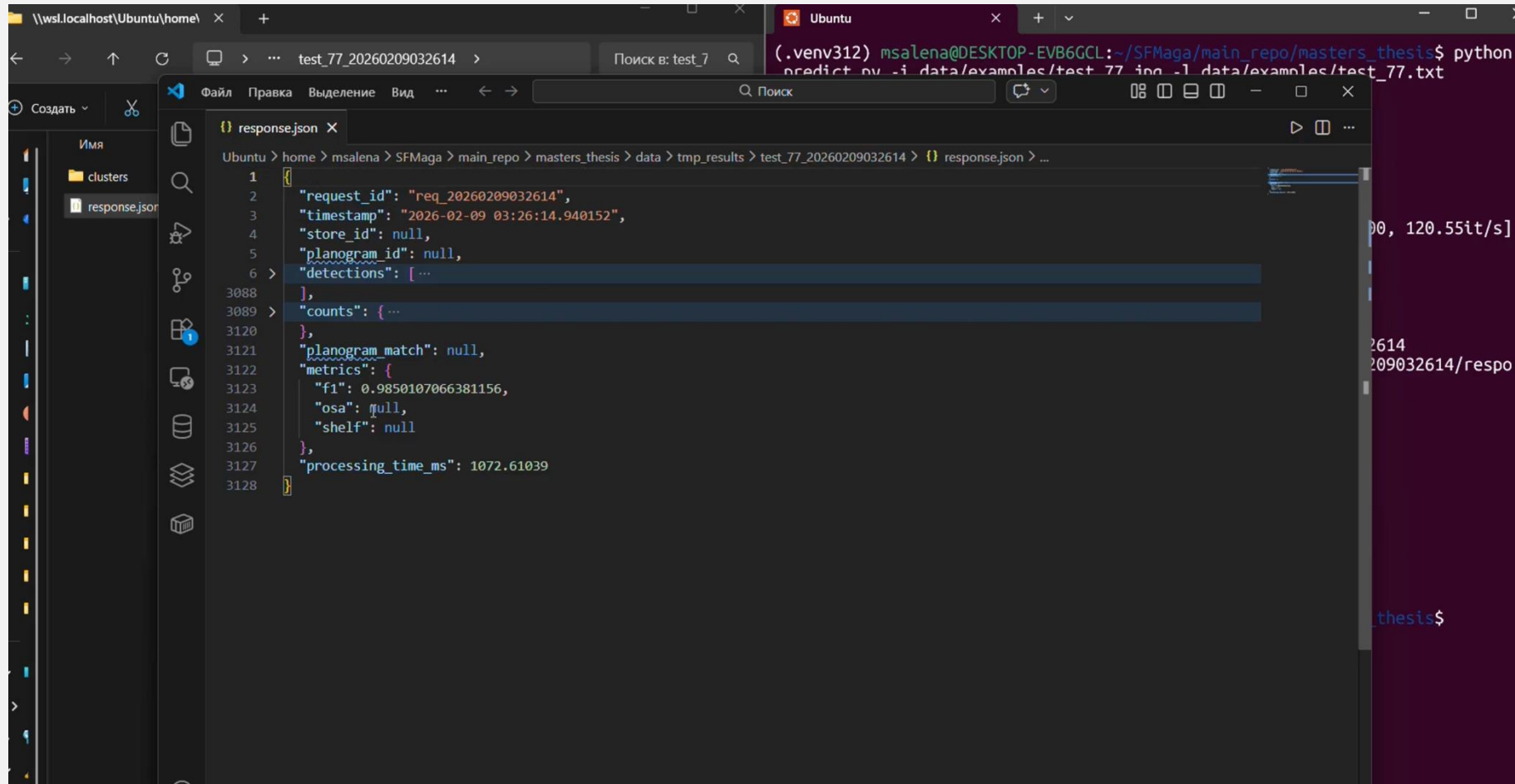
Validation set (20 эпох):

- F1-score: 0.859
- Precision: 0.8958
- Recall: 0.8258

Отложенная test выборка:

- F1-score: 0.868
- Precision: 0.9064
- Recall: 0.8336

# Демонстрация



The screenshot shows a development environment with three main components:

- File Explorer:** Located on the left, it shows a directory structure with folders like "clusters" and "response.json".
- Code Editor:** The central pane displays the content of "response.json". The JSON data includes fields for "request\_id", "timestamp", "store\_id", "planogram\_id", "detections", "counts", "planogram\_match", "metrics" (with "f1", "osa", and "shelf" values), and "processing\_time\_ms".
- Terminal:** On the right, it shows a command prompt session. The command executed is `python predict.py -i data/examples/test_77.jpg -l data/examples/test_77.txt`. The output shows a prediction result: `00, 120.55it/s]`. Below this, there is a line indicating the file path: `2614 209032614/respo`. The terminal prompt is `thesis:$`.

# Выводы и roadmap

## Итоги

- Среднее время анализа одного изображения до 2 секунд при разрешении 1280×720
- Метрики качества F1-score 0,834 свидетельствует о балансе между полнотой и точностью детекции.
- Особое внимание уделено обработке сложных случаев.
- Решение дополнено прототипом будущей брендовой классификации – кластеризацией алгоритмом HDBSCAN на эмбедингах CLIP
- Модульная архитектура

## Развитие проекта

- Доработка архитектуры
- Усовершенствование детекции и классификации в разрезе брендов и планограмм
- Возможный учет глубины полочного пространства
- Разработка системы активного дообучения на новых линейках и контрагентах.
- REST API





Национальный  
исследовательский  
Томский  
государственный  
университет



# Спасибо за внимание!

**Христофорова Полина**  
Студентка программы