Система трансформации таблиц[1]

А.О. Шигаров, И.В. Бычков, Г.М. Ружников, А.Е. Хмельнов, Р.К. Федоров

Аннотация. Работа посвящена вопросам анализа логической компоновки таблицы в задаче структурирования табличной информации, содержащейся в неструктурированных документах и изначально предназначенной для восприятия человеком, а не для машинной обработки. Предлагается система трансформации таблицы от слабоструктурированного представления к отношению реляционной модели данных. Эта система обеспечивает полуавтоматическое восстановление используемых в таблице измерений (доменов). Трансформация ориентирована на таблицы, сформированные из баз данных.

Ключевые слова: анализ и распознавание документов, извлечение информации из таблиц, анализ и обработка таблиц, преобразование таблиц.

Введение

В настоящее время в мире накоплено большое количество документов, которые изначально адресованы для восприятия человеком, например, отчеты, научные статьи, счета, формы, газеты, e-mail сообщения. Обычно такие документы содержат неструктурированную или слабоструктурированную информацию, не предназначенную для машинной обработки. Как показано в недавних исследованиях [1, 5], количество таких документов продолжает быстро увеличиваться.

Одним из наиболее используемых способов представления информации в документах являются таблицы. Например, в статистических и финансовых отчетах таблицы часто являются основным способом представления информации. В задачах прогнозирования, планирования и принятия решений часто необходимо анализировать и обрабатывать такие таблицы. Для автоматизации таких задач информация, представленная в таблицах, должна быть структурирована. Однако обычно публикуемые таблицы не содержат вовсе или содержат только часть метаданных, необходимых для машинной обработки табличной информации. Даже высокоуровневое представление таблицы, например, в виде HTML разметки или объекта текстового процессора Word, не включает информацию о разделении ячеек на данные и заголовки, связях между ячейками, а также об используемых типах данных. Для преобразования такой таблицы к структурированному виду, например, к отношениям в терминах реляционной модели данных, необходимо восстановить отсутствующие метаданные.

Массовое ручное структурирование информации из таблиц является трудоемким процессом, связанным с большим количеством ошибок обработки. Для автоматизации этого процесса требуются методы и системы анализа и обработки таблиц, обеспечивающие преобразование табличной информации к структурированному виду. При этом задачи, которые необходимо решать, зависят от уровня представления исходных данных. Для высокоуровневого представления одной из важнейших задач структурирования является анализ логической компоновки (logical layout analysis, в терминах работы [6]), т.е. присвоение каждому элементу таблицы смыслового значения.

[1] Работа выполнена при финансовой поддержке РФФИ грант № 12-07-31051 и Совета по грантам Президента РФ СП-3387.2013.5

	Sent			Received		
	FY2010	FY2011	2011/2010 (%)	FY2010	FY2011	2011/2010 (%)
	Letters					
EU						
Spain	462.9	469.4	101.4	556.3	576.4	103.6
Cyprus	82.9	89.7	108.2	97.1	101.7	104.7
Belgium	352.3	341.1	96.82	387.2	366.1	94.5
Middle East						
Lebanon	21.1	21.5	101.9	19.8	19.5	98.5
Israel	353.8	483.0	136.5	365.8	376.0	102.8
	Parcels					
EU						
Spain	102.2	109.3	106.9	134.2	145.4	108.3
Middle East						
Lebanon	12.3	13.1	106.5	11.7	11.3	96.6

Рис.1. Пример таблицы

Настоящая работа посвящена вопросам анализа логической компоновки таблицы в задаче структурирования табличной информации. В работе предлагается система трансформации таблицы (CTT) от слабоструктурированного представления, содержащего информацию о ячейках и связях между ними, к отношению реляционной модели данных. Предлагаемая система обеспечивает полуавтоматическое восстановление измерений (в терминах OLAP**,** Online Analytical Processing), определяющих типы данных табличных заголовков. Рассматриваемая трансформация ориентирована на таблицы (пример показан на Рис. 1.), являющиеся результатом использования систем генерации отчетов, сводных таблиц (pivot table) в OLAP системах или в табличных процессорах (например, Excel), а также систем кросс-табуляции (например, “TPL Tables”). Правила их оформления описаны в документах [3, 12]. Такие таблицы могут иметь сложную компоновку с вложенными и охватывающими заголовками, перерезами (заголовками внутри тела таблицы), объединенными ячейками. Рассматриваемая система трансформации таблицы (CTT) является развитием технологии извлечения табличной информации из документов, предложенной авторами в работе [15].

1. Связанные работы

В последние годы активно развивается область исследований, называемая анализом и распознаванием документов, АРД (Document Analysis and Recognition, DAR). В работе [6] цель АРД формулируется, как автоматическое извлечение информации, представленной на бумаге и изначально адресованной для восприятия человеком. Существенной частью АРД является анализ и распознавание таблиц (АРТ). Важная задача АРТ - преобразование табличной информации к реляционному представлению. В работах [2, 10] такое преобразование называется *канонизацией таблицы* (*table canonicalization*).

В [2] приводится определение *канонической формы* (*canonical form*) *таблицы*, под которой понимается отношение в терминах реляционных баз данных. Авторы определяют, что такая таблица, изображенная в двумерном “матричном” виде, имеет *каноническую компоновку* (*canonical layout*). Ими предлагается метод интерпретации таблиц, которые содержатся в документах (спецификациях), используемых в строительной промышленности. Метод [2] предназначен для *канонизации* таких таблиц, т.е. преобразования их к канонической форме. Для этого Douglas и др. предлагают использовать обработку естественного языка. В частности, для разделения табличных заголовков и данных, а также для отнесения заголовков к определенным доменам, предлагается использовать онтологию предметной области (подъязык спецификаций строительной промышленности).

В работе [10] описывается платформа TANGO (Table ANalysis for Generating Ontologies, [9]), для инкрементальной генерации концептуальной онтологии из таблиц, содержащихся в web-страницах. Tijerino и др. [10] приводят формальное определение канониче-

ской таблицы, под которой они, как и авторы [2], понимают отношение в терминах реляционной модели данных. В TANGO канонизация таблицы является первым этапом в процессе генерации онтологии. Предлагаемый ими способ канонизации основан на использовании библиотеки фреймов данных (*data frames*, в терминах работы [10]), содержащей знания о лексическом содержании таблиц. Каждый фрейм данных описывает один абстрактный тип данных, например, числовой, денежный, координаты (широта долгота), проценты, географические названия, единицы измерения, религии. Такой фрейм данных используется как распознаватель для отнесения выражения на естественном языке (табличных заголовков и значений) к определенному типу данных (измерению). Для описания типов данных предлагается использовать регулярные выражения, словари, лексическую базу данных английского языка WordNet [11] и другие ресурсы. Например, для идентификации географических названий среди табличных заголовков и данных предлагается использовать WordNet, а также доступные в Интернете словари географических названий (газеттиры).

В известном обзоре по проблематике обработки таблиц [4] на основе формального определения канонической таблицы из работы [10] приводится формальное определение понятия — *понимание таблицы* (*table understanding*). Embley и др. определяют понимание таблицы как восстановление отношения в терминах реляционной модели данных из информации, представленной в таблице. Другими словами, канонизация таблицы является процессом понимания таблицы в терминах работы [4].

Следует отметить, что приведение к канонической форме используется для таблиц со сложной компоновкой, например, для кросс-таблиц (*cross-tabulation*), которая отличается от "решеточной" ("матричной"), используемой по умолчанию в табличных процессорах, например, Excel. При "решеточной" компоновке табличные строки непосредственно соответствуют записям таблицы в базе данных, а столбцы — полям. Для обработки таблиц с "решеточной" компоновкой, как правило, можно обойтись стандартными средствами импорта данных, например, DTS (Data Transformation Services), включенным в систему управления базами данных (СУБД) "SQL Server". Кроме того, каждая канонизируемая таблица может иметь уникальную компоновку. Это отличает их от слабоструктурированных табличных форм (например, счетов, бланков), которые имеют одинаковое расположение и функции полей на просвет в тех случаях, когда они являются экземплярами одного шаблона. Для автоматизации ввода таких форм в базы данных существует ряд программных продуктов на основе систем оптического распознавания символов (например, "OmniPage Capture SDK").

2. Структурное описание таблицы

Предлагаемая система предназначена для канонизации таблицы, которая представлена в слабоструктурированном виде, включающем информацию о декомпозиции таблицы на ячейки и их содержимом (заголовках и данных), связях между табличными данными и заголовками и отношениях подчиненности между заголовками. Модель такого представления предложена авторами в работах [14, 15], в последней из них она называется *структурным описанием таблицы*.

В СТТ используется измененная модель таблицы, которая включает дерево заголовков столбцов, дерево заголовков строк и множество связанных с заголовками фактов (значений данных). На Рис. 2 приводится пример структурного описания таблицы, показанной на Рис. 1. От модели, предлагавшейся в работах [14, 15], она отличается отсутствием отдельного дерева перерезов. В настоящей модели перерезы (заголовки внутри тела таблицы, например, "Letters" и "Parcels" в таблице на Рис. 1) рассматриваются как охватывающие заголовки строк.

В этой модели табличные заголовки образуют узлы соответствующего дерева, а отношения подчиненности между заголовками вида (h_1, h_2), где заголовок h_2 вложен в охватывающий заголовок h_1, соответствуют его ребрам. Корнем такого деверева является пустой узел (мнимый заголовок), в который вложены заголовки самого верхнего уровня. Одно из

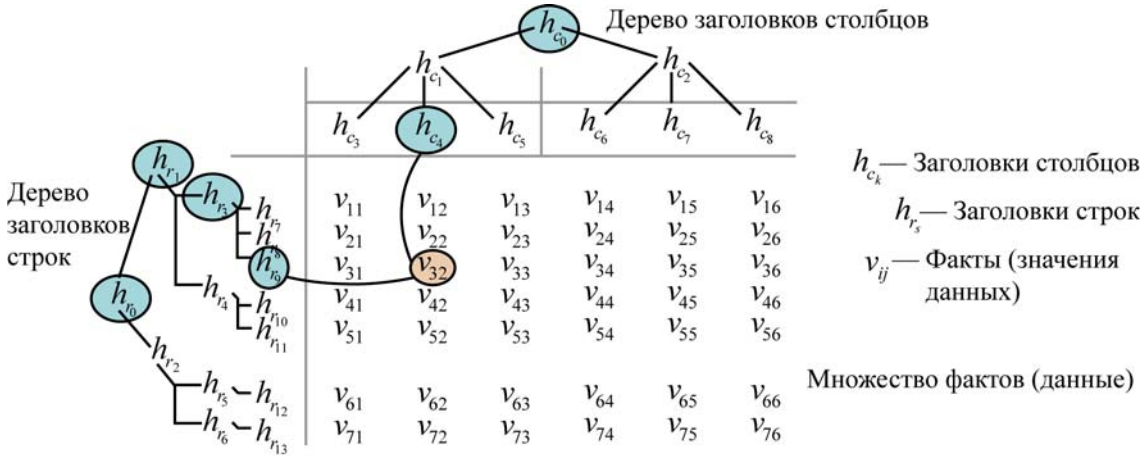


Рис.2. Структурное описание таблицы из Рис.1

деревьев в таблице может быть *вырожденным*, т.е. состоять только из одного корня. Это соответствует случаям, когда у таблицы полностью отсутствуют либо заголовки строк, либо заголовки столбцов.

В структурном описании каждый факт (значение данных) *связан* с парой заголовков, т.е. для каждого факта определена тройка следующего вида:

$$\left(v_{ij}, h_{c_k}, h_{r_s}\right),$$

где v_{ij} — факт, а h_{c_k} — заголовок столбца и h_{r_s} — заголовок строки. Например, в таблице из Рис. 1 факт v_{32}="341.1" связан с парой заголовков: $h_{c_4} = \text{"2011"}$ и $h_{r_7} = \text{"Belgium"}$. Вместе они образуют отношение $\left(v_{32}, h_{c_4}, h_{r_7}\right)$. Если одно из деревьев заголовков вырождено, тогда один из двух заголовков, связанных со значением данных, является корнем этого дерева.

Если в дереве заголовок h_1 охватывает подзаголовок h_2, который связан с фактом v, то считается, что охватывающий заголовок h_1 также *связан* с фактом v. Другими словами, если факт v *связан* с заголовком h_2, то он *связан* со всеми его предками. Дополнительно, чтобы отличить непосредственную связь заголовка с фактом от связи через подзаголовки, предполагается, что в каждой тройке вида $\left(v, h_2, h_3\right)$ факт v *связан напрямую* с заголовками h_2 и h_3, но не связан напрямую с их предками.

3. Трансформация структурного описания таблицы

Предлагаемая трансформация состоит в отображении табличной информации от представления в виде структурного описания к отношению реляционной модели данных. Структурное описание не содержит необходимую для канонизации таблицы информацию об используемых в ней измерениях (типах данных). В СТТ выполняется анализ логической компоновки таблицы, целью которого является восстановление измерений.

Проблемы извлечения табличной информации из неструктурированных документов и формирование структурного описания таблицы подробно рассматриваются в работах [7, 8, 13, 15]. В частности, метод обнаружения таблиц (поиска в документе ограничивающих прямоугольников таблиц) предлагается в работах [7, 13, 15]. Анализ физической компоновки (сегментации таблицы на столбцы, строки и ячейки) и часть анализа логической компоновки таблицы (разделение ячеек на заголовки и данные, определение связей между ячейками) рассматриваются в работах [8, 15].

В данном разделе предлагается дальнейшая обработка и канонизация таблицы. Рассматриваемая трансформация структурного описания

включает следующие выполняемые последовательно этапы: предобработка; разметка заголовков; восстановление измерений; формирование таблицы реляционной базы данных.

3.1. Предобработка

Заголовки внутри одной или нескольких таблиц могут различаться по написанию, но иметь одно лексическое значение, т.е. могут являться синонимами. Например, следующие заголовки: “2010”, “FY2010”, “Year 2010” и “Previous Year”, “2010 г.”, “Текущий год” могут быть синонимами, означающими 2010 год. В качестве эталонного значения приведенных заголовков может использоваться выражение “2010”. Предобработка начинается с замены значений таких заголовков на их эталоны. Такая замена позволяет сделать последующую разметку заголовков и восстановление измерений более простыми.

Предобработка выполняется автоматически и основана на использовании *словаря эталонов*, в котором естественно-языковые выражения и регулярные выражения сопоставляются с эталонными выражениями. Этот словарь формируется вручную и содержит набор отношений вида (R_1, R_2), где R_1 — исходное выражение, которое необходимо заменить на эталонное выражение, а R_2 — целевое эталонное выражение. Например, для автоматического приведения следующих значений заголовков “FY2010”, “Year 2010” и “Previous Year”, “2010 г.”, “Текущий год” к эталонному значению “2010” необходимо добавить в словарь соответствующие сопоставления: (“FY2010”, “2010”), (“Year 2010”, “2010”),…, (“Текущий год”, “2010”). Информация о том, что “Previous Year” или “Текущий год” означают 2010 год, задается пользователем, исходя из контекста обрабатываемой таблицы. При обработке таблиц добавленные сопоставления будут использоваться в автоматическом приведении заголовков к эталонам до тех пор, пока не будут отключены, изменены или удалены. Для замены заголовков на эталоны также могут быть использованы регулярные выражения. Например, если в словаре задать отношения вида (“FY[2][0][0-1][0-9]”, “[2][0][0-1][0-9]”), то все заголовки, соответствующие регулярному выражению “FY[2][0][0-1][0-9]”, т.е. “FY2000”, …, “FY2010”, “FY2011”,…, “FY2019”, будут заменены на соответствующие эталоны “2000”,…, “2010”, “2011”,…, “2019”. В частности, такую замену можно использовать в случае обработки таблицы из Рис. 1. После такого приведения к эталонам заголовки столбцов h_{c_3}, h_{c_4}, h_{c_6} и h_{c_7} из структурного описания (Рис. 2) примут следующий вид: h_{c_3} =“2010”, h_{c_4} =“2011”, h_{c_6} =“2011” и h_{c_7} =“2011”.

В таблице некоторые заголовки могут тиражироваться, т.е. в таблице может содержаться несколько заголовков с одинаковым значением. Например, в таблице из Рис. 1 часть заголовков строк (“EU”, “Spain” и т.д.) тиражируется для каждого перереза (“Letters” и “Parcels”), так же как и часть заголовков столбцов (“FY2010”, “FY2011” и “2011/2010(%)”) тиражируется для охватывающих заголовков “Sent” и “Received”. Предполагается, что заголовки с одинаковым значением являются экземплярами одного растиражированного заголовка. В этих случаях в структурном описании таблицы некоторые факты будут связаны с разными заголовками, имеющими одинаковое значение. Например, в структурном описании из Рис. 2 в отношениях $\left(v_{11}, h_{c_3}, h_{r_7}\right)$ и $\left(v_{14}, h_{c_6}, h_{r_7}\right)$ заголовки столбцов h_{c_3} и h_{c_6} имеют одинаковое значение — “FY2010”.

Для дальнейшей обработки структурного описания необходимо, чтобы оно содержало только уникальные по значению заголовки. Для этого в структурном описании таблицы все экземпляры $h_2, \ldots, h_n$ одного растиражированного заголовка h_1 заменяются на этот заголовок h_1. При этом каждый факт v, связанный напрямую с заменяемым заголовком $h_i, i = \{2, \ldots n\}$, связывается взамен этого с заголовком h_1. Например, в структурном описании из Рис. 2 в дереве заголовков столбцов заголовок h_{c_6} =“2010” заменяется на заголовок h_{c_3} =“2010”. При этом отношения $\left(v_{14}, h_{c_6}, h_{r_7}\right), \ldots, \left(v_{74}, h_{c_6}, h_{r_7}\right)$ приводятся к виду $\left(v_{14}, h_{c_3}, h_{r_7}\right), \ldots, \left(v_{74}, h_{c_3}, h_{r_7}\right)$. После предобра-

ботки деревья заголовков в структурном описании из Рис. 2 примут вид, показанный на Рис. 3.

3.2. Разметка заголовков

После предобработки выполняется разметка заголовков, которая состоит в сопоставлении заголовку метки, задающей тип данных: принадлежность заголовка некоторому заданному измерению или вычислимость (относительность) связанных с заголовком данных. Эта операция выполняется полуавтоматически. Разметка основана на использовании *словаря измерений*, в котором естественно-языковые выражения и регулярные выражения сопоставляются с заранее заданными измерениями. Словарь измерений формируется вручную и содержит набор отношений вида (R, D_i), где R — значение заголовка (строковая константа или регулярное выражение), которому требуется присвоить метку его измерения, а D_i — измерение (домен), описываемый выражением R.

Часто таблицы в документах (например, отчетах) генерируются из баз данных и OLAP систем. При этом значения измерений, используемых в OLAP, становятся заголовками в публикуемых отчетных таблицах. Например, данные в таблицах из статистических отчетов часто имеют пространственно-временную привязку. Заголовки в таких таблицах часто являются значениями измерений времени (даты, месяцы, годы) и географических названий (регионы, страны, районы).

Для того чтобы восстановить измерения из обрабатываемой таблицы, заголовкам присваиваются метки, задающие типы данных. Этот процесс выполняется с помощью словаря по аналогии с описанным приведением заголовков к эталонам. Предполагается, что измерение можно описать с помощью наборов естественно-языковых или регулярных выражений. Например, в некоторых таблицах измерение годы может описываться как “[0-9]{4}”, а измерение районы как “.+ район”.

В частности, в таблице из Рис. 1 заголовки относятся к следующим измерениям: D_1 — произведенная операция (“Sent”, “Received”); D_2 — время (“FY2010”, “FY2011”), D_3 — тип отправления (“Letters” “Parcels”); D_4 — регионы (“EU”, “Middle East”); D_5 — страны (“Spain”, “Cyprus”, …, “Israel”). Эти измерения можно описать следующим набором сопоставлений: (“Sent”, D_1), (“Received”, D_1), (“FY[2][0][0-1][0-9]”, D_2), (“Letters”, D_3), (“Parcels”, D_3), (“EU”, D_4), (“Middle East”, D_4), (“Spain”, D_5), (“Cyprus”, D_5),…, (“Israel”, D_5).

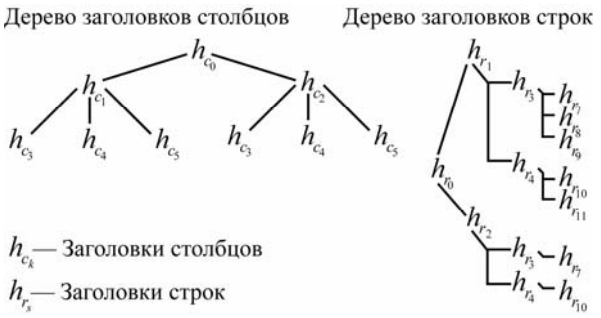


Рис.3. Предобработка

Заголовки сравниваются с выражениями из словаря. Если заголовок удовлетворяет некоторому выражению, то он помечается как значение соответствующего измерения. Например, в структурном описании из Рис. 2, если заданы метки измерений $D_1,\ldots, D_5$, то заголовки помечаются следующим образом: h_{c_1}, h_{c_2} — D_1; h_{c_3}, h_{c_4} — D_2; h_{r_1}, h_{r_2} — D_3; h_{r_3}, h_{r_4} — D_4; h_{r_7}, $h_{r_8},\ldots,h_{r_{11}}$ — D_5.

В рассматриваемом преобразовании некоторые данные, содержащиеся в обрабатываемой таблице, могут являться ненужными для представления в целевом виде. Например, в таблицах часто используются вычислимые (относительные) данные, которые можно вычислить по другим данным этой же таблицы или других доступных таблиц. Такие данные можно исключать из дальнейшего преобразования таблицы к целевой форме. Разметка заголовков также используется для того, чтобы пометить заголовки, связанные с игнорируемыми данными. Например, таблицы часто имеют заголовки, содержащие символ “%”, которые описывают процентные соотношения, вычисляемые из других табличных данных, или заголовки “Итого”, “Всего”, “Total”, описывающие итоговые данные, агрегированные из других табличных данных. В частности, в таблице из Рис. 1 данные, связанные с заголовком

$h_{c_5} = \text{"2010/2011(\%)"}$, являются вычислимыми по данным, представленным в этой же таблице в столбцах “FY2010” и “FY2011”.

В этих случаях заголовкам, связанным с игнорируемыми данными, также сопоставляются метки по аналогии с разметкой значений измерений. Такие заголовки можно рассматривать как значения специального измерения игнорируемых данных — *I*. Например, в случае обработки таблицы из Рис. 1 для распознавания заголовков игнорируемых данных в словарь можно добавить следующее сопоставление — (“[0-9]{4}/[0-9]{4}\(%\)”, *I*) или более общее сопоставление — (“.+%.*”, *I*). Кроме того, дополнительно к каждому сопоставлению (R, I), определенному для измерения игнорируемых данных, может быть определено два маркера. Первый из них указывает на способ разметки: помечается только заголовок, соответствующий выражению *R*; дополнительно к этому заголовку также помечаются все его подзаголовки. Второй маркер указывает на способ дальнейшей обработки помеченных заголовков и связанных с ними игнорируемых данных: из дальнейшей обработки исключаются только данные, связанные напрямую с помеченным заголовком; дополнительно к этим данным также исключается сам заголовок. С помощью сопоставления заданным в словаре измерений выражениям выполняется разметка заголовков игнорируемых данных. Например, если для таблицы из Рис. 1 определено приведенное сопоставление с измерением *I*, то заголовок c_3 помечается как значение измерения *I* .

Результаты автоматической разметки заголовков можно корректировать вручную, т.е. изменять, добавлять и отменять метки, задающие заголовкам типы данных. Пример разметки заголовков структурного описания из Рис. 2 показан на Рис. 4.

3.3. Восстановление измерений

После разметки заголовков выполняется восстановление измерений. Прежде всего из обрабатываемого структурного описания исключаются игнорируемые данные и напрямую связанные с ними заголовки по правилам (маркерам), заданным в результате разметки. Затем

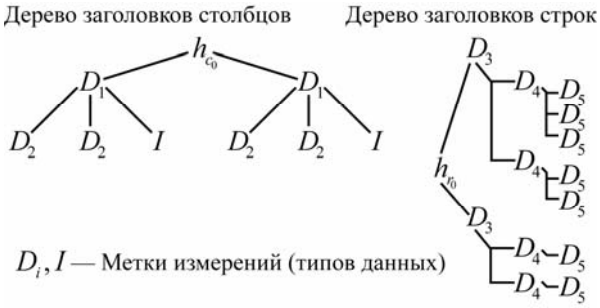


Рис. 4. Разметка заголовков

исключаются заголовки, помеченные как значения остальных измерений. При этом в дереве заголовков дети исключаемого узла становятся детьми родителя исключаемого узла. Каждый такой исключаемый заголовок добавляется в список значений измерения, соответствующего его типу данных. Например, для структурного описания таблицы из Рис. 2 можно восстановить следующие измерения:

1. $D_1 = \left\{h_{c_1} = \text{"Sent"}, h_{c_2} = \text{"Received"}\right\}$ — произведенная операция,
2. $D_2 = \left\{h_{c_3} = \text{"2010"}, h_{c_4} = \text{"2011"}\right\}$ — время,
3. $D_3 = \left\{h_{r_1} = \text{"Letters"}, h_{r_2} = \text{"Parcels"}\right\}$ — тип отправлений,
4. $D_4 = \left\{h_{r_3} = \text{"EU"}, h_{r_4} = \text{"Middle East"}\right\}$ — регионы,
5. $D_5 = \left\{h_{r_7} = \text{"Spain"}, h_{r_8} = \text{"Cyprus"}, \ldots, h_{r_{11}} = \text{"Israel"}\right\}$ — страны,
6. $I = \left\{h_{c_5} = \text{"2011/2010(\%)"}\right\}$ — игнорируемые данные.

Трансформации, производимые над деревьями заголовков структурного описания из Рис. 2 в процессе восстановления измерений, показаны на Рис. 5.

Если в результате разметки каждый заголовок был отнесен к некоторому измерению, то в результате восстановления измерений деревья заголовков станут вырожденными. Однако вовсе не обязательно, чтобы для каждого заголовка было определено измерение (тип данных). Предполагается, что некоторые измерения,

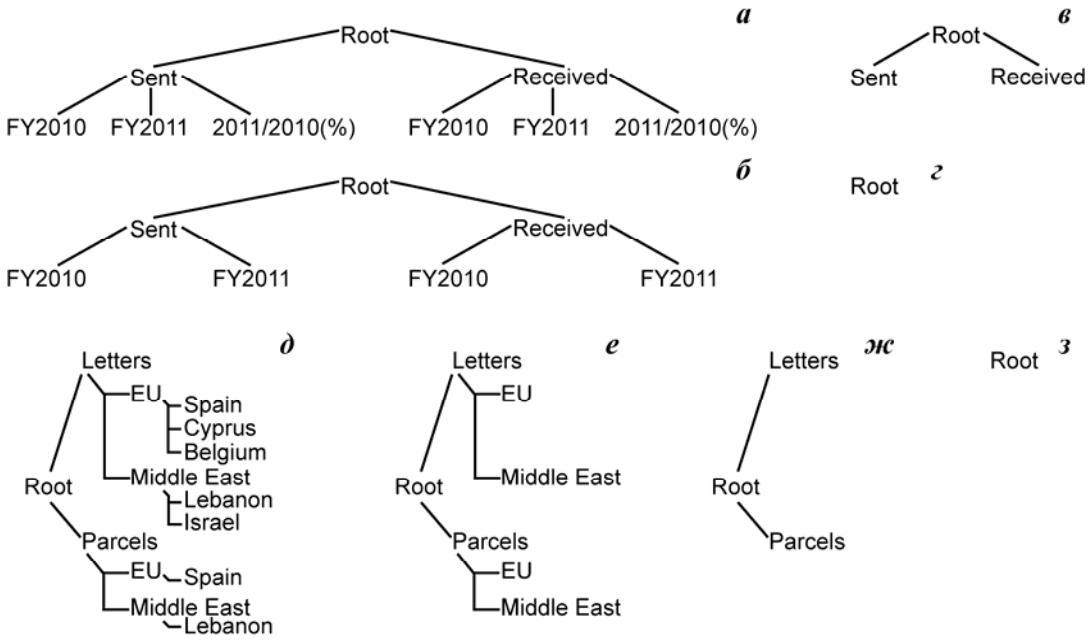


Рис. 5. Трансформации над деревьями заголовков в процессе восстановления измерений

дерево заголовков столбцов: исходное — (*а*), после исключение измерения игнорируемых данных *I* — (*б*), после исключения измерения "времени" D_1 — (*в*), после исключения измерения "операций" D_2 — (*г*); дерево заголовков строк: исходное — (*д*), после исключение измерения "страны" D_3 — (*е*), после исключения измерения "регионы" D_4 — (*ж*), после исключения измерения "типы отправлений" D_5 — (*з*)

присутствующие в таблице, могут быть заданы только частично или не заданы вовсе. В таких случаях после описанного исключения заголовков в дереве заголовков могут образоваться листы или целые поддеревья, которые имеют одного родителя и при этом являются полностью идентичными. Например, если из дерева заголовков столбцов исключить измерение $D_1 = \left\{h_{c_1} = \text{"Sent"}, h_{c_2} = \text{"Received"}\right\}$, но не исключать измерение $D_2 = \left\{h_{c_3} = \text{"2010"}, h_{c_4} = \text{"2011"}\right\}$, то корень будет иметь четыре подузла — два h_{c_3} ="2010" и два h_{c_4} ="2011". В таких случаях идентичные листы или поддеревья объединяются. В результате для каждого родителя листов его дети представляют уникальные заголовки.

В документах важная информация, описывающая табличные данные, часто находится не в самой таблице, а в её контексте. Например, период времени (год, месяц) или территория (регион, страна), к которому относятся табличные данные, могут содержаться в названии отчета или в заголовке раздела документа или в названии таблицы. Для того чтобы учесть такую информацию, в процессе восстановления измерений могут задаваться дополнительные измерения и их значения исходя из контекста таблицы в соответствии с решаемой задачей. Для каждого такого измерения задается единственное значение, которое связывается с каждым фактом в обрабатываемой таблице.

3.4. Формирование таблицы реляционной базы данных

Преобразование структурного описания к отношению реляционной модели данных начинается с того, что невырожденные деревья заголовков (т.е. содержащие, по крайней мере, один узел помимо корня) объединяются в одно дерево, называемое *деревом атрибутов*. Для этого дерево заголовков столбцов и дерево заголовков строк обрабатываемой таблицы выстраиваются в двухуровневом заранее заданном порядке вложенности друг в друга. Вложенное дерево заголовков тиражируется для каждого

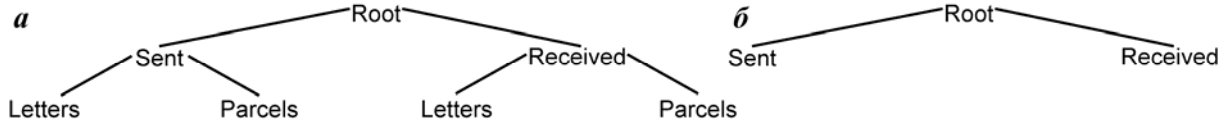


Рис. 6. Примеры деревьев атрибутов, построенных для таблицы из Рис. 1, для следующих случаев: не исключались измерения D_1 и D_3 — (*а*); не исключалось измерение D_1 — (*б*)

листа охватывающего дерева заголовков. При этом корень вложенного дерева исключается из его экземпляров. Экземпляры вложенного дерева становятся поддеревьями с корнями в листах охватывающего дерева. Дети исключенного корня (заголовки) вложенного дерева становятся в таком поддереве детьми соответствующего листа (заголовка) охватывающего дерева. Примеры деревьев атрибутов, полученных для таблицы из Рис. 1, показаны на Рис. 6. В этих примерах, выбран следующий порядок вложенности деревьев заголовков, сверху вниз: уровень 1 — заголовки столбцов, уровень 2 — заголовки строк.

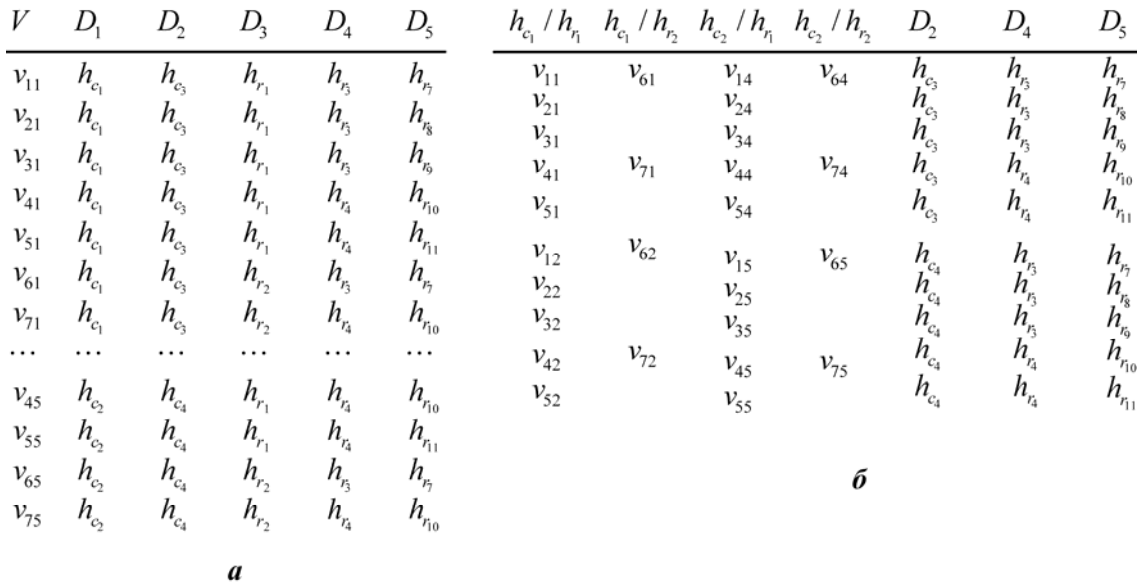
После этого на основе структурного описания, восстановленных измерений и дерева атрибутов формируется каноническая таблица (отношение). Для каждой ветки L от корня h_0 до листа h_n дерева атрибутов, которая образована заголовками: $h_0, h_1, \ldots, h_n$, последовательно вложенными друг в друга: $(h_0, h_1), (h_1, h_2), \ldots, (h_{n-1}, h_n)$, формируется отдельное поле (столбец) фактов $A = A(L)$. Оно образуется всеми фактами, связанными с заголовками $h_1, \ldots, h_n$ из соответствующей ветки дерева атрибутов. Кроме того, все факты, не связанные с заголовками из дерева атрибутов, образуют отдельное поле фактов A_V. Если дерево атрибутов вырождено, то все значения данных формируют одно единственное поле фактов A_V. Также для каждого восстановленного измерения D формируется отдельное поле измерения $A = A(D)$, которое образуется из значений соответствующего измерения.

При формировании канонической таблицы поля фактов и измерений заполняются согласованно между собой таким образом, что в каждый кортеж включаются только те значения данных, которые связаны только с одинаковыми значениями измерений, и включены только те значения измерений, которые связаны с этими фактами. Для этого рассматриваются тройки следующего вида $(v, A(L), H(v))$, где v — факт, $A(L)$ — поле фактов, в которое включено значение данных v, и $H(v) = \{h_1, \ldots, h_n\}$ — набор всех заголовков, которые связаны с фактом v и при этом являются значениями измерений, т.е. они не включены в дерево атрибутов. Сформированная каноническая таблица с полями $A(L_1), \ldots, A(L_n), A(D_1), \ldots, A(D_m)$ будет состоять из кортежей следующего вида:

$$(v_1, \ldots, v_n, h_1, \ldots, h_m),$$

где $v_1, \ldots, v_n$: $v_1, \ldots, v_n \in \{V, \varnothing\}$ — факты с идентичными наборами значений измерений $H(v_1) = H(v_2) = \ldots = H(v_n)$ или пустые значения, а $h_1, \ldots, h_m$ — значения измерений или пустые значения, такие что $h_1, \ldots, h_m \in \{\mathrm{H}(v_1), \varnothing\}$, $h_1 \in \{D_1, \varnothing\}, \ldots, h_m \in \{D_m, \varnothing\}$. Будем считать, что такие значения данных $v_1, \ldots, v_n$ и значения измерений $h_1, \ldots, h_m$ *согласованы* между собой.

Поле фактов $A(L_1)$ заполняется значениями данных, связанными с веткой L_1, в любом порядке. Следующие поля фактов $A(L_2), A(L_3), \ldots, A(L_n)$ заполняются значениями данных, связанными с ветками соответственно $L_2, L_3, \ldots, L_n$, в согласованном порядке с уже сформированными (предыдущими) полями фактов соответственно $A(L_1)$; $A(L_1), A(L_2)$;…; $A(L_1), \ldots, A(L_{n-1})$. Поля измерений $A(D_1), \ldots, A(D_m)$ заполняются значениями измерений соответственно $D_1, \ldots, D_m$ в порядке,



V	D_1	D_2	D_3	D_4	D_5
v_{11}	h_{c_1}	h_{c_3}	h_{r_1}	h_{r_3}	h_{r_7}
v_{21}	h_{c_1}	h_{c_3}	h_{r_1}	h_{r_3}	h_{r_8}
v_{31}	h_{c_1}	h_{c_3}	h_{r_1}	h_{r_3}	h_{r_9}
v_{41}	h_{c_1}	h_{c_3}	h_{r_1}	h_{r_4}	$h_{r_{10}}$
v_{51}	h_{c_1}	h_{c_3}	h_{r_1}	h_{r_4}	$h_{r_{11}}$
v_{61}	h_{c_1}	h_{c_3}	h_{r_2}	h_{r_3}	h_{r_7}
v_{71}	h_{c_1}	h_{c_3}	h_{r_2}	h_{r_4}	$h_{r_{10}}$
…	…	…	…	…	…
v_{45}	h_{c_2}	h_{c_4}	h_{r_1}	h_{r_4}	$h_{r_{10}}$
v_{55}	h_{c_2}	h_{c_4}	h_{r_1}	h_{r_4}	$h_{r_{11}}$
v_{65}	h_{c_2}	h_{c_4}	h_{r_2}	h_{r_3}	h_{r_7}
v_{75}	h_{c_2}	h_{c_4}	h_{r_2}	h_{r_4}	$h_{r_{10}}$

а

h_{c_1}/h_{r_1}	h_{c_1}/h_{r_2}	h_{c_2}/h_{r_1}	h_{c_2}/h_{r_2}	D_2	D_4	D_5
v_{11}	v_{61}	v_{14}	v_{64}	h_{c_3}	h_{r_3}	h_{r_7}
v_{21}		v_{24}		h_{c_3}	h_{r_3}	h_{r_8}
v_{31}		v_{34}		h_{c_3}	h_{r_3}	h_{r_9}
v_{41}	v_{71}	v_{44}	v_{74}	h_{c_3}	h_{r_4}	$h_{r_{10}}$
v_{51}		v_{54}		h_{c_3}	h_{r_4}	$h_{r_{11}}$
v_{12}	v_{62}	v_{15}	v_{65}	h_{c_4}	h_{r_3}	h_{r_7}
v_{22}		v_{25}		h_{c_4}	h_{r_3}	h_{r_8}
v_{32}		v_{35}		h_{c_4}	h_{r_3}	h_{r_9}
v_{42}	v_{72}	v_{45}	v_{75}	h_{c_4}	h_{r_4}	$h_{r_{10}}$
v_{52}		v_{55}		h_{c_4}	h_{r_4}	$h_{r_{11}}$

б

Рис. 7. Примеры канонических таблиц: случай, когда дерево атрибутов вырождено — (*а*), и не вырождено — (б)

согласованном с уже сформированными полями фактов $A(L_1),\ldots,A(L_n)$. Если в кортеж формируемого отношения для поля фактов $A(L_k)$ добавлен факт v_1: $(v_1, A(L_k), H(v_1))$, а для поля фактов $A(L_{k+s})$ нет факта v_2: $(v_2, A(L_k), H(v_2))$, связанного с тем же набором значений измерений, т.е. $H(v_1) = H(v_2)$, то в кортеж в поле $A(L_{k+s})$ добавляется пустое значение. Если для поля фактов $A(L_k)$ добавлен факт v_1: $(v_1, A(L_k), H(v_1))$, а для поля измерений $A(D_k)$ нет значения измерения из набора $H(v_1)$, то в кортеж в поле $A(D_k)$ добавляется пустое значение.

Если ни одного измерения не было восстановлено, то отношение будет состоять всего из одного кортежа. Напротив, если восстановлены все измерения, содержащиеся в исходной таблице, то целевое отношение будет содержать максимально возможное количество кортежей, равное количеству не исключенных фактов. На Рис. 7 показаны два примера отношений, полученных для исходного структурного описания из Рис. 2: для обозначения ветки используется нотация вида $h_1 / h_2 / \ldots / h_{n-1} / h_n$, где заголовки $h_1,\ldots,h_n$ последовательно охватывают друг друга, h_1 вложен в корень дерева атрибутов, а h_n является листом. Соответствующие примеры таблиц реляционной базы данных показаны на Рис. 8.

Заключение

Содержащиеся в документах таблицы часто формируются с помощью различных генераторов отчетов из баз данных. Ввод табличной информации из неструктурированных документов в базы данных может рассматриваться как обратная задача генерации таблиц. В работе предложена система трансформации таблицы от высокоуровневого представления в виде структурного описания к отношению реляционной модели данных. Трансформация состоит в восстановлении недостающих метаданных о таблице (информации об используемых измерениях), необходимых для ее канонизации.

Система является развитием технологии извлечения табличной информации из неструктурированных документов, предложенной авторами в работе [15]. Разработанный механизм канонизации таблицы в составе этой технологии может использоваться для автоматизации ввода в базы данных табличной информации из неструктурированных документов.

Данные	Операция	Год	Тип отправления	Регион	Страна
462.9	Sent	2010	Letters	EU	Spain
82.9	Sent	2010	Letters	EU	Cyprus
352.3	Sent	2010	Letters	EU	Belgium
21.1	Sent	2010	Letters	Middle East	Lebanon
353.8	Sent	2010	Letters	Middle East	Israel
102.2	Sent	2010	Parcels	EU	Spain
12.3	Sent	2010	Parcels	Middle East	Lebanon
469.4	Sent	2011	Letters	EU	Spain
89.7	Sent	2011	Letters	EU	Cyprus
341.1	Sent	2011	Letters	EU	Belgium
21.5	Sent	2011	Letters	Middle East	Lebanon
483.0	Sent	2011	Letters	Middle East	Israel
109.3	Sent	2011	Parcels	EU	Spain
13.1	Sent	2011	Parcels	Middle East	Lebanon
556.3	Received	2010	Letters	EU	Spain
97.1	Received	2010	Letters	EU	Cyprus
387.2	Received	2010	Letters	EU	Belgium
19.8	Received	2010	Letters	Middle East	Lebanon
365.8	Received	2010	Letters	Middle East	Israel
134.2	Received	2010	Parcels	EU	Spain
11.7	Received	2010	Parcels	Middle East	Lebanon
576.4	Received	2011	Letters	EU	Spain
101.7	Received	2011	Letters	EU	Cyprus
366.1	Received	2011	Letters	EU	Belgium
19.5	Received	2011	Letters	Middle East	Lebanon
376.0	Received	2011	Letters	Middle East	Israel
145.4	Received	2011	Parcels	EU	Spain
11.3	Received	2011	Parcels	Middle East	Lebanon

а

Sent/ Letters	Sent/ Parcels	Received/ Letters	Received/ Parcels	Год	Регион	Страна
462.9	102.2	556.3	134.2	2010	EU	Spain
82.9		97.1		2010	EU	Cyprus
352.3		387.2		2010	EU	Belgium
21.1	12.3	19.8	11.7	2010	Middle East	Lebanon
353.8		365.8		2010	Middle East	Israel
469.4	109.3	576.4	145.4	2011	EU	Spain
89.7		101.7		2011	EU	Cyprus
341.1		366.1		2011	EU	Belgium
21.5	13.1	19.5	11.3	2011	Middle East	Lebanon
483.0		376.0		2011	Middle East	Israel

б

Рис. 8. Таблицы реляционной базы данных

Литература

1. Bohn R.E., Short J.E. How Much Information? 2009. Report on American Consumers // Global Information Industry Center. University of California, San Diego. 2010.
2. Douglas S., Hurst M., Quinn D. Using Natural Language Processing for Identifying and Interpreting Tables in Plain Text // In: Proceedings of the 4th annual symposium on document analysis and information retrieval, Las Vegas, 15–17 April 1995, pp 535–546.
3. «Chicago Manual of Style, 15th Edition». University of Chicago Press, 2003. 948 p.
4. Embley D.W., Hurst M., Lopresti D., Nagy G. Table-processing paradigms: a research survey // International Journal on Document Analysis and Recognition. 2006. Vol. 8, No. 2. P. 66-86.
5. Lyman P., Varian H.R. How Much Information? Technical Report. University of California at Berkeley. 2003.
6. Machine Learning in Document Analysis and Recognition // Series: Studies in Computational Intelligence, Vol. 90. Marinai S., Fujisawa H. (Eds.). 2008. 434 p.

7. Shigarov A.O., Bychkov I.V., Hmelnov A.E., Ruzhnikov G.M. A method for table detection in metafiles // Pattern Recognition and Image Analysis. 2009. Vol. 19, No 4. P. 693-697.
8. Shigarov A.O., Hmelnov A.E. Analisis and segmentation of tables from electronic unstructured documents // In Proc. Int. Conf. on Mathematical and Informational Technologies (Zbornik radova konferencije MIT 2009). Kopaonik, Serbia, Budva, Montenegro, 2009. P. 373-376.
9. TANGO (Table ANalysis for Generating Ontologies), http://tango.byu.edu
10. Tijerino, Y., Embley, D., Lonsdale, D., Nagy, G.: Towards ontology generation from tables. World Wide Web: Internet and Web Information Systems. 2005. Vol. 8, No 3. P. 261-285.
11. WordNet, http://wordnet.princeton.edu
12. ГОСТ 2.105-95 ЕСКД «Общие требования к текстовым документам». М.: ИПК Издательство стандартов, 2001. 27 с.
13. Бычков И.В., Ружников Г.М., Хмельнов А.Е., Шигаров А.О. Эвристический метод обнаружения таблиц в разноформатных документах // Вычислительные технологии. 2009. Т. 14, № 2 С. 58-73.
14. Хмельнов А.Е., Шигаров А.О. Метод извлечения таблиц из неформатированного текста // Вычислительные технологии. 2008. Т. 13. Спец. выпуск 1. С. 93-101.
15. Шигаров А.О. Технология извлечения табличной информации из электронных документов разных форматов. Дис. канд. тех. наук. Иркутск. 2010.142 с.

Шигаров Алексей Олегович. Научный сотрудник Института динамики систем и теории управления Сибирского отделения РАН. Окончил Иркутский государственный университет в 2004 году. Кандидат технических наук. Автор 35 печатных работ. Область научных интересов: анализ и распознавание документов, извлечение информации, геоинформационные системы, web-технологии. E-mail: shigarov@icc.ru

Бычков Игорь Вячеславович. Директор Института динамики систем и теории управления Сибирского отделения РАН. Окончил Иркутский государственный университет в 1983 году. Доктор технических наук, член-корреспондент РАН. Автор 169 печатных работ. Область научных интересов: икусственный интеллект, геоинформационные системы, проблемно-ориентированные базы данных, системы интеллектуального анализа данных, распределенные вычисления, грид. E-mail: bychkov@icc.ru.

Ружников Геннадий Михайлович. Заместитель директора Института динамики систем и теории управления Сибирского отделения РАН. Окончил Иркутский государственный университет в 1970 году. Кандидат технических наук, старший научный сотрудник. Автор 147 печатных работ. Область научных интересов: искусственный интеллект, распознавание образов, геоинформационные системы, web-технологии, системы интеллектуального анализа данных.
E-mail: rugnikov@icc.ru

Хмельнов Алексей Евгеньевич. Заведующий лабораторией Института динамики систем и теории управления Сибирского отделения РАН. Окончил Московский физико-технический институт в 1991 году. Кандидат технических наук, доцент. Автор 67 печатных работ. Область научных интересов: формальные спецификации, обработка и представление информации, геоинформационные системы, информационно-поисковые системы, базы данных, web-технологии.
E-mail: hmelnov@icc.ru

Федоров Роман Константинович. Ведущий научный сотрудник Института динамики систем и теории управления Сибирского отделения РАН. Окончил Иркутский государственный университет в 1999 году. Кандидат технических наук. Автор 35 печатных работ. Область научных интересов: обработка изображений, сигналов и сцен, распознавание образов, геоинформационные технологии, искусственный интеллект и принятие решений. E-mail: fedorov@icc.ru