Московский государственный университет имени М.В. Ломоносова
Механико-математический факультет
Кафедра Математической теории интеллектуальных систем (МаТИС)

Курсовая работа

Сравнительный анализ функций потерь для верификации лиц с использованием сиамских нейронных сетей

Макаров Илья Александрович
Научный руководитель:
Миронов Андрей Михайлович

Москва, 2025

Содержание

Введение **2**
Актуальность . . . 2
Практическая значимость . . . 2

Исторический обзор **3**
2000-е годы . . . 3
2010-е годы . . . 3
2020-е годы . . . 3

Формальная постановка задачи верификации лиц **4**
Исходные данные . . . 4
Постановка задачи . . . 4
Основные метрики . . . 4

Основные определения и теория **5**
Эмбеддинг . . . 5
Сиамские нейронные сети . . . 5
Triplet Loss . . . 5
Триплетный майнинг в задачах метрического обучения . . . 6
Cluster Loss . . . 6
Компонента компактности кластеров . . . 6
Компонента разделения кластеров . . . 7
Метрики качества . . . 7
Точность (Accuracy) . . . 7
VAL@FAR(10^{-2}) . . . 7

Подготовка экспериментальной части **8**
Датасет . . . 8
Обработка датасета . . . 8

Экспериментальная часть **9**
Целевые метрики . . . 9
Формирование выборки для стохастического градиентного спуска . . . 9
Валидация . . . 9
Архитектура . . . 9

Результаты **10**
Обучение с Cluster loss . . . 10
Обучение с Triplet loss random mining . . . 11
Обучение с Triplet loss semi-hard mining . . . 12
Обучение с Triplet loss hard mining . . . 13

Выводы **14**

Возможные улучшения Cluster Loss **15**
Пересмотр понятия центра временного кластера . . . 15
Автоматизация подбора гиперпараметров . . . 15
Детектирование вбросов . . . 15

Введение

Актуальность

Актуальность исследования функций потерь в сиамских нейронных сетях для верификации лиц обусловлена возрастающей ролью биометрической аутентификации в современных системах безопасности, персонализированных сервисах и автоматизированном контроле доступа. В условиях роста требований к надёжности идентификации ключевое значение приобретает эффективность алгоритмов распознавания, которая во многом зависит от выбора и оптимизации функций потерь. Такие функции, как Contrastive Loss, Triplet Loss и более современные ArcFace, CosFace, SphereFace, определяют, насколько хорошо модель различает признаки человека и, а также насколько устойчива она к изменениям освещения, ракурса и другим искажениям. Однако каждая из этих функций имеет свои ограничения, и их сравнительный анализ необходим для того, чтобы найти компромисс между точностью, скоростью обучения и устойчивостью модели.

Практическая значимость

Практическая значимость работы заключается в том, что её результаты могут быть использованы для улучшения реальных биометрических систем. Оптимизация функций потерь позволяет повысить точность верификации, что критически важно в условиях, где ошибки идентификации могут привести к утечкам данных или несанкционированному доступу. Кроме того, эффективные модели распознавания лиц востребованы в персонализированных сервисах, цифровых ассистентах и автоматизированных системах учёта, где важно быстро и точно определять пользователя. Таким образом, исследование не только углубляет теоретическое понимание работы сиамских сетей, но и предлагает практические решения для внедрения в современные технологии аутентификации и безопасности.

Исторический обзор

2000-е годы

Сиамские нейронные сети [5] зародились как инструмент для сравнения объектов в условиях малого количества данных. Их структура, напоминающая сиамских близнецов (две идентичные подсети с общими весами), позволяла эффективно извлекать признаки из парных входов и оценивать их сходство через евклидово расстояние или другие метрики. Примером раннего применения стала работа с датасетом AT&T Faces, где сети обучались на парах изображений для бинарной классификации "свой/чужой". Основной функцией потерь тогда выступала Contrastive Loss, минимизирующая расстояние между схожими объектами и увеличивающая его для разных.

2010-е годы

С развитием свёрточных нейросетей (CNN) [4] сиамские архитектуры стали использовать CNN в качестве базового блока для извлечения признаков. Например, в работах Google был предложен Triplet Loss [3], где модель обучалась на тройках данных (якорь, позитивный и негативный примеры), что улучшило дискриминативность признаков . Ключевым прорывом стало применение сиамских сетей в коммерческих системах, таких как FaceNet (2015) [3], которая достигла точности 99.63% на LFW (Labeled Faces in the Wild), используя тройки и глубокие CNN.

2020-е годы

Современные исследования сосредоточены на оптимизации углового пространства признаков. В 2018 году Цзянькан Чжан (Университет Торонто) предложил ArcFace, который добавляет угловой зазор между “кластерами” картинок, улучшая межклассовую дистанцию.

Формальная постановка задачи верификации лиц

Исходные данные

Пусть заданы:

- Множество изображений $\mathcal{X} = \{x_1, \ldots, x_N\}$
- Соответствующие идентификаторы классов $\mathcal{Y} = \{y_1, \ldots, y_M\}$, где $y_i \in \{1, \ldots, M\}$
- Функция эмбеддинга $f : \mathcal{X} \to R^d$, преобразующая изображение в вектор признаков
- Функция расстояния $\rho : R^d \times R^d \to R^+$

Постановка задачи

Для произвольной пары изображений $(x_i, x_j) \in \mathcal{X} \times \mathcal{X}$ требуется определить находятся ли на них один и тот же объект или нет, для этого используется следующая последовательность действий:

1. Получить эмбеддинги изображений
2. Посчитать расстояние между ними
3. В зависимости от порога сделать вывод о том, один и тот же объект на фотографии или нет

Основные метрики

- Точность
- VAL@FAR

Основные определения и теория

Эмбеддинг

Эмбеддинг (латентный вектор) - это компактное числовое представление данных (например, изображения, текста или аудио) в скрытом (латентном) пространстве, полученное в результате работы нейронной сети или другого метода машинного обучения.

Сиамские нейронные сети

Сиамская нейронная сеть (англ. *Siamese Neural Network*) — это специальная архитектура глубокого обучения, характеризующаяся следующими свойствами:

- Состоит из двух или более идентичных подсетей (близнецов), имеющих общие веса
- Все подсети принимают разные входные данные, но вычисляют признаки в едином латентном пространстве

Основные применения:

- Задачи верификации (лиц, подписей)
- Поиск дубликатов
- One-shot learning
- Ранжирование в рекомендательных системах

Triplet Loss

Функция потерь на триплетах (Triplet Loss) определяется следующим образом:

$$\mathcal{L}_{\text{triplet}} = \max\left(\|f(A) - f(P)\|_2^2 - \|f(A) - f(N)\|_2^2 + \alpha, 0\right)$$

- $\mathcal{L}_{\text{triplet}}$ - значение функции потерь
- A (**Anchor**) - опорный пример
- P (**Positive**) - позитивный пример (того же класса)
- N (**Negative**) - негативный пример (другого класса)
- $f(\cdot)$ - функция эмбеддинга (например, нейросеть)
- $\|\cdot\|_2$ - L2-норма (евклидово расстояние)
- α - параметр отступа (**margin**)

Триплетный майнинг в задачах метрического обучения

Для обучения с Triplet loss используются три типа триплетов:

- **Сложные триплеты** (Hard triplets):
  - В качестве позитивного примера берут наиболее удаленного от якоря представителя данного класса, в качестве негативного примера берут наиболее близкого представителя другого класса
  - Наиболее информативны для обучения, но могут содержать шум
  - Сложны для оптимизации, могут привести к коллапсу модели
- **Полусложные триплеты** (Semi-hard triplets):
  - Позитивный пример берется наиболее удаленным, а негативный исходя из соотношения: $d(A, N) > d(A, P)$ но $d(A, P) + \alpha > d(A, N)$
  - Оптимальный баланс между информативностью и стабильностью обучения
  - Помогают избежать проблем сходимости
- **Случайные триплеты** (Random triplets):
  - Полностью случайные комбинации якорей, позитивов и негативов
  - Просты в генерации, но содержат много неинформативных примеров
  - Используются на начальных этапах обучения

Cluster Loss

Созданная в рамках данного исследования функция потерь состоит из двух компонент:

$$\mathcal{L}_{\text{cluster}} = \alpha \cdot \mathcal{L}_{\text{compactness}} + \mathcal{L}_{\text{separation}}$$

Компонента компактности кластеров

$$\mathcal{L}_{\text{compactness}} = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k} \text{ReLU}\left(\|\mathbf{c}_k - \mathbf{x}_i^k\|_2 - \delta_{\text{close}}\right)$$

- K – количество кластеров
- N_k – количество элементов в кластере k
- $\mathbf{c}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{x}_i^k$ – центр кластера k
- $\mathbf{x}_i^k$ – i-й элемент кластера k
- $\delta_{\text{close}} = 0.1$ – пороговое расстояние
- $\alpha = 0.4$ – гиперпараметр

Компонента разделения кластеров

$$\mathcal{L}_{\text{separation}} = \frac{1}{K} \sum_{k=1}^{K} \text{ReLU}\left(\delta_{\text{far}} - \|\mathbf{c}_k - \mathbf{c}_{\text{nearest}(k)}\|_2\right)$$

- $\mathbf{c}_{\text{nearest}(k)}$ – центр ближайшего кластера к кластеру k
- $\delta_{\text{far}} = 0.5$ – минимальное желаемое расстояние между центрами

Метрики качества

Точность (Accuracy)

Точность – доля правильно классифицированных примеров среди всех предсказаний:

$$\text{Accuracy} = \frac{\text{Число верно классифицированных пар}}{\text{Общее число пар}}$$

VAL@FAR(10^{-2})

Метрика Verification Accuracy at Fixed False Acceptance Rate (10^{-2}) показывает точность верификации при фиксированном уровне ложных принятий 1%:

$$\text{VAL}(d) = \frac{|\text{TA}(d)|}{|\mathcal{P}_{\text{same}}|}$$

$$\text{FAR}(d) = \frac{|\text{FA}(d)|}{|\mathcal{P}_{\text{diff}}|}$$

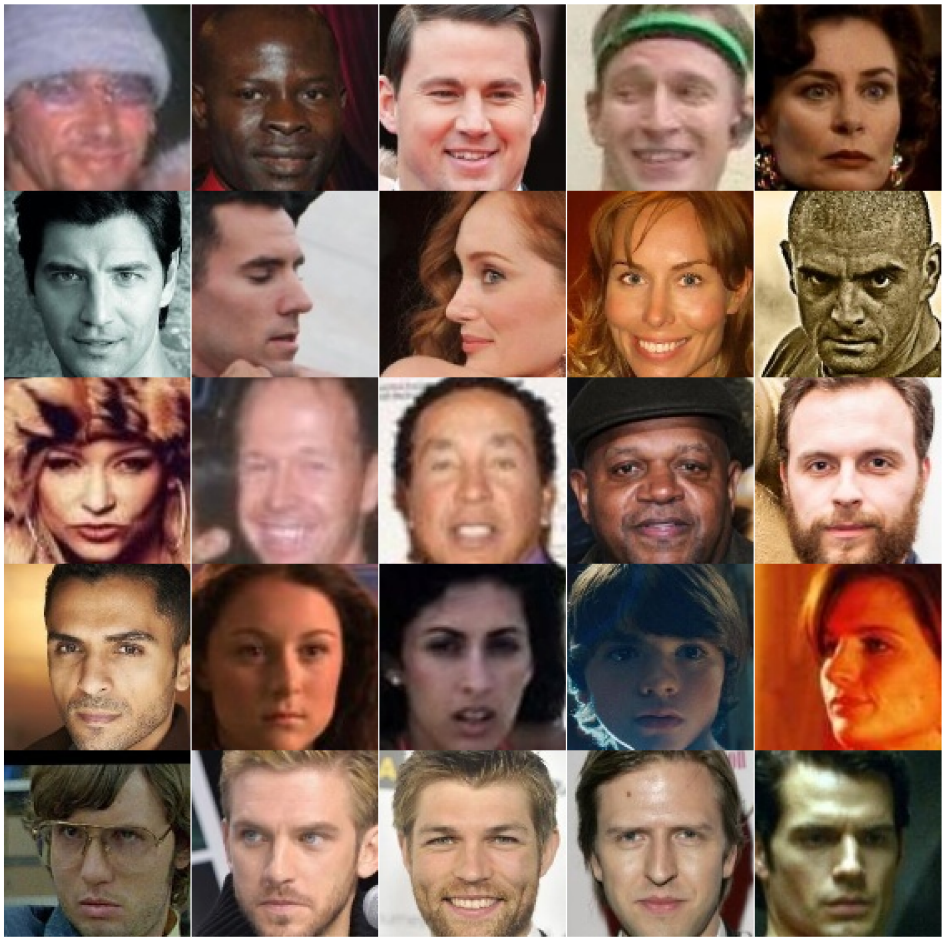
- $\mathcal{P}_{\text{same}}$ – множество всех пар изображений одного человека (genuine pairs)
- $\mathcal{P}_{\text{diff}}$ – множество всех пар изображений разных людей (impostor pairs)
- $\text{TA}(d) = \{(x_i, x_j) \in \mathcal{P}_{\text{same}} \mid \text{dist}(x_i, x_j) \leq d\}$ – множество верно принятых пар
- $\text{FA}(d) = \{(x_i, x_j) \in \mathcal{P}_{\text{diff}} \mid \text{dist}(x_i, x_j) \leq d\}$ – множество ложных принятий

Процедура вычисления:

1. Для всех пар вычисляются расстояния между эмбеддингами
2. Подбирается порог t, при котором $\text{FAR} = 0.01$
3. Вычисляется VAL при найденном пороге t

Подготовка экспериментальной части

Датасет



Используется датасет `CASIA-WEBFACE`, содержащий фотографии людей размером 112×112 пикселей. В нем представлено 10 572 личности, при этом на одну личность приходится от 9 до более чем 1000 фотографий.

Обработка датасета

- Личности, на которых приходится менее 10 фотографий, были исключены из датасета.
- Личности с количеством фотографий от 10 до 16 включены в тестовую выборку.
- Остальные личности вошли в тренировочную выборку.
- Была проведена аугментация изображений: случайные повороты на угол, имеющий нормальное распределение со стадартным отклонением в $\frac{3\pi}{50}$.

Экспериментальная часть

Целевые метрики

- Точность (Accuracy)
- VAL@FAR(10^{-2})

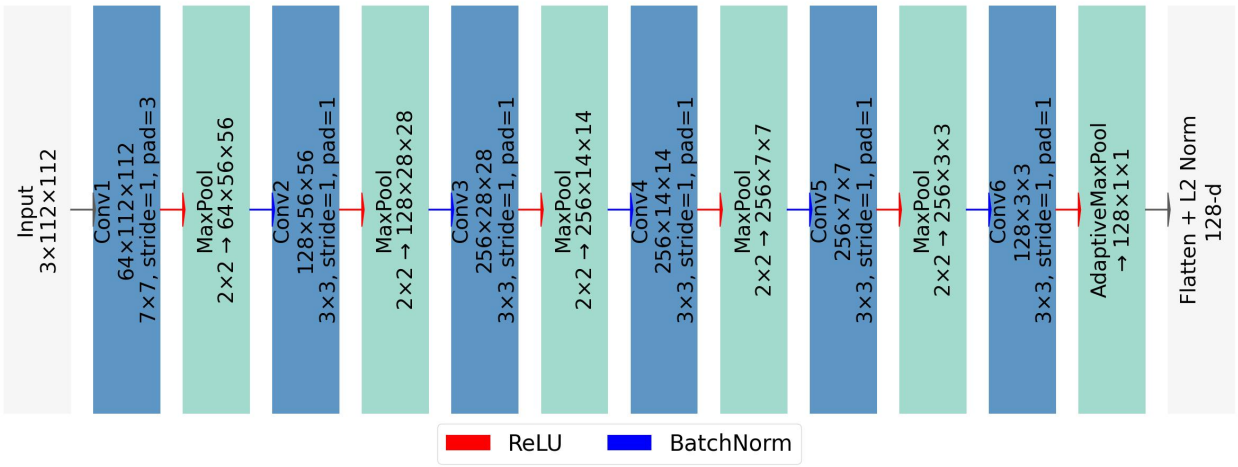
Формирование выборки для стохастического градиентного спуска

- Для **cluster loss**:
  - Из тренировочной выборки извлекаются 96 личностей с вероятностями, пропорциональными числу изображений на каждую из них.
  - Для каждой личности равновероятно извлекаются 15 изображений.
- Для **triplet loss**:
  - Аналогично *cluster loss* извлекаются 192 личности по 15 изображений на каждую.
  - Для каждой личности выбираются 5 якорей (A).
  - По оставшейся выборке формируются 5 триплетов согласно способу майнинга (большее кол-во приводит к переизбытку одинаковых или похожих триплетов и малой информативности каждого из них).
  - В итоге получается 960 триплетов.

Валидация

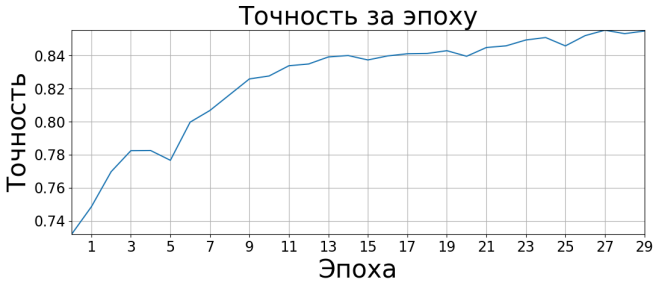
- Формируются пары в равной пропорции: фотографии одного и того же класса и фотографии разных классов (чтобы избежать дисбаланса классов).
- Вычисляются оптимальный порог (*threshold*) и значения метрик.

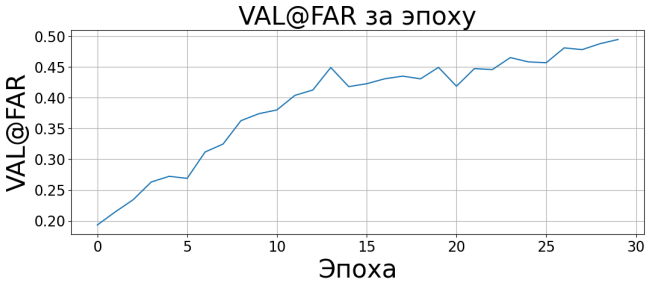
Архитектура

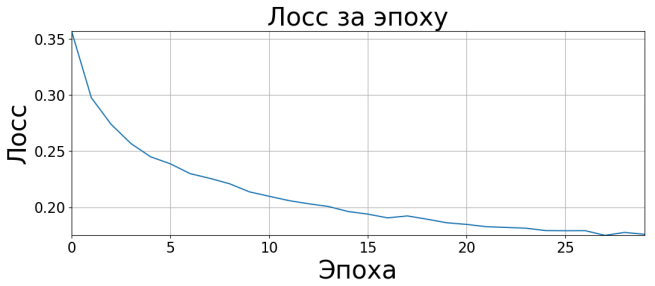


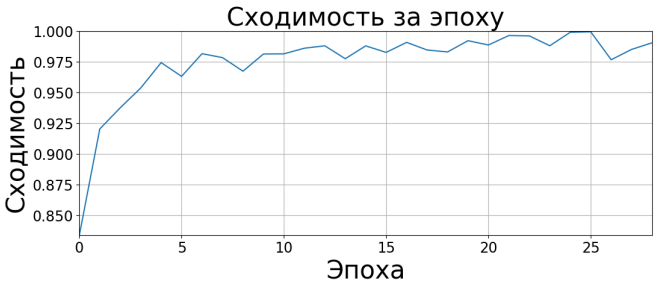
Результаты

Обучение с Cluster loss



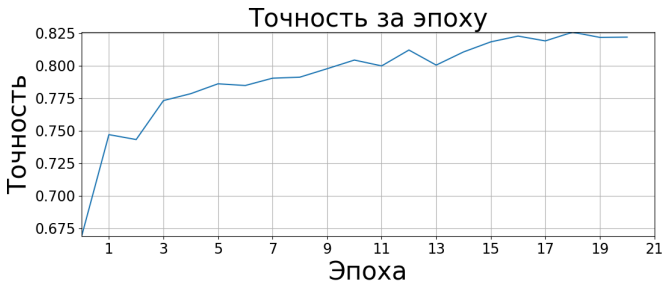


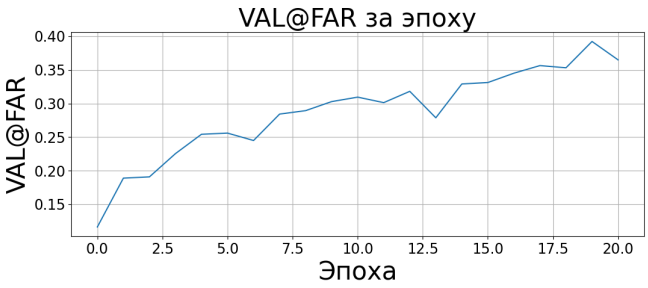


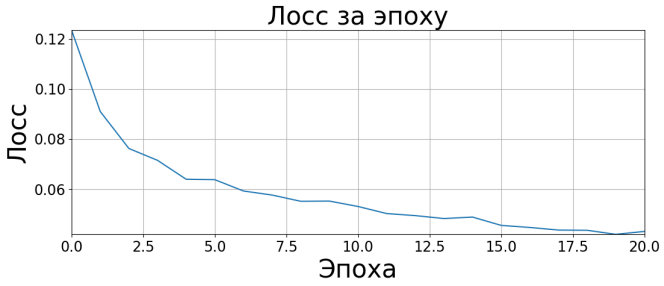


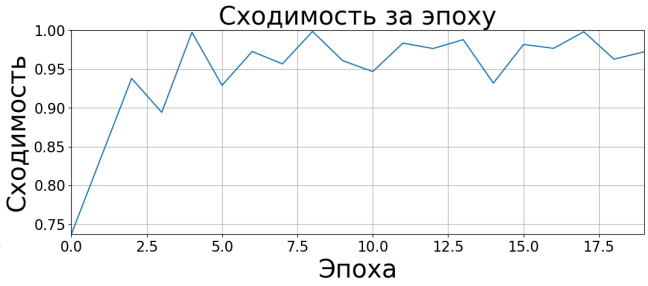
Точность	VAL@FAR(10^{-2})	Порог (threshold)	Среднее время на эпоху
0.86	0.48	0.82	162

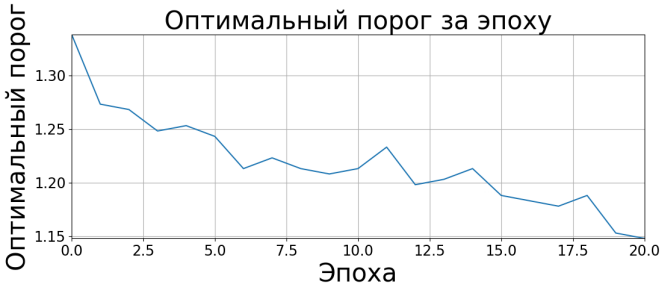
Обучение с Triplet loss random mining

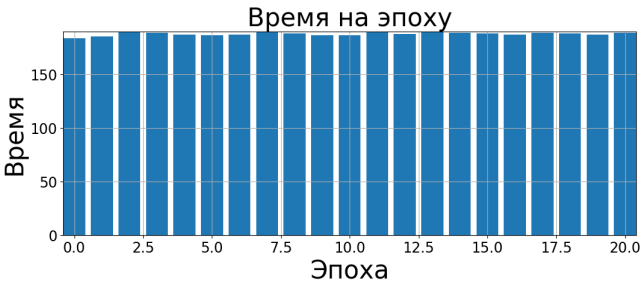






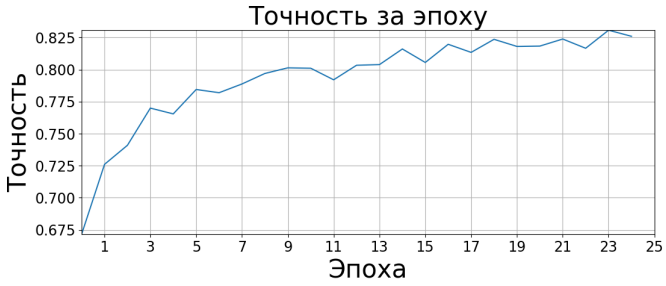


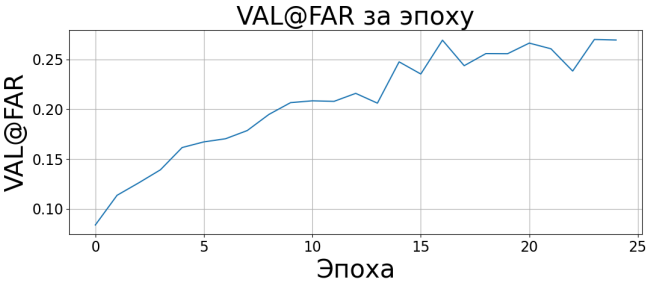


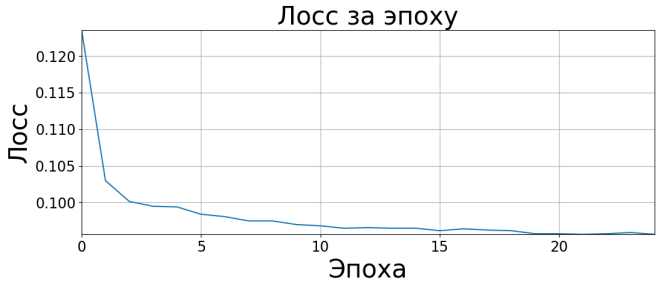


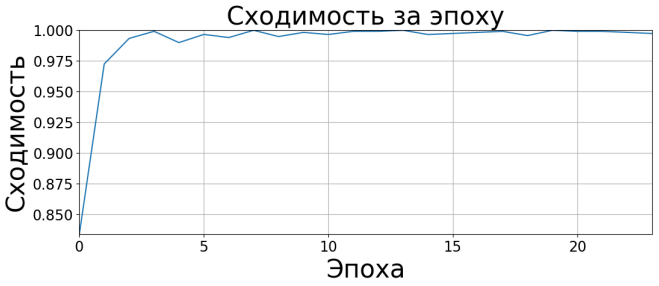
Точность	VAL@FAR(10^{-2})	Порог (threshold)	Среднее время на эпоху
0.83	0.35	1.19	188

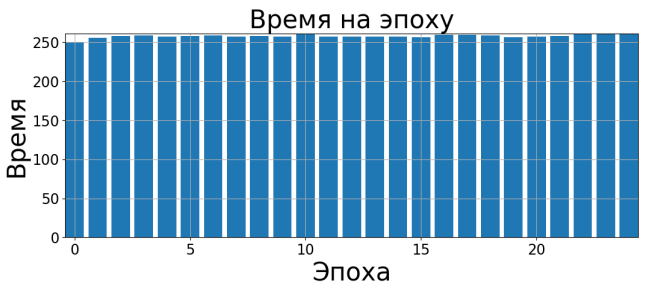
Обучение с Triplet loss semi-hard mining





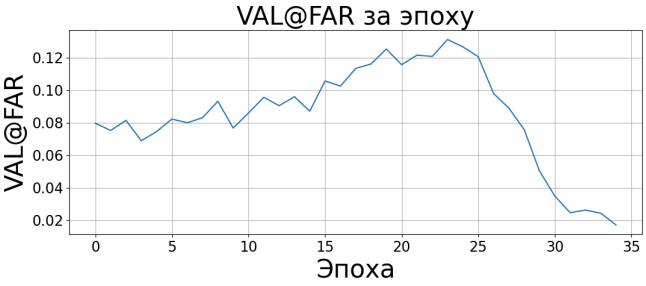


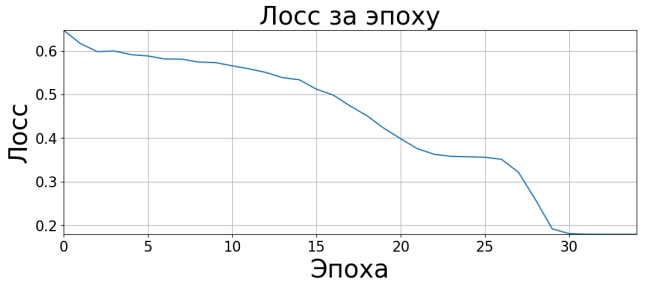


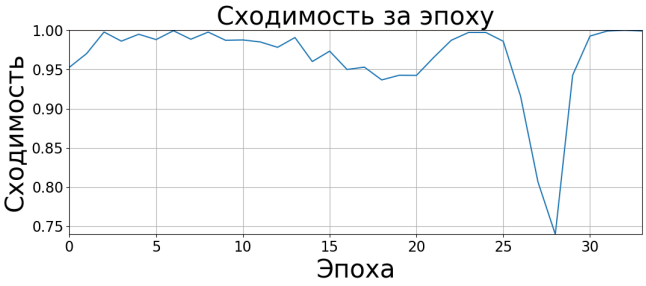


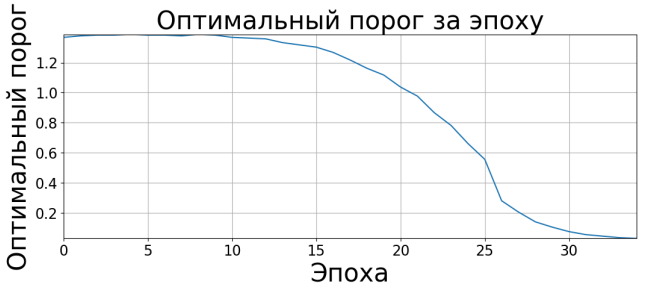
Точность	VAL@FAR(10^{-2})	Порог (threshold)	Среднее время на эпоху
0.83	0.27	1.03	258

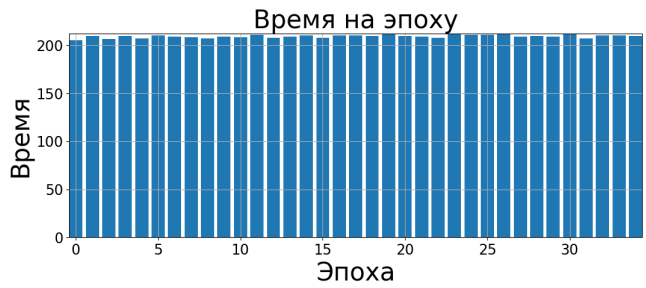
Обучение с Triplet loss hard mining











Точность	VAL@FAR(10^{-2})	Порог (threshold)	Среднее время на эпоху
0.72	0.12	0.56	209

Выводы

1. Введённый в данной работе `Cluster loss` показал лучшие результаты из представленных, как по значениям метрик, так и по скорости работы. Помимо этого, он продемонстрировал стабильность и простоту применения.

2. `Cluster loss` прекрасно работает "из коробки", не требуя дополнительной настройки стратегий майнинга, в отличие от `triplet loss`.

3. `Random mining` на удивление показал лучший результат среди способов майнинга триплетов, что идёт вразрез с результатами, представленными в работе [3].

4. Модель, обученная на hard треплетах, не сошлась вопреки теоретическим ожиданиям. Возможными причинами могли быть использование слишком маленького размера батча (увеличение размера батча могло бы привести к улучшению результатов), а также низкое качество датасета (проблема могла бы быть исправлена ручной проверкой и удалением "плохих" изображений, например, низкого качества).

Возможные улучшения Cluster Loss

Для повышения эффективности новой функции потерь (*Cluster Loss*) можно рассмотреть следующие направления

Пересмотр понятия центра временного кластера

- Использование адаптивных методов вычисления центра кластера, например, экспоненциального скользящего среднего (EMA).
- Учет не только пространственной близости, но и временной динамики изменения центров.
- Введение весов для объектов кластера в зависимости от их "возраста"или степени уверенности.

Автоматизация подбора гиперпараметров

- Применение методов байесовской оптимизации для автоматического выбора оптимальных параметров в процессе обучения.
- Использование адаптивных стратегий, подобных *learning rate scheduling*, но для параметров кластеризации.
- Внедрение механизмов мета-обучения (meta-learning) для настройки гиперпараметров на лету.

Детектирование вбросов

- Интеграция статистических методов (например, анализ межквартильных размахов) для выбросов.
- Использование методов, основанных на плотности (DBSCAN-like подходы) для игнорирования шумовых точек.

Список литературы

[1] ArcFace: Additive Angular Margin Loss for Deep Face Recognition. https://arxiv.org/abs/1801.07698

[2] Understanding the Behaviour of Contrastive Loss. https://arxiv.org/abs/2012.09740

[3] FaceNet: A Unified Embedding for Face Recognition and Clustering. https://arxiv.org/abs/1503.03832

[4] ImageNet Classification with Deep Convolutional Neural Networks.

[5] "Signature Verification using a 'Siamese' Time Delay Neural Network".