БИБЛИОТЕЧНЫЕ КАТАЛОГИ И ИНФОРМАЦИОННО-ПОИСКОВЫЕ СИСТЕМЫ

УДК 025.355:025.4.03 + 025.355:004
https://doi.org/10.33186/1027-3689-2025-2-144-161

Разработка алгоритма автоматизации ретроконверсии для создания электронного каталога

В. А. Коробковский[1], Н. Н. Горлушкина[2], М. А. Белинская[3]

[1, 2]Национальный исследовательский университет информационных технологий, механики и оптики, Санкт-Петербург, Российская Федерация
[3]Библиотека Российской академии наук, Санкт-Петербург, Российская Федерация

[1]vkorobkovskiy@gmail.com
[2]nagor@itmo.ru, https://orcid.org/0000-0002-6549-1723
[3]masha_belinskaya@mail.ru

Аннотация. При создании электронных каталогов, значительно упрощающих читателям доступ к нужной информации, возникают определённые сложности. Проблемы, связанные с созданием современного цифрового фонда, особенно актуальны для библиотек, имеющих длительную историю и большие фонды хранения. В статье рассматривается вопрос расширения возможностей библиографического поиска по фондам российских библиотек на основе пополнения электронных каталогов информацией со сканов каталожных бумажных карточек. Описаны существующие способы перевода бумажных карточек в электронный каталог.

В рамках исследования были проанализированы преимущества и недостатки различных методов создания электронного каталога, а также проведён обзор различных технических средств, которые могли бы подойти для решения задачи автоматизации создания или пополнения электронного каталога. С помощью «дообучения» и применения нейронных сетей был реализован алгоритм на языке программирования Python, позволяющий выполнять задачи предобработки, локализации необходимых областей, распознавания текста и, что самое главное, конвертирование считанного текста на поля и подполя формата RUSMARC. С его помощью решение задач ретроконверсии библиографических данных происходит значительно быстрее по сравнению с ручным вводом.

Ключевые слова: организация электронных библиотек, электронный каталог, автоматизированные информационные системы, алгоритм, ретроконверсия библиографических данных, Python, программирование, нейронные сети, библиографический поиск, библиографическая карточка, RUSMARC

Для цитирования: Коробковский В. А., Горлушкина Н. Н., Белинская М. А. Разработка алгоритма автоматизации ретроконверсии для создания электронного каталога // Научные и технические библиотеки. 2025. № 2. С. 144–161. https://doi.org/10.33186/1027-3689-2025-2-144-161

LIBRARY CATALOGS AND INFORMATION RETRIEVAL SYSTEMS

UDC 025.355:025.4.03 + 025.355:004
https://doi.org/10.33186/1027-3689-2025-2-144-161

Development of an algorithm for automating retroconversion for creating an electronic catalog

Vadim A. Korobkovsky[1], Natalia N. Gorlushkina[2] and Maria A. Belinskaya[3]

[1, 2]National Research University for Information Technologies, Mechanics and Optics, St. Petersburg, Russian Federation
[3]Library of Russian Academy of Sciences, St. Petersburg, Russian Federation

[1]vkorobkovskiy@gmail.com
[2]nagor@itmo.ru, https://orcid.org/0000-0002-6549-1723
[3]masha_belinskaya@mail.ru

Abstract. The authors substantiate the need for electronic catalogs that significantly simplify users' access to relevant information. They formulate the difficulties of this process. The mentioned problems become especially acute for the libraries with a long history and large collections when they start conversion to the digital. The authors discuss the possibilities of bibliographic search expansion through scanning paper catalog cards. The ways to convert paper cards into digital format are described.

As part of the study, the advantages and disadvantages of each method for acquiring e-catalog were analyzed, and different technical tools were reviewed to find the most efficient solution for developing e-catalogs. Based on the analysis, through additional training and with the neural networks, the algorithm in the Python language was implemented, which allows to perform preprocessing tasks, to localize the necessary areas, to recognize text and, most importantly, to convert the scanned text into RUSMARC format fields and subfields. This algorithm accelerates retroconversion of bibliographic data as compared to the manual entry.

Keywords: electronic libraries, electronic catalog, automated information system, algorithm, retroconversion of bibliographic data, Python, programming, neural networks, bibliographic search, bibliographic card, RUSMARC

Cite: Korobkovsky V. A., Gorlushkina N. N., Belinskaya M. A. Development of an algorithm for automating retroconversion for creating an electronic catalog // Scientific and technical libraries. 2025. No. 2, pp. 144–161. https://doi.org/10.33186/1027-3689-2025-2-144-161

Введение. Российские библиотеки активно занимаются переводом бумажных карточных каталогов в электронный вид [**1**, **2**] для простоты получения читателями доступа к нужной им информации. Электронный каталог позволяет пользователям быстро и эффективно находить необходимые материалы [**3**], осуществлять электронный заказ и получать заказанные издания. Поиск в электронном каталоге по различным параметрам, таким как автор, заглавие, ключевые слова, год издания, существенно сокращает время, необходимое для нахождения нужного произведения, и значительно влияет на качество найденной информации.

Например, фонд Библиотеки Российской академии наук (БАН) составляет более 20 млн единиц хранения [**4**], а в электронном каталоге содержится около 2 млн записей, что составляет лишь 10% от общего фонда. Полноценная книговыдача существенно затруднена. Кроме того, нельзя исключать чрезвычайные ситуации (пожар, наводнение и др.), в результате которых хранимая информация может быть частично повреждена или полностью утеряна.

У электронных копий есть существенные преимущества. Они позволяют уменьшить износ оригинала и открывают возможности для

межбиблиотечного обмена [**5**]. К тому же в случае необходимости оригинал можно будет реставрировать по имеющейся копии.

Одной из наиболее популярных в России систем автоматизации, предназначенной для создания и ведения электронной библиотеки, является ИРБИС64+ [**6**]. Она позволяет поддерживать любое количество баз данных, составляющих электронный каталог, а также обеспечивает работу с видео- и фотоматериалами. Библиографические данные добавляются в автоматизированную библиотечную информационную систему (АБИС) на основе коммуникативного формата RUSMARC [**7**] – адаптации формата UNIMARC [**8**].

Ретроконверсия библиографических данных является проблемой: например, отечественная часть генерального алфавитного каталога БАН содержит более 6 млн бумажных каталожных карточек. На текущий момент есть два способа ввода информации с карточек библиографического описания в систему: ручной и автоматический. Каждый из них имеет свои преимущества и недостатки. Ручной ввод является самым точным, однако занимает огромное количество времени и сил. Для такого вида работ необходимо дополнительно привлекать сотрудников, обладающих знанием как системы ИРБИС64+, так и библиотечных ГОСТов. Кроме того, потребуется контролировать проделанную работу. Автоматический способ намного быстрее, однако он не гарантирует стопроцентной точности из-за большого количества нюансов. К тому же у библиотек нет ни специалистов для написания таких программ, ни средств для покупки готовых решений [**9**].

Цель исследования. Разработка алгоритма автоматизации переноса информации с отсканированных каталожных карточек в информационную систему библиотеки, поддерживающую формат RUSMARC, для расширения возможностей библиографического поиска.

Методы и материалы исследования. Проблема ретроконверсии информации с бумажных носителей не нова. Впервые о ней заговорили в начале XXI в. [**10**], и с тех пор она остаётся предметом интереса и исследований [**1**, **5**, **9**, **11**]. Причиной этого может быть как технологическая отсталость некоторых библиотек, так и высокая стоимость имеющихся решений [**12**], которые абсолютно точно не являются полностью автоматизированными. На рынке России в области ретроконверсии лидирует корпорация ЭЛАР [**13**]. И хотя её технологии являются

коммерческой тайной, в описаниях работ нет указаний на автоматизацию процесса ретроконверсии.

В статье [**14**] описываются три способа решения проблемы: клавиатурный набор текста, заимствование базы данных из других библиотек и сканирование. Стоит отметить, что в двух последних способах может использоваться ручной клавиатурный набор в случае наличия недостающих записей в заимствованных базах данных, плохого качества или состояния карточек и наличия рукописного или плохо различимого текста.

Таким образом, единого решения для автоматизации ретроконверсии не существует. Предлагаемые варианты требуют как финансовых, так и иных ресурсов, причём точность и полнота полученных результатов всё равно не гарантируются.

Авторами был проанализирован процесс ретроконверсии и выделены необходимые этапы его реализации – предобработка текста, распознавание текста и конвертация полученной информации в формат RUSMARC.

Предобработка. Проблемы, возникающие при предобработке изображений, были изучены на основе моделирования и принятия возможных решений. Предобработке уделено особое внимание, так как она существенно влияет на итоговый результат.

Первый этап – удаление тёмных краёв, возникающих в результате сканирования. Для изображений, переведённых в чёрно-белый вариант, применялась функция Гаусса с размером окна 5 x 5 пикселей для сильного размытия и поиска контуров [**15**]. После этого для размытых карточек выполнялась бинаризация. Приоритет был отдан методам, которые ищут порог для разделения автоматически. Конечно, указывать размер окна и константу для вычитания всё равно необходимо вручную, однако в данном случае это не имеет особого значения из-за решаемой задачи. Несмотря на то, что самым популярным вариантом является метод Оцу, он не был выбран из-за плохой работы на изображениях с тенями. По этой причине использовались методы адаптивной бинаризации ADAPTIVE_THRESH_MEAN_C и ADAPTIVE_THRESH_GAUSSIAN_C [**16**] с размером окна 5 x 5 пикселей и константой 2, а также метод THRESH_BINARY с вручную выставленным пороговым значением, равным 64. Они применялись последовательно, каждый последующий метод использовался только в том случае, если с помо-

щью предыдущего контур по заданным условиям не был найден. Чтобы избежать возможной потери необходимой информации, проверяется размер изображения по длине и ширине (уменьшение возможно не более чем на 30%).

Все последующие этапы предобработки проводятся на изображениях, полученных в результате локализации. По итогам выполнения этого процесса получались три обрезанных изображения, содержащих в себе информацию с разных полей формата RUSMARC.

Второй этап – выравнивание текста. Изображение переводится из цветного в градации серого, после чего вновь применяется бинаризация. Использовались те же методы, что и в случае с удалением тёмных краёв, и метод Оцу [**17**], поскольку в данном случае наличие теней не влияет на результат, в отличие от этапа по удалению тёмных краёв. Далее для каждого из углов в диапазоне от –5 до 5 градусов с дельтой 0.1 происходят поворот изображения и вычисление гистограммы суммы значений пикселей по вертикали. Затем путём вычисления суммы квадратов разностей между значениями гистограммы получается оценка для конкретного угла. После рассмотрения всех углов находятся максимальное значение и соответствующий ему угол. Финальным шагом являются нахождение центра изображения и его поворот на найденный угол с применением интерполяции Ланцоша для сохранения качества изображения и параметра BORDER_REPLICATE для заполнения образующихся при повороте тёмных краёв значениями близлежащих пикселей.

Третий этап – применение билатерального фильтра для удаления шумов на изображении. Фильтр является нелинейным и не размывает границы объектов, что позволяет сохранить качество текста на уровне, близком к исходному. Билатеральный фильтр выбран в результате сравнительного анализа с медианным фильтром. Линейные фильтры не рассматривались вовсе, так как они не сохраняют границы текста, что негативно влияет на процесс распознавания.

Четвёртый этап – проведение бинаризации, поскольку чёрный текст будет лучше распознаваться на белом фоне. Был выбран метод адаптивной бинаризации ADAPTIVE_THRESH_GAUSSIAN_C, который автоматически ищет порог для разделения с помощью вычитания из взвешенной по Гауссу суммы значений пикселей в квадратном окне размером N x N пикселей некой заданной вручную константы. Исполь-

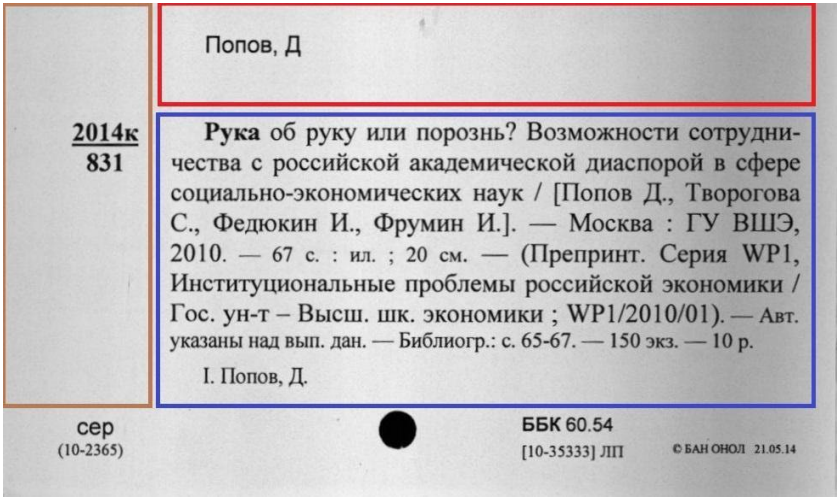
зование алгоритмов пороговой бинаризации было нежелательным из-за необходимости подбора порогового значения для каждого изображения. Данный метод достаточно хорошо справляется с бинаризацией участков изображения, на которых есть тень, сохраняя при этом нужную информацию. Экспериментально были определены параметры окна и констант. Увеличение окна с 5 до 15 поможет улучшить результат, а вот небольшое увеличение вычитаемой константы с 2 до 4 связано с большим количеством рукописной информации. Если поставить большую константу, то эта информация с большой долей вероятности будет удалена при бинаризации.

Распознавание текста. В начале работы над данным этапом был проведён анализ инструментов, используемых для считывания информации с изображения, определены возникающие проблемы, смоделированы возможные решения.

Самым распространённым вариантом инструмента для распознавания текста являются OCR-библиотеки для различных языков программирования [**18**]. Наиболее известными инструментами для Python являются pytesseract и EasyOCR. Первая библиотека представляет собой адаптацию Tesseract OCR под Python от Google, вторая является комплексным многоязычным решением с открытым исходным кодом. Оба инструмента бесплатные, однако результаты их работы существенно различаются и зависят от качества изображения и структуры текста на нём. Для таких решений очень важна предобработка, однако универсального метода, который можно применять к любому изображению, не существует. В конечном счёте предпочтение было отдано библиотеке Pytesseract, так как она работает немного быстрее и лучше распознаёт знаки препинания, которые очень важны для проведения конвертации текста в формат RUSMARC.

Важный момент при распознавании текста – локализация структурных частей карточки. Несмотря на то, что изображения имеют некое сходство в структурном плане, в большинстве своём они достаточно сильно отличаются друг от друга, а из-за их большого количества сложно выделить какие-то закономерности. Общими являются основные области, а именно поля автора, заглавия, шифра и хранения. И хотя расположение этих полей может незначительно отличаться, приведённую на рисунке структуру можно считать постоянной: красным цветом отражена область с полем автора, синим цветом – область с

полем заглавия, а оранжевым – область с полем шифра и условий хранения. Части изображения, не вошедшие ни в одну из выделенных областей, для конвертации не используются и могут быть обрезаны.



Структура карточки

Для анализа имеющихся инструментов были выбраны нейронные сети YOLOv8 и EfficientNet, обнаруживающие объекты на изображениях. YOLOv8 [**19**] имеет простую архитектуру, так как является одноэтапной нейронной сетью, предобучена на объёмном датасете, подходит для поиска самых разных объектов и работает достаточно быстро, что очень важно при наличии большого количества данных. EfficientNet [**20**] чуть более точна, но работает медленнее и хуже показывает себя в распознавании объектов новых самостоятельно созданных классов. Поэтому предпочтение было отдано YOLOv8.

Разметка изображений была проведена с помощью интерактивного инструмента CVAT [**21**]. На 2723 изображениях были отмечены поля автора, заглавия и шифра и хранения как отдельные классы. Инструмент является бесплатным и позволяет сохранить информацию о выделенных областях в различных форматах, в том числе и подходящем для моделей YOLO. После разметки и скачивания файлов можно сразу приступать к настройке модели и её дообучению. Для дообучения была

выбрана модель YOLOv8n, которая специализируется на обнаружении объектов и при этом является самой быстрой.

По результатам 300 эпох обучения существующей модели на новых данных результат оказался достаточно хорошим – точность модели составила 85%. Она отлично справляется с «идеальными» карточками, на которых присутствуют все поля и нет пересекающегося текста. Проблемы возникают лишь с частными случаями, такими как отсутствие области одного или более полей; карточки с двумя и более авторами, информация о которых расположена в разных местах; карточки с большим объёмом текста, пересекающим границы разных областей. Исправить ситуацию можно дополнительным дообучением модели на соответствующих размеченных данных. На текущий момент результаты уверенно можно считать более чем хорошими.

Второй частью при распознавании текста стало получение информации из выделенных областей. Проблема заключалась в наличии как рукописных пометок, так и полностью написанных от руки карточек. Для её решения предполагалось использовать модель оптического распознавания рукописных символов Shiftlab OCR. Тип текста, от которого зависит используемый метод распознавания, определяла дообученная нейронная сеть ResNet-50.

Shiftlab OCR [**22**] является библиотекой для сегментации рукописного текста и распознавания рукописных символов с открытым исходным кодом. Её использование в качестве готового решения обусловлено тем, что создание нейронной сети для распознавания рукописного текста с нуля является достаточно сложной и объёмной задачей, для которой необходим датасет с большим количеством данных. Однако результаты использования данной библиотеки оказались совершенно неудовлетворительными. На большей части карточек инструмент не смог распознать ни единого символа, хотя с этим отчасти справлялась библиотека Pytesseract, не предназначенная для этого. К тому же в вывод зачастую попадала информация, не имеющая ни смысла, ни отношения к реальному тексту, поэтому «пустого» вывода не было ни в одном из рассматриваемых случаев. По этим причинам было принято решение продолжать использовать библиотеку Pytesseract даже для таких случаев. Несмотря на то, что это решение не является оптимальным, данное средство позволяет распознавать рукописные цифры и написанные от руки печатные буквы достаточно точно.

ResNet-50 [**23**] является свёрточной нейронной сетью для классификации объектов на изображении, имеет возможность дообучения с помощью библиотеки Keras (что и было реализовано). Датасет состоял из объектов трёх классов, а именно из рукописных и печатных областей, а также областей с обоими типами текста. В качестве данных использовались изображения, полученные в результате проведения локализации. Было проведено 150 эпох обучения на 3510 изображениях, разделённых на тренировочный и валидационный датасеты в размере 3159 и 351 изображений соответственно. Итоговая точность оказалась равна 86%, что является хорошим результатом. Однако датасет на момент обучения был несбалансированным по областям, так как рукописный текст и оба типа текста зачастую встречаются в области шифра и хранения. Из-за этого таких изображений получалось в 3–4 раза больше, чем для областей автора и заглавия.

Конвертация полученной информации в формат RUSMARC. Для понимания важности решения этой задачи необходимо детально изучить формат RUSMARC. Его структура представляет набор полей и подполей, содержащих информацию о различных аспектах библиографической записи. Каждое поле имеет свой уникальный номер, который определяет его тип и содержание. Подполя используются для более детального описания информации внутри полей. Разбиение текста в соответствии с этой структурой облегчает поиск информации в электронном каталоге.

Конвертация полученной информации в формат RUSMARC является самой сложной из рассмотренных проблем. Она подразумевает работу с распознанным текстом, который изначально уже может содержать ошибки, опечатки, пропуски символов (в особенности знаков препинания), что негативно сказывается на процессе конвертации. Данное решение достаточно сложно описать, поскольку речь идёт о написании алгоритма для разбиения токенизированного по определённым правилам текста на поля формата RUSMARC.

Информация из области автора является самой простой для разбиения. Зачастую фамилия отделена от имени и отчества запятой, а сами авторы могут быть разделены как запятыми, так и союзом «и». На основании этого авторы делятся по ФИО полностью с помощью каждой второй запятой или союза «и». После этого рассматриваются имя и отчество (при наличии). Если они состоят из одного-двух симво-

лов, то информация добавляется как инициалы. В противном случае они добавляются как полные, а инициалы подтягиваются на основании сокращения до первой буквы и добавления точки в конце. Также проверяется наличие скобок, в которых указываются годы жизни. Например, полная библиографическая запись для строки «Пушкин, Александр Сергеевич (1799–1837)» будет выглядеть как «#700: ^АПушкин^ВА.С.^GАлександр Сергеевич^D1799-1837».

Информацию из области основного текста разбить по подполям уже сложнее. Связано это с привязкой к знакам препинания, которые могут быть либо распознаны с ошибками, либо отсутствовать вовсе. Во втором случае автоматизированное разбиение провести невозможно. На текущий момент реализовано разделение текста на токены с помощью символов «. –». Все необходимые области идут в определённом порядке, однако некоторые из них могут и отсутствовать. На текущий момент при разбиении учитываются:

название на языке оригинала;

название на иностранном языке с указанием языка;

перечень авторов, художников, составителей, редакторов и т. д.;

информация о публикации, производстве и распространении;

физические характеристики;

серия;

библиография;

международный стандартный книжный номер ISBN.

Последней является область шифра и хранения. Информация из неё также сложно разбивается на подполя, так как встречается несколько возможных вариаций. Она не является обязательной для добавления, однако для самой библиотеки может быть чрезвычайно важна, поэтому все правила разбиения необходимо узнавать у сотрудников. Поскольку информация считывается построчно, то её удобно делить на токены для дальнейшей работы. Если встречается хоть один токен, содержащий скобочку, это говорит о наличии информации о периодичности выпуска материала. Например, для строки «1 11563 192)» итоговая запись будет иметь вид «#910: ^G1^N192^R11563». В противном случае каждый из токенов рассматривается по отдельности. Если он содержит в себе не только буквы, то относится к шифру. При наличии двух таких токенов, идущих друг за другом, они записываются через слэш, то есть объединяются в один шифр: «2014к» и

«831» становятся «#910: ^A0^DOсн.ф.^R2014к/831». В случае появления токенов, состоящих только из букв, они записываются как информация о хранении: «СБО» и «Ак.с.» записываются как «#910: ^A0^DСБО» и «#910: ^A0^DАк.с.».

В конце каждой отдельной записи также добавляется информация о номере карточки и номере ящика, в котором она хранится. Отделение записей друг от друга производится с помощью пяти символов «*», идущих подряд.

Результаты, выводы и рекомендации. При разработке алгоритма управления библиографическим поиском в организации библиотечных систем был не только создан код для выполнения описанных задач, но и исследованы возникающие в процессе работы проблемы, для каждой из которых предложено возможное решение. После проведения анализа и выбора необходимых инструментов с помощью дообучения и применения нейронных сетей был реализован итоговый алгоритм на языке программирования Python.

Отдельное внимание стоит уделить результатам, касающимся типов карточек. В случае с полностью печатным вариантом проблем почти не возникает (точность распознавания и конвертации составляет по 95%). Основная сложность заключается в необходимости более глубокой конвертации в формат RUSMARC, что, конечно, выполнимо. Однако для полностью печатных карточек без разделителей в виде знаков препинания автоматизированную конвертацию для области заглавия на текущий момент провести невозможно.

Карточки с рукописным текстом распознаются довольно плохо (точность меньше 20%), исключением являются лишь цифры в области шифра и хранения. Точность конвертации составляет примерно 35%, так как неправильное распознавание символов выливается в ошибки. Сегодня они исправляются вручную, в дальнейшем может быть создана модель нейронной сети для улучшения распознавания рукописного текста.

У так называемых «смешанных» карточек всё зависит от распределения типов текста. В основном встречаются карточки с печатной областью заглавия и рукописными областями автора, шифра и хранения (точность распознавания и конвертации составляет 80% и 75% соответственно).

«Пустые» карточки было решено пропускать в связи с отсутствием полезной информации, которую необходимо заносить в систему ИРБИС64+. Стоит отметить, что такие карточки идеально распознаются, однако могут возникать проблемы с конвертацией из-за ошибок в локализации областей, возникающих по причине малого количества таких примеров в тренировочном датасете.

Точность распознавания и конвертации карточек старого образца составляет примерно 60%. Это связано с тем, что карточки данного типа отличаются большим количеством шумов, большой яркостью текста и малым межстрочным интервалом. Всё это осложняет предобработку и само распознавание, поскольку иногда библиотека Pytesseract при распознавании одного слова выдаёт сразу несколько результатов, включая при этом слова, находящиеся под ним, что сильно усложняет процесс конвертации. На данный момент такие ошибки исправляются вручную, в дальнейшем будут рассмотрены новые способы предобработки карточек данного типа, улучшающие читаемость текста и, следовательно, точность распознавания.

Наконец, карточки, на которых текст представлен на нескольких языках, имеют среднюю точность распознавания и конвертации – по 70%. Это может произойти из-за отсутствия модели для конкретного языка, неточности в его определении или плохой работы Pytesseract на данных примерах. Поскольку доля карточек данного типа от общего количества достаточно мала, они легко обрабатываются вручную.

Практическая значимость исследования заключается в том, что предложенный алгоритм позволяет автоматизировать обработку библиографических карточек для размещения в систему. Это поспособствует увеличению скорости процесса в несколько раз: с нормы в 10 карточек в час при ручном вводе до 60 и более карточек в час при автоматическом вводе в зависимости от типа текста и количества информации на изображении без учёта возможного внесения исправлений вручную. При этом остаётся большое пространство для улучшения: ускорение работы алгоритма, добавление нового функционала (например, распознавание рукописного текста) или улучшение имеющегося для более полного и чёткого разбиения.

Электронный каталог, содержащий качественные библиографические записи, даёт возможность организовать более полный и эффективный поиск необходимой информации. На момент написания статьи

автоматизированная программа находится на тестировании в Библиотеке Российской академии наук, по результатам которого будут внесены изменения и улучшения. В дальнейшем возможно её тиражирование, что позволит упростить процесс создания электронных каталогов и ускорить обмен библиографическими сведениями между библиотеками страны.

Список источников

1. **Стукалова А. А.** Основные направления развития электронных каталогов ГПНТБ СО РАН // Труды ГПНТБ СО РАН. 2018. № 13–2. С. 185–192. DOI 10.20913/2618-7515-2018-2-185-192.

2. **Скарук Г. А.** Электронные каталоги библиотек в борьбе за пользователя: «старые» и новые подходы // Библиосфера. 2016. № 2. С. 7–15. DOI 10.20913/1815-3186-2016-2-7-15.

3. **Довбня Е. В.** Проблемы тематического поиска в электронном каталоге научной библиотеки: обзор исследований // Библиотековедение. 2020. № 69 (4). С. 367–374. DOI 10.25281/0869-608X-2020-69-4-367-374.

4. **Белинская М. А., Елкина Н. Н.** Основные задачи Библиотеки Российской академии наук в направлении от «буквы к цифре» // Буква и цифра: библиотеки на пути к цифровизации: сборник докладов Третьей научно-практической конференции «Библио Питер-2022» (г. Санкт-Петербург, 6–8 апреля 2022 г.). С. 12–17. DOI 10.33186/978-5-85638-249-4-12-17.

5. **Степанов В. К.** Манифест библиотек цифровой эпохи. 2014. URL: http://www.calameo.com/read/0034547383b7da70af379 (дата обращения: 28.08.2024).

6. **Бродовский А. И., Сбойчаков К. О., Соколовский В. В.** Перспективы развития системы ИРБИС: новый продукт ИРБИС64+ // Научные и технические библиотеки. 2017. № 11. С. 65–74. DOI 10.33186/1027-3689-2017-11-65-74.

7. **Российский** коммуникативный формат представления библиографических записей в машиночитаемой форме (российская версия UNIMARC). URL: http://www.rusmarc.ru/rusmarc/format.html (дата обращения: 28.08.2024).

8. **Скворцов В. В.** Форматы MARC21, UNIMARC, RUSMARC, их настоящее и будущее. URL: http://www.rusmarc.ru/publish/mar.htm (дата обращения: 30.08.2024).

9. **Вакал Т. С.** Электронные библиотеки: проблемы создания и перспективы развития // Молодой учёный. 2022. № 9 (404). С. 226–228. URL: https://moluch.ru/archive/404/89221/ (дата обращения: 28.07.2024).

10. **Сергеева О. В.** Ретроконверсия каталогов: современный опыт и проблемы применения // Теория и практика общественно-научной информации. 2004. № 19.

URL: https://cyberleninka.ru/article/n/retrokonversiya-katalogov-sovremennyy-opyt-i-problemy-primeneniya (дата обращения: 10.08.2024).

11. **Ретроконверсия** карточных каталогов: основные методы : методические рекомендации / Архангельская областная научная библиотека имени Н. А. Добролюбова; Отдел формирования документ. фонда и организации каталогов; [сост.: М. Ф. Зотова, К. С. Петрова]. Архангельск, 2020. 17 с.
URL: https://biblioteka29.ru/upload/medialibrary/928/retrokonversiya_katalogov.pdf
(дата обращения: 08.06.2024).

12. **Воройский Ф. С.** Организация и технология переработки карточных каталогов в машиночитаемую форму для создания электронных каталогов.
URL: https://www.gpntb.ru/win/ntb/ntb99/1/f01_14.html (дата обращения: 06.08.2024).

13. **ЭЛАР**. Сводный электронный каталог. URL:
https://elar.ru/resheniya/biblioteki/elektronnye_katalogi_i_kollektsii/svodnyy_elektronnyy_katalog/ (дата обращения: 15.08.2024).

14. **Стукалова А. А.** Ретроспективная конверсия карточных каталогов: опыт российских библиотек // Библиосфера. 2012. № 3.
URL: https://cyberleninka.ru/article/n/retrospektivnaya-konversiya-kartochnyh-katalogov-opyt-rossiyskih-bibliotek (дата обращения: 05.07.2024).

15. **Гауссова** фильтрация. URL: https://russianblogs.com/article/7930400611/
(дата обращения: 01.07.2024).

16. **OpenCV** Python Tutorials. Image Thresholding.
URL: https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html (In Eng.).
(дата обращения 10.07.2024).

17. **Обнаружение** объектов методом Оцу. URL: https://habr.com/ru/articles/112079/
(дата обращения: 10.07.2024).

18. **Марцинкевич В. И., Ларионова Г. С., Терещенко В. В., Ситникова К. А., Горлушкина Н. Н.** Анализ возможностей парсинга электронных текстовых документов для автоматизации нормоконтроля // Экономика. Право. Инновации. 2022. № 3. С. 39–49. DOI 10.17586/2713-1874-2022-3-39-49.

19. **Ultralytics** YOLOv8 Docs. URL: https://docs.ultralytics.com/ (In Eng.)
(дата обращения: 18.06.2024).

20. **EfficientNet** PyTorch. URL: https://github.com/lukemelas/EfficientNet-PyTorch (In Eng.)
(дата обращения: 18.06.2024).

21. **CVAT**. URL: https://www.cvat.ai/ (In Eng.) (дата обращения: 18.06.2024).

22. **Shiftlab** OCR. URL: https://github.com/konverner/shiftlab_ocr (In Eng.)
(дата обращения: 25.07.2024).

23. **ResNet** (34, 50, 101): «остаточные» CNN для классификации изображений.
URL: https://neurohive.io/ru/vidy-nejrosetej/resnet-34-50-101/#pll_switcher
(дата обращения: 28.07.2024).

References

1. **Stukalova A. A.** Osnovny`e napravleniia razvitiia e`lektronny`kh katalogov GPNTB SO RAN // Trudy` GPNTB SO RAN. 2018. № 13–2. S. 185–192. DOI 10.20913/2618-7515-2018-2-185-192.

2. **Skaruk G. A.** E`lektronny`e katalogi bibliotek v bor`be za pol`zovatelia: «stary`e» i novy`e podhody` // Bibliosfera. 2016. № 2. C. 7–15**.** DOI 10.20913/1815-3186-2016-2-7-15.

3. **Dovbnia E. V**. Problemy` tematicheskogo poiska v e`lektronnom kataloge nauchnoi` biblioteki: obzor issledovanii` // Bibliotekovedenie. 2020. № 69 (4). C. 367–374. DOI 10.25281/0869-608X-2020-69-4-367-374.

4. **Belinskaia M. A., Elkina N. N.** Osnovny`e zadachi Biblioteki Rossii`skoi` akademii nauk v napravlenii ot «bukvy` k tcifre» // Bukva i tcifra: biblioteki na puti k tcifrovizatcii: sbornik docladov Tret`ei` nauchno-prakticheskoi` konferentcii «Biblio Peter-2022» (g. Sankt-Peterburg, 6–8 aprelia 2022 g.). S. 12–17. DOI 10.33186/978-5-85638-249-4-12-17.

5. **Stepanov V. K.** Manifest bibliotek tcifrovoi` e`pohi. 2014. URL: http://www.calameo.com/read/0034547383b7da70af379 (data obrashcheniia: 28.08.2024).

6. **Brodovskii` A. I., Sboi`chakov K. O., Sokolovskii` V. V.** Perspektivy` razvitiia sistemy` IRBIS: novy`i` produkt IRBIS64+ // Nauchny`e i tekhnicheskie biblioteki. 2017. № 11. C. 65–74**.** DOI 10.33186/1027-3689-2017-11-65-74.

7. **Rossii`skii`** kommunikativny`i` format predstavleniia bibliograficheskikh zapisei` v mashinochitaemoi` forme (rossii`skaia versiia UNIMARC). URL: http://www.rusmarc.ru/rusmarc/format.html (data obrashcheniia: 28.08.2024).

8. **Skvortcov V. V.** Formaty` MARC21, UNIMARC, RUSMARC, ikh nastoiashchee i budushchee. URL: http://www.rusmarc.ru/publish/mar.htm (data obrashcheniia: 30.08.2024).

9. **Vakal T. S.** E`lektronny`e biblioteki: problemy` sozdaniia i perspektivy` razvitiia // Molodoi` uchyony`i`. 2022. № 9 (404). S. 226–228. URL: https://moluch.ru/archive/404/89221/ (data obrashcheniia: 28.07.2024).

10. **Sergeeva O. V.** Retrokonversiia katalogov: sovremenny`i` opy`t i problemy` primeneniia // Teoriia i praktika obshchestvenno-nauchnoi` informatcii. 2004. № 19. URL: https://cyberleninka.ru/article/n/retrokonversiya-katalogov-sovremennyy-opyt-i-problemy-primeneniya (data obrashcheniia: 10.08.2024).

11. **Retrokonversiia** kartochny`kh katalogov: osnovny`e metody` : metodicheskie rekomendatcii / Arhangel`skaia oblastnaia nauchnaia biblioteka imeni N. A. Dobroliubova; Otdel formirovaniia dokument. fonda i organizatcii katalogov; [sost.: M. F. Zotova, K. S. Petrova]. Arhangel`sk, 2020. 17 s. URL: https://biblioteka29.ru/upload/medialibrary/928/retrokonversiya_katalogov.pdf (data obrashcheniia: 08.06.2024).

12. **Voroi`skii` F. S.** Organizatciia i tekhnologiia pererabotki kartochny`kh katalogov v mashinochitaemuiu formu dlia sozdaniia e`lektronny`kh katalogov**.** URL: https://www.gpntb.ru/win/ntb/ntb99/1/f01_14.html (data obrashcheniia: 06.08.2024).

13. **E`LAR.** Svodny`i` e`lektronny`i` katalog. URL: https://elar.ru/resheniya/biblioteki/elektronnye_katalogi_i_kollektsii/svodnyy_elektronnyy_katalog/ (data obrashcheniia: 15.08.2024).

14. **Stukalova A. A.** Retrospektivnaia konversiia kartochny`kh katalogov: opy`t rossii`skikh bibliotek // Bibliosfera. 2012. № 3. URL: https://cyberleninka.ru/article/n/retrospektivnaya-konversiya-kartochnyh-katalogov-opyt-rossiyskih-bibliotek (data obrashcheniia: 05.07.2024).

15. **Gaussova** fil`tratciia. URL: https://russianblogs.com/article/7930400611/ (data obrashcheniia: 01.07.2024).

16. **OpenCV** Python Tutorials. Image Thresholding. URL: https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html (In Eng.). (Accessed: 10.07.2024).

17. **Obnaruzhenie** ob``ektov metodom Otcu. URL: https://habr.com/ru/articles/112079/ (data obrashcheniia: 10.07.2024).

18. **Martcinkevich V. I., Larionova G. S., Tereshchenko V. V., Sitneykova K. A., Gorlushkina N. N.** Analiz vozmozhnostei` parsinga e`lektronny`kh tekstovy`kh dokumentov dlia avtomatizatcii normokontrolia // E`konomika. Pravo. Innovatcii. 2022. № 3. S. 39–49. DOI 10.17586/2713-1874-2022-3-39-49.

19. **Ultralytics** YOLOv8 Docs. URL: https://docs.ultralytics.com/ (In Eng.) (Accessed: 18.06.2024).

20. **EfficientNet** PyTorch. URL: https://github.com/lukemelas/EfficientNet-PyTorch (In Eng.) (Accessed: 18.06.2024).

21. **CVAT**. URL: https://www.cvat.ai/ (In Eng.) (Accessed: 18.06.2024).

22. **Shiftlab** OCR. URL: https://github.com/konverner/shiftlab_ocr (In Eng.) (Accessed: 25.07.2024).

23. **ResNet** (34, 50, 101): «остаточные» CNN для классификации изображений. URL: https://neurohive.io/ru/vidy-nejrosetej/resnet-34-50-101/#pll_switcher (Accessed: 28.07.2024).

Информация об авторах / Authors

Коробковский Вадим Андреевич – студент, магистрант Национального исследовательского университета информационных технологий, механики и оптики, Санкт-Петербург, Российская Федерация

vkorobkovskiy@gmail.com

Vadim A. Korobkovsky – Student of Master Level, National Research University for Information Technologies, Mechanics and Optics, St. Petersburg, Russian Federation

vkorobkovskiy@gmail.com

Горлушкина Наталия Николаевна – канд. техн. наук, доцент Национального исследовательского университета информационных технологий, механики и оптики, Санкт-Петербург, Российская Федерация

nagor@itmo.ru

Белинская Мария Александровна – заведующая научно-исследовательским отделом информатики и автоматизации Библиотеки Российской академии наук, Санкт-Петербург, Российская Федерация

masha_belinskaya@mail.ru

Natalia N. Gorlushkina – Cand. Sc. (Engineering), Associate Professor, National Research University for Information Technologies, Mechanics and Optics, St. Petersburg, Russian Federation

nagor@itmo.ru

Maria A. Belinskaya – Head, Department for Information Technologies and Automation, Library of Russian Academy of Sciences, St. Petersburg, Russian Federation

masha_belinskaya@mail.ru