

Обучение нейронных сетей на новых доменах и малых выборках в задаче определения группы крови¹

П.М. Пищев, Д.А. Шарапов, С.А. Корчагин, Е.И. Ершов

Институт проблем передачи информации, Российская академия наук, Москва, Россия

Поступила в редколлегию 20.08.2024 г. Принята 01.10.2024 г.

Аннотация—Современные алгоритмы глубокого обучения позволяют достичь высоких результатов предсказания в различных задачах классификации изображений. Как правило, для этого требуется достаточно большое количество данных из каждого класса в обучающей выборке. Однако сбор нового набора размеченных данных для решения похожих задач классификации может быть трудозатратным и дорогостоящим. Для решения такой проблемы существуют методы адаптации к новому домену на основании данных из исходного домена. В данной работе исследуются техники доменной адаптации в приложении к распознаванию реакции агглютинации крови в лунках серологических планшетов в условиях малого количества данных в целевом домене. В качестве доменов выступают лунки круглой и овальной форм с двумя классами: наличия или отсутствия реакции агглютинации. В сценарии обучения без исходного домена удалось достигнуть точности классификации в 95,6% на 32 круглых лунках и 94,2% на 32 овальных лунках. В сценарии обучения без разметки на целевом домене удалось достигнуть точности классификации в 95,8% на 32 круглых лунках и 95,5% на 32 овальных лунках.

КЛЮЧЕВЫЕ СЛОВА: доменная адаптация, определение групп крови, обучение на малых данных.

DOI: 10.53921/18195822_2024_24_3_262

1. ВВЕДЕНИЕ

Согласно данным Всемирной Организации Здравоохранения [1], ежегодно в мире собирается около 118,5 миллиона донаций крови. Значительную их часть используют для переливания пациентам — это может как спасти человеческую жизнь, так и нанести серьезный вред здоровью: переливание несовместимой группы способно привести к разрушению эритроцитов и летальному исходу. Вследствие чего, корректное тестирование групп крови людей имеет высокую значимость во многих медицинских процессах.

Один из наиболее популярных способов определения групп крови людей — визуальная оценка наличия или отсутствия реакции агглютинации (склеивания эритроцитов). Для этого сначала смешивают образцы крови с различными реагентами, после чего по комбинации реакций определяется группа крови.

Существуют три основных подхода для определения группы крови, основанных на агглютинации: ручной, автоматический и полуавтоматический. В ручном подходе смешиванием реагентов и визуальной оценкой присутствия реакции занимается сам врач. Главными недостатками этого подхода являются относительно высокая вероятность ошибки врача и невысокая скорость работы. В автоматических подходах за смешивание реагентов и крови, а также за получение результата тестирования отвечает устройство-анализатор. Точность оценки у такого

¹ Исследование осуществлено в рамках Госзадания № FFNU-2022-0028 ИППИ РАН

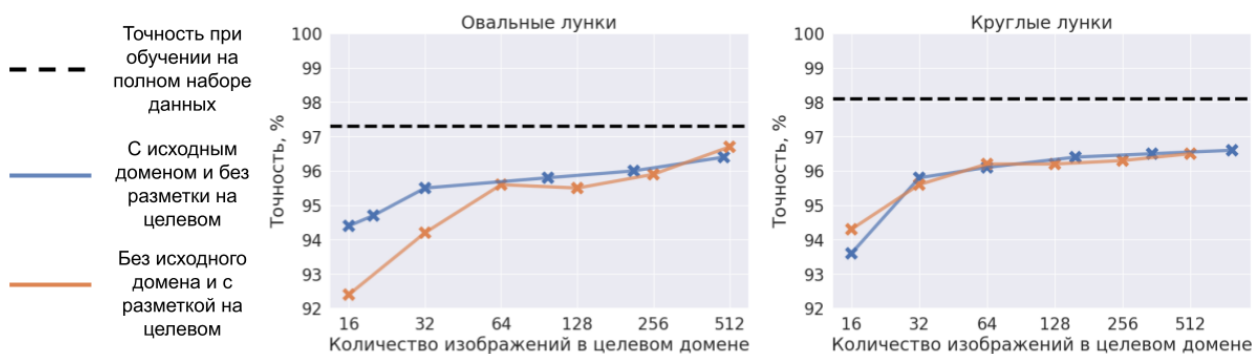


Рис. 1. Достигнутая точность работы моделей в двух исследуемых сценариях и двух целевых доменах

подхода выше, однако само устройство и реагенты для него могут быть достаточно дорогостоящими для небольших клиник.

В этом исследовании рассматривается третий, полуавтоматический подход [51–53]. Его использование востребовано для большинства клиник, так как позволяет сохранить баланс между низкой стоимостью ручного и высокой скоростью автоматического подхода. Рассматриваемая процедура определения группы крови предполагает, что врач самостоятельно смешивает образцы крови с реагентами. После возникновения реакций медицинская система автоматически анализирует изображение планшета на предмет наличия агглютинации. Общий результат исследования затем формируется как групповая принадлежность анализируемых образцов и заполняется в информационную систему лаборатории.

В качестве метода автоматического анализа планшета с лунками на агглютинацию предлагается использовать свёрточные нейронные сети. На текущий момент разработанную систему удалось настроить для работы с двумя типами планшетов: планшеты «Медиклон» (круглая форма лунок) и «Башлай» (овальная форма лунок). Эти типы лунок отличаются формой и глубиной дна, что дает различные особенности проявления реакции и не позволяет использовать сеть, обученную на одном домене для использования на другом.

Таким образом, если собирать данные только для одного типа планшетов, то нейронная сеть после обучения на этих данных сможет эффективно распознавать агглютинацию только на нём. Это означает, что одной из главных проблем в использовании нейронных сетей для данной задачи является большое разнообразие возможных форм лунок (см. рис. 2). Сбор большого набора данных для каждого существующего типа планшетов сопровождается большими трудозатратами. С целью их уменьшить в данной работе рассматриваются различные методы доменной адаптации и методы обучения на малой тренировочной выборке, позволяющие достигнуть максимальной точности. В данной работе рассматриваются техники обучения на малых данных на примере доменов лунок круглой и овальной форм в условиях малых размеров тренировочного набора данных.

Обучение на малых данных может быть проведено как с использованием исходного домена, так и без него. Таким образом, в данной работе рассматриваются два сценария: обучение без дополнительной информации (без набора данных исходного домена) и перенос знаний от одного домена к другому.

В данной работе рассмотрены методы обучения на малых данных и доменной адаптации. В условиях отсутствия исходного домена на 64 изображениях в целевом домене удалось уменьшить количество ошибок в 2,4 раза по сравнению с не нейросетевым решением, основанным на сопоставлении ключевых точек. При использовании большого исходного домена

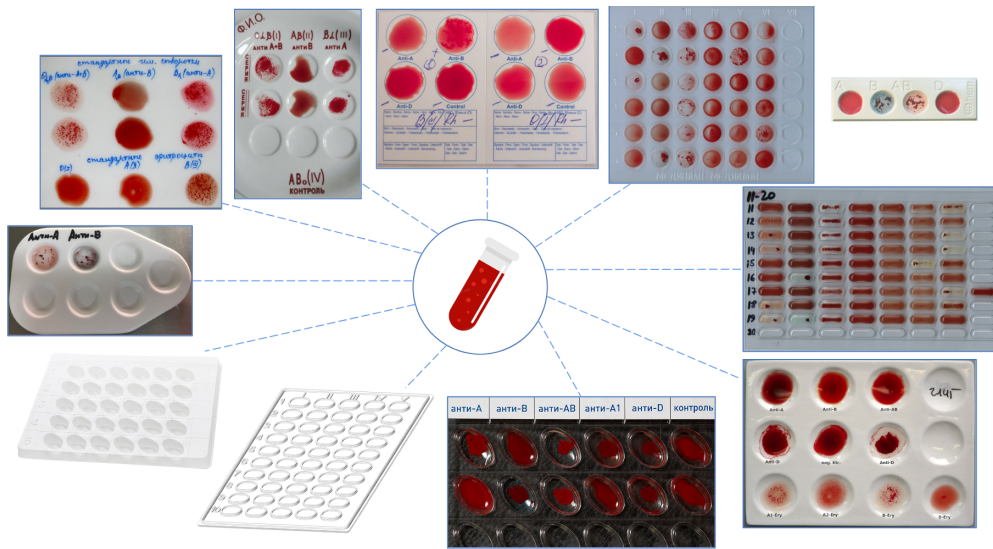


Рис. 2. Пример разнообразия тест-систем на похожую реакцию агглютинации крови

и целевого домена, состоящего из 30 неразмеченных изображений удалось достичь точности предсказания класса в 95,8% при адаптации от овальных лунок к круглым и 95,5% при адаптации от круглых лунок к овальным.

2. ОБЗОР ЛИТЕРАТУРЫ

За последнее десятилетие в задаче классификации изображений при обучении с учителем были достигнуты значительные успехи [20–23]. Чаще всего эти подходы требуют наличия достаточно большого количества размеченных данных каждого класса в обучающей выборке. Другим препятствием для обучения модели является различие условий эксперимента: формы лунок, используемые реагенты, освещение. Сочетание различных условий сбора данных задаёт некоторый домен. Известно, что после обучения на одном домене, точность работы нейронной сети на других доменах значительно ниже, поэтому для каждого домена требуется собирать набор данных заново. Однако сбор обширного набора данных для каждого домена в отдельности может быть затратным по времени или усилиям, особенно в приложениях к медицинским задачам.

Одним из первых основных методов обнаружения агглютинации было извлечение признаков изображения с реакцией агглютинации заранее заданным алгоритмом и их использование в основе метода принятия решения. В работах [29–33] авторы предложили для своих методов находить оптимальный порог разделения на классы, основываясь на собранном наборе данных. Используемые решения часто включают в себя различную локальную статистику изображения [29, 32, 33], количество компонентов связности в карте границ Кэнни [30, 33] или анализ ключевых точек изображения [31, 35]. Эти подходы просты в реализации, но сам порог очень сильно зависит от используемого набора данных и плохо применим за пределами полученных доменов.

Рассмотренные методы хоть и позволяют классифицировать изображения без использования большого набора данных, однако их точность может быть неудовлетворительна. По этой причине в настоящее время большую популярность набирают техники обучения на малых данных [36] и доменной адаптации [3], позволяющие достигать качество работы сильно превосходящую классические алгоритмы, используя экстремально малые и неразмеченные наборы

данных. Данный обзор поделён на две части: в первой будут рассмотрены методы обучения на малых данных, а во второй части будут рассмотрены методы доменной адаптации.

2.1. Методы обучения на малых тренировочных выборках

Обучение нейронной сети на малых выборках без использования дополнительных доменов сопряжено с рядом трудностей [36]. Большое количество параметров в нейронной сети и малая обучающая выборка приводят к переобучению. В результате после обучения модель делает хорошие предсказания на обучающей выборке, но предсказания на других данных остаются неудовлетворительными. Для преодоления этого эффекта были разработаны различные методы обучения модели на малых данных.

Аугментации. Один из самых простых способов улучшить предсказательную способность модели — это использование техник аугментаций [45]. Как правило, аугментации представляет собой достаточно простые преобразования изображений, такое как поворот, растяжение или попиксельные изменения цвета. Существуют также более сложные методы аугментаций, использующие предварительно обученные нейронные сети, которые могут изменять более сложные характеристики изображения, такие как фон [37] и цветовой тон основных объектов на изображении [38]. Основные преимущества использования аугментаций заключаются в том, что они просты в реализации и могут использоваться в сочетании с другими подходами.

Метрическое обучение. Альтернативный подход к обучению на небольших выборках — метрическое обучение [39]. При метрическом обучении нейронные сети обучаются не предсказывать классы, а создавать векторные представления, похожие на векторные представления объектов того же класса. При обучении модели с помощью этого метода функция потерь представляет собой метрику расстояния между векторами. Обучение происходит следующим образом: нейронная сеть генерирует векторные представления для нескольких изображений и сравнивает их с помощью метрики. Такой подход повышает качество модели за счет дополнительной регуляризации: метрическое обучение вводит дополнительный штраф за то, что векторные представления объектов одного класса могут быть далеки друг от друга. В предлагаемом методе можно варьировать количество одновременно сравниваемых изображений [40], метрика может быть как заранее заданной, например, чаще всего используется L_2 -норма между векторами, или обучаемой, как например в сетях сходства [41]. Также можно сравнивать представление изображения с усредненным по классу векторным представлением (прототипом), а не с векторным представлением двух изображений [42]. Другим подходом обучения на малой выборке является мета-обучение. Этот подход заключается в использовании нескольких больших наборов данных для получения таких весов модели, которые позволяют в среднем лучше всего дообучить модель на целевом наборе данных. Это особенно важно для обучения при малой обучающей выборке, поскольку в этом случае обучение модели обычно приводит к локальному оптимуму функции потерь. Целью мета-обучения является обучение генератора гиперпараметров, и классические методы включают MAML [43], Reptile [44]. MAML определяет направление глобальной оптимизации, вычисляя направление оптимизации для каждой задачи. По сравнению с MAML, Reptile может обновлять меньше параметров одновременно.

2.2. Методы, требующие использование исходного домена

Доменная адаптация. Существует обширная классификация подходов к решению задач доменной адаптации [3]. В частности, к ним относятся статистические методы, основанные на идеях прикладной статистики. Одним из примеров является техника адаптации DAN [16], которая использует критерий проверки статистической гипотезы равенства распределений признаков векторов. Её известным аналогом является метод CORAL [8] и его расширение на

архитектуру глубокого обучения [8]. Он основывается на минимизации нормы разницы матриц ковариации признаков векторов для сохранения зависимостей признаков при адаптации модели к новому домену.

Другим важным и развивающимся направлением в доменной адаптации можно выделить методы на состязательных функциях потерь. Одним из фундаментальных примеров таких методов является техника DANN [19], которая при обучении использует классификатор метки домена экземпляров для дискриминации экстрактора признаков. Такой процесс позволяет обучить целевую модель извлекать домен-инвариантные признаки.

В настоящее время большое внимание уделяют техникам субдоменной адаптации, где в качестве доменов выступают экземпляры данных с одной и той же классовой меткой. Переход к такому роду задач адаптации имеет важное значение, потому что сопоставление глобальных распределений может смешать между собой нерелевантные данные разных классов, что в конечном итоге принесёт дополнительные ошибки при классификации.

Перечисленные выше техники имеют свои модификации в задаче субдоменной адаптации. В качестве одного из примеров, DSAN [17], модификация DAN, которая сопоставляет классовые распределения признаков векторов между двумя доменами. Идея используется та же: воспользоваться критерием проверки гипотезы равенства распределений, но уже между субдоменами.

Модификации на субдоменную адаптацию существуют и для состязательных методов. Например, в техниках MEDA [10] и DAAN [11] предложено обучать не только глобальный дискриминатор, предсказывающий метки доменов для всех экземпляров, но и для каждого класса в отдельности обучать дополнительный локальный дискриминатор, нацеленный на предсказание меток доменов для экземпляров конкретного класса. Такой подход позволяет получить домен-инвариантные представления нейронной сети в терминах субдоменов. Более того, интересно отметить, что эти техники являются «динамическими» — в них приведён способ переключения между адаптацией глобального или локального распределений, что на практике повышает классификационную точность в различных прикладных задачах.

Приложения адаптации в медицине. Сбор обширного набора данных для каждого домена в отдельности может быть затратным по времени или усилиям, особенно в приложениях к медицинским задачам. Поэтому в настоящее время в этой области большую популярность набирают техники доменной адаптации [4], позволяющие достигать неплохие результаты предсказания классов на неразмеченных наборах медицинских данных.

На примере задачи классификации рака груди [5] была продемонстрирована эффективность использования техник сопоставления вторых моментов признаков векторов (CORAL) [8], проектирования на выделенные подпространства (TCA) [6] и взвешивания глобального и локального распределений доменов (BDA) [9]. В работе [18] был предложен фреймворк с применением доменного дискриминатора [19], значительно повысившего качество предсказания в задаче классификации текстуры лёгких. В работе по исследованию доменов в капсульной эндоскопии [13] была показана возможность перехода между средами при помощи дообучения триплетной функции потерь на достаточно малом количестве новых данных.

С наиболее известной и распространенной техникой переноса знаний [2] связывают использование предобученных нейросетей [24]. Для этого выбирают некоторую архитектуру, обучают её на одном из доменов, а затем копируют все слои (кроме классификационных), донастраивая их новым значением шага градиентного спуска [25]. В случае установки значения шага нулём донастройку называют *заморозкой* слоёв. Такой подход имеет примеры эффективного приложения на медицинских наборах данных — при использовании блочной донастройки слоёв в классификации опухолей головного мозга [26] или заболевания Альцгеймера [27].

3. НАБОР ДАННЫХ

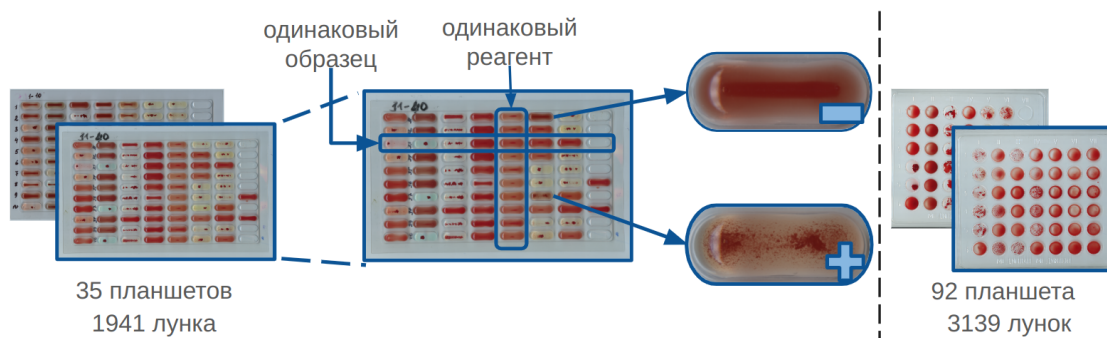


Рис. 3. Используемые наборы данных

В работе исследуются два набора данных, состоящих из изображений лунок с кровью. Первый набор данных состоит из 92 изображений серологических планшето «Медиклон». Каждый планшет состоит из круглых лунок, расположенных в виде условной сетки из 6 рядов по 7 столбцов. Каждая строка представляет разные образцы крови, а каждый столбец соответствует разным реагентам. Таким образом, каждая непустая лунка содержит смесь одного образца крови и одного реагента. Набор данных хорошо сбалансирован и включает в себя 3139 лунок с кровью (1586 с агглютинацией и 1553 без неё). Каждой непустой лунке сопоставлен один из двух классов: наличие или отсутствие агглютинации. Комбинация реакций одного и того же образца крови на различные реагенты однозначно задаёт группу крови этого образца.

В наборе данных представлены реагенты для определения группы крови по трём системам:

- ABO. Наиболее значимая система групп крови. Несовпадение групп крови по этой системе проявляется наиболее заметно. Используются реагенты A, B.
- Rh (Резус). Эта система также очень значима, особенно реакция на антиген D (резус-фактор). Используются реагенты D, C, c, E, e, C^w.
- Kell. Менее распространенная система групп крови. Используются реагенты K, k.

Аналогично устроен и второй набор данных. Он состоит из 35 планшетов «Башлай» с овальными лунками. На каждый планшет находится 10 рядов по 8 столбцов. Набор данных состоит из 1941 изображения и немного несбалансирован (851 лунка с агглютинацией и 1090 без неё).

В этом наборе данных также как и в прошлом представлены реагенты для трёх систем групп крови:

- ABO. Реагенты A, B, anti-A, anti-B.
- Rh (Резус). Реагенты D, C, c, E, e, C^w.
- Kell. Антигены K, k.

Далее в работе будем использовать следующую нотацию: «К» — набор лунок круглой формы и «О» — набор лунок овальной формы. Для оценки точности исследуемых методов 50% каждого набора было отведено под тестовую выборку, которая не изменялась от эксперимента к эксперименту.

Эти типы лунок отличаются глубиной и формой дна, что дает особенности проявления агглютинации в этих типах лунок и не позволяет легко перенести знания, полученные на одном домене на другой, поэтому эти два набора данных можно использовать как два разных домена для одной и той же задачи.

4. МЕТОДЫ ОБУЧЕНИЯ

Доменом \mathcal{D} называют пару $\{\mathcal{X}, \mathbb{P}(x)\}$, $x \in \mathcal{X}$, где \mathcal{X} — пространство признаков, $\mathbb{P}(x)$ — вероятностное распределение на нём. Далее по тексту будут представлены техники адаптации исходного домена $\mathcal{D}_s = \{\mathcal{X}_s, \mathcal{S}\}$ к целевому $\mathcal{D}_t = \{\mathcal{X}_t, \mathcal{T}\}$. Эти методы так или иначе используют архитектуры глубокого обучения, поэтому введём необходимые обозначения основных их компонент в рамках задачи классификации изображений.

Будем предполагать, что исследуемые модели обучаются на данных $x \in \mathcal{X}$, которые преобразуются в d -мерные векторы признаков $z = G_f(x)$ отображением $G_f : \mathcal{X} \rightarrow \mathbb{R}^d$, называемым *экстрактором признаков*. Полученные признаки преобразуются *классификатором* $G_c : G_f(\mathcal{X}) \rightarrow [0, 1]^C$ в вектор оценок \hat{y} вероятностей принадлежности к классам $c \in \{0, \dots, C-1\}$. Стоит отметить, что $G_f(\cdot, \theta_f)$ и $G_c(\cdot, \theta_c)$ являются частью обучаемых моделей, поэтому они дополнительно содержат в себе векторы θ_f и θ_c обучаемых параметров. На основании их композиции рассчитывается классификационная *функция потерь* $\mathcal{L}_c(G_c(z), y)$ с учётом известной метки класса y экземпляра x , примером которой в глубоком обучении нередко служит кросс-энтропия

$$\mathcal{L}(\theta_f, \theta_c) = -\mathbb{E}_{(x,y) \in \mathcal{D}} \sum_{c=0}^{C-1} \mathbf{1}_{\{y=c\}} \ln \hat{y}_c, \quad (1)$$

где \hat{y}_c — компонента $c \in \{0, \dots, C-1\}$ вектора $\hat{y}(x) = G_c(G_f(x, \theta_f), \theta_c)$.

Кроме того, далее в тексте будет предполагаться существование вероятностных распределений на признаковых векторах исходного и целевого доменов:

$$p := \{G_f(x_s, \theta_f) \mid x_s \sim \mathcal{S}\}, \quad q := \{G_f(x_t, \theta_f) \mid x_t \sim \mathcal{T}\} \quad (2)$$

4.1. Метод определения агглютинации без использования нейронных сетей

При обучении на малых данных может оказаться так, что нейронная сеть справляется с задачей хуже, чем простые алгоритмы, основанные на классическом компьютерном зрении. Это означает, что необходимо выбрать такой алгоритм компьютерного зрения, который позволяет достигать наилучшей точности работы. В данной статье был реализован для сравнения метод определения агглютинации из [34], основанный на сопоставлении особых точек с эталонным изображением. Он включает в себя следующие этапы:

- Использование алгоритма ORB [50] для получения ключевых точек изображения и их дескрипторов (векторных представлений).
- Одно изображение с агглютинацией используется в качестве эталона для сравнения с другими изображениями.
- Для эталона и выбранного изображения попарно сопоставляются ключевые точки. Если дескрипторы достаточно похожи по расстоянию Хэмминга (числу отличающихся бит), то считаем, что найдено соответствие.
- Если количество соответствий превышает некоторый порог, то считаем, что агглютинация присутствует.

Из всего набора данных выбирается одно случайное изображение с положительным результатом реакции в качестве эталона. Остальные изображения из обучающей выборки используются для подбора порогового числа количества соответствий. Остальные параметры алгоритма остаются неизменными.

4.2. Аугментации

Основная проблема, которая возникает при обучении, заключается в недостатке данных. Чтобы разнообразить набор, не прибегая к сбору дополнительных данных, используют аугментации [45]. Не всегда их использование приводит к улучшению обучения, поэтому нужно использовать только те из них, которые будут уместны в рассматриваемой задаче. Исследование, посвященное этой проблеме проводилось ранее в [46] и на его основании были выбраны следующие аугментации:

- Поворот на произвольный угол;
- Смещение по каждому измерению на $\pm 3\%$ от исходного изображения;
- Обрезание изображения вплоть до 90% по ширине и высоте от исходного;
- Гауссовский шум с дисперсией 102;
- Гамма-коррекция изображения с коэффициентом γ от 0,8 до 1,3;
- Наклон изображения вплоть до 30 градусов;
- Растяжение/сжатие до 30% по каждой из осей;
- Вырезание из изображения от 1 до 5 квадратов с линейными размерами до 12% от исходного изображения;
- Разделение изображения на 8×8 квадратов и их последующее перемешивание,

несколько из которых представлены на рис. 4.

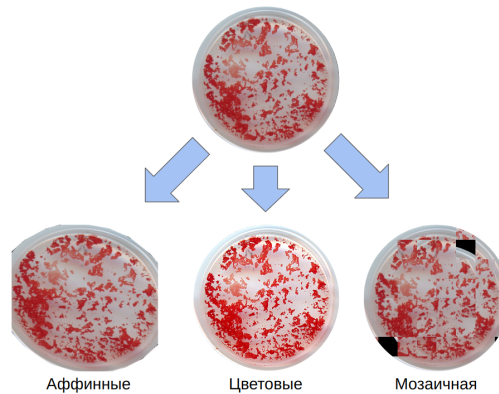


Рис. 4. Результат применения используемых аугментаций

4.3. Методы обучения с учителем только на одном домене

Перенос знаний. Одним из широко используемых методов переноса знаний является использование предварительно обученных нейронных сетей [28]. Этот подход включает в себя обучение нейронной сети на одном наборе данных (в исходном домене), а затем дообучение этой сети на другом наборе данных (в целевом домене).

Обычно нейронные сети представляют собой композицию нескольких простых слоёв:

$$G_f = G_{f,1} \circ G_{f,2} \circ \dots \circ G_{f,n-1} \circ G_{f,n}, \tag{3}$$

где у каждого слоя есть свой набор параметров θ_i . При классическом обучении нейронной сети множество обучаемых параметров θ_f представляет собой объединение $\theta_1, \dots, \theta_n$. При дообучении часто некоторые слои, например, все слои кроме нескольких k последних, оставляют неизменными, чтобы понизить число обучаемых параметров. В этом случае для θ_i получаем

$$\begin{aligned} \theta_i &= const, \quad i \leq n - k \\ \theta_i &\in \theta_f, \quad i > n - k \end{aligned} \tag{4}$$

Такой подход к уменьшению числа обучаемых параметров называют заморозкой слоёв. Однако при уменьшении числа обучаемых слоёв вместе с снижением числа обучаемых параметров также уменьшается максимально возможная допустимая точность классификации модели, поэтому необходимо найти оптимальное количество обучаемых слоёв. В данной работе начальные веса модели заимствуются из модели, обученной на наборе данных ImageNet-1K [48], поскольку он содержит большое количество разнообразных данных.

Метрическое обучение. Обычно модель классификации учится предсказывать класс объекта по его признакам, но часто используется и другой подход, который заключается в обучении кодировщика $G_f(x, \theta_f)$, который по изображению генерирует его векторное представление. Для этого векторные представления сравниваются с использованием функции схожести $\rho(z_1, z_2, \theta_\rho)$, где θ_ρ — набор возможных обучаемых параметров функции. Модель обучается так, чтобы значение функции схожести для векторных представлений объектов из одного класса было существенно меньше чем значение функции схожести для векторов из разных классов. Такой подход к обучению нейронных сетей называют метрическим обучением и часто используют при малом количестве данных [39].

Классификация при метрическом обучении. Стоит отметить, что данный подход позволяет получить только кодировщик, классификатор требуется обучать отдельно. Для всех методов метрического обучения в качестве классификатора будет использоваться SVM [49]. Алгоритм SVM в случае линейной разделимости множеств заключается в таком подборе параметров, чтобы разделяющая линия была равноудалена от ближайших точек каждого класса и это расстояние было максимальным. Это условие, а также условие корректной классификации приводит к следующей задаче:

$$\begin{cases} w^T w \rightarrow \min \\ y(w^T z - b) \geq 1, \end{cases} \quad (5)$$

где w и b — обучаемые параметры, z — векторные представления, генерируемые кодировщиком. Для метрического обучения получаем двухэтапный алгоритм: сначала проводится обучение кодировщика, затем на основании векторных представлений, генерируемых кодировщиком, проводится обучение классификатора SVM.

Триплетная функция потерь. «Triplet loss» — это один из наиболее часто используемых методов метрического обучения, а также одноименная функция потерь [55]. Для использования триплетной функции потерь требуется три изображения: якорное x_a , положительное x_p (Из того же класса, что и якорное. Обозначается как x_p) и отрицательное x_n (Из другого класса). Сама функции ошибки строится на основе функции схожести между двумя векторами и устроена так, чтобы якорный и положительный примеры были как можно ближе по заданной функции, а якорный и отрицательный как можно дальше:

$$\mathcal{L}_{\text{triplet}} = \sum_{z_a, z_p \in \mathbf{Z}_{y=1}} \sum_{z_n \in \mathbf{Z}_{y=0}} \max\{0, \alpha - \rho(z_a, z_n) + \rho(z_a, z_p)\}, \quad (6)$$

где ρ — функция схожести (в нашем случае $\rho(z_1, z_2) = \|z_1 - z_2\|_2$), α — гиперпараметр, характеризующий требуемую разницу между средними межклассовыми и внутриклассовыми расстояниями в признаковом пространстве кодировщика G_f , $z_i = G_f(x_i, \theta_f)$ — векторное представление изображения.

Прототипные сети. Одной из проблем предыдущих подходов является шумность такой функции ошибки. Так, например, если в наборе данных есть элемент, на котором нейронная сеть работает неудовлетворительно, то минимизация расстояния между ним и другими элементами приводит к ухудшению обучения. Одним из способов устранения этой проблемы является использование прототипных сетей (prototype networks) [42]. Прототип класса — это

усредненное векторное представление набора элементов из этого класса:

$$\bar{z}_c = \mathbb{E}_{z \sim p^{(c)}} z, \quad (7)$$

где $p^{(c)}$ — распределение класса c в признаковом пространстве кодировщика G_f . Метод заключается в том, чтобы заменить обучение модели на близость элементов из одного класса, на обучение близости элементов класса к прототипу этого класса. Для этого метода весь батч s с изображениями случайно делится на две равные, сбалансированные по классам половины: первая половина используется для оценки прототипов классов \bar{z}_c :

$$\bar{z}_c \approx \frac{4}{N} \sum_{\substack{y=c \\ i < N/2}} z_i, \quad (8)$$

где N — количество изображений в батче, y — класс объекта x . Вторая половина батча будет использована, чтобы обучить кодировщик генерировать на них близкие к прототипам векторные представления:

$$\mathcal{L}_{\text{prot}} = \sum_{c=0,1} \left(\sum_{\substack{y_i=c \\ i \geq N/2}} \rho(\bar{z}_c, z_i) + \sum_{\substack{y_i \neq c \\ i \geq N/2}} \max\{0, \alpha - \rho(\bar{z}_c, z_i)\} \right), \quad (9)$$

где функция схожести $\rho(z_1, z_2) = \|z_1 - z_2\|_2$.

Сети сходства. Другим часто используемым подходом являются сети сходства [41]. Они отличаются от трех упомянутых выше методов тем, что ее функция сходства рассчитывается с использованием обучаемой нейронной сети. В таком подходе ставится задача классификации, где в роли исходных данных выступают пары (или триплеты, если используется триплетная функция потерь) элементов из набора данных, а в качестве класса схожесть. Отличительной чертой сети сходства является то, что она позволяет найти и использовать наиболее адекватную для задачи функцию расстояния. Функция потерь для сети сходства в данной работе основана на триплетной функции потерь:

$$\mathcal{L}_{\text{relation}} = \sum_{z_a, z_p \in \mathbf{Z}_{y=1}} \sum_{z_n \in \mathbf{Z}_{y=0}} \max\{0, \alpha - \rho(z_a, z_n, \theta_\rho) + \rho(z_a, z_p, \theta_\rho)\}, \quad (10)$$

где z_a, z_p, z_n — векторные представления якорного, положительного и негативного образца соответственно, а θ_ρ — обучаемые параметры функции схожести.

4.4. Методы обучения без учителя на двух доменах

Доменная адаптация. Пусть имеются исходный домен \mathcal{D}_s , обученная на нём модель f_s и целевой домен \mathcal{D}_t . Тогда задача адаптации домена \mathcal{D}_s к домену \mathcal{D}_t состоит в улучшении показателей целевой модели f_t на домене \mathcal{D}_t с использованием накопленной информации из f_s и \mathcal{D}_s .

Важно подчеркнуть, что под неравенством доменов $\mathcal{D}_s \neq \mathcal{D}_t$ обычно подразумевается различие признаковых пространств $\mathcal{X}_s \neq \mathcal{X}_t$ или вероятностных распределений $\mathcal{S} \neq \mathcal{T}$ на них. Эти различия носят обобщённое название *доменный сдвиг*, а подходы к его устранению — *доменной адаптацией*.

Обученные на одном домене нейронные сети с высоким качеством предсказания могут давать неудовлетворительные результаты на экземплярах из другого домена. Такое поведение

связано с доменным сдвигом, обусловленным, например, неравенством распределений $\mathcal{S} \neq \mathcal{T}$ данных из доменов. Так как на эти распределения невозможно воздействовать напрямую, в глубоком обучении устраняют этот сдвиг на уровне распределений признаков векторов со скрытых слоёв нейронных сетей. Для этого сопоставляют эти распределения как можно ближе друг другу — пример такой адаптации на признаковых векторах изображений лунок круглой и овальной форм приведён на рис. 5.

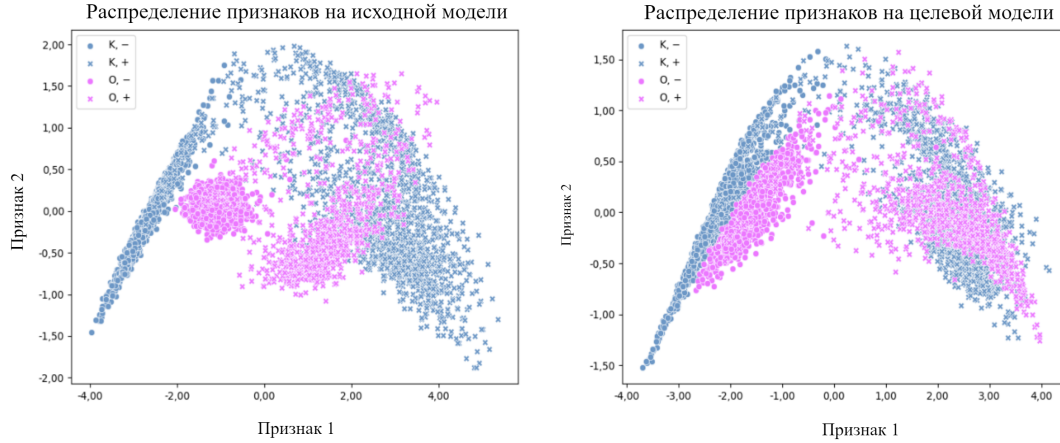


Рис. 5. Пример сопоставления распределений признаковых векторов одной из техник доменной адаптации. Здесь исходным доменом (синий цвет) выступают лунки круглой формы «К»; целевым доменом (фиолетовый цвет) — лунки овальной формы «О». Круглыми метками обозначены метки отсутствия агглютинации, крестообразными — метки наличия агглютинации

Deep Adaptation Network. Как говорилось выше, доменный сдвиг может быть устранён путём сопоставления распределений признаковых векторов исходного и целевого доменов. Для этого применяются множество различных подходы из прикладной статистики.

Пусть даны две случайные величины $\xi \sim p$ и $\eta \sim q$, заданные на одном вероятностном пространстве и принимающие значения из топологического пространства \mathfrak{X} . Пусть также задан абстрактный класс \mathcal{F} измеримых функций вида $f : \mathfrak{X} \rightarrow \mathbb{R}$. Для проверки гипотезы равенства распределений

$$H_0 : p = q, \quad H_1 : p \neq q \quad (11)$$

вводят интегральную вероятностную метрику

$$d[p, q] := \sup_{f \in \mathcal{F}} \{ \mathbb{E}_{\xi \sim p} f(\xi) - \mathbb{E}_{\eta \sim q} f(\eta) \}, \quad (12)$$

имеющую следующий смысл: если распределения p и q равны, то на правильно подобранном классе функций \mathcal{F} её значение должно быть близко к нулю. Действительно, согласно [14] выбор единичного шара $\mathcal{F} = \{f \in \mathcal{H} : \|f\|_{\mathcal{H}} \leq 1\}$ в универсальном гильбертовом пространстве \mathcal{H} с воспроизводящим ядром k даёт критерий

$$d[p, q] = 0 \quad \Leftrightarrow \quad p = q \quad (13)$$

и позволяет переписать метрику (12) в более удобную форму

$$d[p, q] = \|\mathbb{E}_{\xi \sim p} \phi(\xi) - \mathbb{E}_{\eta \sim q} \phi(\eta)\|_{\mathcal{H}}. \quad (14)$$

Теперь, при наличии реализаций двух простых выборок $x = (x_1, \dots, x_n)$ и $y = (y_1, \dots, y_m)$ случайных величин ξ и η , метрика (14) принимает более простой для вычисления вид

$$\begin{aligned} \tilde{d}_k^2(x, y) = & -\frac{2}{nm} \sum_{i,j=1}^{n,m} k(x_i, y_j) \\ & + \frac{1}{n(n-1)} \sum_{i \neq j}^n k(x_i, x_j) + \frac{1}{m(m-1)} \sum_{i \neq j}^m k(y_i, y_j), \end{aligned} \tag{15}$$

При выборе подходящего ядра [15] несмещённая оценка (15) позволяет удобно и эффективно различить два распределения. Эту идею используют при построении критериев проверки гипотезы (11) или при обучении домен-инвариантных нейронных сетей.

Как было подмечено, на доменный сдвиг $\mathcal{S} \neq \mathcal{T}$ напрямую повлиять невозможно, т.к. он вызван природой самих доменов. Но при этом, технически, можно построить экстрактор признаков из экземпляров домена и исследовать распределения на нём. В такой постановке понятен и процесс обучения: минимизируют функцию, состоящую из потерь классификации \mathcal{L}_c на доверительной метке y и линейной комбинации метрик (15) между представлениями z_s^l и z_t^l экземпляров x_s и x_t на множестве скрытых слоёв $l \in L$ нейронной сети с некоторым постоянным весом $\lambda > 0$:

$$\mathcal{L}(z_s, y, z_t) = \mathcal{L}_c(G_c(z_s), y) + \lambda \sum_{l \in L} \tilde{d}_k^2(z_s^l, z_t^l). \tag{16}$$

Описанный подход носит название DAN [16] и является основой для современных техник доменной адаптации.

Deep Correlation Alignment. Естественным требованием на свойства модели f_t может быть сохранение зависимостей между признаками после адаптации к целевому домену. Удовлетворить эти требования позволяет другая распространённая техника доменной адаптации CORAL [7]. Она основана на сопоставлении вторых моментов распределений признаков векторов путём минимизации расстояния

$$\mathcal{L}_{\text{CORAL}}(x, y) = \frac{1}{4d^2} \|R_x - R_y\|_F, \tag{17}$$

где R_x, R_y — ковариационные матрицы векторов x и y размерности d , $\|\cdot\|_F$ — норма Фробениуса.

Обучение модели на функции потери (17) между признаковыми векторами z_s и z_t совместно с классификационной функцией потерь \mathcal{L}_c в зарубежной литературе носит название DeepCORAL [8] и по аналогии с (16) может быть применительно к нескольким скрытым слоям L нейронной сети одновременно

$$\mathcal{L}(z_s, y, z_t) = \mathcal{L}_c(G_c(z_s), y) + \lambda \sum_{l \in L} \mathcal{L}_{\text{CORAL}}[z_s^l, z_t^l]. \tag{18}$$

с постоянным весовым параметром $\lambda > 0$.

Domain-Adversarial Neural Network. Два предыдущих подхода решают задачу устранения доменного сдвига $\mathcal{S} \neq \mathcal{T}$ путём сопоставления распределений (2) с использованием подходов из прикладной статистики. Похожий результат можно получить при помощи одновременном решения сразу нескольких задач оптимизации [19].

Зададимся целью построить архитектуру нейронной сети таким образом, чтобы после обучения она была способна выделять домен-инвариантные признаки. Для этого возьмём экстрактор признаков $G_f(\cdot, \theta_f)$, классификатор метки классов $G_c(\cdot, \theta_y)$ и выберем некоторую классификационную функцию потерь $\mathcal{L}_c(\theta_f, \theta_y)$, которую будем минимизировать параметрами θ_f и

θ_y на данных из исходного домена (см. рис. 6). Такой процесс обучения обеспечит классификационную производительность нейронной сети на целевом домене.

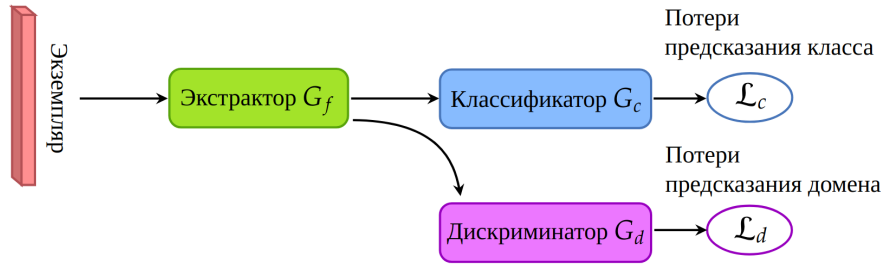


Рис. 6. Схематичное описание архитектуры DANN. Здесь экстрактор признаков G_f преобразует экземпляр данных в вектор признаков. Доменный дискриминатор G_d предсказывает домен экземпляра, а классификатор G_c преобразует вектор в вероятности принадлежности к классам. На основе их результатов рассчитываются функции потерь предсказания метки класса \mathcal{L}_c и предсказания метки домена \mathcal{L}_d

Кроме того, рассмотрим бинарный классификатор $G_d : \mathcal{X} \rightarrow [0, 1]$ с параметрами θ_d , который будет предсказывать метки самих доменов (например, метка «0» — лунки круглой формы, метка «1» — лунки овальной формы). В качестве функции потерь выберем бинарную кросс-энтропию:

$$\begin{aligned} \mathcal{L}_d(\theta_f, \theta_d) = & - \mathbb{E}_{x_s \in \mathcal{D}_s} \ln [G_d(G_f(x_s, \theta_d), \theta_d)] \\ & - \mathbb{E}_{x_t \in \mathcal{D}_t} \ln [1 - G_d(G_f(x_t, \theta_d), \theta_d)]. \end{aligned} \quad (19)$$

Для получения домен-инвариантных признаков на этапе обучения подбирают параметры θ_f так, чтобы *максимизировать* функцию потерь предсказания метки домена (сопоставляя распределения признаков как можно ближе), в то время как параметры θ_d *минимизируют* потери классификатора доменов. Такой процесс носит *состязательный* характер и обеспечивает построение домен-инвариантных признаков.

Осталось объединить две эти идеи в одну функцию потерь, поставив на общую функцию потерь

$$\mathcal{L}(\theta_f, \theta_c, \theta_d) = \mathcal{L}_c(\theta_f, \theta_c) - \lambda \mathcal{L}_d(\theta_f, \theta_d) \quad (20)$$

с весовым параметром $\lambda > 0$ минимаксную задачу оптимизации

$$\begin{aligned} \hat{\theta}_f, \hat{\theta}_c &= \arg \min_{\theta_f, \theta_c} \mathcal{L}(\theta_f, \theta_c, \hat{\theta}_d), \\ \hat{\theta}_d &= \arg \max_{\theta_d} \mathcal{L}(\hat{\theta}_f, \hat{\theta}_c, \theta_d). \end{aligned} \quad (21)$$

Оптимальные параметры $\hat{\theta}_d$ минимизируют функцию потерь классификации доменов \mathcal{L}_d , поскольку она входит в (20) с отрицательным знаком и в (21) поставлена задача на максимум. В то же время параметры $\hat{\theta}_f$ минимизируют потери предсказания меток классов \mathcal{L}_c и одновременно максимизируют потери классификации доменов \mathcal{L}_d . Такой процесс позволяет получить домен-инвариантные признаки. С учетом этого факта, параметры $\hat{\theta}_c$ обеспечивают высокую точность классификации нейронной сети не только на целевом, но уже и на исходном домене, т.к. они минимизируют функцию потерь предсказания меток классов \mathcal{L}_c .

Субдоменная адаптация. Рассмотренные ранее техники доменной адаптации преимущественно направлены на устранение *глобального доменного сдвига* распределений $\mathcal{S} \neq \mathcal{T}$, что не позволяет учитывать отношения между отдельными классами внутри доменов. В результате

адаптации может происходить потеря информации о межклассовых распределениях внутри доменов. Это, в свою очередь, может приводить к дополнительным ошибкам классификации на данных из целевого домена. Поэтому современные методы адаптации уделяют особое внимание *локальному доменному сдвигу* распределений $\mathcal{S}(y_s | x_s) \neq \mathcal{T}(y_t | x_t)$ классов. Такой подход позволяет более детально учитывать внутрикласовые распределения в доменах (см. рис. 7).

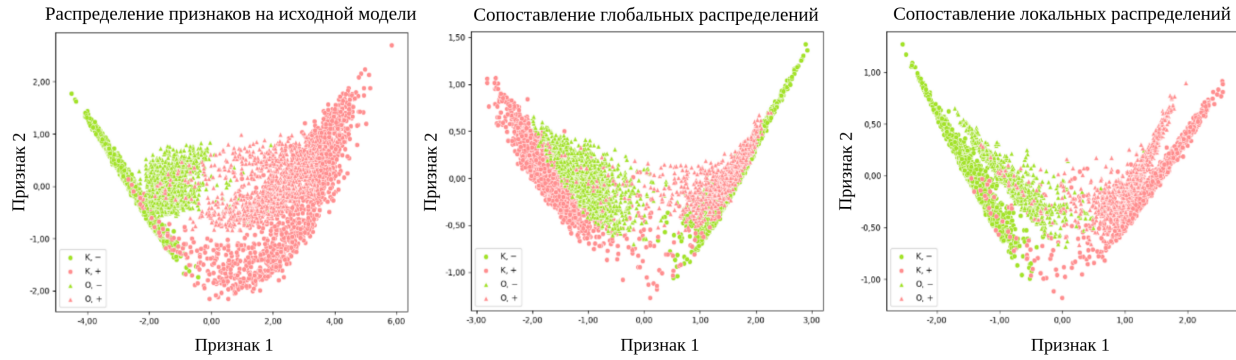


Рис. 7. Пример смещения классов при сопоставлении глобальных распределений доменов. На всех трёх изображениях в качестве исходного домена (обозначен круглыми метками) выступает набор лунок круглой формы «К»; в качестве целевого (обозначен треугольными метками) — набор лунок овальной формы «О». Классы наличия/отсутствия агглютинации обозначены красным и зелёным цветами соответственно. На левом изображении представлено распределение признаков векторов на исходной модели; На центральном изображении представлено неудачное сопоставление распределений — смешались разные классы исходного и целевого доменов (красный и зелёный). На правом изображении продемонстрирован результат работы одной из техник сопоставления локальных распределений

Проще говоря, вместо сопоставления глобальных распределений исходного и целевого доменов, эти техники фокусируются на группах данных, принадлежащих к одному и тому же классу. Разобьем домены \mathcal{D}_s и \mathcal{D}_t на C субдоменов $\mathcal{D}_s^{(c)}$ и $\mathcal{D}_t^{(c)}$ в зависимости от меток классов $c \in \{0, \dots, C - 1\}$. Классовые распределения признаков векторов обозначим за $p^{(c)}$ и $q^{(c)}$ и будем называть их *локальными* распределениями классов. Будем решать ту же задачу адаптации, но уже между субдоменами $\mathcal{D}_s^{(c)}$ и $\mathcal{D}_t^{(c)}$. Такая постановка задачи носит название *субдоменной адаптации*.

Стоит отметить, что интерес изучения следующих методов субдоменной адаптации побудил именно пример неудачного сопоставления глобальных распределений, изображенный на рис 7.

Deep Subdomain Adaptation Network. В качестве непараметрического¹ способа проверки равенства распределений метрика (14) широко используется в техниках адаптации для измерения расхождений между исходным и целевым доменами. Она позволяет сопоставлять глобальные распределения доменов, однако может не учитывать взаимоотношения между соответствующими субдоменами в разрезе каждого класса.

Для устранения локального сдвига в работе [17] была введена модификации метрики (14), учитывающая расстояния между локальными распределениями соответствующих классов:

$$d [p^{(c)}, q^{(c)}] = \mathbb{E}_c \left\| \mathbb{E}_{\xi \sim p^{(c)}} \phi(\xi) - \mathbb{E}_{\eta \sim q^{(c)}} \phi(\eta) \right\|_{\mathcal{H}}^2. \tag{22}$$

¹ Не требующего знаний о виде или свойствах распределений случайных величин

Помимо этого, было экспериментально показано, что минимизация функции потерь с учётом (22) сближает локальные распределения признаков векторов на выходах скрытых слоёв нейронных сетей.

Вместо выражения (22) запишем его оценку на дискретных наборах данных из доменов \mathcal{D}_s и \mathcal{D}_t :

$$\hat{d} [p^{(c)}, q^{(c)}] = \frac{1}{C} \sum_{c=0}^{C-1} \left\| \sum_{x_s \in \mathcal{D}_s} w_{x_s}^{(c)} \phi(x_s) - \sum_{x_t \in \mathcal{D}_t} w_{x_t}^{(c)} \phi(x_t) \right\|_{\mathcal{H}}^2, \quad (23)$$

где $w_{x_s}^{(c)}$ и $w_{x_t}^{(c)}$ — веса принадлежности векторов x_s и x_t к классу c . Для экземпляров из исходного домена \mathcal{D}_s в качестве веса вектора x_s с меткой класса y_s берут оценку соотношения классов

$$w_{x_s}^{(c)} = \frac{\mathbf{1}_{[y_s=c]}}{\sum_{(x_s, y_s) \in \mathcal{D}_s} \mathbf{1}_{[y_s=c]}}. \quad (24)$$

При этом возникает проблема вычисления весов $w_{x_t}^{(c)}$, так как метки классов y_t недоступны в домене \mathcal{D}_t . Такую проблему возможно решить на уровне реализации архитектур глубокого обучения. Для этого вычисляют ответ обучаемой нейронной сети $f : \mathcal{X}_t \rightarrow [0, 1]^C$ в виде вектора оценок соотношения классов $\hat{y} = f(x_t)$, а затем берут компоненту \hat{y}_c в качестве веса $w_{x_t}^{(c)}$.

Остаётся записать оценку (22)

$$\hat{d}_k(x, y) = \frac{1}{C-1} \sum_{c=0}^{C-1} \left[-2 \sum_{i,j=1}^{n,m} w_{x_i}^{(c)} w_{y_j}^{(c)} k(x_i, y_j) + \sum_{i \neq j}^n w_{x_i}^{(c)} w_{x_j}^{(c)} k(x_i, x_j) + \sum_{i \neq j}^m w_{y_i}^{(c)} w_{y_j}^{(c)} k(y_i, y_j) \right] \quad (25)$$

и на её основе функцию потерь, применительно к представлениям z_s^l и z_t^l экземпляров x_s и x_t на множестве скрытых слоёв $l \in L$ нейронной сети с весовым параметром $\lambda > 0$:

$$\mathcal{L}(z_s, y, z_t) = \mathcal{L}_c(G_c(z_s), y) + \lambda \sum_{l \in L} \hat{d}_k^2(z_s^l, z_t^l). \quad (26)$$

Dynamic Adversarial Adaptation Network. Идеи сопоставления локальных распределений нашли применение и в методах состязательной адаптации. В таких подходах улучшение показателей достигается либо согласованием глобальных [19], либо согласованием локальных распределений [56]. Более того, экспериментально было показано, что совместное сопоставление этих распределений приводит к лучшим результатам [12]. Однако в прикладных задачах эти распределения могут вносить совершенно различный вклад в различия между доменами, поэтому в современных техниках адаптации предлагают использование динамических подходов [11], позволяющих количественно оценивать важность маргинального и условного распределений для адаптации доменов.

Приведём формальное описание архитектуры (см. рис. 8). Как и раньше, пусть заданы экстрактор признаков G_f , классификатор метки классов G_c и некоторая классификационная функция потерь \mathcal{L}_c — например, кросс-энтропия (1).

Для сопоставления глобальных распределений используется введённый *глобальный* доменный дискриминатор G_d (DANN) с той же функцией потерь

$$\begin{aligned} \mathcal{L}_g(\theta_f, \theta_d) = & - \mathbb{E}_{\mathbf{x}_s \in \mathcal{D}_s} \ln [G_d(G_f(\mathbf{x}_s), \theta_f), \theta_d] \\ & - \mathbb{E}_{\mathbf{x}_t \in \mathcal{D}_t} \ln [1 - G_d(G_f(\mathbf{x}_t), \theta_f), \theta_d]. \end{aligned} \quad (27)$$

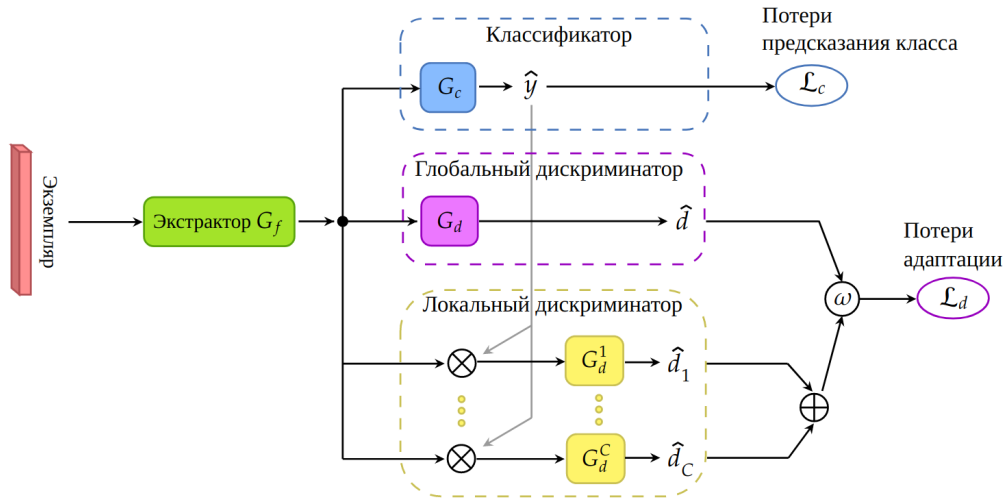


Рис. 8. Схематичное описание архитектуры DAAN. Она состоит из экстрактора признаков G_f , классификатора меток классов G_c , глобального доменного дискриминатора G_d и C локальных субдоменных дискриминаторов G_d^c . Символом \otimes условно обозначена операция умножения; символом \oplus условно обозначена операция сложения. Также за \hat{y} обозначен вектор оценок вероятностей принадлежности к классам; \hat{d} и \hat{d}_c — предсказанные метки доменов

Архитектура также содержит C локальных доменных дискриминаторов G_d^c , отвечающих за сопоставление распределений классов c в доменах. Они представляют из себя те же классификаторы меток домена, но применяемые к данным конкретного класса. В такой постановке появляется необходимость определения принадлежности к классу данных x для использования нужного классификатора. Поэтому, в силу отсутствия меток классов в целевом домене, используют предсказания $\hat{y} = G_c(z)$ в качестве весов признаковых векторов $G_f(x)$ для более точного учета вкладов каждого классификатора в общую функцию потерь

$$\mathcal{L}_l(\theta_f, \theta_d^c)_{c=1}^C = \sum_{c=1}^C \{ -\mathbb{E}_{\mathbf{x}_s \in \mathcal{D}_s} \ln [G_d^c(\hat{y}_c G_f(\mathbf{x}_s, \theta_f), \theta_d^c)] - \mathbb{E}_{\mathbf{x}_t \in \mathcal{D}_t} \ln [1 - G_d^c(\hat{y}_c G_f(\mathbf{x}_t, \theta_f), \theta_d^c)] \}. \quad (28)$$

Обучение нейронной сети происходит минимизацией функции потерь

$$\mathcal{L}(\theta_f, \theta_y, \theta_d, \theta_d^c)_{c=1}^C = \mathcal{L}_c - \lambda((1 - \omega)\mathcal{L}_g + \omega\mathcal{L}_l) \quad (29)$$

с динамическим фактором $\omega \in [0, 1]$. Он имеет смысл некоторого переключателя режимов нейронной сети: если $\omega \rightarrow 0$, то это означает, что преобладает большое расхождение между глобальными распределениями доменов, и в таком случае нейронная сеть должна быть переключена в режим сопоставления глобальных распределений (DANN). Если же $\omega \rightarrow 1$, то глобальные распределения сопоставлены достаточно близко, и в таком случае должен преобладать режим сопоставления локальных распределений классов. Фактор ω можно было бы рассматривать как гиперпараметр нейронной сети и подбирать его значение на валидационной выборке, однако целевой домен не содержит меток классов, поэтому приходится вычислять его другими способами.

В теории доменной адаптации известна ещё одна мера несоответствия распределений [57], называемая \mathcal{A} -расстоянием

$$d_{\mathcal{A}}(\mathcal{D}_s, \mathcal{D}_t) = 2(1 - 2\epsilon(h)), \quad (30)$$

где $\epsilon(h)$ — обобщение ошибки бинарного классификатора h , различающего домены \mathcal{D}_s и \mathcal{D}_t . На её основе в работе [11] предлагается использовать ошибки, учитывающие глобальную и

локальную структуру доменов одновременно:

$$d_{A,g}(\mathcal{D}_s, \mathcal{D}_t) = 2(1 - 2\mathcal{L}_g), \quad d_{A,l}(\mathcal{D}_s^{(c)}, \mathcal{D}_t^{(c)}) = 2(1 - 2\mathcal{L}_l^c). \quad (31)$$

Здесь \mathcal{L}_l^c — функция потерь локального субдоменного дискриминатора класса c . Наконец, динамический фактор ω определяют как

$$\hat{\omega} = \frac{d_{A,g}(\mathcal{D}_s, \mathcal{D}_t)}{d_{A,g}(\mathcal{D}_s, \mathcal{D}_t) + \sum_{c=0}^{C-1} d_{A,l}(\mathcal{D}_s^{(c)}, \mathcal{D}_t^{(c)})}. \quad (32)$$

5. ЭКСПЕРИМЕНТЫ

5.1. Метод определения агглютинации без использования нейронных сетей

Для того, чтобы использование нейронных сетей для решение задачи было оправданным, необходимо чтобы они решали задачу лучше, чем это делают более простые подходы, не требующие сбора набора данных. В дальнейшем все решения, основанные на нейронных сетях, будут сравниваться по точности с рассматриваемым здесь. Будет необходимо получить такой метод обучения нейронных сетей, который позволит получить модель с точностью выше, чем у предлагаемого решения и среднего врача.

Исследуемый метод основан на алгоритме поиска ключевых точек ORB [50]. Для него будут использоваться параметры из статьи [34]:

- Длина дескриптора — 512 бит
- Порог для коэффициента корреляции между отобранными ключевыми точками равен 0.3
- В качестве расстояния между дескрипторами используется расстояние Хэмминга, т.е. количество отличающихся бит

Для использования предложенного метода требуется выбрать эталонное изображение с агглютинацией. Точность определения агглютинации для этого метода определяется как средняя точность по эталонам и изображениям из тестовой выборки. Порог количества соответствий с эталоном будет выбирается по всем изображениям в наборе данных так, чтобы достигалось наилучшая точность. Полученный порог используется для классификации на тестовой выборке.

Таблица 1. Точность предсказания метода, основанного на ORB

Тип лунок	Количество лунок в обучающей выборке		
	16	32	64
Круглые лунки	91,3 %	92,0 %	91,4 %
Овальные лунки	88,7 %	89,4 %	89,5 %

5.2. Методы не требующие использование исходного домена

В данном разделе рассматривается только обучение на одном домене с малой обучающей выборкой. Её размеры составляют 16, 32 и 64 изображения, причем классы сбалансированы. Используемая модель - MobileNetV3 large [54], поскольку эта модель представляет наилучшее качество на полном наборе данных [52]. Размер входного тензора составляет $512 \times 512 \times 3$,

а количество обучаемых свёрточных слоёв — 28. Используемые аугментации были описаны ранее.

Перенос знаний. Перенос знаний — один из простейших методов дообучения нейронной сети. Обучение будет проводиться для разного количества замороженных слоёв, всего будет 1 – 4 незамороженных слоя. Этот эксперимент нужен для двух целей: во-первых, это один из методов, который может быть использован для обучения на малых данных. Во-вторых, полученное оптимальное количество незамороженных слоёв будет сохраниться и в следующем эксперименте. Для предобучения был использован набор данных ImageNet-1K [48] Его главным преимуществом является большое количество изображений, что позволяет эффективно с него дообучаться даже на сильно отличающемся домене.

Для обучения без дополнительных данных параметры следующие:

- Для оптимизации был использован ADAM [47];
- В качестве функции потерь используется бинарная перекрёстная энтропия;
- Обучение происходило в течение 240 эпох;
- Learning rate в первые 120 эпох равен 10^{-4} , в следующие 60 эпох равен 10^{-5} , и далее понижается до 10^{-6} ;
- Weight decay (параметр L2-регуляризации) равен $5 \cdot 10^{-5}$;
- Считаем, что наблюдается агглютинация, если выход сети превышает 0,5.

Результаты обучения представлены в таблицах:

Таблица 2. Точность модели после переноса знаний на круглых лунках

Количество обучаемых слоёв	Количество лунок в обучающей выборке		
	16	32	64
1	89,6 %	91,8 %	93,4 %
2	92,6 %	93,1 %	93,7 %
3	92,3 %	94,1 %	94,4 %
4	91,8 %	93,8 %	94,3 %

Таблица 3. Точность модели после переноса знаний на овальных лунках

Количество обучаемых слоёв	Количество лунок в обучающей выборке		
	16	32	64
1	89,2 %	91,4 %	92,8 %
2	92,0 %	92,8 %	93,3 %
3	91,7 %	93,7 %	93,6 %
4	91,4 %	93,3 %	93,8 %

Из таблиц 2 и 3 видно, что оптимальное количество незамороженных слоёв растёт при увеличении количества изображений в обучающей выборке и составляет 2/3/4 слоя для 16/32/64 изображений соответственно. При обучении оптимального числа слоёв получается превзойти

метод, основанный на ключевых точках: в худшем случае (для 16 круглых лунок) количество ошибок уменьшилось всего в 1,18 раз.

Метрическое обучение. Этот метод может дать дополнительный прирост в точности, поскольку требует не просто разделимости двух классов в признаковом пространстве, но и требует близости элементов из одного класса. Для дообучения будет использовано оптимальное количество слоёв, полученное в прошлом эксперименте (т.е. 2/3/4 слоя для 16/32/64 лунок соответственно).

Для обучения без дополнительных данных параметры следующие:

- Для оптимизации был использован ADAM [47];
- В качестве функции потерь используется триплетная функция потерь;
- Обучение происходило в течение 120 эпох;
- Learning rate в первые 60 эпох равен 10^{-4} , в следующие 30 эпох равен 10^{-5} , и далее понижается до 10^{-6} ;
- Weight decay (параметр L2-регуляризации) равен $5 \cdot 10^{-4}$;
- По векторам, которые возвращает нейронная сеть, обучается классификатор SVM с weight decay (параметр L2-регуляризации) равным 1,0.

Результаты обучения представлены в таблицах:

Таблица 4. Точность модели после метрического обучения на круглых лунках

Метод	Количество лунок в обучающей выборке		
	16	32	64
Триплетная функция потерь	94,6 %	95,3 %	95,9 %
Сеть сходства	94,2 %	95,6 %	96,1 %
Сеть прототипов	94,3 %	95,5 %	96,2 %

Таблица 5. Точность модели после метрического обучения на овальных лунках

Метод	Количество лунок в обучающей выборке		
	16	32	64
Триплетная функция потерь	93,0 %	93,8 %	95,3 %
Сеть сходства	92,2 %	94,1 %	95,6 %
Сеть прототипов	92,4 %	94,2 %	95,6 %

Из таблиц 4 и 5 видно, что удалось существенно превзойти как метод, основанный на ключевых точках, так и перенос знаний.

Таким образом, в сценарии без использования исходного домена для любого размера обучающей выборки наилучшее качество показывает метрическое обучение и точность достигает 96,2 % для 64 круглых лунок и 95,6 % для 64 овальных лунок. Таким образом, количество ошибок и для круглых и для овальных лунок уменьшилось примерно в 2,4 раза по сравнению с методом, основанным на ключевых точках. Использование более маленькой выборки даёт уменьшение числа ошибок только в 1,5 раза для круглых лунок и 1,4 раза для овальных лунок. Несмотря на более скромные результаты по сравнению с обучением на 64 лунках, прирост точности по сравнению с методом, основанным на ключевых точках существенный.

5.3. Методы доменной адаптации

Для чистоты наблюдения эффекта адаптации были обучены исходные модели f_s на данных только из одного домена с константой $\lambda = 0$ в функции потерь (16). На основании их средней точности предсказания будет проводиться оценка степени адаптации нейронных сетей к целевому домену.

В таблице 6 приведены результаты средней точности предсказания полученных моделей на каждом из двух доменов в отдельности. Нотации вида «К → К» означают, что модели обучались на исходном домене «К» лунок круглой формы и качество их предсказания измерялось на том же домене. Нотации вида «К → О» обозначают те же самые модели, но с измерением качества их предсказания на домене «О» лунок овальной формы.

Таблица 6. Качество распознавания (ассигасу) моделей на исходных доменах, где производилось обучение с учителем на тренировочных выборках размера 50% от общего числа данных

Адаптация	К → К	К → О	О → О	О → К
Точность, %	98,1	93,4	97,3	93,2

Средняя точность предсказания представлена шестью моделями, обученными на данных из исходного домена по одной и той же схеме со случайной инициализацией весов. Важно отметить, что оценка производилась на фиксированных во всех экспериментах тестовых выборках, состоящих из 1570 экземпляров лунок круглой формы и 972 экземпляров лунок овальной формы.

Результаты показывают, что средняя точность предсказания меток классов ухудшается на неизвестном (целевом) домене. Более того, ответы шести классификаторов становятся менее стабильными, о чем свидетельствует увеличение дисперсии в сравнении с известным (исходным) доменом.

Адаптация круглых лунок к овальным. Пусть исходным доменом является набор лунок круглой формы, а целевым — набор лунок овальной формы. Средняя точность предсказания моделей на целевом домене в зависимости от размера обучающей выборки представлена в таблице 7. С учётом дисперсии в каждой колонке (т.е. при фиксированном размере тренировочной выборки) жирным выделен лучший результат адаптации.

Таблица 7. Качество распознавания адаптированных моделей на целевом домене лунок овальной формы. Исходным доменом является набор лунок круглой формы. В заголовке указано процентное соотношение общего набора в тренировочной выборке. Через дробь указано соответствующее процентам количество лунок

Метод	50% / 970	30% / 486	10% / 214	5% / 98	1,6% / 32	1% / 20
DAN	96,3	96,0	95,7	95,2	94,4	93,8
CORAL	95,6	95,3	95,2	95,0	94,1	94,0
DANN	96,5	96,4	95,9	95,5	95,5	93,8
DSAN	96,6	96,4	96,0	95,8	95,4	94,4
DAAN	96,8	96,4	96,0	95,4	95,1	94,7

Эффект доменной адаптации проявляется уже при наличии 20 неразмеченных экземпляров лунок овальной формы в тренировочной выборке: средняя точность возросла с 93,8%

до 94,7%. Более того, для достижения точности 95,8% достаточно 100 экземпляров, а точность 96,0% достигается при использовании 214 лунок овальной формы. Наилучший результат адаптации составил 96,8% при использовании техники субдоменной адаптации DAAN на тренировочной выборке из 970 экземпляров.

Адаптация овальных лунок к круглым. Теперь исходным доменом возьмём набор лунок овальной формы, а целевым — набор лунок круглой формы. Средняя точность предсказания моделей на целевом домене в зависимости от размера обучающей выборки представлена в таблице 8. Как и прежде, жирным выделен лучший пример адаптации.

Таблица 8. Качество распознавания адаптированных моделей на целевом домене лунок круглой формы. Исходным доменом является набор лунок овальной формы. В заголовке указано процентное соотношение общего набора в тренировочной выборке. Через дробь указано соответствующее процентам количество лунок

Метод	50% / 1568	30% / 784	10% / 346	5% / 158	2% / 64	1% / 32
DAN	95,9	95,8	95,8	95,6	95,3	95,2
CORAL	96,2	96,0	95,9	95,6	95,2	95,1
DANN	96,5	96,2	96,0	95,7	95,3	95,2
DSAN	96,8	96,6	96,3	96,3	95,6	95,1
DAAN	97,2	96,8	96,5	96,4	96,1	95,8

Удивительно, что наилучший результат на каждом размере тренировочной выборки показал метод DAAN. Наличие 32 незамеченных лунок круглой формы позволяет ему повысить точность предсказания с 93,2% до 95,8%. При этом для преодоления качества 96,0% ему было достаточно 160 экземпляров. Самый высокий результат адаптации составил 97,2% качества при использовании тренировочной выборке из 1568 экземпляров.

Обсуждение. Как показали результаты, методы доменной адаптации DAN и CORAL способны демонстрировать эффективность только на достаточно больших наборах тренировочных данных, где им можно отдать предпочтение в силу вычислительной простоты. На практике данного исследования результаты этих методов начинают значительно увеличиваться на выборках от 100 экземпляров и выше, но всё ещё уступают дискриминативной модели DANN и современным техникам субдоменной адаптации.

Техника DANN, хоть и является простой с точки зрения реализации, даёт весомые результаты адаптации моделей, иногда превосходящие даже её модификации (см. выборку 30% таблицы 7). В силу простоты реализации и обучения можно отдавать ей предпочтение и на довольно малых объёмах (меньше 100 экземпляров) тренировочных выборок.

Техники субдоменной адаптации показывали наилучший результат во время проведения экспериментов. Их использование на больших тренировочных выборках, возможно, может быть чересчур оправданным, потому что их результаты в пределах погрешности несильно отличаются от техник доменной адаптации DAN (см. выборку 50% таблицы 7) и DANN (см. выборку 50% таблицы 8), но при этом на их обучение требуется больше времени в силу сложности архитектур. При этом техники DSAN и DAAN значительно выделяются на фоне остальных на объёмах выборок меньше ста экземпляров. Поэтому, при достаточно хороших вычислительных возможностях, основываясь на результатах экспериментов, можно отдавать предпочтение техникам субдоменной адаптации.

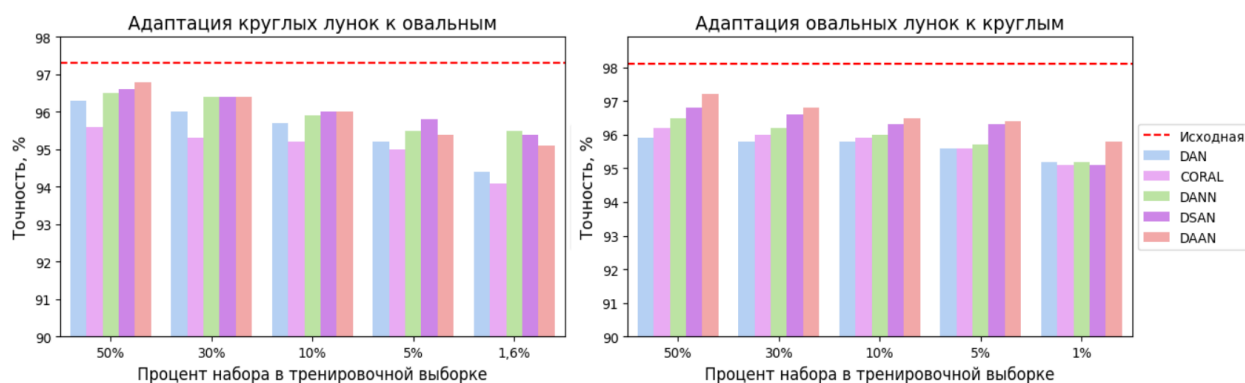


Рис. 9. Графическая иллюстрация сравнения качества распознавания адаптированных моделей на целевых доменах. Слева приведены результаты адаптации лунок круглой формы к овальной в зависимости от размеров тренировочных выборок. Справа — результаты адаптации лунок овальной формы к круглой. Красной штриховой линией показаны результаты предсказания моделей на исходных доменах

6. ЗАКЛЮЧЕНИЕ

Проведенное исследование является первой работой, направленной на изучение обучения нейронных сетей при малых данных и доменной адаптации для задачи определения агглютинации крови. Удалось достичь точности в 96,2 % для круглых лунок и 95,6 % для овальных лунок при использовании только 64 изображений не используя информации о других доменах. Используя информацию о реакции агглютинации на одном домене можно адаптировать к другому домену, достигнув схожей точности при обучении на 32–64 неразмеченных экземпляров в тренировочной выборке. Полученные результаты имеют важное значение для компьютерной медицинской диагностики, поскольку позволяют упростить разработку алгоритмов определения групп крови, инфекционных заболеваний и других разновидностей тест-систем, основанных на реакции агглютинации. Возможными направлениями дальнейших работ являются использование методов обучения с известным исходным доменом и с известной разметкой и изучение свойств адаптации моделей в контексте других тест-систем, использующих реакцию агглютинации. Другой научный интерес может быть вызван изучением доменной адаптации на других архитектурах нейронных сетей и изучением гиперпараметров обучения.

СПИСОК ЛИТЕРАТУРЫ

1. World Health Organization et al. Global status report on blood safety and availability 2021. – World Health Organization, 2022.
2. S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.
3. Wang M., Deng W. Deep visual domain adaptation: A survey // *Neurocomputing*. – 2018. – Т. 312. – С. 135-153.
4. Guan H., Liu M. Domain adaptation for medical image analysis: a survey // *IEEE Transactions on Biomedical Engineering*. – 2021. – Т. 69. – №. 3. – С. 1173-1185.
5. D. Kumar, C. Kumar, and M. Shao, “Cross-database mammographic image analysis through unsupervised domain adaptation,” in *IEEE International Conference on Big Data*, 2017, pp. 4035–4042.
6. S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, “Domain adaptation via transfer component analysis,” *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011
7. B. Sun, J. Feng, and K. Saenko, “Return of frustratingly easy domain adaptation.” in *Association for the Advancement of Artificial Intelligence*, vol. 6, no. 7, 2016, p. 8.

8. B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in European conference on computer vision. Springer, 2016, pp. 443–450.
9. J. Wang, Y. Chen, S. Hao, W. Feng, and Z. Shen, "Balance distribution adaption for transfer learning," 2017.
10. J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in 2018 ACM International Conference on Multimedia (ACM MM). ACM, 2018, pp. 402–410.
11. Yu C. et al. Transfer learning with dynamic adversarial adaptation network //2019 IEEE international conference on data mining (ICDM). – IEEE, 2019. – С. 778-786.
12. Long M. et al. Deep transfer learning with joint adaptation networks //International conference on machine learning. – PMLR, 2017. – С. 2208-2217.
13. Laiz P., Vitria J., Segui S. Using the triplet loss for domain adaptation in WCE //Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. – 2019. – С. 0-0.
14. A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Scholkopf, and A. Smola, "A kernel two-sample test," The Journal of Machine Learning Research, vol. 13, no. 1, pp. 723–773, 2012.
15. Schölkopf B., Tsuda K., Vert J. P. (ed.). Kernel methods in computational biology. – MIT press, 2004.
16. M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in International conference on machine learning. PMLR, 2015, pp. 97–105.
17. Zhu Y. et al. Deep subdomain adaptation network for image classification //IEEE transactions on neural networks and learning systems. – 2020. – Т. 32. – №. 4. – С. 1713-1722.
18. J. Yang, T. Vetterli, P. P. Balte, R. G. Barr, A. F. Laine, and E. D. Angelini, "Unsupervised domain adaption with adversarial learning (UDAA) for emphysema subtyping on cardiac CT scans: The mesa study," in 16th IEEE International Symposium on Biomedical Imaging (ISBI), 2019, pp. 289–293.
19. Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In International Conference on Machine Learning, pages 1180–1189, 2015.
20. He K. et al. Deep residual learning for image recognition //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 770-778.
21. Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks //Advances in neural information processing systems. – 2012. – Т. 25
22. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition //arXiv preprint arXiv:1409.1556. – 2014.
23. Szegedy C. et al. Going deeper with convolutions //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 1-9.
24. Yosinski J. et al. How transferable are features in deep neural networks? //Advances in neural information processing systems. – 2014. – Т. 27.
25. Chu B. et al. Best practices for fine-tuning visual classifiers to new domains //Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14. – Springer International Publishing, 2016. – С. 435-442.
26. Anilkumar B., Kumar P. R. Tumor classification using block wise fine tuning and transfer learning of deep neural network and KNN classifier on MR brain images // Int. J. Emerg. Trends Eng. Res. 2020. V. 8(2). P. 574–583.
27. Anilkumar B., Kumar P. R. Tumor classification using block wise fine tuning and transfer learning of deep neural network and KNN classifier on MR brain images // Int. J. Emerg. Trends Eng. Res. 2020. V. 8(2). P. 574–583.
28. Weiss K., Khoshgoftaar T. M., Wang D. D. A survey of transfer learning //Journal of Big data. – 2016. – Т. 3. – С. 1-40.

29. Ferraz A. Automatic system for determination of blood types using image processing techniques //2013 IEEE 3rd Portuguese meeting in bioengineering (ENBENG). – IEEE, 2013. – С. 1-6.
30. Rahman S. et al. Blood group detection using image processing techniques : дис. – BRAC University, 2017.
31. Sahastrabuddhe A. P., Ajjij S. D. Blood group detection and RBC, WBC counting: an image processing approach //Int. J. Eng. Comput. Sci. – 2016. – Т. 5. – №. 10. – С. 18635-18639.
32. Yamin A. et al. Image processing based detection and classification of blood group using color images //2017 International Conference on Communication, Computing and Digital Systems (C-CODE). – IEEE, 2017. – С. 293-298.
33. ATICI H., Koçer H. E., Kader S. Determination of blood group by image processing using digital images //Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Dergisi. – 2020. – Т. 7. – №. 2. – С. 649-659.
34. Odeh N. et al. An Efficient System for Automatic Blood Type Determination Based on Image Matching Techniques //Applied Sciences. – 2021. – Т. 11. – №. 11. – С. 5225.
35. Sklavounos A. A. et al. Digital microfluidic hemagglutination assays for blood typing, donor compatibility testing, and hematocrit analysis //Clinical chemistry. – 2021. – Т. 67. – №. 12. – С. 1699-1708.
36. Song Y. et al. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities (2022) //arXiv preprint arxiv:2205.06743.
37. Kwitt R., Hegenbart S., Niethammer M. One-shot learning of scene locations via feature trajectory transfer //Proceedings of The IEEE conference on computer vision and pattern recognition. – 2016. – С. 78-86.
38. Chen Z. et al. Semantic feature augmentation in few-shot learning //arXiv preprint arXiv:1804.05298. – 2018. – Т. 86. – №. 89. – С. 2.
39. Kaya M., Bilge H. Ş. Deep metric learning: A survey //Symmetry. – 2019. – Т. 11. – №. 9. – С. 1066.
40. Huang S. et al. Adaptively weighted k-tuple metric network for kinship verification //IEEE Transactions on Cybernetics. – 2022.
41. Sung F. et al. Learning to compare: Relation network for few-shot learning //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 1199-1208.
42. Snell J., Swersky K., Zemel R. Prototypical networks for few-shot learning //Advances in neural information processing systems. – 2017. – Т. 30.
43. Finn C., Abbeel P., Levine S. Model-agnostic meta-learning for fast adaptation of deep networks //International conference on machine learning. – PMLR, 2017. – С. 1126-1135.
44. Nichol A., Schulman J. Reptile: a scalable metalearning algorithm //arXiv preprint arXiv:1803.02999. – 2018. – Т. 2. – №. 3. – С. 4.
45. Xu M. et al. A comprehensive survey of image augmentation techniques for deep learning //Pattern Recognition. – 2023. – Т. 137. – С. 109347.
46. Пищев П. М. и др. Определение достаточного объема обучающей выборки для достижения заданной точности на примере задачи определения группы крови //ИТиС 2023: Сборник трудов 47й междисциплинарной школы-конференции ИППИ РАН "Информационные технологии и системы 2023". – М.: Институт проблем передачи информации им. А.А. Харкевича РАН, 2023
47. Kingma D. P., Ba J. Adam: A method for stochastic optimization //arXiv preprint arXiv:1412.6980. – 2014.
48. Deng J. et al. Imagenet: A large-scale hierarchical image database //2009 IEEE conference on computer vision and pattern recognition. – Ieee, 2009. – С. 248-255.
49. Cortes C., Vapnik V. Support-vector networks //Machine learning. – 1995. – Т. 20. – С. 273-297.
50. Rublee E. et al. ORB: An efficient alternative to SIFT or SURF //2011 International conference on computer vision. – Ieee, 2011. – С. 2564-2571.

51. Корчагин С. А. и др. Алгоритм определения групп крови по изображениям серологических планшетов //Компьютерная оптика. – 2023. – Т. 47. – №. 6. – С. 958-967.
52. Korchagin S. et al. Image-based second opinion for blood typing //Health Information Science and Systems. – 2024. – Т. 12. – №. 1. – С. 28.
53. Gupta S. et al. Machine Learning to Automate the Visual Interpretation of Chemical Agglutination Tests //2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops). – IEEE, 2022. – С. 230-235.
54. Howard A. et al. Searching for mobilenetv3 //Proceedings of the IEEE/CVF international conference on computer vision. – 2019. – С. 1314-1324.
55. Chechik G. et al. Large scale online learning of image similarity through ranking //Journal of Machine Learning Research. – 2010. – Т. 11. – №. 3.
56. *Pei Z., Cao Z., Long M., Wang J.* Multi-adversarial domain adaptation // Thirty-Second AAAI Conference on Artificial Intelligence. 2018.
57. *Ben-David S.* [et al]. Analysis of representations for domain adaptation // Advances in neural information processing systems. 2006. Т. 19.

Domain adaptation for neural networks with small training sets for the blood typing problem

P. Pishchev, D. Sharapov, S. Korchagin, E. Ershov

Modern deep learning algorithms can achieve high prediction performance in various image classification tasks. Typically, a sufficiently large amount of data from each class is required in the training dataset. However, collecting a new set of labeled data for similar classification tasks can be labor-intensive and expensive. To address this issue, there are methods for adapting to a new domain based on data from the source domain. This paper investigates domain adaptation techniques in the context of recognizing blood agglutination reactions in wells of serological plates under conditions of limited data in the target domain. The domains consist of round and oval wells with two classes: presence or absence of agglutination reaction. In a scenario of training without a source domain, a classification accuracy of 95.6% was achieved on 32 round wells and 94.2% on 32 oval wells. In a scenario of training without labels in the target domain, a classification accuracy of 95.8% was achieved on 32 round wells and 95.5% on 32 oval wells.

KEYWORDS: domain adaptation, Blood typing, few-shot learning