ISSN 2658-5014 (Print) СИИТ ISSN 2686-7044 (Online)
http://siit.ugatu.su 2026. Т. 8, № 4(28)
СИСТЕМНАЯ ИНЖЕНЕРИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
Научный обзор

УДК 004.056 DOI: 10.54708/SIIT-2026-no4-p12 EDN: PYQOZS

Автоматизация процессов извлечения данных из таблиц электронных документов неструктурированного формата: методы и инструментальные средства

А. О. Шигаров

Институт динамики систем и теории управления имени В. М. Матросова Сибирского отделения Российской академии наук

Представлен обзор исследований по созданию методов автоматизированного понимания таблиц (АПТ) и комплекса инструментальных средств на их основе для упрощения разработки прикладного программного обеспечения извлечения данных из документных таблиц с машиночитаемым текстовым содержимым, представленных в неструктурированном виде, за счет поддержки произвольности табличной структуры. Решены следующие задачи: проанализирована совокупность задач АПТ и усовершенствована ее структура; выявлены ограничения текущего исследования вопросов АПТ и сформулированы актуальные направления его развития; предложен метод автоматизации распознавания таблиц в НПОД, т. е. конвертирования их в редактируемый формат РК; предложен метод автоматизации анализа и интерпретации таблиц РК, т. е. извлечения из них наборов записей в канонической форме; разработаны инструментальные средства на основе предлагаемых методов; выполнена оценка производительности реализованных решений и сравнение их с аналогами; изучены возможности практической применимости предлагаемых методов и инструментальных средств. Методы исследования основаны на применении эвристик, машинного обучения, систем исполнения правил, моделей данных, формальных грамматик, трансляции программ и тестов производительности

Электронные документы; неструктурированные форматы; документные таблицы; автоматизированное понимание таблиц; PDF; PostScript; Excel; Sheets.

Введение

Таблицы являются способом представления реляционных данных. Когда они заключены в электронных документах *неструктурированного формата* (т. е. без схемы данных), например, в растровых изображениях, печатно-ориентированных описаниях страниц (PDF[1], PostScript[2] и др.), веб-страницах (HTML[3]) или рабочих книгах (Excel[4], Sheets[5] и др.), в научной литературе их часто называют *документными таблицами* (Document Table). К настоящему времени в мире накоплен большой массив такой информации. Например, по некоторым оценкам, опубликованным в научной литературе, количество только HTML-таблиц с реляцион-

[1] https://pdfa.org/resource/iso-32000-pdf
[2] https://www.adobe.com/jp/print/postscript/pdfs/PLRM.pdf
[3] https://www.w3.org/html
[4] https://www.microsoft.com/en-us/microsoft-365/excel
[5] https://www.google.ru/intl/ru/sheets/about

Шигаров А. О. Автоматизация процессов извлечения данных из таблиц электронных документов неструктурированного формата: методы и инструментальные средства // СИИТ. 2026. Т. 8, № 4(28). С. 12-36. DOI: 10.54708/SIIT-2026-no4-p12. EDN: PYQOZS.

Shigarov A. O. “Automation of data extraction processes from tables of unstructured electronic documents: methods and tools” // SIIT. 2026. Vol. 8, no. 4(28), pp. 12-36. DOI: 10.54708/SIIT-2026-no4-p12. EDN: PYQOZS. (In Russian).

ными данными в открытой части веб исчисляется по крайней мере сотнями миллионов. Предполагается, что из них можно извлечь сотни миллиардов фактов. Объем всех документных таблиц в мире неизвестен, но, вероятно, значительно превосходит указанные оценки.

Документные таблицы являются ценным источником информации в различных приложениях, в том числе системах поддержки принятия решений, интеллектуальном анализе данных, конструировании баз знаний и информационном поиске. Для того чтобы табличная информация могла быть проиндексирована, запрошена и применена в перечисленных приложениях, прежде всего она должна быть приведена к структурированному представлению (базам данных или графам знаний). Последнее согласуется с целью *извлечения информации*, которая состоит в получении структурированных данных, а именно сущностей и связей между ними, из неструктурированных источников. Однако применить существующий инструментарий *извлечения информации*, ориентированный главным образом на текст, напрямую к таблицам, как правило, невозможно. Поскольку, в отличие от текста, рассматриваемая информация выражена не только с помощью естественного языка, но также и посредством размещения ее частей в структуре ячеек.

В литературе сложившуюся проблематику извлечения структурированных данных из документных таблиц принято называть *автоматизированным пониманием таблиц* (АПТ, Automated Table Understanding). Базовая задача АПТ может быть сформулирована следующим образом. Имеется документная таблица t, доступная как часть неструктурированного источника d, такая что представленные в ней данные составляют один или несколько *наборов записей* $R_1, \dots, R_n$ со схемами соответственно $S_1, \dots, S_n$, описывающими их атрибуты. Будем называть *канонической формой* набора записей R_i такую таблицу, в которой заголовок с именами атрибутов схемы S_i занимает первую строку, каждая запись из набора R_i полностью размещается в одной из последующих строк (в одной строке — одна запись), каждому атрибуту схемы S_i соответствует отдельный столбец (в нем ячейка заголовка содержит его имя, а остальные ячейки — его значения, характеризующие соответствующие записи). Требуется извлечь наборы записей $R_1, \dots, R_n$ в канонической форме, доступной в машиночитаемом формате, из неструктурированного источника d. Расширенная формулировка может также предусматривать дальнейшие действия, направленные на нормализацию извлеченных данных и сопоставление их с внешними словарями и схемами.

Сложность АПТ обусловлена двумя основными факторами: во-первых, наличием большого разнообразия способов компоновки, форматирования и наполнения таблиц; во-вторых, ограниченностью форматов их представления. Являясь частью неструктурированного источника, документная таблица обычно не сопровождается формальной моделью, позволяющей интерпретировать представленную там информацию в соответствии со смыслом, заложенным в нее автором. Обычно неизвестны местоположение документной таблицы внутри источника, структура ячеек, логический порядок чтения данных и пр. В общем случае процесс АПТ включает следующие этапы: *распознавание* — обнаружение местоположения таблицы и ее ячеек в источнике; *анализ* — восстановление логического порядка чтения данных; *интерпретация* — восстановление соответствующего ей набора записей в канонической форме, приведение его к некоторой совокупности отношений в терминах реляционной модели данных и сопоставление их с внешними словарями и схемами. При текущем уровне развития информационных технологий данные процессы в общем случае не могут выполняться без участия человека. Поэтому автоматизация этих процессов нацелена на сокращение операций, производимых человеком.

Исследование вопросов данной проблематики началось еще в середине 1990-х. За три последних десятилетия были защищены десятки диссертаций, опубликованы тысячи научных статей и зарегистрированы по крайней мере сотни патентов.

Несмотря на продолжительную историю, данная проблематика до сих пор остается открытой, нуждаясь в выработке общих теоретических основ и создании технологических решений,

применимых для различных сред и форматов представления табличной информации. Наблюдаемый рост количества публикаций, посвященных ее вопросам, показывает, что интерес к ней со стороны научного сообщества продолжает усиливаться. Последнее десятилетие ознаменовалось всплеском публикаций, предлагающих решения задач АПТ на основе новых техник глубокого обучения, векторного представления слов и сущностей, а также связанных открытых данных. В рамках ведущих конференций (ICDAR[6], ISWC[7], NeurIPS[8], VLDB[9] и др.) стали проводиться тематические секции, семинары и соревнования по отдельным задачам АПТ с участниками со всего мира. Сегодня некоторые элементы АПТ уже представлены в популярных сервисах извлечения данных из документов от крупнейших технологических компаний: «Amazon Textract», «IBM Smart Document Understanding», «Google Document AI», «Microsoft Azure AI Document Intelligence» и др.

Современные обзоры тематической литературы показывают, что все известные решения автоматизации извлечения данных из документных таблиц являются частными. Их общим ограничением является отсутствие поддержки произвольности структуры документных таблиц. Известные методы полагаются на *типичную компоновку* (т. е. небольшое количество — от одного до пяти — компоновочных типов, наиболее распространенных в открытой части веб), атомарность ячеек и плоскую структуру заголовков, игнорируя большое количество случаев, когда эти предположения не выполняются. Таким образом, автоматизация процессов извлечения данных из документных таблиц произвольной структуры является актуальной научной проблемой. В частности, ее решение имеет важное хозяйственное значение для массовой обработки таких источников табличной информации, как *нередактируемые печатно-ориентированные документы* (НПОД), представленные в форматах языков описания страниц (PDL): PDF, PostScript и др., а также *рабочие книги* (РК), представленные в форматах табличных процессоров: Excel, Sheets и др.

Цель исследования состоит в создании методов АПТ и комплекса инструментальных средств на их основе для упрощения разработки прикладного программного обеспечения извлечения данных из документных таблиц с машиночитаемым текстовым содержимым, представленных в неструктурированном виде, за счет поддержки произвольности табличной структуры. Для достижения поставленной цели были решены следующие задачи: проанализирована совокупность задач АПТ и усовершенствована ее структура; выявлены ограничения текущего исследования вопросов АПТ и сформулированы актуальные направления его развития; предложен метод автоматизации распознавания таблиц в НПОД, т. е. конвертирования их в редактируемый формат РК; предложен метод автоматизации анализа и интерпретации таблиц РК, т. е. извлечения из них наборов записей в канонической форме; разработаны инструментальные средства на основе предлагаемых методов; выполнена оценка производительности реализованных решений и сравнение их с аналогами; изучены возможности практической применимости предлагаемых методов и инструментальных средств.

Обзор проблематики и современного состояния исследования АПТ

Структура совокупности задач АПТ. Последовательность из пяти этапов АПТ, включая обнаружение таблицы, распознавание ячеек, ролевой (в англоязычной литературе обычно используется термин «functional analysis») и структурный анализ, а также интерпретацию, впервые была сформулирована в основополагающей работе M. Hurst [Hur00]. На основе этой формулировки в настоящей работе предлагается усовершенствованная структура совокупности задач АПТ (рис. 1). В отличие от постановки M. Hurst, выделены уровни: задач, подзадач

[6] https://icdar.org
[7] https://semanticweb.org
[8] https://nips.cc
[9] https://vldb.org

и методов. На верхнем уровне выделяются три основные задачи: распознавание, анализ, интерпретация (рис. 2).

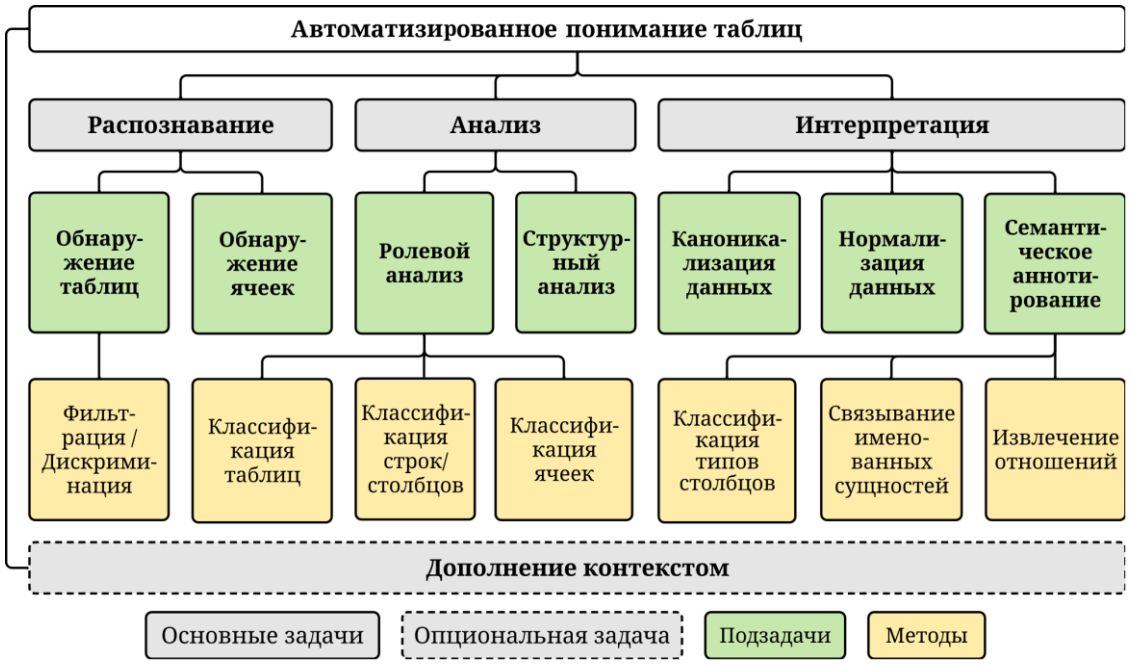


Рис. 1 Структура совокупности задач АПТ

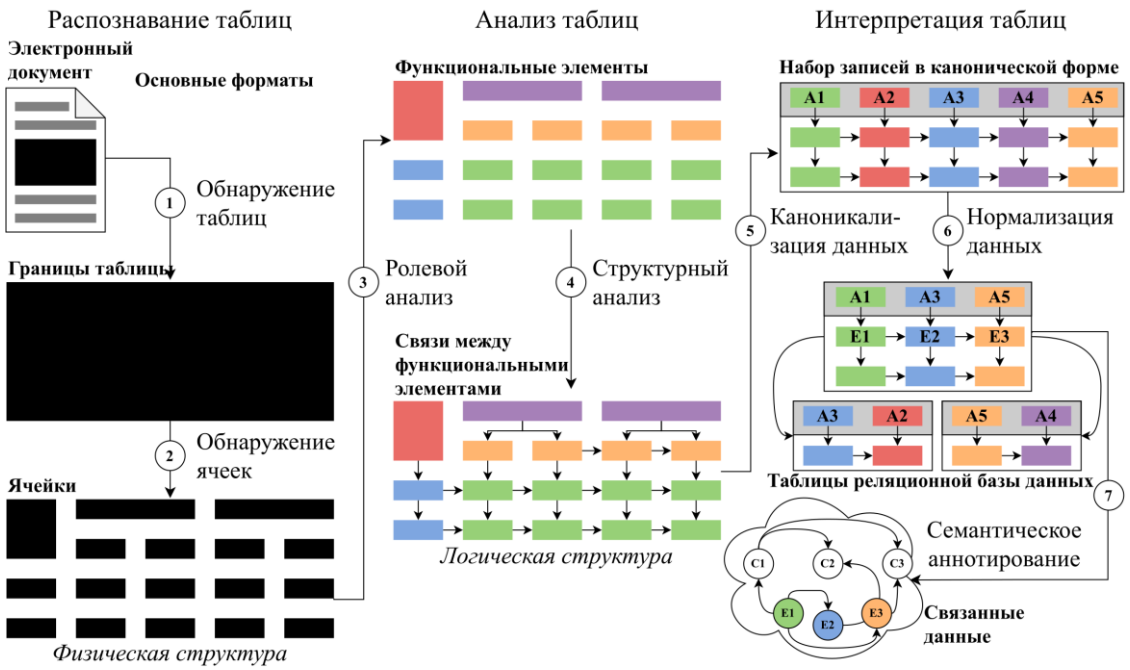


Рис. 2 Последовательность этапов АПТ полного цикла:
от неструктурированного источника данных до структурированного целевого представления

Задача распознавания таблиц (РТ): имеется неструктурированный источник, потенциально содержащий одну или несколько таблиц; требуется извлечь из него *физическую структуру* (т. е. ячейки с определенными координатами в пространстве строк и столбцов, характеристиками форматирования, текстовым и иным содержимым) каждой из них. Включает две взаимосвязанные подзадачи: *обнаружение таблиц* и *обнаружение ячеек*. Первая из них формулируется как выделение той части источника, которая представляет единственную таблицу

и ничего другого. Вторая состоит в определении всех частей источника, которые представляют отдельные ячейки одной таблицы. Конкретные постановки обеих подзадач зависят от исходного формата данных. В общем случае обеспечивается возможность представления результатов в редактируемом формате: HTML, CSV или др.

Методы РТ можно разделить на нисходящие и восходящие. Первые вначале обнаруживают границы таблицы внутри источника, а затем сегментируют выделенную часть на ячейки. Вторые, напротив, сперва обнаруживают отдельные ячейки внутри всего источника, а затем составляют их в таблицу. В недавнем прошлом известные методы традиционно полагались на правила и машинное обучение. Сегодня основное направление их развития связано с применением техник глубокого обучения.

Среди печатно-ориентированных источников наибольшее внимание уделяется скан-копиям и PDF-документам. Следует отметить, что самые последние разработки показывают высокое качество результатов РТ на устоявшихся соревновательных коллекциях данных (ICDAR 2013–2021 [Yep21]). Однако другие тесты производительности, включая SciTSR и IAIS, в целом выявляют недостаточную эффективность доступных решений.

Для веб-страниц предлагаются методы дискриминации HTML-таблиц (т. е. классификации на подлинные и неподлинные случаи). Известные методы преимущественно полагаются на корректную HTML-разметку. Имеются единичные предложения РТ на основе их визуального представления. Методы, имеющие дело с РК, начали развиваться только недавно, ограничиваясь пока в основном вопросами обнаружения таблиц.

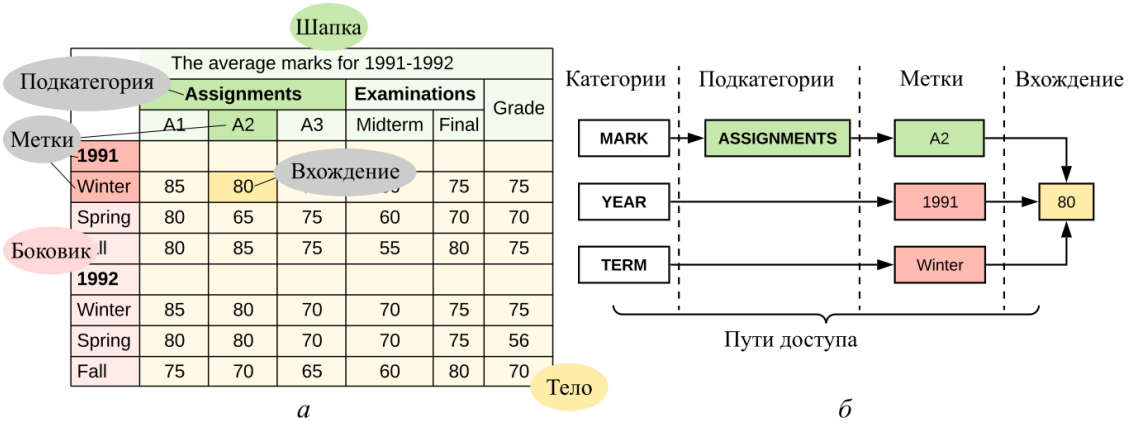
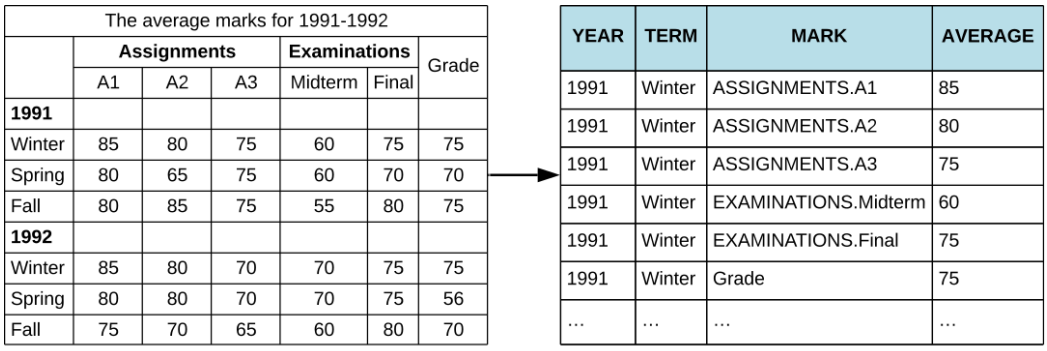


Рис. 3 Восстановление логического порядка чтения: исходная таблица (*а*); порядок чтения, соответствующий вхождению «80», составлен из так называемых путей доступа (*б*)

Задача анализа таблиц (АТ): имеется физическая структура таблицы; требуется восстановить соответствующую *логическую структуру*, т. е. функциональные компоненты и связи между ними. Известные постановки этой задачи можно разделить по тому, каким образом они определяют функциональные компоненты: первые ориентируются на целые регионы ячеек, вторые — на отдельные ячейки, третьи — на функциональные элементы внутри содержимого ячеек. Все они сходятся в том, что целевое представление должно обеспечивать возможность декодирования данных в логическом порядке, ориентированном на чтение человеком (рис. 3).

Анализ таблиц принято делить на *ролевой* и *структурный*. Первый состоит в восстановлении функциональных компонентов или элементов таблицы, а второй — их связей. Например, ролевая часть АТ может сопоставлять ячейкам функциональные роли («данные», «заголовки» и пр.), а структурная — сопоставлять логически связанные друг с другом ячейки («данные» с «заголовками» и пр.). Известные методы в основном адресуются ролевому АТ. Одни предлагают классифицировать таблицы, строки, столбцы и ячейки на основе техник машинного обучения, в том числе глубокого. Другие эксплуатируют эвристики и кластерный анализ.

Вопросы структурного АТ изучены в меньшей степени. Имеются единичные работы, посвященные вопросам распознавания связей между функциональными элементами внутри структурированного содержимого ячейки или иерархического заголовка. Внимание именно к ролевой части АТ объясняется тем, что текущие исследовательские работы предпочитают иметь дело исключительно с типичной компоновкой веб-таблиц. В таких случаях структурный АТ может сводится к ряду тривиальных процедур. Обычно предлагается выводить связи между ячейками по правилам, ориентированным на известные таксономии типов таблиц.



The average marks for 1991-1992

	Assignments			Examinations		Grade
	A1	A2	A3	Midterm	Final	
1991						
Winter	85	80	75	60	75	75
Spring	80	65	75	60	70	70
Fall	80	85	75	55	80	75
1992						
Winter	85	80	70	70	75	75
Spring	80	80	70	70	75	56
Fall	75	70	65	60	80	70

YEAR	TERM	MARK	AVERAGE
1991	Winter	ASSIGNMENTS.A1	85
1991	Winter	ASSIGNMENTS.A2	80
1991	Winter	ASSIGNMENTS.A3	75
1991	Winter	EXAMINATIONS.Midterm	60
1991	Winter	EXAMINATIONS.Final	75
1991	Winter	Grade	75
...	...	...	...

а *б*

Рис. 4 Каноникализация данных: исходная таблица (а) и соответствующий ей набор однотипных записей в канонической форме (б)

Задача интерпретации таблиц (ИТ): доступна логическая структура таблицы; требуется вывести заключенные в ней данные в некоторой структурированной форме (рис. 4). В зависимости от конечной цели может включать в себя *каноникализацию*, *нормализацию* и *семантическое аннотирование*. Первая часть выполняет вывод логической структуры таблицы в набор записей канонической формы; вторая приводит его к некоторой нормальной форме; третья сопоставляет его схему и записи с внешним графом знаний. Сегодня в основном рассматриваются вопросы интерпретации *набора однотипных сущностей* (т. е. набора записей, в котором каждая запись представляет единственную сущность, а все перечисленные там сущности принадлежат одному классу), соответствующего отношению в третьей нормальной форме. Предлагаются различные методы семантического аннотирования заголовка, записей и отношений между полями такого набора однотипных сущностей на основе поисковых SPARQL-запросов к внешним графам знаний общего назначения (DBpedia, Wikidata и др.), а также моделей векторного представления сущностей (RDF2Vec, KGloVe и др.). Однако они непригодны для работы со случаями, которые не являются наборами однотипных сущностей в третьей нормальной форме. Например, класс таких случаев составляют наборы записей, извлекаемые из так называемых *многомерных таблиц*, предназначенных для выявления взаимосвязей между несколькими категориальными переменными.

Актуальные направления развития. Результатом литературного обзора стало выявление ряда ограничений современного исследования АПТ и перспективных путей их преодоления. В частности, к наименее проработанным вопросам следует отнести изучение возможности применения PDL-специфичной информации в процессе РТ в НПОД. Сегодня основное направление исследовательских работ связано с растровыми изображениями. В действительности НПОД можно растеризовать, сведя таким образом задачу к более общей. Однако данный процесс неизбежно сопровождается потерей информации. По сравнению с растром, PDL предоставляет дополнительную информацию (порядок отрисовки текста в графическом контексте, следы перемещения пера и пр.), которая может улучшить качество результатов РТ в НПОД. В частности, она может использоваться на этапе предобработки для сегментации страницы НПОД (обнаружения заголовков, колонок и абзацев текста), а также на этапе посто-

бработки для фильтрации кандидатных случаев. Таким образом, *первым актуальным направлением* является изучение возможности применения PDL-специфичной информации для РТ в НПОД.

Известные решения АТ документных таблиц веб-страниц и РК ограничиваются типичной компоновкой. При этом многие приемы оформления остаются неохваченными. Важным свойством, которое игнорируется абсолютным большинством современных исследовательских работ, является структурированность самой ячейки, когда она содержит несколько значений одновременно. Практически все известные решения ориентируются на атомарность ячеек, полагая, что любая из них всегда соответствует единственному значению. Другим общим допущением является игнорирование иерархических заголовков, но именно такой прием часто применяется при генерации многомерных таблиц. Очевидно, что такие решения не являются полными относительно всего разнообразия всевозможных способов оформления документных таблиц. Таким образом, *вторым актуальным направлением* является поддержка произвольности структуры, включая свободную компоновку, структурированность ячеек и иерархичность заголовков.

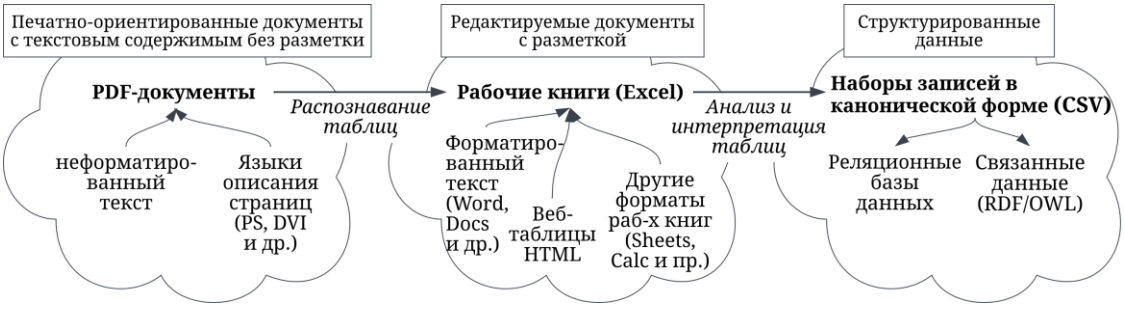


Рис. 5 Возможные цепочки преобразования данных в процессе АПТ

Таким образом, в результате обобщения известных постановок задач АПТ и согласования их терминологии была предложена усовершенствованная структура совокупности задач АПТ, предназначенная для проектирования программных систем извлечения данных из документных таблиц. Кроме того, обзор тематической литературы выявил пробелы в изучении некоторых вопросов; были предложены актуальные направления развития, нацеленные на преодоление ограничений современного исследования, которым до сих пор не уделялось должного внимания. Именно этим направлениям адресованы методы и программные средства, разработанные в настоящей работе. В качестве источников они поддерживают НПОД (PDF) и РК (Excel/Sheets). При этом иные форматы источников документных таблиц с машиночитаемым текстовым содержимым могут быть преобразованы к поддерживаемым форматам с помощью существующих инструментов (рис. 5).

Предлагаемый метод автоматизации распознавания таблиц

Постановка задачи:

- Имеется страница НПОД p с машиночитаемым набором символьных позиций S, часть из которых может визуально представлять одну или несколько таблиц: $t_1, \dots, t_n$.
- Необходимо распознать их физические структуры: $\phi(t_1), \dots, \phi(t_n)$.

Выполняются следующие предположения. Исходные данные представлены *символьными позициями* (т. е. символами, отрисованными на странице в заданных координатах с помощью установленных шрифтов), а также *следами перемещения пера* (т. е. линиями, выводимыми на странице в заданных координатах с помощью установленных параметров контура и заливки). Страница имеет так называемую манхэттенскую компоновку. Наличие разграфки не обязательно. Распознанная физическая структура таблицы (строки, столбцы и ячейки) должна быть доступна для кодирования в редактируемом формате.

Предлагаемый метод:

- Найти непересекающиеся наборы символьных позиций $S_1 \subset S, \dots, S_n \subset S$, принадлежащие размещенным на странице p таблицам $t_1, \dots, t_n$ соответственно.
- Найти для каждой таблицы t_i, состоящей из ячеек $c_1, \dots, c_m$, непересекающиеся наборы символьных позиций $S_{i_1} \subset S_i, \dots, S_{i_m} \subset S_i$, принадлежащие ее ячейкам $c_1, \dots, c_m$ соответственно.
- Пусть каждая символьная позиция $s\colon s \in S_i$ сопоставлена с некоторой ячейкой $c\colon c \in C_i$, тогда восстановлены физические структуры таблиц: $\phi(t_1), \dots, \phi(t_n)$.

Организация взаимодействия компонентов предлагаемого метода схематично показана на рис. 6. Сперва выполняется обнаружение ограничивающей рамки таблицы внутри источника (обеспечивается выбор набора символьных позиций S_i), а затем сегментация выделенной части на отдельные ячейки (обеспечивается отображение: $S_i \to C_i$). Первая часть включает предварительную сегментацию страницы документа на основе синтеза блоков текста и последующее предсказание ограничивающих рамок таблиц посредством либо попарного сопоставления кандидатных табличных строк, либо применения искусственных нейронных сетей (ИНС) обнаружения объектов на изображениях. Вторая часть выполняется за счет либо сегментации пробельного пространства, либо анализа компонент связности блоков текста.

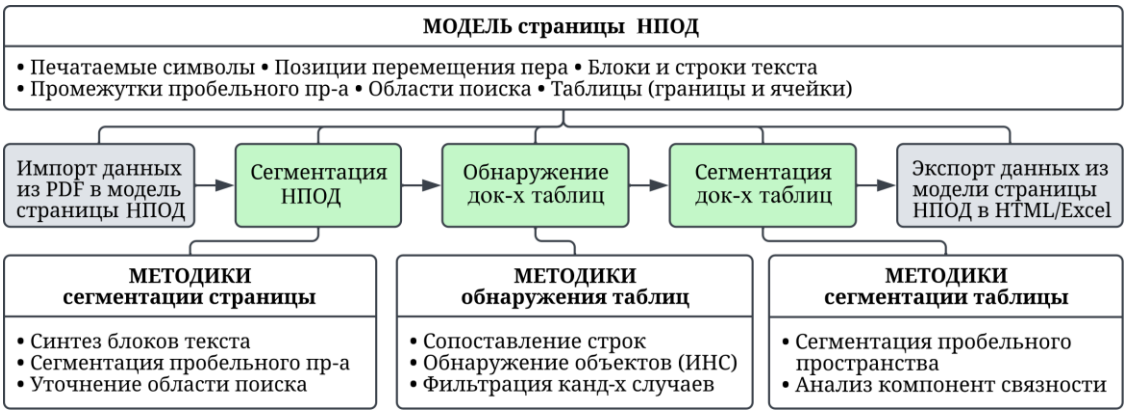
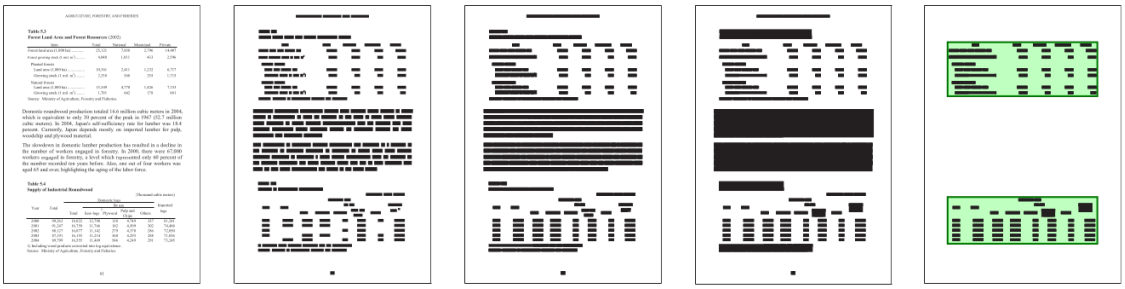


Рис. 6 Метод автоматизации РТ: основные шаги и компоненты

Сегментация страницы документа. Доступные изначально напечатанные символы объединяются в однокомпонентные блоки текста — «слова» (рис. 7, *а, б*), те в свою очередь группируются в многокомпонентные блоки — «строки» и «абзацы» (рис. 7, *в, г*). Предлагается адаптация известного метода «кластеризации слов» в неформатированном тексте T-Recs [Kie98] к специфике НПОД (PDF). Он включает два этапа: первый — начальное формирование многокомпонентных и многострочных блоков; второй — коррекция типовых ошибочных случаев, которые могут возникнуть в результате выполнения первого этапа. На первом этапе адаптированная версия использует PDL-специфичную информацию в качестве ограничений, накладываемых на восстанавливаемый блок (рис. 8, *а, б*). В частности, предполагается, что внутри любого блока сохраняется порядок отрисовки символов текста в графическом контексте (т. е. PDL-инструкции их отрисовки идут одна за другой) и отсутствуют пересечения со следами перемещения пера (т. е. PDL-инструкции не перемещают перо поверх блока текста). Второй этап дополнен новыми алгоритмами коррекции ошибочных результатов применения предлагаемой адаптации (рис. 8, *в, д*).

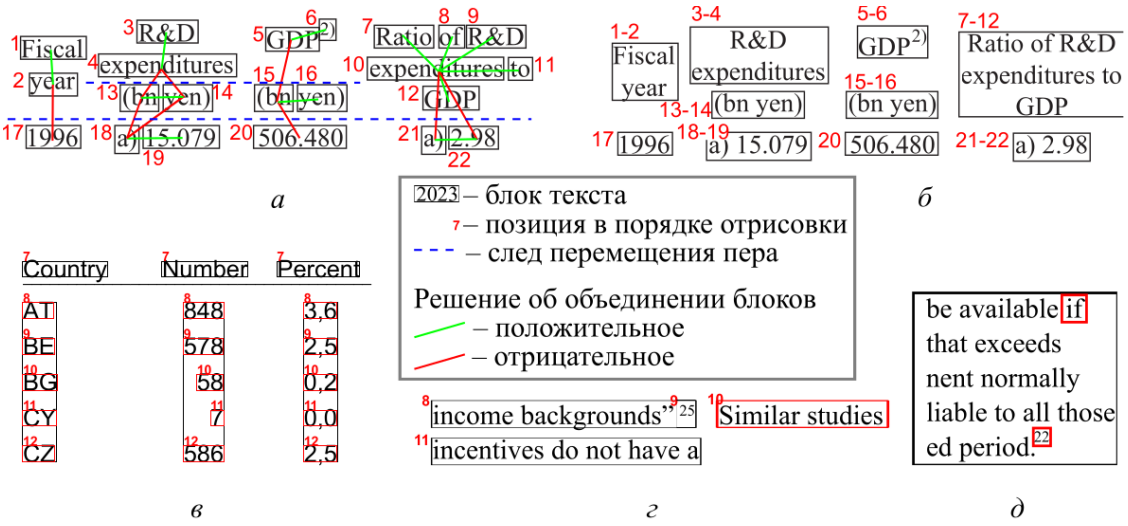
С помощью восстановленных блоков текста выполняется анализ компоновки страницы. Сегментируется пробельное пространство; среди полученных сегментов выбираются промежутки между колонками текста. Некоторые блоки распознаются как разделители, а именно заголовки — по ключевым словам («таблица», «рисунок» и др.) и «крупные» многострочные абзацы текста, занимающие всю ширину страницы. В результате область поиска таблиц может

быть сужена до некоторой части внутри страницы. Одна страница может быть разделена на несколько изолированных областей поиска.



а *б* *в* *г* *д*

Рис. 7 Синтез блоков текста: символы (*а*); слова (*б*); строки (*в*); абзацы (*г*); обнаружение ограничивающих рамок таблиц (*д*)



а *б* *в* *г* *д*

Рис. 8 Начальное формирование блоков: исходные (*а*); целевые (*б*). Типы ошибочных случаев: дупликация порядка отрисовки текста в графическом контексте (*в*), изоляция (*г*) и наложение (*д*) блоков текста

Обнаружение таблиц на основе правил. В заданной области поиска блоки текста группируются в кандидатные табличные строки по пересечению проекций на ось *Y* (рис. 9). Среди них выбираются только многокомпонентные (т. е. включающие по два и более блока).

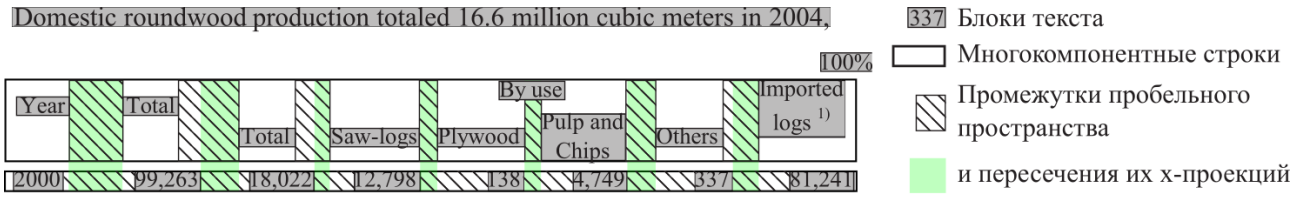


Рис. 9 Обнаружение таблиц на основе сопоставления кандидатных табличных строк

Предполагается, что любые соседние строки одной таблицы имеют схожее размещение промежутков пробельного пространства. На первом этапе группируются соседние многокомпонентные строки; на втором аналогичным образом — группы строк. Каждая из полученных групп принимается за одну целую таблицу.

Обнаружение таблиц на основе машинного обучения. Другая методика состоит в использовании ИНС *обнаружения объектов* на изображениях с последующей фильтрацией

кандидатных случаев. Первая часть может базироваться на настройке доступных ИНС известной архитектуры «Faster R-CNN», а именно обучению двух полносвязных слоев классификации объектов и регрессии координат регионов на небольших наборах предметных данных. Во второй части предлагается прибегнуть к бинарной классификации графовых представлений кандидатных случаев на ложноположительные и истинно положительные. (Вершины такого графа соответствуют блокам текста, ребра — пересечениям их проекций на ось X.) Показано, что бинарная классификация может быть реализована с помощью ансамбля деревьев решений. Как и в случае с настройкой ИНС классификатор кандидатных случаев может быть обучен на небольших выборках.

Сегментация таблицы. Предлагается две методики на основе правил: *сегментация пробельного пространства* и *анализ компонент связности*. В первом случае сперва восстанавливаются линейки разграфки таблицы по промежуткам пробельного пространства, затем формируются границы ячеек по пересечениям полученных линеек (рис. 10). Во втором случае выбираются блоки текста, односвязные по проекции на ось X, каждая компонента связности соответствует одному столбцу (рис. 11, *а*). Аналогично выбираются блоки, односвязные по проекции на ось Y, каждая компонента связности соответствует одной строке (рис. 11, *б*). Когда несколько ячеек пересекают один блок текста, то они объединяются (рис. 11, *в*).

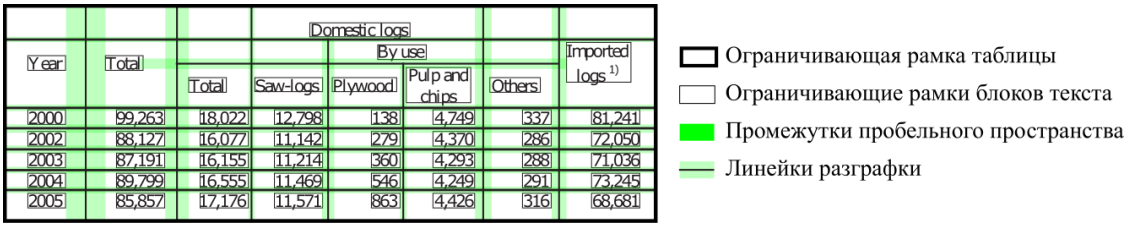


Рис. 10 Обнаружение ячеек на основе сегментации пробельного пространства

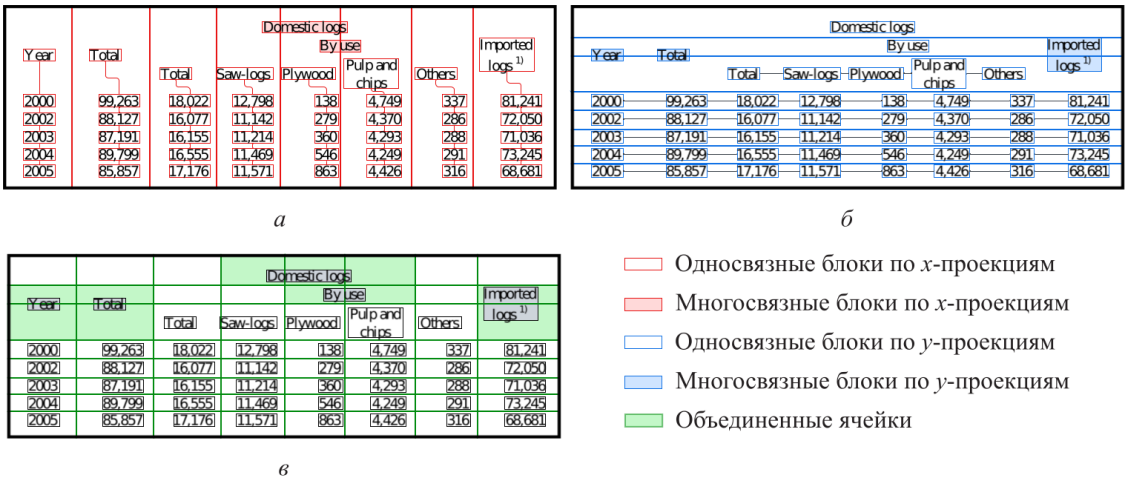


Рис. 11 Обнаружение ячеек на основе анализа компонент связности: разделение на столбцы по x-проекциям блоков текста (*а*); разделение на строки по y-проекциям блоков текста (*б*); объединение ячеек, содержащих части одного блока текста (*в*)

Полученные в результате применения как первого, так и второго способа сегментации таблицы, границы ячеек, позволяют сопоставить каждой ячейке координаты в пространстве строк и столбцов таблицы, а также блоки текста, составляющие ее текстовое содержимое. Сформированная таким образом физическая таблицы представляется в редактируемом формате.

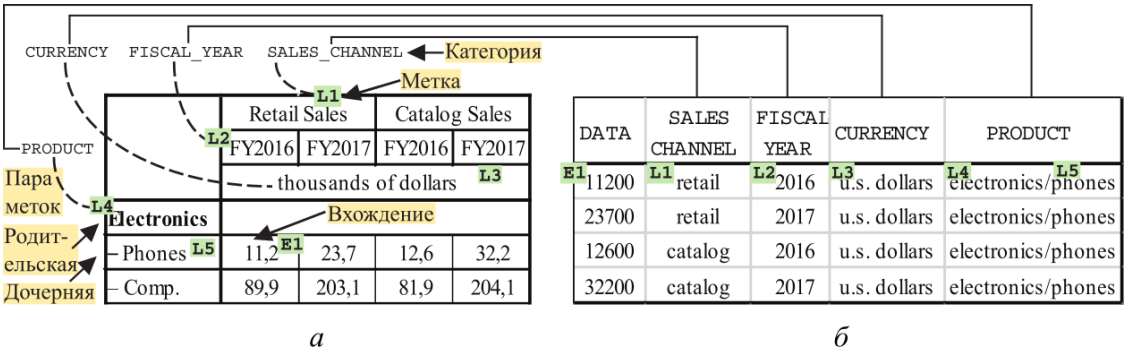
Таким образом, предложенный метод позволяет реализовывать решения прикладных задач РТ в НПОД за счет организации взаимодействия программ сегментации страниц, обнаружения таблиц и ячеек на основе использования PDL-специфичной информации: порядка отрисовки текста в графическом контексте, следов перемещения пера и пр. Насколько известно автору,

никто до сих пор не предлагал использовать именно такие особенности PDL-представления НПОД в означенном ключе. Данное предложение позволило адаптировать к поставленной задаче некоторые известные подходы и методы, изначально ориентированные на растровые изображения и неформатированный текст, включая «кластеризацию слов» T-Recs, обнаружение строк «rows first», сегментацию пробельного пространства и анализ компонент связности. В совокупности они обеспечивают РТ в НПОД, приводимых к формату PDF. Результаты РТ доступны в редактируемом формате РК.

МЕТОД АВТОМАТИЗАЦИИ АНАЛИЗА И ИНТЕРПРЕТАЦИИ ТАБЛИЦ НА ОСНОВЕ ПОЛЬЗОВАТЕЛЬСКОГО ПРОГРАММИРОВАНИЯ ПРАВИЛ

Постановка задачи:

- Имеется таблица t с доступной физической структурой $\phi(t)$, которая визуально представляет набор записей R с общей схемой S.
- Необходимо извлечь набор записей R в канонической форме из t.



а *б*

Рис. 12 Исходная таблица (*а*) и фрагмент ее канонической формы (*б*)

Исходные данные представлены набором ячеек, составляющих полностью одну таблицу (рис. 12, *а*). Любая из них размещается в одной или нескольких соседних строках/столбцах. Она может иметь форматирование (шрифт, выравнивание и т. д.). Ее содержимое может представлять один или несколько функциональных элементов двух типов: вхождения и метки. Под *вхождениями* понимаются значения данных, а под *метками* — значения *категорий*, описывающих вхождения. Любое вхождение должно быть ассоциировано с одной единственной меткой в каждой категории. Любая метка должна принадлежать некоторой категории, но только одной. Пара меток с одинаковой категорией может составлять родительско-дочернюю связь. Целевой набор записей в канонической форме предполагает, что каждой категории соответствует отдельное поле, а каждому вхождению — отдельная запись (рис. 12, *б*).

Предлагаемый метод:

- Пусть T — класс таблиц с общими свойствами компоновки, форматирования и наполнения такими, что каждому из них можно сопоставить правило κ, отображающее элементы физической структуры в элементы логической структуры.
- Создать набор правил $K = \{\kappa_1, \dots, \kappa_n\}$, который позволяет отобразить физическую структуру $\phi(t)$ любой таблицы $t: t \in T$ в ее логическую структуру $\lambda(t)$.
- Применить набор правил K к таблице $t: t \in T$, в результате чего должна быть восстановлена логическая структура $\lambda(t)$, обеспечивающая автоматический вывод набора записей R в канонической форме.

Организация взаимодействия компонентов предлагаемого метода схематично показана на рис. 13. Предполагается, что структуру любой таблицы можно представить в виде фактов. Сопоставление их с предоставленными правилами должно отражать физическую часть структуры в логическую. Каждый набор правил составляет решение некоторого частного случая приведения таких таблиц к канонической форме. Подобные правила можно выражать

на одном из имеющихся формальных языков (DRL[10], CLIPS[11] или др.), а их исполнение производить с помощью совместимой машины вывода (Rete, Leaps или др.). Тем не менее, чтобы упростить разработку правил конечными пользователями, предлагается собственный проблемно-ориентированный язык. Правила, записанные на нем, можно транслировать на языки правил общего назначения или процедурного программирования. Для представления структуры таблицы в виде базы фактов предлагается собственная модель. Она определяет допустимые условия и действия, которые могут использоваться в правилах.

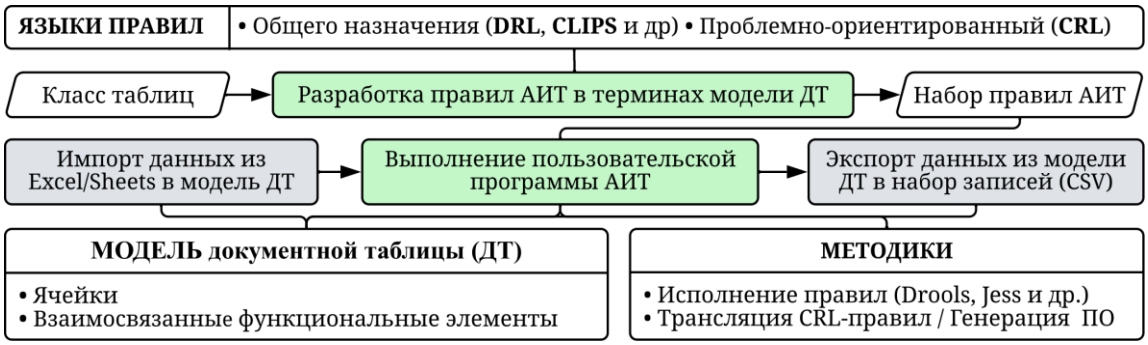


Рис. 13 Метод автоматизации АИТ: основные шаги и компоненты

Рис. 14 Двухуровневая модель документной таблицы

Модель документной таблицы состоит из двух уровней (рис. 14): физического и логического. Физический уровень представляет ячейки, каждой из которых сопоставлены координаты в пространстве строк и столбцов, характеристики форматирования, текстовое содержимое, а также пользовательский дескриптор (тег), обычно означающий некоторое функциональное назначение (например, заголовок или данные). Логический уровень представляет вхождения, метки и категории, связанные между собой парами трех типов: вхождение-метка, метка-метка и метка-категория. Каждая метка принадлежит одной-единственной категории при наличии связи типа метка-категория. Метки одной категории составляют иерархию

[10] https://www.drools.org
[11] https://www.clipsrules.net

посредством связи типа метка-метка. Каждое вхождение связано с одной-единственной меткой в каждой категории через связь типа вхождение-метка. Порождаемые функциональные элементы сопоставлены ячейкам. Предлагаемая модель основана на концепциях модели X. Wang [Wan96], предназначенной для генерации документных таблиц.

Проблемно-ориентированный язык правил АИТ, именуемый CRL (Cell Rule Language), реализует продукционную модель. Левая часть CRL-правила (`when`[12]) состоит из одного или нескольких взаимосвязанных условий, каждое из которых принадлежит одному из двух видов:

- существует по крайней мере один факт:

```
cell | entry | label | category присваивание: ограничения;
```

- не существует ни одного факта:

```
no cells | no entries | no labels | no categories: ограничения
```

Условие первого вида проверяет наличие фактов указанного типа: `cell` — ячеек, `entry` — вхождений, `label` — меток и `category` — категорий. Блок присваивания определяет ссылку на запрашиваемые факты, которая может использоваться в других условиях и действиях данного правила. Блок ограничений, которым должны удовлетворять запрашиваемые факты, является конъюнкцией логических выражений языка программирования Java, перечисленных через запятую. Условие второго вида, напротив, проверяет отсутствие соответствующих фактов.

Правая часть CRL-правила (`then`) перечисляет действия, направленные на изменение имеющихся и создание новых фактов. Язык CRL допускает следующие действия, разделенные на четыре группы по виду их назначения: очистка ячеек, ролевой анализ, структурный анализ и интерпретация.

Очистка ячеек: `merge` — объединение двух соседних ячеек; `split` — разделение ячейки по строкам и столбцам; `set text` — редактирование текста внутри ячейки; `set indent` — изменение индентации текста внутри ячейки. Иногда структура и текст исходных ячеек содержат ошибки, в частности, возникающие в результате ручного набора или редактирования. В некоторых случаях они могут быть исправлены напрямую в процессе исполнения CRL-правил с помощью указанных действий.

Ролевой анализ: `set tag` — присваивание ячейке пользовательского дескриптора; `new entry` / `new label` — создание вхождения/метки в ячейке. Действие `set tag` обеспечивает возможность пользовательской разметки ячеек. В частности, общий дескриптор можно сопоставить ячейкам, выполняющим одинаковую функцию, для того чтобы применить к ним отдельное подмножество правил в последующем выводе. Действия `new entry` и `new label` пополняют базу фактов соответственно вхождениями и метками, порождаемыми из результатов обработки текста ячеек.

Структурный анализ: `add label` — ассоциирование вхождения с меткой; `set parent` — ассоциирование дочерней метки с родительской. Действие `add label` может ассоциировать вхождение с меткой до ее категоризации. Метка выбирается либо напрямую из базы фактов, либо с помощью поиска в некоторой категории. Последняя форма позволяет создавать метки из контекста, описанного непосредственно в правилах. Действие `set parent` связывает пару меток, родительскую и дочернюю, формируя некоторый путь в дереве меток иерархической категории. В выходной канонической форме такой путь записывается в виде составного значения.

Интерпретация: `set category` — включение метки в именованную категорию; `group` — группирование двух меток в анонимную категорию. Действие `set category` предлагает два способа категоризации меток. В первом категория выбирается из базы фактов. Во втором выполняется ее поиск по имени и создание в случае отсутствия. Действие `group` группирует две метки. Если одна из них уже принадлежит некоторой группе, то вторая также добавляется

---

[12] Здесь и далее моноширинным шрифтом выделены ключевые слова языка CRL.

в эту группу. Если они принадлежат разным группам, то обе группы объединяются в одну. Каждая группа, метки которой не сопоставлены какой-либо именованной категории, составляет отдельную анонимную категорию.

Таким образом, предложенный метод позволяет реализовывать решения прикладных задач извлечения данных из документных таблиц РК за счет организации взаимодействия программ трансляции и исполнения пользовательских правил АИТ. Созданный язык программирования CRL обеспечивает упрощенный синтаксис правил АИТ, ориентированный на конечных пользователей. По сравнению с языками правил общего назначения, он скрывает несущественные детали и позволяет сфокусироваться исключительно на реализации логики процесса АИТ. Один набор CRL-правил может обеспечить обработку целой коллекции таблиц с общими свойствами компоновки, форматирования и наполнения. Сложность его реализации зависит от разнообразия этих свойств.

Комплекс инструментальных средств (КИС)

Часть КИС для РТ (TabbyPDF) в НПОД PDF составлена кодовой базой Java-программ[13,14], реализующих методики обнаружения и сегментации таблиц, а также предобработки (сегментации страниц) и постобработки (фильтрации кандидатных случаев). На ее основе разработано модельное веб-ориентированное приложение[15,16] интерактивного конвертирования таблиц нередактируемого формата PDF в редактируемые форматы (HTML/Excel). Оно имеет распределенную клиент-серверную архитектуру. Клиентская часть обеспечивает загрузку входных, визуализацию и пользовательское редактирование выходных данных. Серверная часть выполняет операции автоматической обработки данных. Взаимодействие двух частей реализовано посредством веб-ориентированного интерфейса прикладного программирования, реализующего архитектурный подход REST API. Сценарий использования данного приложения предполагает следующий порядок действий. Пользователь выбирает документ и запускает его обработку. Приложение обнаруживает ограничивающие рамки таблиц. Они возвращаются клиенту, в графическом интерфейсе которого отрисовываются поверх страниц документа. При необходимости пользователь может изменить границы таблиц, а затем повторно запустить обработку документа. Приложение распознает структуру ячеек внутри предоставленных ограничивающих рамок. Размеченные таблицы возвращаются клиенту. Ознакомившись с ними, пользователь может скачать архив полученных результатов в формате Excel/HTML. Само функциональное ядро извлечения таблиц также может использоваться в пакетном режиме (без участия пользователя) через командную строку.

Кроме того, разработан набор Python-скриптов[17] для автоматизации рабочего процесса трансферного обучения ИНС обнаружения таблиц на платформе TensorFlow, включая следующие шаги: доступные наборы данных в нативных форматах приводятся к единому формату Pascal VOC; изображения страниц документов размываются с помощью преобразования расстояний; набор данных дополняется синтетическими примерами, полученными путем аффинных преобразований исходных изображений; генерация обучающей и тестовой выборки в формате «TF Records»; обучение модели ИНС на платформе TensorFlow.

Часть КИС для АИТ (TabbyXL) представлена платформой[18] разработки прикладного ПО извлечения данных из таблиц РК на основе пользовательского программирования правил; базируется на методе автоматизации АИТ. Платформа предоставляет интерфейс прикладного

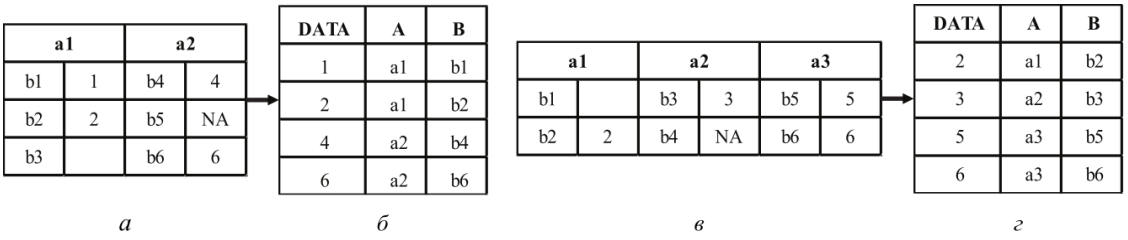
---

[13] https://github.com/tabbydoc/tabbypdf
[14] https://github.com/tabbydoc/tabbypdf2
[15] https://github.com/tabbydoc/tabbypdf-web
[16] https://github.com/tabbydoc/tabbypdf-front
[17] https://github.com/tabbydoc/dl4td
[18] https://github.com/tabbydoc/tabbyxl

программирования, обеспечивающий доступ к экземплярам объектной модели документной таблицы. Ядро АИТ восстанавливает логический уровень такого экземпляра, используя одну из двух опций: либо исполнение правил (CRL, DRL или др.), либо генерацию Java-приложений из CRL-правил и их выполнение в виртуальной машине Java. В результате обращения к любой из них экземпляр дополняется логической структурой, которая может быть выгружена в набор записей канонической формы. Первая опция осуществляется с помощью некоторой системы исполнения правил, совместимой со стандартом JSR-94[19] (по умолчанию Drools). Правила должны быть представлены на соответствующем формальном языке. (Реализована автоматическая трансляция CRL-правила в формат DRL, поддерживаемый Drools.) Вторая опция выполняется следующим образом: синтаксический анализ CRL-правил; создание экземпляра объектной модели каждого CRL-правила; генерация соответствующего исходного кода Java, готового к компиляции; сборка исполняемого Java-приложения. Реализована сериализация полученного исходного кода в виде Maven-проекта.

В составе ПО TabbyXL разработана утилита, предназначенная для автоматического сопоставления физической структуры ячеек заголовка с его визуальной структурой и внесение необходимых корректировок в тех случая, когда они не совпадают. Применение данной утилиты в процессе предобработки исходных данных позволяет улучшить качество результатов АИТ. Также разработано веб-ориентированное приложение интерактивной демонстрации провенанса данных, извлеченных посредством ПО TabbyXL. Оно предоставляет пользовательский интерфейс для отслеживания связей между ячейками исходных таблиц и значениями целевых наборов записей.

Исследование пользовательской разработки CRL-правил показало, что конечные пользователи могут успешно разрабатывать наборы CRL-правил для модельных задач, одна из которых приводится в качестве иллюстративного примера (рис. 15).



1. Очистка ячеек NA

```
when cell $c: text == "NA"
then set text "" to $c
```

2. Создание меток в нечетных столбцах

```
when cell $c: (cl % 2) == 1
then new label $c
```

3. Создание вхождений в четных столбцах

```
when cell $c: (cl % 2) == 0, !blank
then new entry $c
```

4. Связывание вхождений с метками по столбцам

```
when
  entry $e
  label $l: cell.cr == $e.cell.cr
then add label $l to $e
```

5. Связывание вхождений с метками по строкам

```
when
  entry $e
  label $l: cell.rt == $e.cell.rt, cell.cl == $e.cell.cl - 1
then add label $l to $e
```

6. Связывание меток 1-й строки с категорией A

```
when label $l: cell.rt == 1
then set category "A" to $l
```

7. Связывание меток ниже 1-й строки с категорией A

```
when label $l: cell.rt > 1
then set category "B" to $l
```

д

Рис. 15 Иллюстративный пример: исходные таблицы ($а$, $в$) и соответствующие им целевые наборы записей ($б$, $г$), где $1, \ldots, n$ — вхождения, $a_1, \ldots, a_m$ — столбцевые метки категории A; $b_1, \ldots, b_k$ — строчные метки категории B; набор CRL-правил для конвертирования первых во вторые ($д$)

[19] https://www.jcp.org/ja/jsr

Демонстрируется набор CRL-правил, позволяющий автоматически вывести из исходных таблиц с общими свойствами (рис. 15, *а*, *в*) соответствующие наборы записей (рис. 15, *б*, *г*). Следует отметить, что представленная компоновка взята из реальных форм сбора информации об электротехническом оборудовании. Набор CRL-правил (рис. 15, *д*) выполняет следующие действия: очистка текстового содержимого ячеек, генерация вхождений/меток, ассоциирование вхождений с метками строк/столбцов и категоризация меток.

Таким образом, разработанное ПО демонстрирует реализуемость теоретических основ, рассмотренных в предыдущих разделах. Инструментальные средства TabbyPDF/TabbyXL могут использоваться для разработки цифровых продуктов (программ для ЭВМ, баз данных, графов знаний и пр.) как совместно, так и независимо друг от друга. В первом случае доступна интеграция на уровне данных, поскольку используется общий промежуточный формат РК. Во втором случае они могут войти в состав сторонних решений прикладных задач АПТ. Весь исходный код опубликован в открытом доступе под свободными лицензиями.

Оценка производительности и сравнение с аналогами

Показатели качества результатов РТ/АИТ. Оцениваются *точность* (P), *полнота* (R) и F_1-*мера* (F_1), которые определяются следующим образом:

$$P = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fp}},\ R = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fn}},\ F_1 = 2\frac{PR}{P + R},$$

где количество tp — истинно положительных, fp — ложноположительных и fn — ложноотрицательных исходов, значения которых вычисляются путем сопоставления результатов — набора однотипных объектов RS, произведенных тестируемым решением, с эталонными данными теста производительности — набором однотипных объектов GT. Исход сравнения некоторого объекта из RS считается *истинно положительным*, если есть идентичный ему объект в наборе GT и *ложноположительным* — в противном случае. Исход сравнения некоторого объекта из GT считается *ложноотрицательным*, если среди объектов набора RS такового нет.

Оценка производительности решений РТ выполнена с помощью известной методики M. Göbel и др. [Göb12]. Вычисляются средние значения *точности* (P_{doc}) и *полноты* (R_{doc}) среди документов следующим образом:

$$P_{\mathrm{doc}} = \frac{P_1 + \cdots + P_n}{n},\ R_{\mathrm{doc}} = \frac{R_1 + \cdots + R_n}{n},$$

где P_i — точность, а R_i — полнота, измеренная на i-м документе. Оценивается качество обнаружения и сегментации/распознавания таблиц. В первом случае сопоставляются печатаемые символы внутри предсказанных и эталонных ограничивающих рамок таблиц, а во втором — отношения соседства текстового содержимого обнаруженных ячеек.

Результаты оценки методик РТ получены на тесте ICDAR-2013 (табл. 1). Обнаружение таблиц с помощью ИНС дает лучшие результаты по сравнению с методикой на основе сопоставления строк. Однако при высокой полноте (около 98 %) точность остается низкой (менее 87 %) из-за большого количества ложноположительных предсказаний ИНС. Фильтрация кандидатных случаев позволила значительно поднять точность (до 97 %). Методики сегментации пробельного пространства и анализа компонент связности дают близкое качество обнаружения ячеек.

Сравнение решений РТ с аналогами выполнено на трех доступных тестах производительности. На предметно-независимом тесте ICDAR-2013 показано, что TabbyPDF дает результаты, близкие к лучшим академическим аналогам, немного уступая некоторым индустриальным продуктам (табл. 2).

Таблица 1

Оценка производительности решений PT на тесте ICDAR-2013

Показатели	Обнаружение таблиц			Сегментация таблиц[1]	
	CTC	OO	OO+Ф	СПП	АКС
P_{doc}	0,7605	0,8651	0,9703	0,9180	0,9499
R_{doc}	0,8172	0,9795	0,9795	0,9121	0,9233
F_1	0,7878	0,9187	0,9748	0,9150	0,9364

[1] Используются эталонные ограничивающие рамки таблиц.

Оцениваются следующие варианты: CTC — сопоставление табличных строк; OO — обнаружение объектов с помощью ИНС; OO+Ф — OO с фильтрацией кандидатных случаев; СПП — сегментация пробельного пространства; АКС — анализ компонент связности.

Таблица 2

Сравнение с аналогами PT на тесте ICDAR-2013

Решение	Обнаружение таблиц			Решение	Распознавание таблиц[1]		
	P_{doc}	R_{doc}	F_1		P_{doc}	R_{doc}	F_1
FineReader*	0,973	0,997	0,985	FineReader*	0,871	0,883	0,877
Kavasidis et al.	0,981	0,975	0,978	OmniPage*	0,846	0,838	0,842
TabbyPDF[2]	**0,970**	**0,979**	**0,975**	**TabbyPDF[2]**	0,849	0,824	**0,839**
DeepDeSRT	0,974	0,961	0,968	Tabula	0,869	0,808	0,837
TableNet	0,970	0,963	0,966	Tab.IAIS	0,918	0,762	0,832
OmniPage*	0,957	0,964	0,966	**TabbyPDF[3]**	0,834	0,830	**0,832**
Tran et al.	0,964	0,952	0,958	Acrobat*	0,816	0,726	0,768
Silva et al.	0,929	0,983	0,955	Nitro*	0,846	0,679	0,753
Hao et al.	0,922	0,972	0,946	Silva et al.	0,687	0,705	0,696
Nitro*	0,940	0,932	0,936	pdf2table	0,575	0,595	0,585

[1] Используются автоматически обнаруженные ограничивающие рамки таблиц.
[2] На основе ИНС обнаружения объектов, фильтрации кандидатных случаев и анализа компонент связности.
[3] На основе сопоставления строк и сегментации пробельного пространства.
* Индустриальные программные продукты.

Схожие выводы следуют из сравнения с аналогами на двух предметно-ориентированных тестах, а именно SciTSR[20] и IAIS [Ada21] (табл. 3). При этом также используется методика оценки производительности M. Göbel и др., но с расчетом макросредних/микросредних значений точности и полноты:

$$P_{\text{macro}} = \frac{P_1+\cdots+P_n}{n},\ R_{\text{macro}} = \frac{R_1+\cdots+R_n}{n},$$

$$P_{\text{micro}} = \frac{tp_1+\cdots+tp_n}{tp_1+fp_1+\cdots+tp_n+fp_n},\ R_{\text{micro}} = \frac{tp_1+\cdots+tp_n}{tp_1+fn_1+\cdots+tp_n+fn_n},$$

где P_i и R_i, tp_i, fp_i и fn_i — значения, полученные на i-й таблице.

[20] https://github.com/academic-hammer/scitsr

Таблица 3

Сравнение с аналогами PT на тестах SciTSR и IAIS

SciTSR				IAIS			
Решение	Сегментация таблиц[1]			Решение	Распознавание таблиц[2]		
	P_{macro}	R_{macro}	F_1		P_{micro}	R_{micro}	F_1
GraphTSR[3]	0,959	0,948	0,953	FineReader*	0,606	0,609	0,607
TabbyPDF[4]	0,926	0,920	**0,921**	Tab.IAIS	0,6601	0,480	0,556
DeepDeSRT	0,906	0,887	0,890	**TabbyPDF[4]**	0,538	0,561	**0,549**
Acrobat*	0,930	0,784	0,851	Tabula	0,630	0,130	0,215

[1] Используются эталонные ограничивающие рамки таблиц.
[2] Используются автоматически обнаруженные ограничивающие рамки таблиц.
[3] Базируется на извлечении блоков текста из PDF-документа, выполняемого TabbyPDF.
[4] На основе сопоставления строк и сегментации пробельного пространства.
* Индустриальные программные продукты.
Представленные данные опубликованы авторами тестов SciTSR и IAIS.

Качественное сравнение с аналогами PT приводится в табл. 4. Следует отметить, что представленное здесь сравнение охватывает только часть конкурентных методов, которые в целом дают общую картину отличий предлагаемого метода от аналогов по качественным характеристикам.

Таблица 4

Сравнение с аналогами PT по качественным характеристикам

Метод	Подход	Формат	Не требуется		PDL-специфика				
			OCR	Разграфка	Т	Ш	Л	ОТ	ПП
DeepDeSRT	ГО	РИ	-	+	-	-	-	-	-
GraphTSR*	П/ГО	ИО	+	+	+	+	+	+	+
pdf2table	П	ИО	+	+	+	-	-	-	-
Tab.IAIS	П	РИ	-	-	-	-	+	-	-
TabbyPDF	П/ГО	ИО	+	+	+	+	+	+	+
TableNet	П/ГО	РИ	-	+	-	-	-	-	-
Tabula	П	ИО	+	+	+	-	+	-	-
Tran et al.	П	РИ	-	-	-	-	+	-	-

П — правила, ГО — глубокое обучение; РИ — растровые изображения, ИО — PDL-инструкции отрисовки.
Поддерживаемая PDL-специфика: Т — машиночитаемый текст, Ш — шрифтовые свойства, Л — линейки разграфки (в том числе восстанавливаемые из растра), ОТ — порядок отрисовки текста, ПП — следы перемещения пера.

* Базируется на извлечении блоков текста из PDF-документа, выполняемого TabbyPDF.

Оценка производительности решений АИТ выполнена с помощью двух предметно-ориентированных тестов: Troy200[21] и SAUS200[22]. Оба предоставляют реальные таблицы, собранные случайным образом с веб-ресурсов англоязычной государственной статистики. В рамках настоящего исследования были подготовлены эталонные данные в едином формате. В обоих случаях рассчитываются значения точности, полноты и F_1-меры извлечения функциональных элементов и связей между ними суммарно на всех таблицах.

[21] https://doi.org/10.17632/ydcr7mcrtp.6
[22] https://doi.org/10.6084/m9.figshare.14371055.v2

Первый тест включает 200 таблиц, собранных G. Nagy[23] из 10 источников. Каждая из них имеет четыре функциональных части: угловик с именами категорий, шапка и боковик с иерархиями меток, тело с вхождениями. Применяется несколько вариантов компоновки перечисленных частей и форматирования иерархических заголовков с использованием индентации, шрифтового и символьного выделения. Эталонные данные насчитывают около 16900 вхождений, 4900 меток, 35100 пар типа вхождение-метка и 2100 — метка-метка. Тестируемое решение реализовано в виде 16 CRL-правил[24]. Полученные результаты его оценки приведены в табл. 5.

Таблица 5

Оценка производительности решений АИТ на тестах Troy200 и SAUS200

Показатели	Функциональные элементы			Связи		
2-7	вхождения	метки	все	ВМ	ММ	все
Troy200						
P	1,0000	0,9365	0,9849	0,9966	0,9784	0,9956
R	0,9813	0,9979	0,9850	0,9773	0,9389	0,9751
F_1	0,9906	0,9662	0,9849	0,9869	0,9582	0,9852
SAUS200						
P	0,9420	0,9446	0,9423	0,9275	0,8636	0,9248
R	0,9928	0,9360	0,9856	0,9549	0,8391	0,9498
F_1	0,9667	0,9403	0,9635	0,9410	0,8512	0,9371

Извлекаются связи двух типов: ВМ — вхождение-метка и ММ — метка-метка.

Второй тест включает 200 таблиц, собранных Z. Chen и M. Cafarella[25] из корпуса SAUS. По сравнению с примерами первого теста, они оформлены с применением дополнительных приемов, включая перерезы, чередование столбцов с данными и заголовками, выравнивание и декорирование ячеек иерархических заголовков. Особенностью исходных данных является некорректная физическая структура (примерно в половине случаев она не соответствует визуальному представлению). В рамках настоящей работы подготовлена версия данной коллекции с корректной физической структурой. Эталонные данные насчитывают около 136800 вхождений, 20100 меток, 387500 пар типа вхождение-метка и 17900 — метка-метка. Тестируемое решение реализовано в виде 18 CRL-правил[26]. Полученные результаты его оценки приведены в табл. 5.

Сравнение предлагаемого метода АИТ с аналогами. Среди известных методов со схожей целью только следующие имеют дело с РК: Tango[27], Senbazuru[28] и HUSS [Wu23] Результаты количественного сравнения с аналогами приводятся в табл. 6. Предлагаемые наборы CRL-правил дают значительно лучшие результаты по сравнению со специализированным решением Senbazuru, немного уступая лидеру, а именно недавней разработке HUSS. Следует отметить, что последняя в основном лучше справляется с восстановлением родительско-дочерних связей между ячейками заголовка на коллекции SAUS200. Изучение исходных данных показало

[23] https://tc11.cvc.uab.es/datasets/Troy_200_1

[24] https://github.com/tabbydoc/tabbyxl/wiki/troy

[25] https://dbgroup.eecs.umich.edu/project/sheets/datasets.html

[26] https://github.com/tabbydoc/tabbyxl/wiki/saus

[27] https://tango.byu.edu

[28] https://dbgroup.eecs.umich.edu/project/sheets

наличие там разнообразных приемов визуального оформления иерархических заголовков в боковике. Некоторый уровень относительно соответствующего ему подуровня может быть выделен отступом или выступом произвольной длины, выравниванием по центру, шрифтом, пустой строкой, ключевыми словами и пр. При этом перечисленные приемы оформления могут быть двусмысленными, а число уровней вложенности может доходить до десятка. Большая часть допущенных ошибок была вызвана тем, что не были учтены все варианты применения перечисленных приемов. Допустимо предположить, что качество результатов можно улучшить за счет совершенствования тестируемого решения.

Таблица 6

Сравнение с аналогами АИТ на тестах Troy200 и SAUS200

Показатели		Troy200			SAUS200		
		Senbazuru	TabbyXL	HUSS	Senbazuru	TabbyXL	HUSS
P	ВМ	-	0,98	0,99	-	0,96	0,99
	ММ	0,90	0,97	0,99	0,88	0,96	0,98
R	ВМ	-	0,97	0,98	-	0,95	0,98
	ММ	0,89	0,93	0,99	0,88	0,78	0,92
F_1	ВМ	-	0,97	0,98	-	0,95	0,99
	ММ	0,89	0,95	0,99	0,88	0,86	0,95

Извлекаются связи двух типов: ВМ — вхождение-метка и ММ — метка-метка.

Представленные данные опубликованы авторами HUSS.

Качественное сравнение с аналогами АИТ приведено в табл. 7.

Таблица 7

Сравнение с аналогами АИТ по качественным характеристикам

Характеристики	Tango	Senbazuru	HUSS	TabbyXL
Ролевой анализ	+	+/- *	+	+
Структурный анализ	+	+	+	+
Предлагаемый подход	П	МО	П	П
Сложность реализации	ИП	ИП/ОУ	ИП	ДП
Произвольность структуры:				
свободная компоновка	-	-	-	+
структурированность ячеек	-	-	-	+
иерархичность заголовков	+/- **	+	+	+

* Только распознавание функциональных типов строк.
** Только иерархия верхнего заголовка (шапки).
П — правила, МО — машинное обучение.
И/ДП — императивное/декларативное программирование; ОУ — обучение с учителем.

Следует отметить, что применение конкурентных методов ограничено частным случаем, а именно таблицами англоязычной государственной статистики. Рассматриваемая ими модель предполагает, что любая таблица строго делится на четыре функциональных региона: угловик, шапка, боковик и тело. Следует отметить, что данное предположение выполняется далеко не всегда даже в используемых ими тестовых примерах. Будучи ориентированными на наличие заданных функциональных регионов, они не поддерживают произвольность структуры. Хотя ими выполняется анализ иерархических заголовков, однако ни один из них не касается

структурированности самих ячеек. Более того, реализация конкурентных методов основана на императивном программировании и обучении с учителем классификаторов (CRF/SVM). Предлагаемые ими правила реализованы в составе алгоритмов неотделимым образом.

В отличие от аналогов, предлагаемое решение (TabbyXL) выполняет обработку данных посредством пользовательских правил. Они реализуются с помощью декларативного программирования, упрощенного проблемно-ориентированным языком правил CRL. Специфические ограничения выносятся в CRL-правила, позволяющие порождать целевые алгоритмы. Другим преимуществом предлагаемого метода является поддержка свободной компоновки и структурированности ячеек. В отличие от аналогов, используемая им модель не ограничивает структуру таблицы заданными функциональными регионами атомарных ячеек. Напротив, допускается произвольное размещение функциональных элементов в структурированных ячейках.

Таким образом, полученные результаты оценки производительности показывают эффективность предлагаемых методов. Количественное сравнение с аналогами РТ/АИТ свидетельствует о соответствии реализованных решений современному уровню технологического развития в рассматриваемой области. В то же время качественное сравнение выявляет следующие преимущества перед аналогами. Реализация предлагаемого метода РТ не требует предварительной настройки параметров и обучения с учителем. Однако при наличии готовых нейросетевых моделей они могут заменить алгоритмы обнаружения таблиц на основе правил. При этом качество окончательных результатов может быть улучшено за счет применения фильтрации кандидатных случаев. Альтернативные методы РТ либо работают с растром, либо не в полной мере используют PDL-специфику. Предлагаемый метод АИТ для РК является единственным, в котором структура документной таблицы не ограничена типичностью компоновки и атомарностью ячеек. Кроме того, по сравнению с другими методами АИТ, обеспечивается создание более простых по форме решений. На представленных примерах показано, что 16–18 CRL-правил могут заменить специализированные алгоритмы, реализованные с помощью императивного программирования и обучения с учителем. На коллекциях Troy200/SAUS200 качество работы тестируемых решений не достигает 100 % в силу противоречивости приемов оформления таблиц. Однако для случаев, когда все таблицы имеют однотипную компоновку и форматирование, такое качество может быть достигнуто.

Применение в прикладных задачах

Применение в анализе документов включает три прикладных задачи. Первая из них касается паспортов безопасности химической продукции, распространяемых в формате PDF и наполненных документными таблицами. Исходный код TabbyPDF был положен в основу прикладного ПО извлечения информации из такой документации, разработанного компанией «CloudSDS Inc.»[29] (США).

Вторая прикладная задача состоит в извлечении фактов из отчетов *оценки кредитного риска*. Кредитная организация анализирует такие отчеты, чтобы оценить кредитоспособность потенциальных корпоративных заемщиков. Существенная часть фактов, необходимых для принятия решения о кредитовании, представлена в таблицах редактируемого формата. Извлечение таких фактов нуждается в распознавании логических связей между ячейками. Исходный код TabbyXL, а именно модули, реализующие объектную модель документной таблицы, были применены при разработке ПО интеллектуального анализа отчетов оценки кредитного риска, разработанного компанией «Mosaik Risk Solutions Inc.»[30] (Индия).

Третья прикладная задача — разработка решений РТ, ориентированных на интеллектуальный анализ научной литературы формата PDF. Исследователи (Z. Chi и др.) из Пекинского технологического университета разработали новую архитектуру ИНС для распознавания

[29] https://cloudsds.com

[30] http://www.mosaikrisk.com

структуры таблицы — GraphTSR. В качестве входных данных GraphTSR ИНС принимает граф, составленный из блоков текста, извлекаемых из PDF-документов (научных статей) с помощью ПО TabbyPDF. Кроме того, ПО TabbyPDF включено в качестве одного из базовых решений (наряду с Tabula, DeepDeSRT и др.) в два предметно-ориентированных теста производительности решений РТ, а именно SciTSR и IAIS.

Применение в интеграции данных включает три случая. В первом из них создавались тематические слои электронной карты Иркутской области на основе статистических отчетов, публикуемых Иркутскстатом[31]. Решение автоматизации извлечения статистических сведений было реализовано в виде набора CRL-правил, выполняемого с помощью ПО TabbyXL.

Другой случай применения демонстрирует наполнение информационно-аналитической системы (ИАС) Института национального развития Монголии. Источником данных выступают РК со статистическими сведениями по социально-экономическому положению аймаков Монголии. В качестве решения поставленной задачи предложен набор DRL-правил, выполняемый с помощью ПО TabbyXL.

В третьем случае рассматривается прикладная задача интеграции данных по медиапланированию. Сложившаяся практика рекламных кампаний широко применяет РК для представления медиапланов (расписаний показов рекламных материалов). Ввод данных по медиапланированию, подготовленных с помощью различных шаблонов, в единую ИАС оценки эффективности рекламных кампаний требует автоматизации процесса извлечения данных из документных таблиц РК. Для решения данной задачи компания «ООО Адвентум Консалтинг»[32] (Россия) разработала собственное ПО на базе инструментального средства TabbyXL и наборов CRL-правил, соответствующих клиентским шаблонам медиапланов.

Применение в системах, основанных на знаниях, включает два случая. Первый из них касается конструирования онтологии предметной области экспертизы промышленной безопасности[33] (ЭПБ). Отчеты ЭПБ часто содержат таблицы, описывающие результаты технического диагностирования, расчеты долговечности, остаточный ресурс и пр. Представленные там сущности и связи между ними могут составлять фрагменты предметной онтологии ЭПБ. Рассматриваемая задача включает автоматизацию извлечения таких данных из отчетов ЭПБ ИркутскНИИхиммаш[34] с помощью ПО TabbyXL.

Второй случай применения связан с автоматизацией кросс-контекстного обмена бизнес-документами с табличным содержимым (опросными листами, счетами на оплату, коммерческими предложениями и пр.). Сложность обмена ими состоит в том, что, будучи подготовленными в одном контексте, они затем должны интерпретироваться в другом контексте. Поскольку они не сопровождаются формальной моделью, участники обмена вынуждены вручную вводить передаваемые им данные в собственные информационные системы. При этом различия в контекстах могут приводить к неправильной трактовке загруженных данных. В Гуанчжоуском университете группой исследователей под руководством S. Yang разработана технология организации кросс-контекстного обмена такой информацией, в рамках которой реализован компонент извлечения данных из документных таблиц, адаптирующий предлагаемый в настоящей работе метод автоматизации АИТ.

Таким образом, практическая применимость методов, представленных в настоящей работе, показана на восьми прикладных задачах, пять из которых решались в научных и три — в индустриальных проектах.

---

[31] https://38.rosstat.gov.ru

[32] https://www.adventum.ru

[33] http://www.kremlin.ru/acts/bank/11232

[34] http://hm.irk.ru

Заключение

В работе представлены методы и комплекс инструментальных средств АПТ, совокупность которых составляет теоретические и технологические основы решения проблемы упрощения разработки прикладного программного обеспечения извлечения данных из документных таблиц за счет поддержки произвольности табличной структуры:

- Усовершенствована структура совокупности задач АПТ [Shi23], в которой согласована терминология, сложившаяся в родственных направлениях исследований: компьютерном зрении, управлении данными, информационном поиске и «Семантической паутине». По сравнению с имеющимися формулировками АПТ, она является более релевантной за счет согласованной терминологии и многоуровневой декомпозиции.
- Предложен метод автоматизации распознавания таблиц НПОД [Шиг24, Шиг22, Shi11], т. е. конвертирования их в редактируемый формат РК. Впервые данный процесс базируется на использовании PDL-специфичной информации НПОД. Показано, как ее можно задействовать для анализа компоновки страниц, обнаружения и сегментации таблиц НПОД.
- Создана модель таблицы, обеспечивающая представление табличной информации в процессах АПТ [Шиг25, Shi17]. В отличие от известных аналогов, она не ограничивает структуру таблицы предопределенными типами компоновки, атомарностью ячеек и плоскими заголовками.
- Предложен метод автоматизации анализа и интерпретации таблиц редактируемого формата РК [Шиг25, Shi15, Шиг13–Шиг15]. Впервые данный процесс реализуется посредством пользовательских правил. Обеспечена поддержка произвольности структуры таблицы: свободной компоновки, структурированности ячеек и иерархичности заголовков.
- Создан проблемно-ориентированный язык правил анализа и интерпретации таблиц [Shi17]. В отличие от формальных языков общего назначения, он позволяет сфокусироваться исключительно на реализации логики соответствующих этапов АПТ.
- Разработан комплекс инструментальных средств [Shi19, Yur21], реализующий предлагаемые теоретические основы. По сравнению с другими доступными инструментами АПТ он обеспечивает конвертирование таблиц НПОД в редактируемый формат РК с возможностью пользовательской коррекции результатов и извлечение наборов записей в канонической форме из полученных таблиц РК с помощью правил, предоставляемых пользователями.

Экспериментальные результаты показали соответствие количественных оценок предлагаемых методов текущему мировому уровню по данной тематике (на разных тестах F_1-мера составила от 54,9 % до 92,1 % на этапе РТ и от 93,7 % до 98,5 % на этапе АИТ). При этом качественно преодолеваются некоторые ограничения, присущие современному состоянию исследования, главным из которых является отсутствие в конкурентных методах поддержки свободной компоновки таблиц и структурированности ячеек.

Предложенные модели и методы могут быть положены в основу создания новых информационных технологий и программных продуктов сбора и интеграции данных, представленных в документных таблицах. В сравнении с имеющимися инструментальными средствами общего назначения, предлагаемые технологические решения могут позволить сократить затраты на разработку целевого ПО АПТ.

Благодарности и поддержка

Автор выражает глубокую признательность И. В. Бычкову и А. Е. Хмельнову за научное консультирование и обмен идеями. Исследование выполнено при поддержке грантами РНФ 18-71-10001, РФФИ 17-47-380007, 16-57-44034, 15-37-20042, 12-07-31051, Совета по грантам Президента РФ СП-3387.2013.5, Минобрнауки России на выполнение крупного научного проекта по приоритетным направлениям научно-технологического развития 075-15-2024-533.

СПИСОК ЛИТЕРАТУРЫ | REFERENCES

[Ada21] T. Adams, et al. (2021). Benchmarking table recognition performance on biomedical literature on neurological disorders. Bioinformatics. 10.1093/bioinformatics/btab843.

T. Adams, et al. (2021). Benchmarking table recognition performance on biomedical literature on neurological disorders. Bioinformatics. 10.1093/bioinformatics/btab843.

[Göb12] M. Göbel, et al. (2012). A methodology for evaluating algorithms for table understanding in PDF documents. Proc. ACM DocEng. 10.1145/2361354.2361365.

M. Göbel, et al. (2012). A methodology for evaluating algorithms for table understanding in PDF documents. Proc. ACM DocEng. 10.1145/2361354.2361365.

[Hur00] M. Hurst (2000). The interpretation of tables in texts: Ph.D. thesis / Univ. of Edinburgh. http://hdl.handle.net/1842/7309.

M. Hurst (2000). The interpretation of tables in texts: Ph.D. thesis / Univ. of Edinburgh. http://hdl.handle.net/1842/7309.

[Kie98] Kieninger T. (1998). Table structure recognition based on robust block segmentation. Doc. Recognition V.

Kieninger T. (1998). Table structure recognition based on robust block segmentation. Doc. Recognition V.

[Shi11] Shigarov A., Fedorov R. Simple algorithm page layout analysis // Pattern Recognition and Image Analysis. 2011. 21(2), 324–327. 10.1134/S1054661811021008. RHIMIH.

Shigarov A., Fedorov R. Simple algorithm page layout analysis // Pattern Recognition and Image Analysis. 2011. 21(2), 324–327. 10.1134/S1054661811021008. RHIMIH.

[Shi15] Shigarov A. Table understanding using a rule engine // Expert Systems with Applications. 2015. 42(2), 929–937. 10.1016/j.eswa.2014.08.045. UEGHLB.

Shigarov A. Table understanding using a rule engine // Expert Systems with Applications. 2015. 42(2), 929–937. 10.1016/j.eswa.2014.08.045. UEGHLB.

[Shi17] Shigarov A., Mikhailov A. Rule-based spreadsheet data transformation from arbitrary to relational tables // Information Systems. 2017. 71, 123–136. 10.1016/j.is.2017.08.004. XNTUCT.

Shigarov A., Mikhailov A. Rule-based spreadsheet data transformation from arbitrary to relational tables // Information Systems. 2017. 71, 123–136. 10.1016/j.is.2017.08.004. XNTUCT.

[Shi19] Shigarov A., Khristyuk V., Mikhailov A. TabbyXL: Software platform for rule-based spreadsheet data extraction and transformation // SoftwareX. 2019. 10, 100270. 10.1016/j.softx.2019.100270. VIEIBF.

Shigarov A., Khristyuk V., Mikhailov A. TabbyXL: Software platform for rule-based spreadsheet data extraction and transformation // SoftwareX. 2019. 10, 100270. 10.1016/j.softx.2019.100270. VIEIBF.

[Shi23] Shigarov A. Table understanding: problem overview // WIREs Data Mining and Knowledge Discovery. 2023. 13(1), e1482. 10.1002/widm.1482. DVFJBC.

Shigarov A. Table understanding: problem overview // WIREs Data Mining and Knowledge Discovery. 2023. 13(1), e1482. 10.1002/widm.1482. DVFJBC.

[Wan96] X. Wang (1996). Tabular abstraction, editing, and formatting: Ph.D. thesis / Univ. of Waterloo.

X. Wang (1996). Tabular abstraction, editing, and formatting: Ph.D. thesis / Univ. of Waterloo.

[Wu23] X. Wu, et al. (2023). HUSS: A heuristic method for understanding the semantic structure of spreadsheets. Data Intelligence. 10.1162/dint_a_00201.

X. Wu, et al. (2023). HUSS: A heuristic method for understanding the semantic structure of spreadsheets. Data Intelligence. 10.1162/dint_a_00201.

[Yep21] J. Yepes, et al. (2021). ICDAR 2021 Competition on scientific literature parsing. Proc. 16th ICDAR 2021. 10.48550/arXiv.2106.14616.

J. Yepes, et al. (2021). ICDAR 2021 Competition on scientific literature parsing. Proc. 16th ICDAR 2021. 10.48550/arXiv.2106.14616.

[Yur21] Yurin A., Dorodnykh N., Shigarov A. Semi-automated formalization and representation of the engineering knowledge extracted from spreadsheet data // IEEE Access. 2021. 9, 157468–157481. 10.1109/ACCESS.2021.3130172. ZFRJUM.

Yurin A., Dorodnykh N., Shigarov A. Semi-automated formalization and representation of the engineering knowledge extracted from spreadsheet data // IEEE Access. 2021. 9, 157468–157481. 10.1109/ACCESS.2021.3130172. ZFRJUM.

[Шиг13] Шигаров А. О., Бычков И. В. и др. Система трансформации таблиц // Информационные технологии и вычислительные системы. 2013. № 3. С. 15–26. RCDLGP.

Shigarov A. O., Bychkov I. V. et al. Table transformation system // Information Technologies and Computing Systems. 2013. No. 3, pp. 15–26. RCDLGP. (In Russian).

[Шиг14] Шигаров А. О. Восстановление логической структуры таблиц из неструктурированных текстов на основе логического вывода // Вычислительные технологии. 2014. Т. 19, № 1. С. 87–99. RYYHBV.

Shigarov A. O. Restoring the logical structure of tables from unstructured texts based on logical inference // Computational Technologies. 2014. Vol. 19, No. 1. P. 87–99. RYYHBV. (In Russian).

[Шиг15] Шигаров А. О., Бычков И. В. и др. Анализ и интерпретация произвольных таблиц на основе исполнения CRL-правил // Вычислительные технологии. 2015. Т. 20, № 6. С. 87–112. VKAMER.

Shigarov A. O., Bychkov I. V. et al. Analysis and interpretation of arbitrary tables based on the execution of CRL rules // Computational Technologies. 2015. Vol. 20, No. 6. P. 87–112. VKAMER. (In Russian).

[Шиг22] Шигаров А. О., Парамонов В.В. Сегментация текста неразмеченных PDF-документов // Вычислительные технологии. 2022. Т. 27, № 5. С. 69–78. 10.25743/ICT.2022.27.5.007. DAOWQG.

Shigarov A.O., Paramonov V.V. Text segmentation of unmarked PDF documents // Computational technologies. 2022. Т. 27, no. 5, pp. 69–78. 10.25743/ICT.2022.27.5.007. DAOWQG. (In Russian).

[Шиг24] Шигаров А. О. Распознавание таблиц неаннотированных PDF-документов на основе использования PDF-специфичных свойств // Выч. технологии. 2024. Т. 29, № 6. С. 125–146. 10.25743/ICT.2024.29.6.008. CVSWTN.

Shigarov A. O. Recognition of tables in unannotated PDF documents based on the use of PDF-specific properties // Computational Technologies. 2024. Vol. 29, No. 6. P. 125–146. 10.25743/ICT.2024.29.6.008. CVSWTN. (In Russian).

[Шиг25] Шигаров А. О. Извлечение реляционных данных из произвольных таблиц электронных документов редактируемых форматов на основе пользовательских правил // Вычислительные технологии. 2025. Т. 30, № 3. С. 127–144. 10.25743/ICT.2025.30.3.010. XFYDSE.

Shigarov A. O. Extracting relational data from arbitrary tables of electronic documents of editable formats based on user rules // Computational Technologies. 2025. T. 30, no. 3, pp. 127–144. 10.25743/ICT.2025.30.3.010. XFYDSE. (In Russian).

ОБ АВТОРЕ | ABOUT THE AUTHOR

ШИГАРОВ Алексей Олегович
Институт динамики систем и теории управления имени В. М. Матросова Сибирского отделения РАН, Россия.
shigarov@gmail.com ORCID: 0000-0001-5572-5349.
Ведущий научный сотрудник лаб. комплексных информационных систем. Дипл. математик (Иркутский гос. ун-т). Д-р техн. наук (Ин-т динамики систем и теории управления им. В.М. Матросова СО РАН). Труды в обл. программной инженерии, управления и обработки информации.

SHIGAROV Alexey Olegovich
Matrosov Institute of System Dynamics and Control Theory, Siberian Branch of the Russian Academy of Sciences, Russia.
shigarov@gmail.com ORCID: 0000-0001-5572-5349.
Leading researcher of the Laboratory of Complex Information Systems. Certified mathematician (Irkutsk State University). Dr of Tech. Sci. (Matrosov Institute of System Dynamics and Control Theory, Siberian Branch of the Russian Academy of Sciences). Works in the field of software engineering, data management, and information processing.

МЕТАДАННЫЕ | METADATA

Заглавие: Автоматизация процессов извлечения данных из таблиц электронных документов неструктурированного формата: методы и инструментальные средства

Авторы: Шигаров А. О.

Аннотация: Представлен обзор исследований по созданию методов автоматизированного понимания таблиц (АПТ) и комплекса инструментальных средств на их основе для упрощения разработки прикладного программного обеспечения извлечения данных из документных таблиц с машиночитаемым текстовым содержимым, представленных в неструктурированном виде, за счет поддержки произвольности табличной структуры. Решены следующие задачи: проанализирована совокупность задач АПТ и усовершенствована ее структура; выявлены ограничения текущего исследования вопросов АПТ и сформулированы актуальные направления его развития; предложен метод автоматизации распознавания таблиц в НПОД, т. е. конвертирования их в редактируемый формат РК; предложен метод автоматизации анализа и интерпретации таблиц РК, т. е. извлечения из них наборов записей в канонической форме; разработаны инструментальные средства на основе предлагаемых методов; выполнена оценка производительности реализованных решений и сравнение их с аналогами; изучены возможности практической применимости предлагаемых методов и инструментальных средств. Методы исследования основаны на применении эвристик, машинного обучения, систем исполнения правил, моделей данных, формальных грамматик, трансляции программ и тестов производительности.

Ключевые слова: Электронные документы; неструктурированные форматы; документные таблицы; автоматизированное понимание таблиц; PDF; PostScript; Excel; Sheets.

Язык: Русский.

Статья поступила в редакцию 16 июня 2026 г.

Title: Automation of data extraction processes from tables of unstructured electronic documents: methods and tools.

Authors: Shigarov A. O.

Abstract: This paper presents a review of research in the field of creating automated table understanding (ATU) methods and a set of tools based on them to simplify the development of application software for extracting data from document tables with machine-readable text content presented in an unstructured form by supporting the arbitrariness of the table structure. The following problems are solved: a set of ATU problems is analyzed and its structure is improved; limitations of the current ATU research are identified and current directions for its development are formulated; a method for automating table recognition in unstructured document tables, i.e., converting them into an editable RK format, is proposed; a method for automating the analysis and interpretation of RK tables, i.e., extracting record sets from them in canonical form, is proposed; tools are developed based on the proposed methods; the performance of the implemented solutions is assessed and compared with similar ones; the possibilities of practical applicability of the proposed methods and tools are studied. Research methods are based on the use of heuristics, machine learning, rule execution systems, data models, formal grammars, program translation and performance tests.

Key words: Electronic documents; unstructured formats; document tables; automated table understanding; PDF; PostScript; Excel; Sheets.

Language: Russian.

The article was received by the editors on 16 June 2026.