

9. Talalakina E. Developing and Validating an Academic Vocabulary List in Russian: A Computational Approach / E. Talalakina, D. Stukal, M. Kamrotov. — Text : direct // The Modern Language Journal. — 2020. — Vol. 104, № 3. — P. 618-646.

*Д. К. ЗИТЦЕР, А. Г. ИВАШКО*

*Тюменский государственный университет, г. Тюмень*

**УДК 004.912**

## **ПОДСЧЕТ ПОСЕТИТЕЛЕЙ В РИТЕЙЛЕ НА ОСНОВЕ МОДЕЛИ ДЕТЕКЦИИ И АЛГОРИТМА ОТСЛЕЖИВАНИЯ**

***Аннотация.** В работе представлен процесс разработки программного обеспечения для подсчета посетителей в ритейле. Подробно описаны процессы подготовки данных и обучения модели детекции.*

***Ключевые слова:** подсчет посетителей, компьютерное зрение, машинное зрение, нейронные сети, детекция объектов, трекинг объектов.*

**Введение.** Высокий уровень конкуренции в российском ритейле ставит управленцев перед выбором: как улучшать обслуживание клиентов, отслеживать максимум показателей магазина — при этом не разориться на дорогостоящих IT-продуктах. В данной работе будет представлен простой, с точки зрения разработки, способ подсчета посетителей торговой точки.

*Зачем считать посетителей?*

Подсчет посетителей в торговой точке необходим для того, чтобы:

- 1) корректировать график работы персонала по времени и дням недели;
- 2) выстраивать систему мотивации персонала на основе коэффициента конверсии для улучшения качества обслуживания;
- 3) планировать и проводить маркетинговые акции, оценивать их результаты и корректировать бюджет.

*Существующие способы подсчета*

*Инфракрасные счетчики (табл. 1):*

- технология 2000-х гг.;
- наиболее распространенная система;
- установка по бокам от входной группы.

Таблица 1

**Плюсы и минусы инфракрасных счетчиков**

<i>Плюсы</i>	<i>Минусы</i>
<ul style="list-style-type: none"> <li>• компактность</li> <li>• низкая стоимость</li> <li>• не требовательны к искусственному освещению</li> </ul>	<ul style="list-style-type: none"> <li>• высокая чувствительность к среде</li> <li>• погрешность пропорциональна ширине входа</li> <li>• существенное влияние «человеческого фактора»</li> </ul>

**3D (стерео) счетчик (табл. 2):**

- установка над входом;
- обработка видеопотока;
- анализ изменения изображения.

Таблица 2

**Плюсы и минусы 3D-счетчиков**

<i>Плюсы</i>	<i>Минусы</i>
<ul style="list-style-type: none"> <li>• стабильность работы</li> <li>• отсутствие нагрузки на интернет трафик</li> <li>• высокая точность</li> </ul>	<ul style="list-style-type: none"> <li>• высокая стоимость камеры (~ 30 000 руб.)</li> <li>• отсутствие архива видео</li> </ul>

**Подсчет на основе распознавания лиц (табл. 3):**

- современная технология;
- специализированный софт обработки видеопотока;
- установка камеры напротив входа;
- хранение видеопотока.

Таблица 3

**Плюсы и минусы подсчета с распознаванием лиц**

<i>Плюсы</i>	<i>Минусы</i>
<ul style="list-style-type: none"> <li>• функциональность</li> <li>• разделение «сотрудник–посетитель»</li> <li>• подсчет уникальных посетителей</li> </ul>	<ul style="list-style-type: none"> <li>• высокая стоимость программного обеспечения</li> <li>• высокие требования к аппаратной части камеры</li> <li>• сложность запуска</li> <li>• погрешность при плотном графике</li> </ul>

### 2D-видеоаналитика (табл. 4):

- специализированный софт обработки видеопотока;
- установка над входом;
- запись видеопотока.

Таблица 4

#### Плюсы и минусы подсчета на основе видеоаналитики

<i>Плюсы</i>	<i>Минусы</i>
<ul style="list-style-type: none"><li>• функциональность</li><li>• разделение «сотрудник–посетитель»</li><li>• минимальные требования к аппаратной части камеры</li></ul>	<ul style="list-style-type: none"><li>• стоимость системы зависит от опыта разработчика</li></ul>

В данной работе будет рассмотрен процесс разработки программного обеспечения для 2D-видеоаналитики.

**Проблема исследования.** В 2D-видеоаналитике уже существуют готовые системы подсчета посетителей, работающие в ракурсе, когда камера установлена сбоку от проходящего трафика (рис. 1).

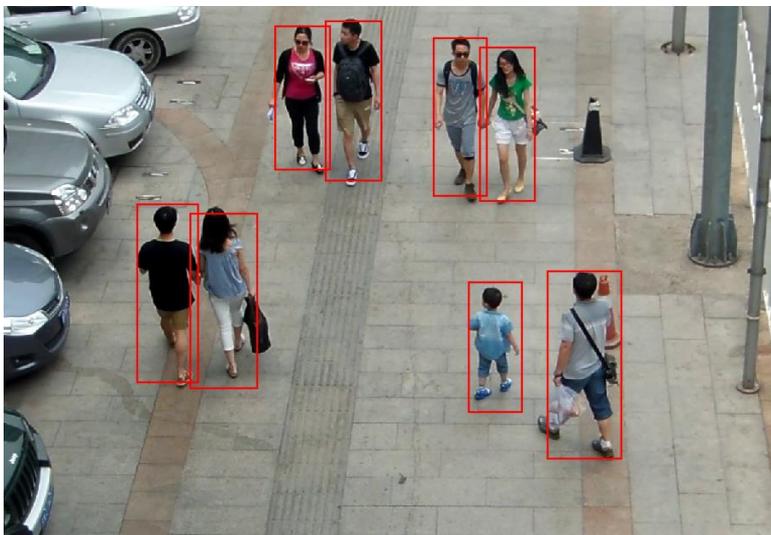


Рис. 1. Пример кадра с камеры подсчета уличного трафика

Такой ракурс съемки обладает рядом недостатков:

- погрешность будет пропорциональна плотности трафика, в толпе люди будут заслонять друг друга;
- невозможность единообразной установки камер в магазинах с разной планировкой входной группы.

По причинам выше, описанное в данной статье решение работает с расположением камеры над входом (рис. 2).



Рис. 2. Пример кадра с камеры подсчета над входом в магазин

Описание идеи подсчета:

- модель детекции выдает координаты прямоугольника (бокса) с человеком в кадре;
- зная положения боксов на последовательных кадрах, мы группируем их по принадлежности к тому или другому человеку;
- таким образом у каждого человека появляется история перемещений, по которой можно понять входит он или выходит.

**Материалы и методы.** Любая задача машинного обучения всегда состоит из двух этапов:

1. Сбор, обработка и подготовка обучающих входных данных для модели.
2. Обучение и тонкая настройка гиперпараметров модели.

В направлении компьютерного зрения уже существуют разнообразные готовые обучающие алгоритмы, которым необходимо лишь правильно подать на вход данные. Поэтому в большинстве задач машинного зрения основной упор приходится делать на работу с данными.

### *Сбор «сырых» видеоданных*

Для того, чтобы модель детекции могла находить на изображениях позицию человека в кадре, ее необходимо обучить. Такого рода обучение относится к типу задач обучения с учителем. В роли входных признаков будут выступать изображения, а роли отклика будут выступать координаты прямоугольников с человеком, которые необходимо разметить (рис. 3).

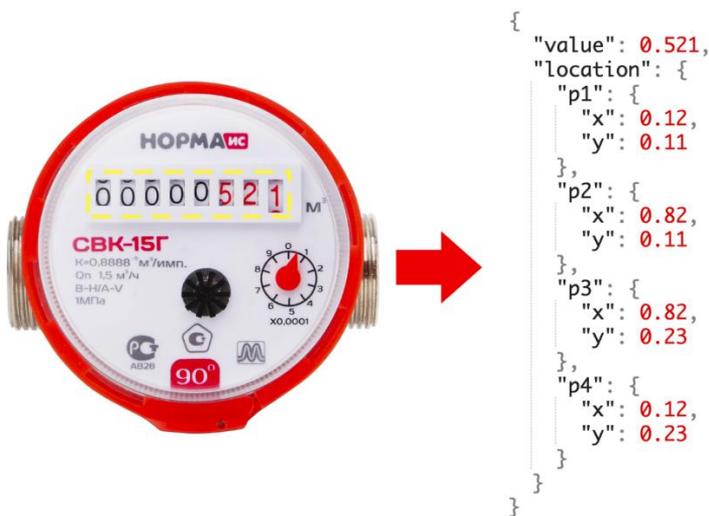


Рис. 3. Пример изображения и аннотации к нему

Процесс сбора первичных данных (видео) выглядел следующим образом:

- в 18 магазинах были установлены камеры подчета;
- с целью внесения разнообразия в обучающую выборку все камеры были разбиты на две группы: в первой группе были выставлены средние настройки качества, а во второй — высокие (рис. 4);

- с камер были получены 4-часовые фрагменты видео (по одному фрагменту с камеры), содержащие разное время суток (всего 73 GB видео).

Город	Название магазина	Разрешение	Частота кадров	Качество/Битрейт
Пермь	Солдатово	1280x720	25	Супер/2048
Пермь	Рязанская	1280x720	25	Супер/2048
Пермь	Верхнемулинская	1280x720	25	Супер/2048
Тюмень	Пафиловцев	1280x720	25	Супер/2048
Тюмень	Ожогово	1280x720	25	Супер/2048
Челябинск	Рылеева	1280x720	25	Супер/2048
Челябинск	Мопра	1280x720	25	Супер/2048
Москва	Молокова	1280x720	25	Супер/2048
Екатеринбург	Расточная	1280x720	25	Супер/2048
Москва	Химки (сходня)	1280x720	25	Среднее/1024
Пермь	Соликамская	1280x720	25	Среднее/1024
Тюмень	Рич	1280x720	25	Среднее/1024
Москва	Пушкино	1280x720	25	Среднее/1024
Екатеринбург	Пышма (Бажова)	1280x720	25	Среднее/1024
Тюмень	Республики	1280x720	25	Среднее/1024
Тюмень	Домостроителей	1280x720	25	Среднее/1024
Тюмень	Федюнинского	1280x720	25	Среднее/1024
Тюмень	Пермякова 2	1280x720	25	Среднее/1024
Тюмень	Жуковского	1280x720	25	Среднее/1024

Рис. 4. Конфигурации камер на момент начала сбора данных

Имея «сырые» видеоданные, необходимо трансформировать их размеченные кадры. Конвейер трансформации выглядит следующим образом:

- покадрово читаем каждое видео;
- избавляемся от статичных кадров: сохраняем только те, на которых есть какое-либо движение (подробнее в следующем разделе);
- создаем пулы разметки (пул разметки — это чаще всего несколько сотен кадров, помещенных в отдельную папку для удобства последующей разметки);
- разметка кадров из пула.

#### *Отбор кадров с движениями*

Этот шаг решает следующую проблему: в большей части записанных видео в кадре отсутствуют люди. В свою очередь для эффективной разметки требуются кадры, где с высокой долей вероятности они присутствуют. Поэтому для отсеивания статичных кадров было решено сравнивать последовательные соседние кадры друг с другом

и анализировать процент измененных пикселей, так как в нашем случае изменения могут быть вызваны только проходящими в кадре людьми (рис. 5).

```
for frame in video:
    # вырезаем область, в которой ожидаем посетителя
    curr_area = get_area(frame)
    # практика показывает, что лучше сравнивать слегка
    # размытые кадры
    curr_area = blur(curr_area)

    if percent_of_changes(prev_area, curr_area) > threshold:
        save_frame(frame)

    prev_area = curr_area
```

Рис. 5. Псевдокод отбора кадров с движением в конкретной области

Всего из видео было извлечено 303 654 кадра с движениями. Далее из них формировались пулы разметки. Один пул разметки — это случайная выборка без повторений 500 изображений для удобства последующей разметки.

#### *Разметка данных*

Существует 4 способа разметки изображений:

- предварительная разметка на основе существующих моделей детекции, либо с помощью классических алгоритмов, с последующей небольшой ручной корректировкой результатов работы алгоритма;
- in-house разметка — это выполнение разметки силами команды разработчиков;
- crowdsourcing — процесс привлечения большого количества внешних исполнителей к разметке данных за небольшую плату;
- выполнение разметки «под ключ» в, специализирующихся на этом, компаниях за существенную плату.

В данной работе опробованы первые три способа.

#### Автоматическая разметка с использованием библиотеки OpenCV

Идея данного подхода в следующем (рис. 6):

- попиксельно сравниваем последовательные соседние кадры;
- области изменения выделяем прямоугольными рамками.

К минусам такого подхода стоит отнести:

- большое количество ложных срабатываний;
- выделение любых движущихся объектов, теней.

К сожалению, от такого способа предразметки было бы больше вреда, нежели пользы. Пришлось бы потратить время на удаления ложных боксов и редактирование неточных.

```
prev_frame = video.get_frame()
prev_frame = blur(prev_frame)

for frame in video:
    curr_frame = blur(frame)
    diff = compare_frames(prev_frame, curr_frame)
    # помещаем области изменений в прямоугольные контуры
    contours = find_cointours(diff)

    for contour in contours:
        # фильтруем контуры малой площади
        if area(contour) > min_area_value:
            draw_rectangle(curr_frame, contour)

    prev_frame = curr_frame
```

Рис. 6. Псевдокод автоматической разметки

Результат работы представлен на рис. 7.

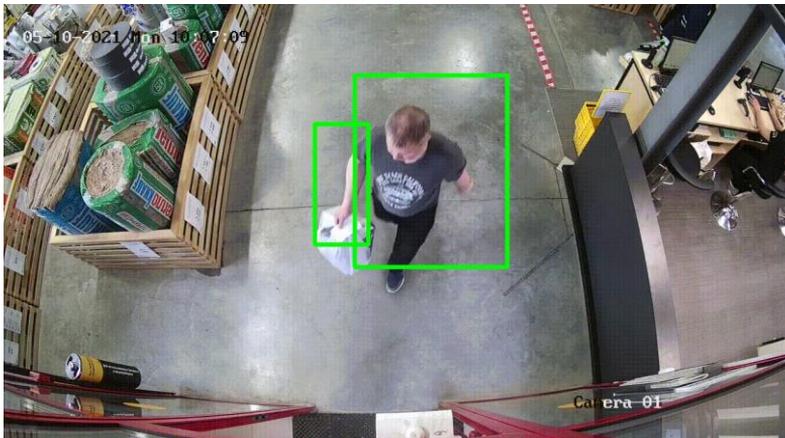


Рис. 7. Визуализация работы автоматической разметки

## Краудсорсинг, Яндекс.Толока

Напомню, что краудсорсинг — это привлечение большого количества внешних исполнителей для выполнения внутренних задач компании с применением информационных технологий. На Российском рынке представлена одна краудсорсинговая платформа, разработанная компанией Яндекс. Платформа называется Яндекс.Толока. Она позволяет заказчикам публиковать разного рода задачи, с которыми не справляются компьютеры, для широкого круга исполнителей. У исполнителей, выполняющих простые задания, появляется возможность заработка в интернете.

Был создан и запущен на платформе тестовый пул, состоящий из 500 изображений. Ниже приведу небольшую статистику процесса разметки исполнителями:

- 83 человека взялись за выполнения задания;
- 32 секунды — среднее время разметки одного изображения;
- 6 страниц в среднем разметил каждый исполнитель;
- 14 долларов было заплачено исполнителям;
- все 500 изображений были размечены за 11 минут.

На краудсорсинговых платформах часто можно встретить ботов — специальные программы, выполняющие автоматически случайную разметку изображений. Для борьбы с ними используется техника отложенной приемки — это когда исполнитель получает вознаграждение, только после принятия задания заказчиком. Интерфейс отложенной приемки оказался непродуманным:

- каждый раз, когда приходилось отклонять неудовлетворительную разметку ассессора, требовалось указать причину отклонения;
- из-за сетевых задержек на загрузка изображения уходило порядка 2 секунд, что на 500 изображениях превращалось в 16 минут простоя.

По причинам выше на проверку 500 изображений было потрачено время сопоставимое с временем самостоятельной разметки изображений. Дальнейший процесс разметки проводился силами команды.

Какой можно сделать вывод: если на изображениях в среднем будут находиться 3 и более объектов, то с точки зрения экономии времени эффективнее будет воспользоваться Толокой.

### in-house разметка

in-house разметка — это разметка изображений чаще всего силами команды разработки, реже — специально обученными людьми в штате компании. Существует разнообразие среди инструментов ручной разметки, ниже приведены наиболее популярные:

- CVAT;
- VGG Image Annotator;
- LabelImg.

В результате сравнения, выбор пал на LabelImg (рис. 8):

- простая установка;
- понятный интерфейс;
- удобные «горячие» клавиши.

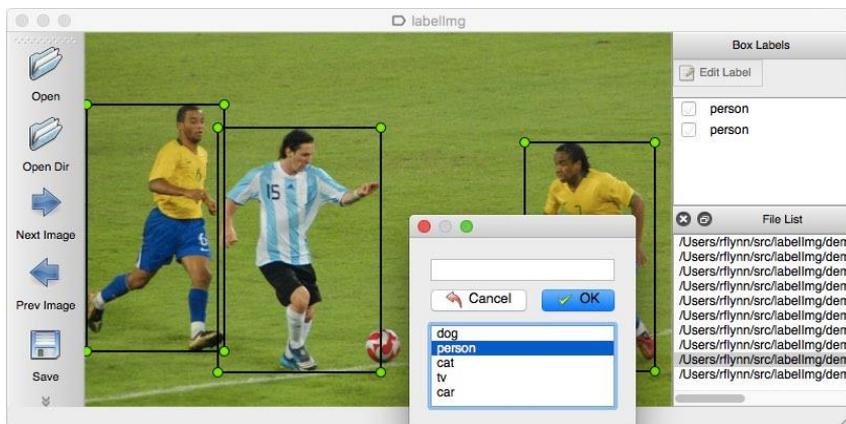


Рис. 8. Пример интерфейса программы LabelImg

Важно отметить, что ключевым моментом в скорости разметки стала декомпозиция задачи: на первой итерации все люди выделялись как посетители, а на второй — сотрудникам проставлялась правильная метка. Пример разметки представлен ниже (рис. 9).

В результате был получен набор данных с 5985 изображениями, на которых присутствуют:

- 9047 посетителей;
- 2453 сотрудника магазина.



Рис. 9. Пример выделения посетителя прямоугольной рамкой

### *Обучение модели*

Для обучения модели детекции посетителей и сотрудников было решено воспользоваться фреймворком с открытым исходным кодом TensorFlow Object Detection API. Он упрощает процесс создания и обучения моделей детекции, обладает подробной документацией, большим комьюнити и «зоопарком» готовых архитектур моделей, обученных на базе снимков ImageNet. Это очень важно, потому что сверточные нейронные сети, используемые в задачах компьютерного зрения, обладают переносимостью изученных признаков между разными задачами.

Из всего «зоопарка» моделей было выбрана модель `ssd_mobilenet_v2/fpnlite_320x320`, по причине того, что она может работать в real-time, то есть обрабатывать больше 25 кадров в секунду.

Обучение моделей заняло около трех часов и проходило на локальном компьютере со следующими характеристиками:

- CPU — AMD Ryzen 9 3900XT 12-Core Processor

- GPU — NVIDIA GeForce RTX 3070 8GB
- RAM — Kingston HyperX FURY 32GB

Для замера качества использовалась наиболее популярная метрика в задачах детекции объектов — mean average precision. Ниже представлено значение этой метрики на тестовом наборе данных (рис. 10).

На рис. 10 единицей закодирован класс «посетители», двойкой — класс «сотрудники».

Что касается производительности, то данная модель при запуске на графическом ускорителе NVIDIA GeForce RTX 3070 за одну секунду успевала обрабатывать 71 кадр.

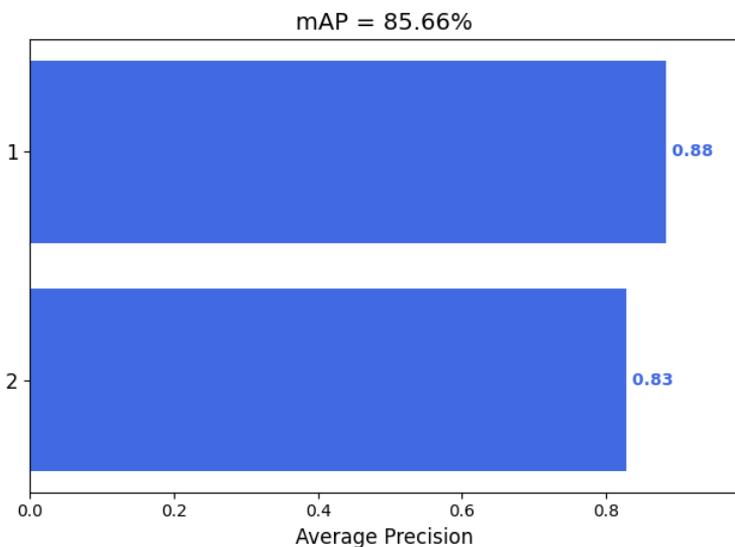


Рис. 10. Значение метрики mAP обученной модели

### *Эвристика подсчета посетителей*

На данном этапе у нас имеется модель детекции посетителей и сотрудников, работающая с кадром из видеопотока. Для того, чтобы построить маршрут передвижения людей на последовательности кадров необходимо сопоставлять детекции с соседних кадров между собой. Для этого была написана эвристика подсчета (рис. 11).

```

# объекты с предыдущего кадра
tracked_objects = [obj1, obj2]

for frame in video:
    # поиск объектов в кадре
    boxes = model.predict(frame)

    for box in boxes:
        index, distance = find_nearest(tracked_objects, box)
        if distance < threshold:
            tracked_objects[index].update_position(box)
        else:
            register_new_object(tracked_objects, box)

```

*Рис. 11. Псевдокод эвристики подсчета посетителей*

**Результаты.** Тестирование системы подсчета проводилось на 500 коротких видео с разных магазинов продолжительностью не более 20 секунд с моментами входа или выхода одного или нескольких человек. Для каждого видео было вручную посчитано количество входящих и выходящих людей, после чего правильные ответы сравнивали с предсказаниями системы подсчета с помощью метрики классификации — доля правильных ответов (accuracy). Получили следующие значения метрики:

- на входящем трафике — 92%
- на выходящем трафике — 94%

**Заключение.** Менее чем за месяц удалось разработать точную систему подсчета посетителей, исключаящую из подсчета сотрудников магазинов. В настоящее время такого функционала «из коробки» нет ни одного вендора, занимающегося видеоаналитикой и представленного на Российском рынке. Систему подсчета можно запускать как локально из магазина, в котором ведется подсчет, так и в «облаке», обслуживая при этом сразу всю торговую сеть магазинов.

## СПИСОК ЛИТЕРАТУРЫ

1. Жерон О. Прикладное машинное обучение с помощью Scikit-Learn, Keras и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем / О. Жерон. — Санкт-Петербург : ООО «Диалектика», 2020. — 1040 с. — Текст : непосредственный.

2. Шолле Ф. Глубокое обучение на Python / Ф. Шолле. — Санкт-Петербург : Питер, 2019. — 400 с. — Текст : непосредственный.
3. TensorFlow Object Detection API / GitHub. — Текст : электронный // GitHub: Where the world builds software. — 2022 : [сайт]. — URL: [https://github.com/tensorflow/models/tree/master/research/object\\_detection](https://github.com/tensorflow/models/tree/master/research/object_detection) (дата обращения: 05.05.2022).
4. mAP (mean Average Precision) / GitHub. — Текст : электронный // GitHub: Where the world builds software. — 2022 : [сайт]. — URL: <https://github.com/Cartucho/mAP> (дата обращения: 05.05.2022).

*А. О. КОЗЛОВА, А. М. ВОРОБЬЕВ, О. А. ИВАНОВ*

*Тюменский государственный университет, г. Тюмень*

**УДК 004.052.3**

## **МОДЕЛЬ СЕРВИСА РАСПРЕДЕЛЕННЫХ EDGE-ВЫЧИСЛЕНИЙ НА ОСНОВЕ ПРОТОКОЛА AMQP**

***Аннотация.** В данной статье рассмотрена модель сервиса распределенных Edge-вычислений на основе протокола AMQP, который активно применяется в различных решениях управления в области IoT. Описано создание механизма «выгрузки» заданий на ресурсы, предоставленные акторами “shared ecomoty”, рассмотрено их взаимодействие на основе стандартизированных шин передачи сообщений.*

***Ключевые слова:** экономика общественного потребления, AMQP, Edge-вычисления, импортозамещение.*

**Введение.** Отечественная отрасль информационных технологий в 2022 г. столкнулась с внезапным и резким недостатком вычислительных мощностей, что поставило под угрозу экстенсивные компоненты планов развития профильных компаний и государственных учреждений. Недостаток серверного оборудования можно проиллюстрировать письмом руководства социальной сети “VK” в адрес Министерства цифрового развития, связи и массовых коммуникаций, в котором авторы просили оказать содействие в поиске поставщика, готового поставить «десятки тысяч серверов» для плановой замены и расширения возможностей социальной сети [1].