УДК 004.422+004.9

АТРИБУЦИЯ АРХИВНЫХ РУКОПИСНЫХ ПИСЕМ С ИСПОЛЬЗОВАНИЕМ СИАМСКИХ НЕЙРОННЫХ СЕТЕЙ

Н. М. Пронина[0009-0008-1336-4512]

НИУ «Высшая школа экономики», г. Москва, Россия

natalka-pronina@mail.ru

Аннотация

Предложен метод автоматической атрибуции архивных рукописных писем на основе сиамской нейронной сети, решающий ключевую проблему цифровой гуманитаристики – установление авторства исторических документов. Актуальность исследования обусловлена массовой оцифровкой архивов XVII–XIX вв., атрибуция которых затруднена из-за неполных исходных сведений об авторах.

Метод адаптирован к работе с реальным корпусом текстов и учитывает характерные для архивов проблемы: некачественные оцифровки, значительную вариативность почерка и выраженный дисбаланс классов (от 1 до 50 и более образцов на автора). Применение сиамской архитектуры позволяет получать дискриминативные векторные представления, эмбеддинги, на основе которых выполняется не только классификация документов известных авторов, но и эффективно выявляются рукописи, не принадлежащие ни одному из них. Это сужает круг кандидатов для последующей экспертной проверки.

Представлен алгоритм предобработки данных и проведено сравнительное исследование двух подходов к анализу текста: на уровне фрагментов изображения (300 × 300 пикселей) и уровне отдельных строк. Разработанный инструмент предлагает архивным работникам и филологам эффективное решение для предварительной сортировки и атрибуции крупных массивов рукописных документов.

Ключевые слова: *сиамская нейронная сеть, идентификация, верификация, атрибуция, рукописный текст, архивные документы, сверточная нейронная сеть, рекуррентная нейронная сеть.*

ВВЕДЕНИЕ

С развитием цифровых технологий все большее значение приобретает автоматизация анализа исторических документов. Особую актуальность приобретает задача атрибуции рукописных текстов – установления авторства и проверки подлинности на основе уникальных характеристик почерка (см., например, рис. 1). Традиционно эта задача решалась экспертами-палеографами, однако автоматические методы анализа могут значительно ускорить и упростить этот процесс, особенно при работе с большими архивами оцифрованных документов.

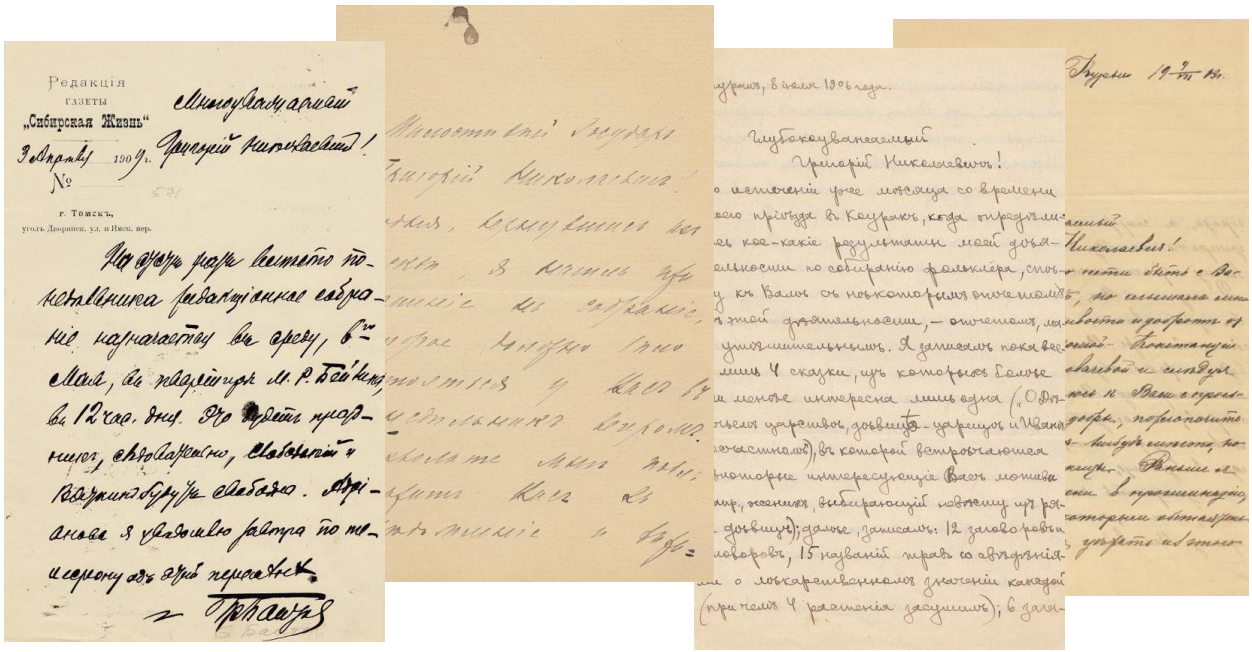


Рис. 1. Письма к Г. Н. Потанину.

Имеется несколько особенностей автоматической атрибуции рукописных текстов:

- дисбаланс количества образцов для разных авторов. Метод должен качественно выделять авторов и с малым количеством примеров в архиве;

- учет индивидуальных особенностей почерка при наличии значительной вариативности даже у одного автора;
- различение авторов со схожими стилями письма;
- обработка документов разного качества.

В настоящей работе предложен метод атрибуции, основанный на использовании сиамских нейронных сетей для сравнения и анализа уникальных характеристик почерка, стиля письма. Такой способ обучения позволяет получать мощные дискриминационные признаки изображения, эмбеддинги, на основе которых можно произвести качественную классификацию почерка автора, обойдя проблемы, описанные выше.

Проведено сравнение двух архитектур:

- картиночной модели на основе сверточной сети ResNet18 [1], анализирующей локальные паттерны [2];
- строчной модели, состоящей из сверточного энкодера и рекуррентного модуля LSTM, который рассматривает строку как последовательность столбцов пикселей.

Актуальность настоящей работы обусловлена тем, что машинный поиск текстов с почерком определенного автора в составе больших баз данных растровых изображений рукописных документов, во-первых, значительно расширяет возможности исследователей, в частности литературоведов или историков, в верификации того или иного лица в архивных собраниях текстов. Во-вторых, он дает архивным работникам удобный инструмент для автоматической классификации: ранжированный список сформированных авторов, выявленный программой, может служить основой для описания рукописи и внесения соответствующей информации в архив.

ПОСТАНОВКА ЗАДАЧИ

В качестве объекта исследования взят архив рукописных писем к Григорию Николаевичу Потанину[1]. В архиве содержатся сканы первых страниц писем (рис.

[1] Григорий Николаевич Потанин (21 сентября 1835 г. – 30 июня 1920 г.) — русский географ, этнограф, один из основателей и крупнейший идеолог сибирского областничества, член Императорского Русского географического общества (ИРГО), почетный член совета Томского технологического института.

1), адресованных Г. Н. Потанину (рис. 2), от различных авторов. В архиве упоминается 811 авторов, в том числе «неизвестные», для каждого приведено от 1 до 67 писем. Всего писем – 2604. Писем с неизвестным автором – 40.

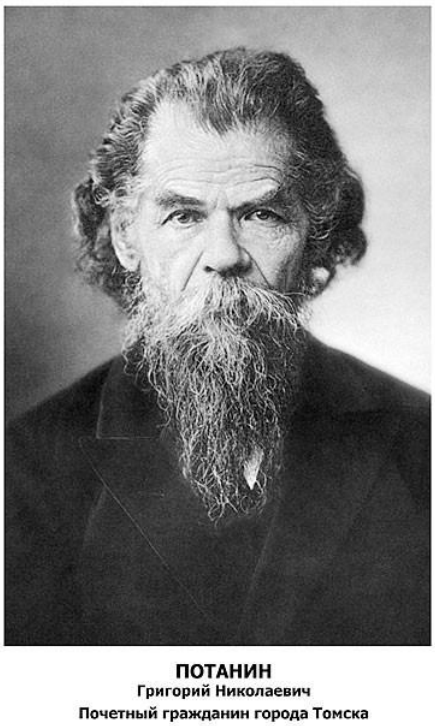


Рис. 2. Григорий Николаевич Потанин.

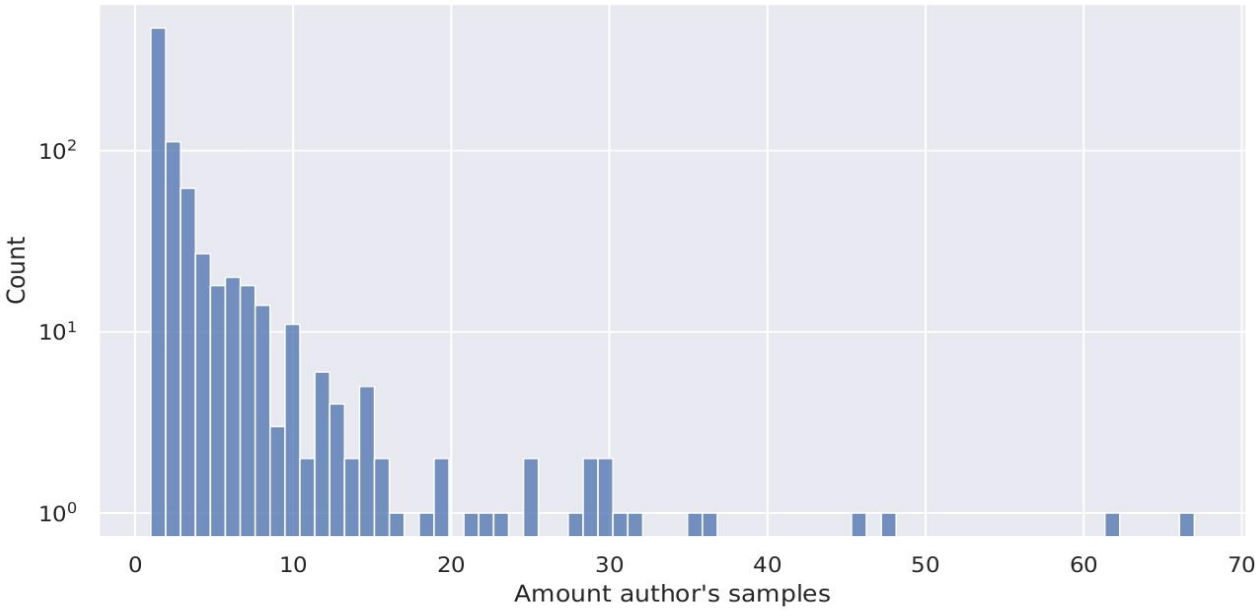


Рис. 3. Распределение количества образцов для каждого автора.

На рис. 3 приведено распределение количества писем. Видно, что подавляющее большинство имеет по одному образцу. Наблюдается сильный дисбаланс классов. Для понимания масштабов проблемы стоит отметить: два и более образца имеют 328 авторов, десять и более – всего 54 автора.

Формальная постановка задачи: требуется разработать алгоритм, входом которого является сканированный неатрибутированный документ, на выходе – ранжированный по убыванию вероятности список возможных его авторов, а также вероятность того, что автор документа неизвестен, т. е. не упоминался в архиве.

ОСНОВНЫЕ ПОНЯТИЯ

Сиамская нейронная сеть

Решить главную проблему – малое количество примеров для большинства авторов – планировалось с помощью алгоритма однократного обучения – *сиамской нейронной сети* (рис. 4). Сиамские сети были впервые представлены в начале 1990-х годов Бромли и Лекуном для решения задачи верификации подписи [3]. Данный тип сети для глубокого обучения использует две или более идентичных подсетей с одинаковой архитектурой, также они используют одни и те же весовые параметры для обучения.

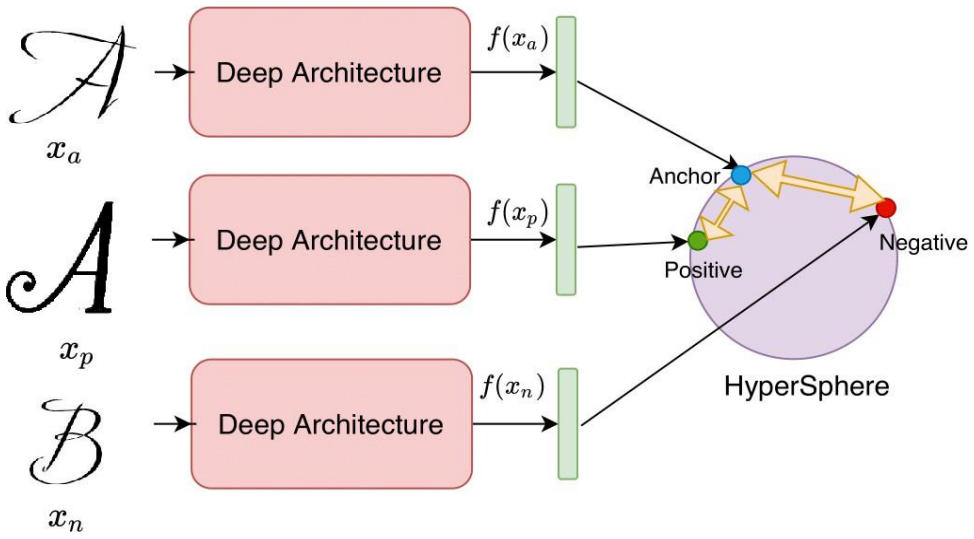


Рис. 4. Обучение сиамской сети с помощью триплетной функции потерь.

Сиамские сети особенно полезны при работе с большим количеством категорий, в каждой из которых представлено лишь несколько объектов. В таких условиях традиционным глубоким сверточным нейросетям не хватает данных для обучения. Кроме того, добавление новых разновидностей потребовало бы изменения архитектуры модели и ее повторного обучения. Вместо этого сиамская сеть решает задачу бинарной классификации пар: определяет, относятся ли два примера к одной группе или разным. Такой подход обеспечивает гибкость и эффективность при ограниченном объеме данных.

Такой вид сетей позволяет получить векторы признаков, эмбеддинги двух объектов, отражающие их семантическое сходство или различие. Примеры приложений для сиамских сетей: верификация подписи [3], распознавание лиц [4], идентификация перефразирования предложений [5].

Метки класса для авторов получаются после обучения простого классификатора на эмбеддингах. В настоящей работе использована двухслойная полносвязная нейронная сеть. Аналогичный метод классификации символов был использован в [6].

Две наиболее популярные функции потерь для обучения сиамских нейронных сетей – контрастивная попарная функция потерь (pairwise contrastive loss function) [7, 8] и триплетная функция потерь (triplet loss function) [9].

Контрастивная попарная функция потерь

Контрастивная попарная функция потерь (pairwise contrastive loss function) принимает на вход пару объектов (*x* и *y*), каждый из которых относится либо к положительному, либо к отрицательному классу:

$$L(x, y) = (1 - Z(x, y))\, d^2(x, y) + Z(x, y) \max(0,\ margin\ -\ d^2(x, y)),$$

$$Z(x, y) = \begin{cases} 0 \mid\ x, y\ \text{ из одного класса}, \\ 1 \mid\ x, y\ \text{ из разных классов}, \end{cases}$$

$$d(x, y) = \| x - y \|_p\ .$$

При минимизации такой функции потерь расстояние между объектами одного класса минимизируется, а расстояние между объектами разных классов стремится стать больше заранее заданного отступа (margin).

Триплетная функция потерь

Улучшением контрастивной функции потерь является триплетная функция потерь (triplet loss function). В отличие от первой здесь используются три объекта: объект рассматриваемого класса; якорь (anchor), с которым будет проводиться сравнение; а также два других объекта: принадлежащий к тому же классу, позитив (*positive*), и объект противоположного класса, негатив (negative):

$$L(a, p, n) = \max\{d(a, p\} - d\{a, n\} + margin, 0\},$$

$$d(x, y) = \| x - y \|_p .$$

При минимизации такой функции потерь расстояние между объектами одного класса (anchor и positive) уменьшается, и увеличивается расстояние между объектами разных классов (anchor и negative).

Параметр margin – это заранее задаваемый параметр, показывающий, за какую разность расстояний следует штрафовать, т. е. при margin = 0 достаточно, чтобы позитивный объект был ближе к якорному, чем негативные. С параметром margin = 1 минимизация будет происходить до тех пор, пока позитивный объект не станет ближе, чем негативный, хотя бы на 1.

При обучении модели с триплетной функцией потерь требуется меньше образцов для сходимости, поскольку сеть обновляется одновременно, используя больше объектов, включая как похожие, так и непохожие образцы. Поэтому в работе была использована данная функция потерь.

Перечислим преимущества в задаче атрибуции [9]:

- эффективное обучение при дисбалансе классов;
- формирование компактных кластеров для одного автора;
- устойчивость к вариациям почерка.

Меры качества

В задачах классификации с большим количеством классов стандартные показатели, такие как правильность (accuracy), недостаточно информативны, поскольку получить высокую правильность модели практически невозможно. Например, если всего классов 100 и модель имеет невысокую долю точных угадываний, но при этом верный класс входит в пять самых вероятных классов, то для пользователя такая модель имеет вполне хорошее качество.

Поэтому в задачах с огромным числом категорий часто используют альтернативные показатели из задач ранжирования и рекомендательных систем. Например, hits@k оценивает, попадает ли истинный класс в первые *k* значений, предсказанных моделью. Этот показатель позволяет гибко оценить качество модели, не требуя точного совпадения единственного предсказания с истиной.

Определение. hits@k описывает долю истинных объектов, которые появляются в первых *k* объектах отсортированного рангового списка:

$$\text{hits@k} = \frac{1}{|\text{I}|}\sum_{r\in \text{I}} \text{I}[r \le k],$$

где *I* – множество рангов. Нетрудно заметить, что hits@1 полностью совпадает с правильностью (accuracy).

В задачах классификации с сильным дисбалансом классов правильность (accuracy) может вводить в заблуждение, т. к. она не учитывает распределение объектов по категориям. Например, если один класс составляет 90% данных, модель с константным предсказанием достигнет точности в 90%, хотя на самом деле она не решает задачу классификации для малых классов. Поэтому в таких случаях предпочтительнее использовать показатели, чувствительные к дисбалансу: точность (precision*),* чувствительность *(*recall), F1, а также их макро- (macro) и взвешенные (weighted) варианты. Они позволяют более адекватно оценить качество модели, учитывая как способность правильно детектировать редкие классы (*recall*), так и точность предсказаний (precision).

Определение. Точность (Precision) – это доля верно предсказанных положительных объектов среди объектов, которые модель отнесла к положительному классу:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

Определение. Чувствительность (Recall*)* – это доля верно предсказанных положительных объектов среди всех реальных положительных объектов:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

Определение. *F1* – это гармоническое среднее точности и чувствительности, которое позволяет оценить баланс между ними:

$$\text{F1} = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}.$$

Определение. Precision@k, Recall@k, F1@k – аналоги соответствующих показателей, рассчитанные с поправкой предсказаний модели: если истинный класс находится в первых *k* самых вероятных классах, то предсказанный моделью класс заменяется на истинный. Далее соответствующий показатель вычисляется на исправленных предсказаниях.

Определение. Macro-Precision@k, Macro-Recall@k, Macro-F1@k – усреднение соответствующей метрики по всем классам, *N* – число всех классов:

$$\text{Macro - Precision@k} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}}\left(\text{Precision@k}\right)_{\text{i}},$$

$$\text{Macro - Recall@k} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}}\left(\text{Recall@k}\right)_{\text{i}},$$

$$\text{Macro - F1@k} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}}\left(\text{F1@k}\right)_{\text{i}}.$$

МЕТОД РЕШЕНИЯ

Бинаризация

Чтобы предотвратить переобучение модели на фон, т. е. плохую обобщающую способность модели по причине того, что в данных присутствует дополнительная информация – фон, от которой не должен зависеть результат, бинаризуем изображения с помощью предобученной нейронной сети DocEnTr [10], состоящей из трансформерного энкодера и декодера (рис. 5). Входное изображение разбивается на фрагменты, которые преобразуются в эмбеддинги, к ним добавляется информация о местоположении фрагмента. Результирующая последовательность векторов подается в энкодер для получения скрытых представлений, далее полученные скрытые представления подаются в декодер для

получения вектора, который линейно проецируется на векторы пикселей, представляющих участки выходного изображения.

Рис. 5. Бинаризация с помощью DocEnTr.

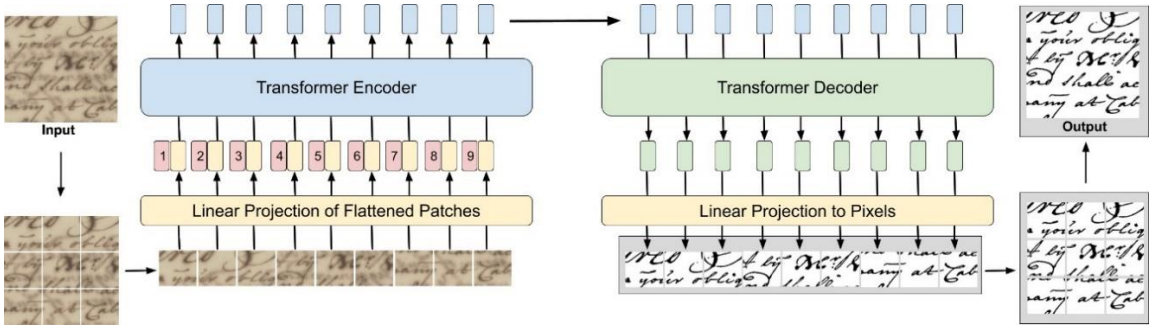


Рис. 6. Архитектура сети для бинаризации (источник: DocEnTr).

На рис. 6 представлен пример бинаризации сложных примеров. Для предотвращения попадания «пустых» изображений в обучающую выборку ауг-

ментация производилась таким образом, чтобы доля белых пикселей была не более 95%.

Нарезка на строки

Для обучения строчной модели необходимо нарезать на строки каждую страницу. Выделение строк производилось с помощью библиотеки kraken для Python [11]. В качестве результата выделения одной строки модуль выдает последовательность координат строки, которая описывает область строки, а также ломаную, задающую линию строки. Все эти ломаные показаны на рис. 7 разными цветами для каждой строки. Видно, что для ровного почерка выделение строк выполняется качественно, но, если строки изгибаются, алгоритм разрезает на экстремально маленькие подстроки, что мешает обучению строчной сети. Возможно, эту проблему можно решить постобработкой результата: слиянием подстрок в одну строку. Наивное решение – слияние строк на основании близости ординат – в некоторых случаях приводило к неправильному слиянию с чужой строкой. Поэтому было принято решение не делать постобработку, а просто удалить выбросы из выборки.

Рис. 7. Выделение строк с помощью библиотеки kraken.

По распределению длин строк (рис. 8) видно, что библиотека kraken выдает очень много коротких строк. Второй пик отражает правильно выделенные строки (средняя ширина изображения как раз около 500 пикселей).

В распределении высот строк (рис. 9) также наблюдаются выбросы. Так, алгоритм выделяет строки, написанные вертикально. Они также удаляются.

Постобработка, использованная в работе, состоит в следующем:

- удаление всех строк с шириной менее 300 пикселей;
- удаление всех строк высотой более 100 пикселей;
- линейное приведение (сжатие/расширение) всех строк к высоте 50 пикселей.

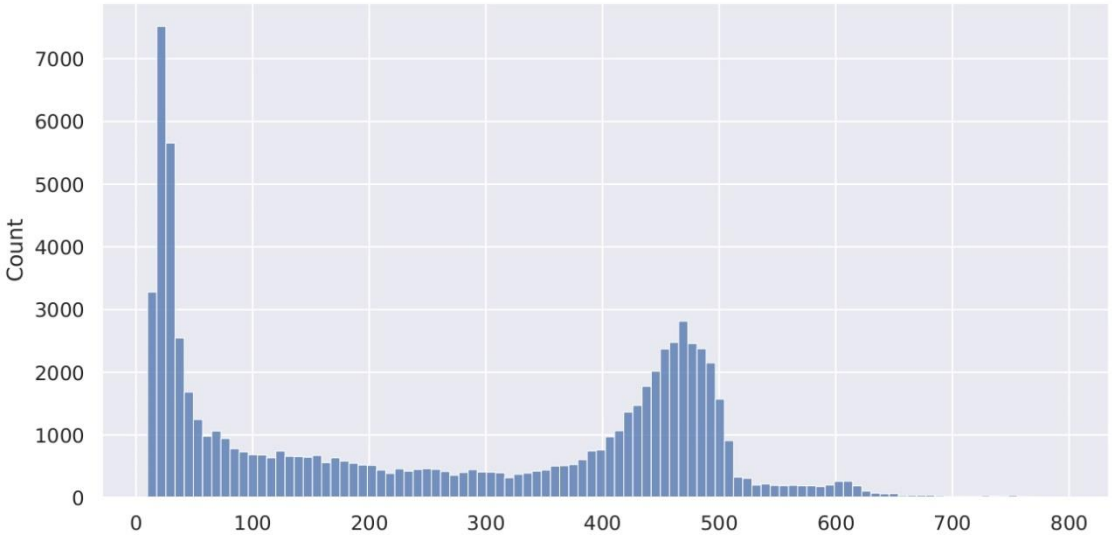


Рис. 8. Распределение длин строк.

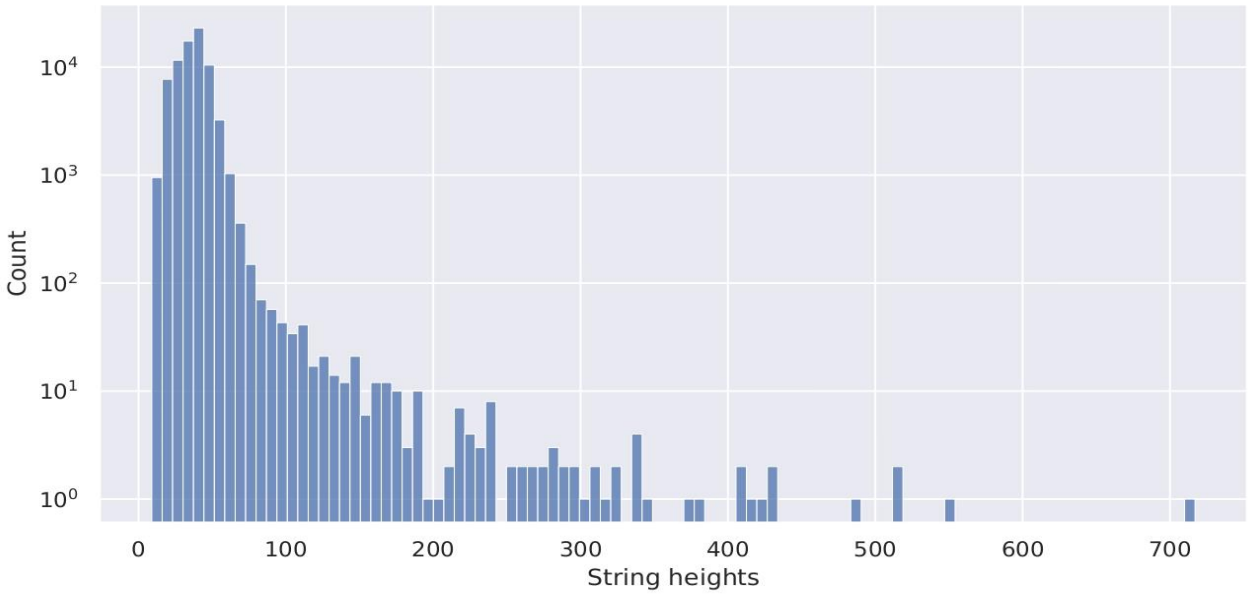


Рис. 9. Распределение высот строк.

Картиночная модель

Была применена идея обучения сиамской сети на бинарных изображениях, фрагментах размера 300 × 300 пикселей. Был взят предобученный ResNet18. Последний слой имеет размерность 1000 на выходе, поэтому сиамская сеть выучила 1000-размерный эмбеддинг изображения. Обучение проводилось только для последних двух слоев, 513000 обучаемых параметров.

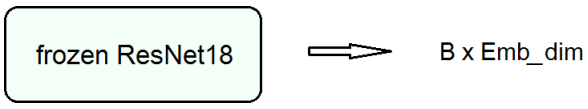


Рис. 10. Архитектура картиночной модели.

При обучении в сеть будут подаваться тройки изображений, триплеты: якорь (anchor), позитив (positive), негатив (negative*)*. В качестве якоря подается очередная строка, в качестве позитива – случайная строка того же автора, в качестве негатива – случайная строка другого автора. При оценке качества сиамской сети правильность (accuracy) вычисляется как доля триплетов, для которых евклидово расстояние между эмбеддингами якоря и позитива меньше, чем между эмбеддингами якоря и негатива.

На полученных эмбеддингах обучим классификатор: двухслойную полносвязную нейронную сеть с 513538 параметрами.

Итоговое предсказание для письма получено с помощью агрегации выходов сети для нескольких случайных фрагментов этого письма. В качестве функции агрегации было выбрано усреднение.

Строчная модель

В качестве строчной модели была выбрана гибридная модель, сочетающая сверточные и рекуррентный слои. Для обработки последовательностей строк она обучалась со случайной инициализацией.

Сверточный блок состоит из трех слоев, включающих: свертку, нормализацию по батчу, активацию и max-пуллинг с асимметричным шагом. Последний выбран, чтобы по высоте строки сжимать меньше, чем по ширине. В качестве

рекуррентного слоя была выбрана однонаправленная LSTM (205507 обучаемых параметров).

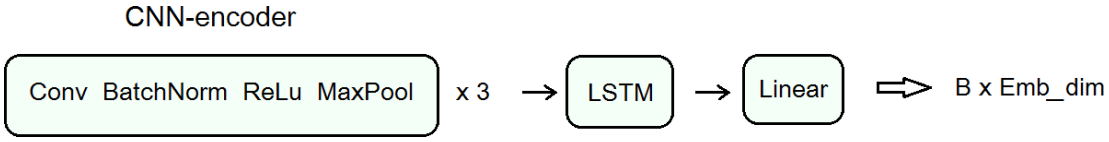


Рис. 11. Архитектура строчной модели.

Классификатор тот же самый – 513538 параметров.

Итоговое предсказание для письма производилось с помощью усреднения выходов сети для всех строк данного письма.

ЭКСПЕРИМЕНТЫ

Были проведены эксперименты с разным числом классов и способом обучения. Всюду далее термин *сбалансированный (balanced)* указывает на то, что при обучении модели объекты подавались равновероятно для разных классов. По умолчанию же использован обычный способ – каждый объект подается равновероятно, поэтому классы с бóльшим числом примеров при обучении встречаются чаще остальных. В таком смысле обучение является *несбалансированным*.

Картиночная модель

Картиночная модель обучается классифицировать фрагменты рукописных документов размера 300 × 300 пикселей, но итоговое предсказание для изображения производится на основе нескольких случайных фрагментов. Было проведено исследование, с целью выяснить какое количество изображений оптимально для качественного определения автора данного письма.

В табл. 1 представлена зависимость качества картиночной модели от числа фрагментов в задачах классификации на 3, 10, 20, 100 классов. Можно заметить, что увеличение количества фрагментов при агрегации практически всегда приводит к увеличению и точности, и *F1*. Поэтому всюду далее используется максимальное число фрагментов для классификации одного документа – 20 шт.

Табл. 1. Accuracy и F1 для картиночной модели

Metric \ Windows num	1	5	10	20
3 classes, Hits@1	0.9209	0.9379	0.9322	0.9492
3 classes, Macro-F1@1	0.9203	0.9385	0.9338	0.9510
10 classes, Hits@1	0.7073	0.7878	0.7854	0.8171
10 classes, Macro-F1@1	0.6856	0.7766	0.7768	0.8091
20 classes, Hits@1	0.6416	0.7621	0.7731	0.7731
20 classes, Macro-F1@1	0.6198	0.7363	0.7598	0.7566
100 classes, Hits@1	0.4415	0.5549	0.5650	0.5882
100 classes, Macro-F1@1	0.3208	0.4239	0.4286	0.4624

Был проведен такой же эксперимент со сбалансированным обучением сети для изучения, какой эффект оно окажет на *F1*. Сравнение табл. 1 и 2 показывает, что сбалансированное обучение дает незначительное улучшение в редких случаях (такие случаи выделены жирным шрифтом), в основном результат становится немного хуже. Вероятно, это объясняется тем, что малые классы и так легко детектируются, поэтому увеличение их частоты встречаемости при обучении не дает сильного прироста качества.

Табл. 2. Accuracy и F1 для картиночной модели при сбалансированном обучении

Metric \ Windows num	1	5	10	20
3 classes, Hits@1	0.9209	0.9209	**0.9435**	0.9209
3 classes, Macro-F1@1	0.9217	0.9217	**0.9469**	0.9220
10 classes, Hits@1	0.6854	0.7488	0.7732	0.7634
10 classes, Macro-F1@1	0.6856	0.7408	0.7656	0.7567
20 classes, Hits@1	0.5696	0.6682	0.6854	0.6964
20 classes, Macro-F1@1	0.5496	0.6331	0.6584	0.6705
100 classes, Hits@1	0.3642	0.4783	0.4993	0.5094
100 classes, Macro-F1@1	0.3146	**0.4251**	**0.4457**	0.4609

Строчная модель

Были проведены эксперименты для строчной модели. В табл. 3 представлено сравнение моделей. Для картиночной модели приведены лучшие результаты, с 20 окнами, при несбалансированном обучении. Для строчной модели уже виден прирост в качестве при переходе к сбалансированному обучению (выделено жирным шрифтом), особенно заметен прирост в *F1* в многоклассовой классификации. В большинстве случаев строчная модель справляется лучше, но по качеству классификации на 100 классов все же уступает.

Табл. 3. Сравнение показателей для картиночной и строчной моделей

Metric \ Model	10 classes			20 classes			100 classes		
	win	str	bal	win	str	bal	win	str	bal
Hits@1	0.817	0.657	**0.694**	0.773	**0.552**	0.515	0.588	0.526	**0.582**
Hits@2	0.915	0.813	**0.851**	0.889	**0.761**	0.727	0.692	0.657	**0.712**
Hits@3	0.946	0.888	**0.908**	0.930	**0.832**	0.815	0.758	0.741	**0.771**
Hits@4	0.968	0.935	**0.943**	0.955	**0.880**	0.859	0.808	0.796	**0.812**
Hits@5	0.980	**0.965**	0.960	0.969	**0.919**	0.905	0.852	0.835	**0.847**
Macro-F1@1	0.809	0.585	**0.676**	0.757	0.484	**0.497**	0.462	0.395	**0.579**
Macro-F1@2	0.913	0.786	**0.843**	0.884	**0.724**	0.722	0.594	0.564	**0.708**
Macro-F1@3	0.948	0.884	**0.908**	0.927	0.801	**0.812**	0.681	0.674	**0.765**
Macro-F1@4	0.969	0.938	**0.946**	0.954	0.861	**0.861**	0.745	0.743	**0.804**
Macro-F1@5	0.981	**0.968**	0.962	0.968	0.904	**0.904**	0.796	0.793	**0.840**

Для картиночной модели была использована предобученная сеть и с бóльшим числом параметров (205507 против 513000). Были проведены эксперименты еще с двумя архитектурами строчной модели: предобученным ResNet18 и измененной архитектурой (10), с 4 слоями в энкодере вместо 3, с двунаправленным LSTM-блоком (770936 обучаемых параметров).

Табл. 4. Сравнение метрик для разных архитектур строчных моделей

Metric \ Model	10 classes			20 classes			100 classes		
	bal	res	str2	bal	res	str2	bal	res	str2
Hits@1	0.694	0.968	0.973	0.515	0.862	0.910	0.582	0.715	0.782
Hits@2	0.851	0.990	0.990	0.727	0.943	0.957	0.712	0.818	0.876
Hits@3	0.908	0.990	0.990	0.815	0.965	0.971	0.771	0.855	0.921

Hits@4	0.943	0.998	0.998	0.859	0.975	0.981	0.812	0.879	0.943
Hits@5	0.960	0.998	0.998	0.905	0.983	0.987	0.847	0.901	0.965
Macro-F1@1	0.676	0.967	0.972	0.497	0.860	0.894	0.579	0.739	0.727
Macro-F1@2	0.843	0.991	0.990	0.722	0.945	0.950	0.708	0.836	0.849
Macro-F1@3	0.908	0.991	0.990	0.812	0.968	0.968	0.765	0.871	0.902
Macro-F1@4	0.946	0.997	0.997	0.861	0.976	0.980	0.804	0.893	0.935
Macro-F1@5	0.962	0.997	0.997	0.904	0.983	0.988	0.840	0.911	0.961

В табл. 4 представлены лучшие результаты строчной модели при сбалансированном обучении и лучшие результаты двух новых строчных моделей. Для ResNet качество лучше при сбалансированном обучении, для большой LSTM модели – при несбалансированном, но в обоих случаях качество заметно выше, чем у картиночной.

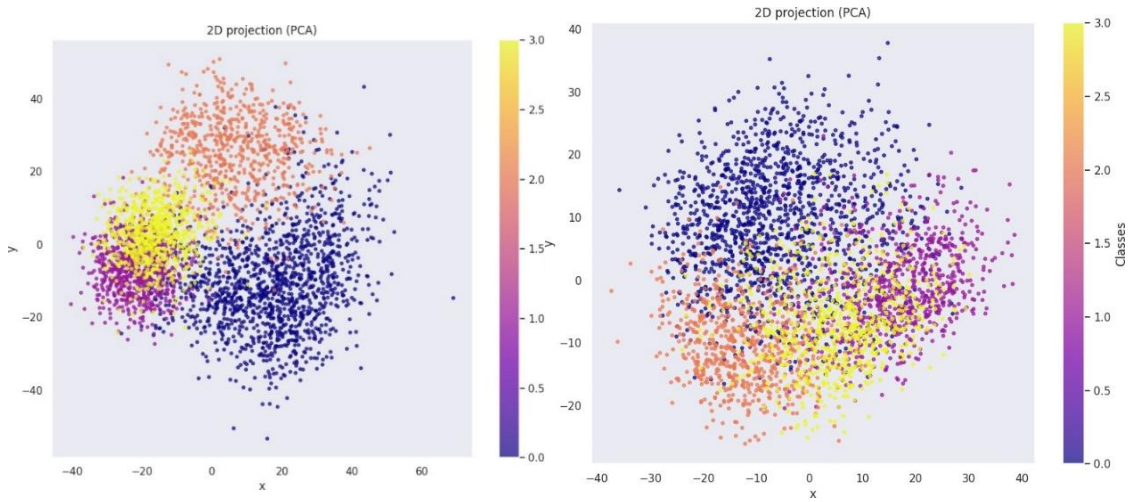


Рис. 12. Две главные компоненты для несбалансированного и сбалансированного обучения сиамской сети.

Кроме того, по визуализации проекции эмбеддингов большой LSTM-модели на двумерную плоскость (с помощью метода главных компонент) видно, что сбалансированное обучение ухудшает разделимость эмбеддингов строк для четырех самых больших классов (рис. 12).

Определение неизвестного класса

После обучения модели на задачу классификации с функцией потерь перекрестной энтропии, модель не в состоянии «отказаться от классификации», если встречает объект класса, который не был представлен в обучающих дан-

ных, но в текущей постановке задачи обязательно нужно иметь такую возможность. Такая классификация называется классификацией в открытом мире (open-world classification), или открытой классификацией (open classification).

Идея перехода к открытой классификации заключается в том, чтобы заменить стандартный softmax-слой, т. к. функция softmax по умолчанию распределяет 100% вероятностей между известными классами, не оставляя места для «неизвестных». Поэтому softmax был заменен на функцию сигмоиды, по одной на каждый класс, функция потерь – на бинарную перекрестную энтропию [12]. Таким образом, задача мультиклассовой классификации, в которой softmax дает взаимоисключающие вероятности, перешла в задачу множественной бинарной классификации, где сигмоиды позволяют объекту принадлежать к нескольким классам или ни к одному.

Итоговое предсказание модели построено следующим образом: если сигмоиды всех классов меньше заранее заданного порога (использовался порог, равный 0.5), то объект классифицируется как неизвестный, иначе выдается класс с наибольшим значением сигмоиды.

Эксперименты показали, что подобная замена последнего softmax-слоя и функции потерь практически никак не влияет на меры качества, полученные на обучающей и валидационных выборках.

Анализ предсказаний

Была визуализирована матрица ошибок для строчной модели, показавшей наивысшее качество, – большой LSTM-модели, чтобы зафиксировать самые частые перепутывания авторов. На рис. 13 видно, что ошибки распределены равномерно по строкам и столбцам, значит, модель одинаково хорошо разделяет всех авторов. Самый яркий пример ошибки – модель атрибутирует некоторые документы Антоновой Валентины Константиновны Киселевой Ольге, но на рис. 14 видно, что эти почерки и правда похожи.

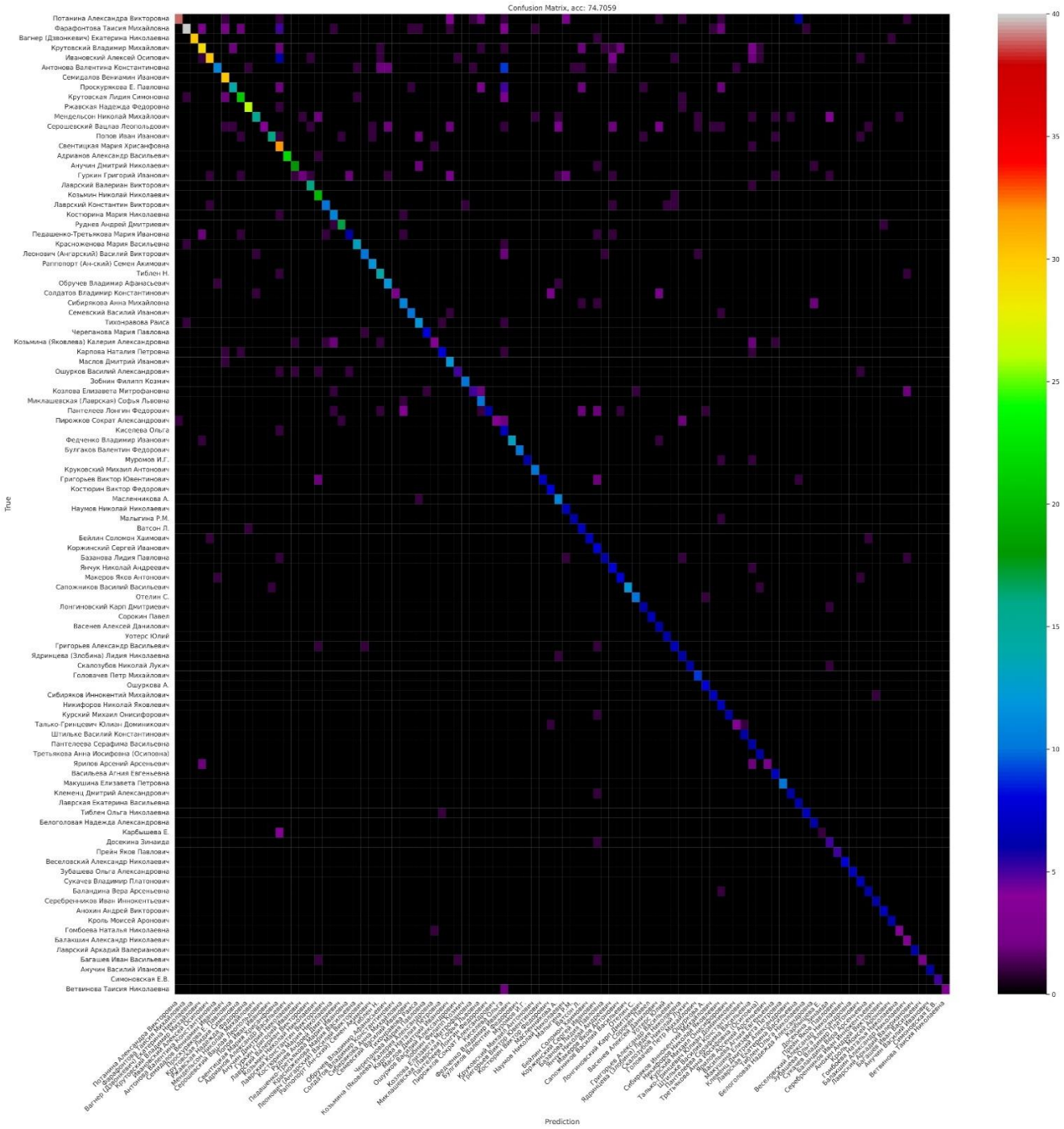


Рис. 13. Матрица ошибок большой LSTM-модели.

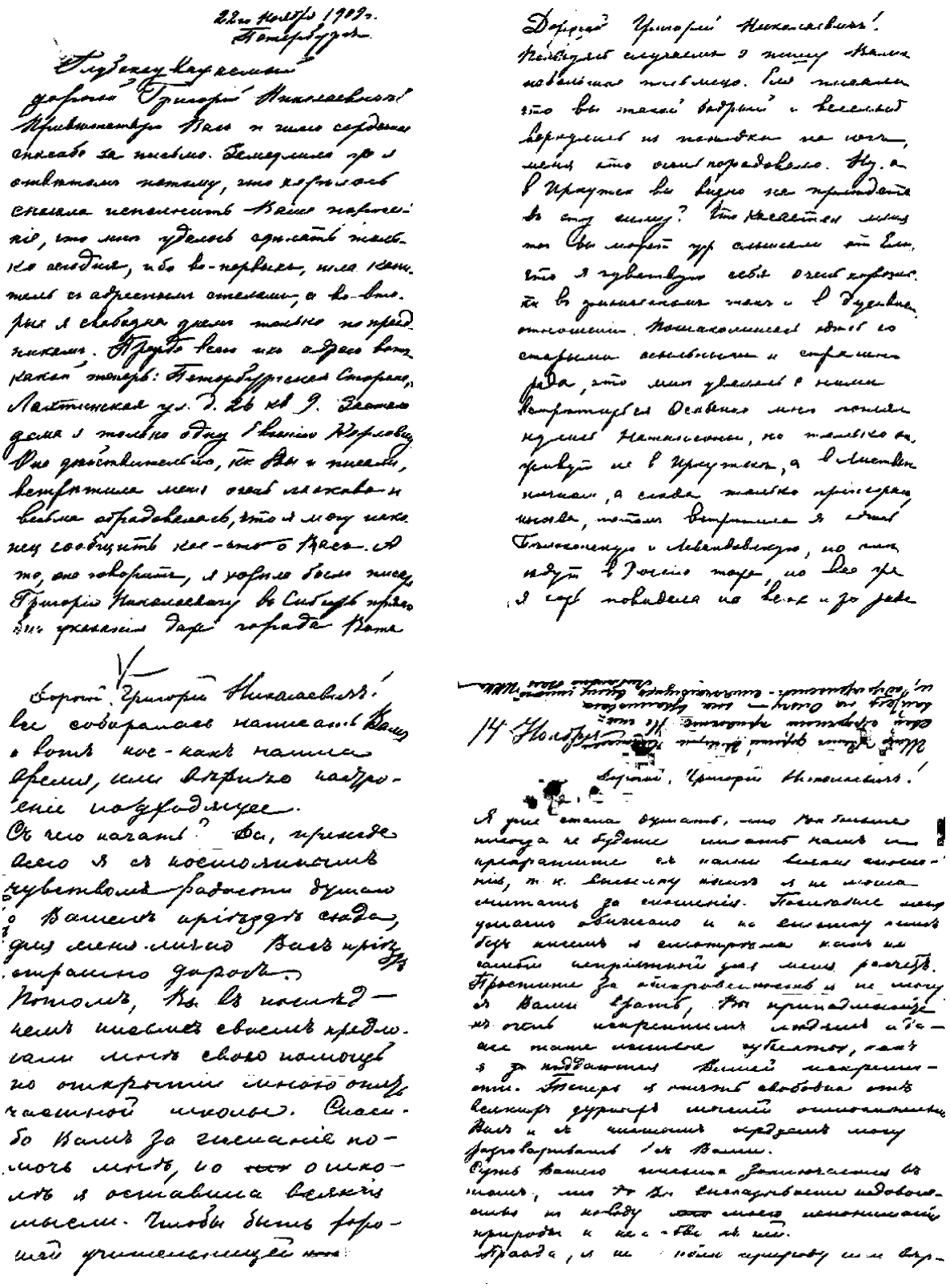


Рис. 14. Два примера для Антоновой Валентины Константиновны и Киселевой Ольги.

Далее были получены вероятности для 40 писем с неизвестным автором. Для большинства изображений максимальная вероятность авторства получилась меньше 30%. Выделяются только два изображения с вероятностью более 85% (рис. 15). Названия типа «КККМ ОФ 9999:9999» обозначают музейный шифр конкретного письма.

КККМ ОФ 7928:2707
0.2767 Проскурякова Е. Павловна
0.2403 Маслов Дмитрий Иванович
0.0958 Потанина Александра Викторовна
0.0691 Веселовский Александр Николаевич
0.0463 Свентицкая Мария Хрисанфовна

КККМ ОФ 7928:2707
0.3495 Проскурякова Е. Павловна
0.1253 Маслов Дмитрий Иванович
0.1250 **Неустановленное лицо**
0.0908 Наумов Николай Николаевич
0.0721 Потанина Александра Викторовна

Рис. 15. Результаты для писем с неизвестным автором.

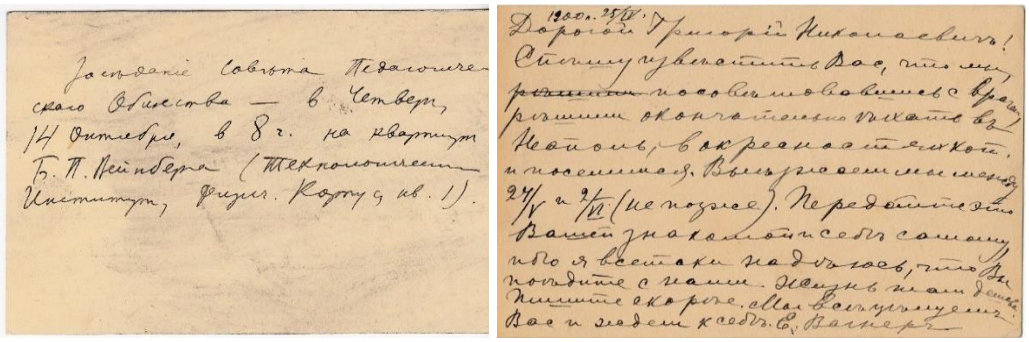


Рис. 16. КККМ ОФ 7928:2480 и почерк Е. Н. Вагнер (Дзвонкевич).

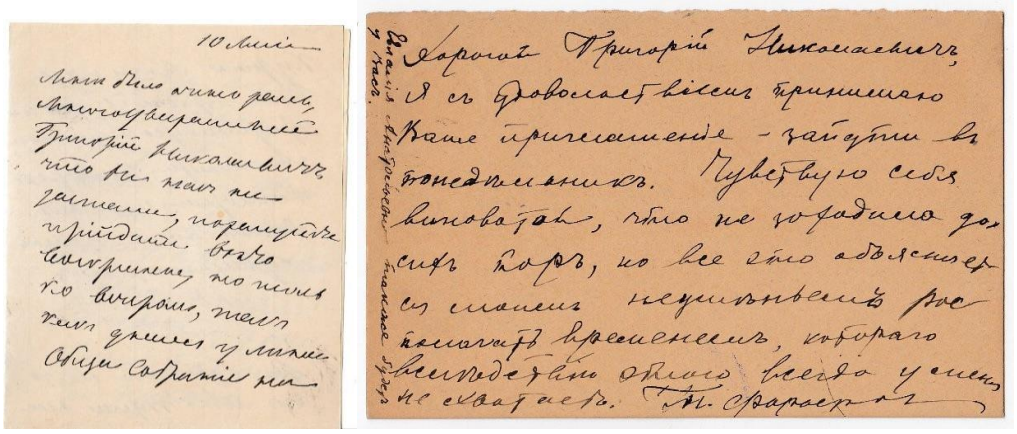


Рис. 17. КККМ ОФ 7928:2719 и почерк Т. М. Фарафонтовой.

На рис. 16 и 17 показано сравнение с почерками соответствующих авторов из размеченных данных. Более детальный анализ эксперта [13] признал оба прогноза реалистичными.

ЗАКЛЮЧЕНИЕ

Предложен подход к задаче атрибуции рукописных писем, основанный на использовании сиамской нейронной сети для сравнения и анализа уникальных характеристик почерка, стиля письма. Предложенное решение позволяет не только сравнить рукописи и определить принадлежность их известным авторам, но и выделить документы, чьи создатели отсутствуют в архиве.

Основным результатом является алгоритм, входом которого является сканированный неатрибутированный документ, а на выходе выдается ранжированный по убыванию вероятности список возможных его авторов, а также вероятность того, что автор документа неизвестен, т. е. не упоминался в архиве. Полученное решение задачи атрибуции включает предобработку сканированных документов и качественное обучение сиамских нейронных сетей. Существенным результатом работы является решение задачи классификации авторов на реальном корпусе текстов с реальными недостатками, с которыми сталкиваются исследователи при работе с архивами: некачественное сканирование, значительная вариативность написания даже для рукописей одного автора, большой дисбаланс
классов.

Исследованы два варианта анализа сканированного текста: анализ фрагментов изображения и анализ каждой строки рукописного текста отдельно. В первом случае, для картиночной модели, в качестве архитектуры сиамской сети использован ResNet18; во втором случае, для строчной модели, использована гиб-ридная модель, сочетающая сверточные и рекуррентный слои для обработки последовательностей строк.

Эксперименты показали, что модели устойчивы к дисбалансу данных, способны работать даже с малым числом образцов почерка, показывая высокое значение *F1*, подтвердили практическую применимость метода для задач архивной атрибуции.

Благодарности

Работа поддержана грантом РНФ № 22-68-00066 «Культурное наследие России: интеллектуальный анализ и тематическое моделирование корпуса рукописных текстов».

СПИСОК ЛИТЕРАТУРЫ

1. *He K., Zhang X., Ren S., Sun J.* Deep residual learning for image recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. P. 770–778. https://doi.org/10.1109/CVPR.2016.90.

2. *Kiselev V., Kropotov D., Pronina N.* Handwritten documents author verification based on the siamese network // The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2024. Vol. XLVIII-2/W5-2024. P. 73–78.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W5-2024-73-2024

3. *Bromley J., Bentz J., Bottou L., Guyon I., Lecun Y., Moore C., Sackinger E., Shah R.* Signature verification using a "siamese" time delay neural network // International Journal of Pattern Recognition and Artificial Intelligence. 1993. Vol. 7, No. 4. P. 669–688. https://doi.org/10.1142/S0218001493000339

4. *Solomon E., Woubie A., Emiru E.S.* Deep learning-based face recognition method using siamese network. 2024. https://doi.org/10.48550/arXiv.2312.14001

5. *Yin W., Schütze H.* Convolutional neural network for paraphrase identification // Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015. P. 901–911. https://doi.org/10.3115/v1/N15-1091

6. *Koch G., Zemel R., Salakhutdinov R. et al.* Siamese neural networks for one-shot image recognition // ICML Deep Learning Workshop. 2015. Vol. 2, No. 1. P. 1–30.

7. *Chopra S., Hadsell R., LeCun Y.* Learning a similarity metric discriminatively, with application to face verification // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). 2005. Vol. 1. P. 539–546.
https://doi.org/10.1109/CVPR.2005.202

8. *Hadsell R., Chopra S., LeCun Y.* Dimensionality reduction by learning an invariant mapping // 2006 IEEE Computer Society Conference on Computer Vision

and Pattern Recognition (CVPR'06). 2006. Vol. 1. P. 1735–1742. https://doi.org/10.1109/CVPR.2006.100

9. *Schroff F., Kalenichenko D., Philbin J.* Facenet: A unified embedding for face recognition and clustering // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015. P. 815–823. https://doi.org/10.1109/CVPR.2015.7298682

10. *Souibgui M.A., Biswas S., Jemni S.K., Kessentini Y., Forn´es A., Llado´s J., Pal U.* Docentr: An end-to-end document image enhancement transformer. 2022. P. 1699–1705. https://doi.org/10.1109/ICPR56361.2022.9956101.

11. *Wood D.E., Salzberg S.L.* Kraken: ultrafast metagenomic sequence classification using exact alignments // Genome Biology. 2014. Vol. 15, No. 1. P. R46. https://doi.org/10.1186/gb-2014-15-3-r46

12. *Shu L., Xu H., Liu B.* Doc: Deep open classification of text documents. 2017. P. 2911–2916. https://doi.org/10.18653/v1/D17-1314.

13. *Киселев В.С., Пронина Н.М.* Машинная атрибуция почерка в решении источниковедческих проблем (на материале переписки Г. Н. Потанина) // Имагология и компаративистика. 2025. № 24.

ARCHIVAL HANDWRITTEN LETTER ATTRIBUTION USING SIAMESE NEURAL NETWORKS

N. M. Pronina[0009-0008-1336-4512]

National Research University Higher School of Economics, Moscow, Russia

natalka-pronina@mail.ru

Abstract

This paper presents a method for the automated attribution of archival handwritten letters based on a Siamese neural network, addressing a key challenge in digital humanities – the authentication of historical documents. The research is motivated by the mass digitization of 17th to 19th-century archives, where attribution is often hindered by incomplete or inaccurate metadata about the authors.

The method is designed for real-world document collections and accounts for challenges typical of archival materials: poor-quality scans, significant handwriting

variation, and substantial class imbalance (from 1 to over 50 samples per author). The use of a Siamese network architecture enables the extraction of discriminative vector representations (embeddings). Based on these embeddings, the method not only classifies documents by known authors but also effectively identifies manuscripts that do not match any known author in the archive. This significantly narrows down the pool of candidates for subsequent expert verification.

The study introduces a data preprocessing algorithm and provides a comparative analysis of two approaches to text analysis: at the image fragment level (300×300 px) and at the individual text line level. The developed tool offers archivists and philologists an effective solution for the preliminary sorting and attribution of handwritten documents large collections.

Keywords: *siamese neural network, identification, verification, attribution, handwritten text, archival documents, convolutional neural network, recurrent neural network.*

REFERENCES

1. *He K., Zhang X., Ren S., Sun J.* Deep residual learning for image recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. P. 770–778. https://doi.org/10.1109/CVPR.2016.90.

2. *Kiselev V., Kropotov D., Pronina N.* Handwritten documents author verification based on the siamese network // The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2024. Vol. XLVIII-2/W5-2024. P. 73–78.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W5-2024-73-2024

3. *Bromley J., Bentz J., Bottou L., Guyon I., Lecun Y., Moore C., Sackinger E., Shah R.* Signature verification using a "siamese" time delay neural network // International Journal of Pattern Recognition and Artificial Intelligence. 1993. Vol. 7, No. 4. P. 669–688. https://doi.org/10.1142/S0218001493000339

4. *Solomon E., Woubie A., Emiru E.S.* Deep learning-based face recognition method using siamese network. 2024. https://doi.org/10.48550/arXiv.2312.14001

5. *Yin W., Schütze H.* Convolutional neural network for paraphrase identification // Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015.

P. 901–911. https://doi.org/10.3115/v1/N15-1091

6. *Koch G., Zemel R., Salakhutdinov R. et al.* Siamese neural networks for one-shot image recognition // ICML Deep Learning Workshop. 2015. Vol. 2, No. 1. P. 1–30.

7. *Chopra S., Hadsell R., LeCun Y.* Learning a similarity metric discriminatively, with application to face verification // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). 2005. Vol. 1. P. 539–546. https://doi.org/10.1109/CVPR.2005.202

8. *Hadsell R., Chopra S., LeCun Y.* Dimensionality reduction by learning an invariant mapping // 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). 2006. Vol. 1. P. 1735–1742. https://doi.org/10.1109/CVPR.2006.100

9. *Schroff F., Kalenichenko D., Philbin J.* Facenet: A unified embedding for face recognition and clustering // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015. P. 815–823. https://doi.org/10.1109/CVPR.2015.7298682

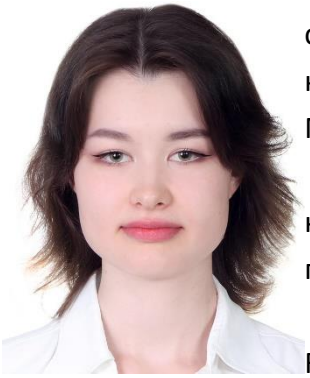
10. *Souibgui M.A., Biswas S., Jemni S.K., Kessentini Y., Forn´es A., Llado´s J., Pal U.* Docentr: An end-to-end document image enhancement transformer. 2022. P. 1699–1705. https://doi.org/10.1109/ICPR56361.2022.9956101.

11. *Wood D.E., Salzberg S.L.* Kraken: ultrafast metagenomic sequence classification using exact alignments // Genome Biology. 2014. Vol. 15, No. 1. P. R46. https://doi.org/10.1186/gb-2014-15-3-r46

12. *Shu L., Xu H., Liu B.* Doc: Deep open classification of text documents. 2017. P. 2911–2916. https://doi.org/10.18653/v1/D17-1314.

13. *Kiselev V., Pronina N.* Machine attribution of handwriting in solving source studies problems (based on the correspondence of G.N. Potanin) // Imagology and Comparative Studies. 2025. No. 24.

СВЕДЕНИЯ ОБ АВТОРЕ



ПРОНИНА Наталия Михайловна – магистр кафедры математических методов прогнозирования факультета вычислительной математики и кибернетики Московского государственного университета имени М.В. Ломоносова.

Область научных интересов: нейронные сети, задачи распознавания образов, расшифровка рукописных текстов. Число научных публикаций – 5.

Nataliia Mikhailovna PRONINA – Master's degree in Mathematical Forecasting Methods Faculty of Computational Mathematics and Cybernetics Lomonosov Moscow State University.

Research interests: neural networks, image recognition problems, handwriting recognition. Number of scientific publications: 5.

email: natalka-pronina@mail.ru

ORCID: 0009-0008-1336-4512

Материал поступил в редакцию 9 ноября 2025 года