Table understanding using a rule engine

Alexey O. Shigarov[a,*]

[a]*Institute for System Dynamics and Control Theory of SB RAS, Lermontov st. 134, Irkutsk, Russia, 664033*

Abstract

The paper discusses issues on the conversion of tabular data from unstructured to structured form. Particularly, we propose an approach to table understanding (i. e. recovering semantic relationships in a table), which is designed for unstructured tabular data integration. Our approach is based on using a rule engine. It is assumed that spatial, style (typographical), and natural language information can be used for table analysis and interpretation. The *CELLS* system based on the approach has been developed for integrating unstructured tabular data presented in Excel spreadsheet format. Experimental results show that the approach and system can be applied to a wide range of tables from statistical and financial reports.

Keywords: table understanding, table canonicalization, information extraction from tables, unstructured tabular data integration, table model

1. Introduction

Nowadays, many researchers in data management (e.g. Doan et al., 2009; Ferrucci & Lally, 2004; Inmon & Nesavich, 2007) note that issues on unstructured data management and integration become increasingly important. The term "unstructured information / data" usually refers to any information that does not have a predefined formal data model or does not fit into a table of a relational database. If unstructured information contains some text (e. g. plain-

*Corresponding author
Email address: shigarov@icc.ru (Alexey O. Shigarov)

text, PDF, or Word documents) then it is called "unstructured textual information / data". More accurate terms "weakly structured" and "semi-structured documents" (Feldman & Sanger, 2006) are used to indicate unstructured textual information.

The documents may contain tables which do not have any formal data model. These tables are intended to be interpreted by humans but not designed for high-level machine processing like SQL queries. Therefore, in the sense defined above, these tables are examples of unstructured textual information. By analogy, they may be called "unstructured tabular information / data".

Automation for transforming tabular information into structured form has important applications in problems of data management, information extraction, and document analysis systems. There are the following problems which can be considered as the conversion of tabular information from unstructured to structured form.

- Table canonicalization (Douglas et al., 1995; Tijerino et al., 2005) is transformation of a table to the canonical form that fits into the table of relational database.

- Information extraction from tables (Embley et al., 2006a) is analogous to the task of information extraction from texts and consists in extracting selectively facts to generate a target database.

- Table understanding (Embley et al., 2006a) consists in recovering relationships among data values, labels (attributes), and dimensions (domains). In general case, as Hurst (2001) notes, the table understanding involves the following steps: (1) table location (to detect positions of a table inside a source), (2) table recognition (to recover individual cells), (3) functional analysis (to find attributes and data in cells, i. e. to recover cell roles), (4) structural analysis (to recover relationships between cells), and (5) interpretation (to extract facts from a table).

The present work is restricted to the issues: how to recover relationships

of table elements (i. e. cell-role, label-value, label-label, and label-dimension pairs). In terms of Hurst (2001), we propose to automate the following steps of table understanding: functional analysis, structural analysis, and interpretation of a table.

Our approach to table understanding is based on the use of a rule engine and table analysis rules. It is expected that facts which are used in the process of logical inference may include information about spatial, style (typographical) and natural language content of tables. The implementation of rule sets for different table forms provides the processing of a wide range of tables having complex structures. The *CELLS* system based on the proposed approach has been developed for integrating unstructured tabular data. It allows extracting data from tables presented in Excel spreadsheet files. The obtained experimental results demonstrate that the system can be applied to input data from tables into a database.

2. Related work

Depending on presentation level of a table, the table understanding requires to solve different tasks (steps), such as location, recognition, analysis, and interpretation of a table, in terms of Hurst (2001). Detailed surveys of methods and systems which are devoted to these problems can be found in the following papers (Embley et al., 2006a,b; Lopresti & Nagy, 2000; e Silva et al., 2006; Zanibbi et al., 2004, 2008).

There is a huge amount of ways to portray a table. Table features originate from typographical standards, corporative practice, ad hoc software, data formats, and human inventiveness. It leads to the complexity of table understanding. The existing methods and systems related with the enumerated above steps of table understanding are based on different approaches, e. g. heuristic, machine learning, dynamic programming, or probabilistic methods. However, all of them use some assumptions about table structures to reduce the complexity of own tasks. Usually, those assumptions are embedded in their algorithms.

It significantly constrains a range of tables that can be efficiently processed by these algorithms.

The current state of research in this area does not allow to say that the problems of table understanding are completely solved. The most studies devoted to the problems of low-level table processing, such as location and recognition of tables from document images and plain-text. Meanwhile, the issues of table understanding (including analysis and interpretation) remain less studied in the case of unstructured tabular information presented in high-level formats of a word processor or spreadsheet.

In the paper, we discuss only methods related with the steps of table analysis and interpretation. Particularly, the following papers (Douglas et al., 1995; Embley et al., 2005; Gatterbauer et al., 2007; Hurst, 2000; Kim & Lee, 2008; Pivk et al., 2007; e Silva et al., 2006; Tijerino et al., 2005; Wang et al., 2012) made significant contribution to solving these problems of table understanding.

In the papers (Douglas et al., 1995; Tijerino et al., 2005) the approaches to table canonicalization are considered. The method for interpretation and canonicalization of tables which are contained in specifications used in construction industry is suggested by Douglas et al. (1995). It is based on natural language processing using domain ontology (i. e. a sub-language of construction industry specifications).

Another technique for table canonicalization proposed by Tijerino et al. (2005) is based on a library of frames containing knowledge about lexical content of tables. Each frame describes a data type using regular expressions, dictionaries, and open resources like the lexical database WordNet[1]. The frame is used to assign data types to table labels and values.

Embley et al. (2005) proposed methods for location of tables in HTML pages, and information extraction from them. It is assumed that a table may have nested tables on linked pages. In particular, in order to detect attributes (labels) and data values in cells they use ontologies developed specifically for

[1] WordNet, http://wordnet.princeton.edu

information extraction. In addition to objects, relationships and constraints an extraction ontology includes a set of data frames which are associated with sets of objects. Those data frames allow binding table content with objects of the ontology using regular expressions. As well, in table analysis they use several table recognition heuristics on table structures and content.

Wang et al. (2012) consider the problem of understanding a web table as associating the table with semantic concepts presented in a knowledge base. In particular, they use Probase[2] as that knowledge base. This method can be applied only for HTML tables with a very simple structure without merged cells, when each row of a table, excluding a single header row, describes a particular entity of the concept associated with this table.

The methods (Douglas et al., 1995; Embley et al., 2005; Tijerino et al., 2005; Wang et al., 2012) use mainly domain knowledge about natural language content of tables. However, it is not always sufficient in practice. There are many cases when the table understanding additionally requires an analysis of spatial and graphical information from tables.

An opposite domain-independent method to extract information from HTML tables is offered by Gatterbauer et al. (2007). It is based on the analysis of only spatial and style information in the CSS2 (Cascading Style Sheets Level 2) format. In particular, they propose to carry out the interpretation of the tables (recovery of semantic relationships) based on assumptions about style information designed for a set of the most common types of web-tables.

Pivk (2006); Pivk et al. (2007) present a methodology and TARTAR system for automatic transforming HTML tables of three typical types into logical structured form (semantic frames) that is intended for using with an inference engine for the query answering and ontology generation. The methodology and system are also independent of domain knowledge. They are based on heuristics on layout and text content of a table.

The paper (Kim & Lee, 2008) proposes a method for extracting logical struc-

[2]Probase, http://research.microsoft.com/en-us/projects/probase

tures (where semantic relationships between attributes and values are presented as tree) from HTML tables and transforming them into a XML representation. Their method is restricted by five types of tables. Kim & Lee (2008) use an analysis of spatial, style and natural language information from a table based on embedded rules and regular expressions.

A detailed description features of several others methods, in particular, (Chen et al., 2000; Hu et al., 2000; Hurst, 2000; Pinto et al., 2003; Yoshida et al., 2001), for functional analysis, structural analysis, and interpretation of a table is given in the paper (e Silva et al., 2006). As a rule, they are based on using some assumptions about table structures in steps of functional or structural analysis of a table. Those assumptions limit a class of tables which can be understood by these methods with a high precision and recall.

3. Class of processed tables

Now, the large volume of unstructured tabular information is presented in high-level document formats, such as Excel, Word, and HTML. The possibilities and constraints of the table presentations in these formats are similar. They allow to present the following information about a table:

- Positions of a cell in row and column coordinates;
- Merged cells (e. g. attributes COLSPAN and ROWSPAN in HTML);
- Cell style (border style, content placement, text metrics, etc.);
- Content of a cell (text, images, etc.).

However, each of the formats has its own features. So a cell can contain other tables in Word and HTML. But Excel does not supported it. HTML allows using the attributes HEADERS, SCOPE of the tags TD and TH to define relationships between headers and values. Excel determines one of the primitive data types (NUMERIC, DATE, STRING, etc.) for cell content.

Based on the listed observations the following set of general assumptions about cells describes the class of processing tables.

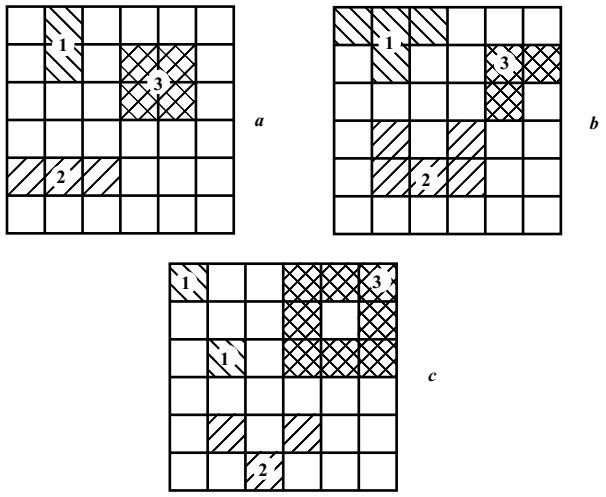

Figure 1: Examples of merging tiles in the three table cells marked as 1, 2, and 3: a cell can combine multiple tiles in Excel, Word, HTML and LaTeX (*a*), a cell can visually include a number of tiles for human reading using graphical lines (*b*), most likely no one presents a cell as shown in (*c*).

- A cell is characterized by the position (coordinates) in the column and row space, style, and content.
- A cell can be located on several consecutive rows and columns, i. e. it can cover a few grid tiles which always form a rectangle as shown in Fig. 1, *a*.
- A cell can contain only text. Although, in practice, it can have a richer content, e. g. RTF (Rich Text Format), images, or formula (in Excel) or contain other tables (cells). However, it is disregarded in this study. It is done to simplify development of data structures and algorithms of the *CELLS* system.

Moreover, the class of processed tables is also restricted by the following general assumptions about relationships of cells.

- A cell can serve as either entry or label. The terms "entry" and "label" correspond to the meaning that was suggested in the paper (Wang, 1996). An entry represents a data value and a label describes (addresses) entries.

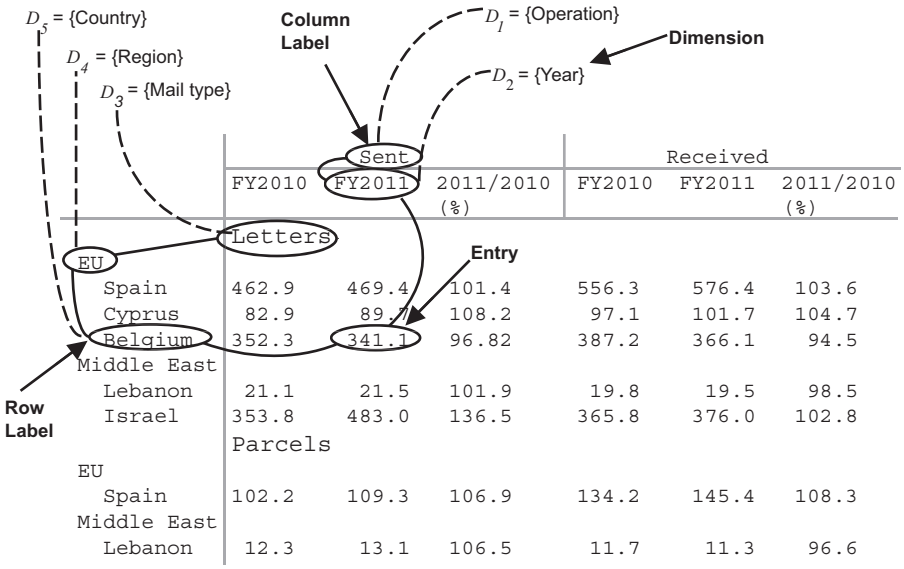

Figure 2: The example of the processed table.

- A label can address entries and other labels either in rows or columns only thus labels can form hierarchical relationships among themselves.
- A label can be a value of a dimension.

An example of a table with those relationships is shown in Fig. 2.

Note, that a table may have some context in the rest of the text of the document (title, unit, footnotes, section name, etc). However, in this study table context is not considered as part of table information for analysis and processing.

4. Table model

The proposed table model is based on the general assumptions described above. It is designed to present facts about tables in process of logical inference.

The model consists of two levels: physical and logical. The first of them presents the visual composition of a table. The second level is intended for presenting the semantic composition of a table.

The physical level describes geometric positions, styles (graphical formatting) and content of cells. This level $T_p = (S_r, S_c, C)$ consists of the following sets.

- S_r is a set of rows and S_c is a set of columns.
- C is a set of cells where each cell — $c = (c', p, G)$ includes:
  - c' is content;
  - $p = (c_l, r_t, c_r, r_b)$ are coordinates in the rows S_r and columns S_c (c_l — a left column, r_t — a top row, c_r — a right column, and r_b — a bottom row);
  - G is a set of style settings (font metrics, colors, text alignment, border styles, etc.).

The logical level presents semantic relationships (i. e. cell-role, label-value, label-label, and label-dimension pairs). This level $T_l = (D, L_r, L_c, E)$ consists of the following sets.

- $D = \{D_i\}$ is a set of dimensions presented in the processed table. Each of them is a set of dimension values $D_i = \{d_j\}$.
- L_r is a tree of row labels and L_c is a tree of column labels. These trees present relationships between their labels. Each label $l = (l')$ has content l' which is not a value of dimensions D_i — $l' \notin \bigcup D_i$.
- E is a set of entries where each entry — $e = (e', D', L')$ includes:
  - e' is content,
  - D' is a set of values from dimensions D_i related with this entry,
  - L' is a set of labels from trees L_r and L_c related with this entry.

5. Presentation and execution of table analysis rules

The major idea of our approach consists in the following. Often, tables from a collection of documents produced by a single vendor have similar layout,

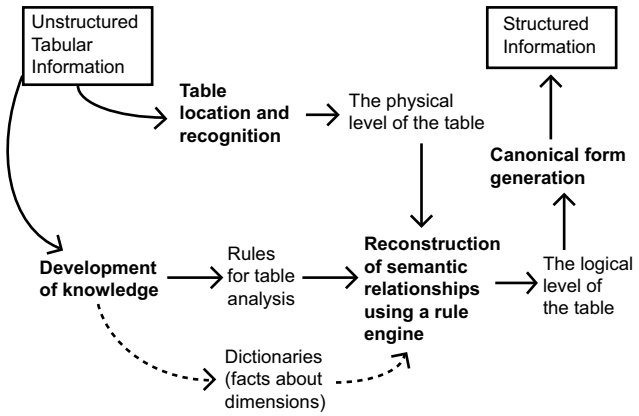

Figure 3: The diagram for the conversion of tabular data from unstructured to structured form using a rule engine.

formatting, and content. These tables are usually created by using uniform standards (e. g. "The Chicago Manual of Style"[3]), templates, or software (e. g. "TPL Tables"[4]). This provides an opportunity to define a set of formalized rules for analysis of tabular information from the document collection, so that they satisfy all or nearly all its tables. Rules can be expressed as a knowledge base, while the recovering of the logical level of a table can be carried out as logical inference. The diagram for the process of table understanding is shown in Fig. 3.

The physical level of a table is produced as the result of table location and recognition. These steps of table understanding are not considered in the paper. It is assumed that they are carried out by third-party systems (e. g. Tabula[5]) which can extract tables from PDF files, plain-text or HTML pages and convert them into Excel format. We expect that the obtained tables presented in Excel can be transformed into representation in terms of the physical level of our model. The physical level is used to generate facts for logical inference.

[3]The Chicago Manual of Style Online, http://www.chicagomanualofstyle.org
[4]TPL Tables, http://www.qqqsoft.com/html/products/tpltables.html
[5]Tabula, http://tabula.nerdpower.org

Moreover, the facts may optionally be supplemented by external information on dimensions.

Table analysis rules map the known information of the physical level (positions, graphical formatting and natural language content of cells) to unknown information of the logical level (relationships among labels, entries, and dimensions). Obtained as the result of the inference, facts about the logical level of a table should be sufficient for its canonicalization.

Logical inference of the rules can be carried out by the Drools Expert[6] rule engine. The Drools Expert system supports the specification for the Java Rule Engine API (JSR-94). It provides opportunity using objects of Java classes as facts in inference. Therefore, in the *CELLS* system, all data structures which represent the proposed model are implemented by Java. While the production rules for table analysis are expressed by the MVEL[7] language. Simplified examples of these rules expressed by MVEL are shown below.

Example 1. If a cell `$c` is located in the 1st column then it serves as a row label.

```
when
  $c : CCell( cl == 1 )
then
  modify ( $c ) { setRole( Role.ROWLABEL ) }
```

Example 2. If a cell `$c1` is directly located above another cell `$c2` spanning it in columns completely then the cell `$c1` is related with the cell `$c2`.

```
when
  $c1 : CCell()
  $c2 : CCell( rt == $c1.rb + 1,
        ( $c1.cl <= cl && cr < $c1.cr ) ||
        ( $c1.cl < cl && cr <= $c1.cr ) )
```

[6]Drools Expert (JBoss Community), http://www.jboss.org/drools/drools-expert.html
[7]MVEL, http://mvel.codehaus.org

```
then
  $c1.addConnectedCell( $c2 )
```

Example 3. If a cell `$c` is completely located in the 1st column and contains a text matching the regular expression `"(?i).*(total)"` then it is disregarded while generating output data.

```
when
  $c : CCell( cl == 1, cl == cr,
              text matches "(?i).*(total)" )
then
  modify ( $c ) { setIgnored( true ) }
```

Example 4. If there is the dimension `$d` which is named `"Religion"` and the cell `$c` contains some text in red (`"#ff0000"`) while the rest of cells located in the same row do not contain text then the cell `$` is related to the dimension `$d`.

```
when
  $d : CDimension( name == "Religion" )
  $c : CCell ( text != null,
       style.getFont().getColor() == "#ff0000" )
  not ( exists CCell ( rt == $c.rt, text != null ) )
then
  $c.setDimension( $d )
```

Example 5. If the cell `$e` serves as an entry while other cell `$l` serves as a column label and they are located in the same column then the cell `$e` is related to the cell `$l`.

```
when
  $l : CCell( role == Role.COLLABEL )
  $e : CCell( role == Role.ENTRY,
              cl == $l.cl, cr == $l.cr )
```

```
then
  $e.addConnectedCell( $l )
```

More examples of rules which are applied to test the *CELLS* system are accessible at http://cells.icc.ru/test.

6. Additional algorithms for the pre- and post-processing of tabular information

In addition to logical inference we also use the number of algorithms for transforming tabular information. They can be divided into pre- and post-processing algorithms in order of their execution with regard to logical inference.

Pre-processing includes optionally the following steps: 1) removing unnecessary whitespaces and special characters from textual content; 2) excluding empty rows and columns; 3) recovering missing style settings of cell borders. The latest procedure is necessary since the visual (human-readable) and physical (machine-readable) boundaries of a cell do not always coincide each other. Visually they may be formed by boundaries of neighbor cells. To simplify table analysis rules the style settings of the physical boundaries are recovered by the corresponding visual boundaries using the styles of the neighbor cells.

Post-processing is applied to the logical level of a table collected as the result of inference. It includes the following steps: 1) transforming textual content of cells into reference values; 2) detecting dimension values among labels; 3) generating canonical forms for processed tables. Here is a brief description of the post-processing algorithms.

Labels can be different natural language expressions with the same lexical meaning, i. e. they can be synonymous. For example, the following label expressions: “2010”, “FY2010”, “Year 2010”, “Previous Year”, “2010”, and “Current year” can be synonymous meaning 2010 year. The expression “2010” can be used as their reference value. In post-processing synonyms are replaced with their reference values through matching with a reference dictionary. It contains a set of relations (S, R) where S is a regular expression to identify synonyms

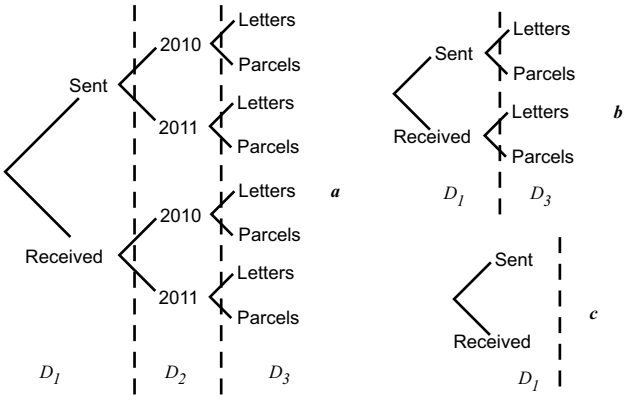

Figure 4: Reduction of the label tree in the process of detecting dimension values among labels: no recovered dimensions (a); the dimension $D_2 = \{2010, 2011\}$ (YEAR) was recovered (b); the dimensions D_2 (YEAR) and $D_3 = \{Letters, Parcels\}$ (MAIL_TYPE) were recovered (c).

and R is the appropriate reference value also specified as regular expression. For example, if the dictionary defines the following pair ("FY[2][0][0-1][0-3]", "[2][0][0-1][0-3]") then all labels matching the regular expression "FY[2][0][0-1][0-3]", i. e. "FY2000", ..., "FY2013" will be replaced with the reference values "2000", ..., "2013" respectively.

To assign labels to dimensions the dictionary that contains a set of relations (S, D_i) is used where S is a regular expression to identify the dimension D_i. As the result the labels which are assigned to the dimension D_i from the set D are excluded from the corresponding trees L_r and L_c as it is shown in Fig. 4. In this case the position that was previously occupied by the excluded label in the tree is filled by its nested labels (Fig. 4). The relationships between entries and the excluded label are replaced by the relationships between these entries and the corresponding values of dimensions D_i. In the ideal case where each label is assigned to a dimension the label trees become degenerate. Note, that the dictionaries of references and dimensions can also be used in inference as facts about tabular information.

The processed logical level of a table is used to generate the canonical form

DATA	OPERA-TION	YEAR	MAIL TYPE	REGION	COUNTRY
462.9	Sent	2010	Letters	EU	Spain
82.9	Sent	2010	Letters	EU	Cyprus
...	...	...	...	...	...
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
...	...	...	...	...	...
11.3	Received	2011	Parcels	Middle East	Lebanon

Figure 5: The canonical form for the table from the Fig. 2. All labels were assigned to dimensions so that the fields `COL_LABEL` and `ROW_LABEL` are missing.

which is include the following fields:

- `DATA` contains data (entries);
- `ROW_LABEL` contains label paths from leaves to roots in the non-degenerate tree L_r;
- `COL_LABEL` contains label paths from leaves to roots in the non-degenerate tree L_c;
- the set of fields `D_1`,...,`D_N` present values of the corresponding dimensions D_i from the set D.

Each tuple in the canonical form presents the relationships between the entry, the label path in the tree L_r, the label path in the tree L_c, and values of the recovered dimensions D_i. Optionally, the field `ROW_LABEL` or `COL_LABEL` can be divided into several separated fields. Each of them corresponds to one level of nesting in the row/column label tree. The example of canonical form is shown in Fig. 5. Generated canonical forms can be exported into a relational database using standard tools of database management systems.

a

Item	Total	National forest	Non-national forest		
			Municipal	Private	Others
Forest land area (1,000 ha)	25 121	7 838	2 796	14 440	46
Forest growing stock (1 mil. m3)	4 040	1 011	433	2 590	5
Planted forests ……..……..					
Land area (1,000 ha) ……..	10 361	2 411	1 232	6 705	12
Growing stock (1 mil. m3) …….	2 338	368	255	1 712	3
Natural forests					
Land area (1,000 ha) ……..	13 349	4 770	1 426	7 126	27
Growing stock (1 mil. m3)……..	1 701	642	178	878	3

b

Company name	Place of incorporation and operation	Activity	Percentage held as of December 31, 2006	Percentage held as of December 31, 2005
CJSC "Sherotel"	Moscow region	Hotel	100,00%	100,00%
OJSC "Terminal"	Moscow region	Project Sheremetyevo-3	100,00%	100,00%
CJSC "Aeroflot Plus"	Moscow region	Airline	100,00%	100,00%
OJSC "Insurance company "Moscow"	Moscow	Captive insurance services	100,00%	100,00%
CJSC "Aeromar"	Moscow region	Catering	51,00%	51,00%
OJSC "Aeroflot-Don"	Rostov-on-Don	Airline	100,00%	51,00%
CJSC "Aeroflot-Nord"	Arkhangelsk	Airline	51,00%	51,00%
CJSC "Aeroflot-Cargo"	Moscow	Cargo transportation services	100,00%	-

c

Kind of seed	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002
	Price per 100 pounds									
	Dollars	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Do llars*	*Dollars*	*Dollars*
Alfalfa, uncertified varieties	152.00	161.00	168.00	185.00	185.00	205.00	184.00	165.00	158 .00	280.00
Potatoes	8,6	10,2	7,9	10,3	7, 6	9,1	8,5	10,45	8,5	10,9
Peanuts	77,3	86,9	79,5	82	81,75	83,6	80,9	81,7	82,6	82,1
Sunflower	300	297	297	313	355	380	400	395	407	407
Cottonseed, all	62,7	63,5	68,2	73	74,9	79,3	82,4	128	154	213
Biotech [1]	.....	.....	.....	.....	.....	.....	.....	.....	217	271
Non-biotech	.....	.....	.....	.....	.....	.....	.....	.....	87	94
Grain sorghum, hybrid	74,5	82,1	78,7	84	92	96	97,6	93	93	96
	Price per bushel									
	Dollars	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Dollars*	*Do llars*	*Dollars*	*Dollars*
Corn, hybrid, all [2]	72,7	73,4	77,1	77,7	83,5	86,9	88,1	87,5	92,2	92
Biotech [1]	.....	.....	.....	.....	.....	.....	.....	.....	110	113
Non-biotec	.....	.....	.....	.....	.....	.....	.....	.....	85,3	85,8
Barley (spring)	5	5,18	5,37	6,49	6,13	6, 04	5,8	5,8	5,8	5,8
Soybeans for seed, all	12,4	13,6	13,4	14,8	16,1	17, 15	17	17,1	20,7	22,5
Biotech [1]	.....	.....	.....	.....	.....	.....	.....	.....	23,9	27
Non-biotec	.....	.....	.....	.....	.....	.....	.....	.....	17,9	15
Flaxseed	7,37	7,74	8	8,14	9,31	10	8,5	7,9	7,6	7,6

d

线路名称	Name	客运量（万人）Passenger Traffic (10 000 persons)	旅客周转量（百万人公里）Passenger-kilometers (million passenger-km)	线路名称	Name	货运量（万吨）Freight Traffic (10 000 tons)	货物周转量（百万吨公里）Freight Ton-kilometers (million ton-km)
京沪线	Beijing-Shanghai	5496	32975	京沈线	Beijing-Shenyang	3438	82790
新石线	Xinjiang-Rizhao			哈大线	Harbin-Dalian	3233	60717
沪杭	Shanghai-Hangzhou	654	6188	津沪线	Tianjin-Shanghai	5304	100909
浙赣线	Hangzhou-Ganzhou	3785	33028	沪杭线	Shanghai-Hangzhou	202	4939
鹰厦线	Yingtan-Xiamen	10869	88717	京广线	Beijing-Guangzhou	7187	131196
京九线	Beijing-Kowloon	814	1906	南北同蒲线	Datong-Taiyuan-Feng	11168	30412
京广线	Beijing-Guangzhou	491	1708	太焦柳线	Taiyuan-Jiaozuo-Liuzl	8206	56729
石太线	Shijiazhuang-Taiyuan	1800	9364	京九线	Beijing-Kowloon	2644	61919
石德线	Shijiazhuang-Dezhou	1575	6452	兰新线	Lanzhou-Urumqi	3366	63348
焦柳线	Jiaozuo-Liuzhou	655	2512	滨洲线	Harbin-Manzhouli	3137	21181
京包线	Bingjing-Baotou	541	1288	滨绥线	Harbin-Suifenhe	1178	16384
包兰线	Baotou-Lanzhou	1245	3615	京包线	Bingjing-Baotou	5881	57077

Figure 6: Samples of the test tables with some typical features: the table (a) has a hierarchy of row labels identified with indents in the stub and a hierarchy of column labels identified with column spanning in the head; in the table (b), the content of the three left columns can be interpreted as labels or as data; the body in the table (c) is crossed by the cut-in heads "Price per 100 pounds" and "Price per bushel"; the table (d) includes labels which are duplicated in English and Chinese, it uses colors to divide labels and data; in (d) the columns with labels are alternated by the columns with data.

7. Experimental results

The experimental evaluation of the approach is made with the *CELLS* system using the Drools Expert as the rule engine. The system implements the described table model and pre- and post-processing algorithms and allows the following: 1) to input unstructured tabular information (test data with the special markup) presented in Excel format; 2) to recover semantic relationships in tables using the rule engine as well as pre- and post-processing algorithms; 3) to output results (canonical form of tables) in Excel format.

To get the experimental evaluation we formed the collection of test data that includes 97 tables in Excel format collected from 7 different sources. The collection is available at address http://cells.icc.ru/test. Its brief description is given in Table 1. The test data sources are weakly structured PDF documents (governmental and financial statistical reports with rich tabular content). To generate the collection the original tables were converted from PDF to Excel format. As far as possible, the graphical formatting of the original tables from PDF was presented in the corresponding generated tables in Excel format. Samples of test tables are shown in Fig. 6.

Each test table has an additional markup to locate it in Excel sheet. Its upper left corner is denoted by the tag `$START`, and bottom right corner is marked by the tag `$END`. In addition, a test table has accurately been decomposed into cells, i. e. visual and physical boundaries of its cells coincide when it is possible. It allows to avoid steps for detection and physical layout analysis of the test table.

In the experiment we evaluate the performance of the recovering entries, labels, and internal relationships between labels only. Evaluation of the recovering external relationships between labels and dimensions is not considered in the paper.

In some cases, the interpretation of relationships in a table is not always obvious even to humans. With this in mind Answering questions about relationship between cells two experts can make different conclusions. In exceptional cases,

Table 1: Test data and the experimental results.

Source	Number of tables	cells	entries	labels	relationships of labels[8]	rules	Time of inference (ms)
JAPAN_STAT[1]	15	1088	734	257	102	10	417
AEROFLOT[2]	13	2047	727	321	167	16	526
BOEING[3]	21	2156	964	470	196	14	663
CHINA_STAT[4]	18	7216	4180	862	551	12	964
CHEVRON[5]	7	812	268	141	89	12	283
USDA_NASS[6]	7	1553	1175	313	174	16	638
TOBACCO[7]	16	2844	2195	508	335	10	730

[1] Statistical Handbook of Japan 2007. Statistics Bureau of Japan. Chapter 5, 8.

[2] OJSC "Aeroflot – Russian Airlines" Consolidated Financial Statements For the Year Ended December 31, 2006. pp. 4-10, 25-26.

[3] Boeing Co, Annual Report 2010. PP. 50-55, 83-85.

[4] China statistical yearbook 2003. National Bureau of Statistics of China. pp. 23-48, 555, 559, 571, 584, 590, 664, 708, 774, 765.

[5] Chevron Corp. News Release November 2, 2012. Chevron Corp. pp. 1, 5-9.

[6] USDA NASS. 2003 Agricultural Statistics Annual. USDA (U.S. Department of Agriculture). National Agricultural Statistics Service. Chapter VI. pp. 5-7, 12.

[7] Tobacco: World Markets and Trade 2005. USDA (U.S. Department of Agriculture). Foreign Agricultural Service.

[8] Excluding relationships from roots of label trees.

we could not find single solution how to interpret the test data (e. g. Fig. 6, *b*). Our experimental evaluation is based on assumption that interpretation used in testing is correct. With this in mind, all entries, labels, and internal relationships were recovered (detected) with 100% recall (i. e. all entries, labels, and internal relationships that are encountered in the test tables are also presented in the resulting tables) and 100% precision (i. e. all entries, labels, and internal relationships that are absent in the test tables are also not presented in the resulting tables).

Logical inference was carried out in the Drools Expert (5.4.0.Final) rule engine on the processor Intel Core 2 Quad, 2,66 GHz. The obtained experimental results are shown in Table 1. They demonstrate the performance of the proposed approach for the wide range of tables from statistical and financial reports.

8. Application

In practice, it is required in many cases to transform tabular data from unstructured to structured form. For example, tables presented in unstructured form are often the only available source of statistical or financial information. But only after transforming information from these tables to databases it is available for using in business intelligence, including online analytical processing, data mining, and knowledge discovery.

Perhaps, the main application of our approach is the unstructured tabular data integration. The described principles of table understanding can be used as a basis for software designed for the conversion of tabular data from unstructured sources to databases.

Particularly, the system *CELLS* with insignificant modifications was used in the Mongolian Development Institute within the joint Russian-Mongolian project (2011-12 years) to form the data warehouse with the socio-economic information about the territories of Mongolia. The system allowed extracting data from more than 40 pivot tables in Excel spreadsheet format.

To extract information from these tables we declared three dimensions which

describe years, the administrative division in Mongolia, and sectors of the economy:

- Years — $D_1 = \{$`19\d\d`, `200\d`, `201[0-2]`$\}$,
- Aimags — $D_2 = \{$`Bayan-Olgii`, `Govi-Altai`, $\ldots$, `Ulaanbaatar`$\}$,
- Sectors — $D_3 = \{$`Agriculture`, `Industry`, `Services`$\}$.

Two additional dimensions "Indicators" and "Units" were formed automatically by extracting table names and units from titles presented in a context of the tables. To extract table names and units from the context we supplemented each table with two additional markers `$NAME` and `$UNIT`. The marker `$NAME` indicates that the next right cell contains the name of a table. Analogically, the marker `$UNIT` shows that the next right cell contains the unit for data in the table.

Furthermore, we developed 9 rules with 81 lines of code in the MVEL language. The left hand side of these rules contains conditions used information about the location of cells and their textual content. Several rules are intended to set ignored labels and entries which are the aggregated data. For example, in the following rule labels, which contain words, such as "total", "region", or "average", are marked as ignored in the further process of table understanding.

```
when
    $c : CCell( role == null, cl == 1, rt > 1, cl == cr,
    text matches "(?i).*(total)|.*(region)|.*(average)|(\\s*)" )
then
    modify ( $c ) { setIgnored( true ) }
```

In several rules, the right hand side has additional expressions to convert numerical data into the unified format using regular expressions and methods of the Math class from the Java API, for example, the following rule.

```
when
    $c : CCell( role == null, rt > 1, cl > 1, cl == cr,
```

```
    text matches "[\\d\\s\\.]+", text != null )
then
    modify ( $c ) { setRole( Role.ENTRY ) }
    modify ( $c ) { setText( String.valueOf( Math.round(
                    Double.valueOf( getText.replaceAll(
                    "[^\\d\\.]", "") ) * 100D) / 100D ) ) }
```

The 40 pivot tables contain more than 15000 not aggregated data values. In the result all values were automatically extracted and loaded into the data warehouse.

9. Conclusions

The use of different assumptions on the structure and content of tables is typical for most of the methods for table analysis and interpretation. Those assumptions (e.g. the top row contains only attributes; all data values are numeric; a data value is described by an attribute which is in the same row and the leftmost column) are embedded in these analysis algorithms. However, they limit a class of tables which can be understood with a high precision and recall.

We also use assumptions about table structures, styles and content. But, in contrast to the known methods for table understanding, we divide assumptions into two parts: general and special. The first constant part, the set of general assumptions, is presented in Section 3. They describe a wide class of tables. Our model is based on them. The second variable part is sets of special assumptions about spatial, style and natural language features of tables. They are expressed as table analysis rules. These special assumptions are combined into sets (knowledge bases) which are designed for different subclasses of tables. That approach allows to reach very high or even absolute precision and recall of table understanding for particular subclasses of tables within the class limited by general assumptions.

Moreover, we propose to automate the table understanding using both domain-specific natural language information and domain-independent spatial and style

(typographical) information. Also, we can analyze and interpret tables using only spatial and style information.

Our approach is based on the supposition that a consistent set of table analysis rules can be developed for one or more similar sources of unstructured tabular information. For example, the analysis of the source "Statistical Handbook of Japan in 2007, Chapter 5, 8" was required to develop 10 rules which consist of 93 lines in the MVEL language. It is assumed that this rule set is also suitable for tables from similar sources (other chapters or editions of this statistical handbook).

Separate sets of rules can be developed to analyze different table structures. However, the development of an unified knowledge base for many various sources is too expensive and is not always possible because of conflicts contained in the sources. Therefore, the approach is intended for data integration tasks, especially for the conversion of tabular information from sets of similar unstructured sources to databases. The *CELLS* system for structuring tabular information is based on the proposed approach. The experimental results show the performance of applying the system to the wide range of tables from statistical and financial reports.

A further research is required to simplify rules through the development of data structures for representing tables and additional algorithms for pre- and post-processing of tabular information. Furthermore, occasionally unknown features associated with presentation of tabular information are discovered during the test of the system. Therefore, the data structures and algorithms may need specifying for processing unstructured tabular information from documents of other formats: Word, HTML, or PDF.

10. Acknowledgements

The work was financially supported by the Russian Foundation for Basic Research (grant no 14-07-00166) and the Council for grants of the President of the Russian Federation (grant no SP-3387.2013.5).

References

Chen, H.-H., Tsai, S.-C., & Tsai, J.-H. (2000). Mining tables from large scale html texts. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1* COLING '00 (pp. 166–172). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/990820.990845.

Doan, A., Naughton, J. F., Ramakrishnan, R., Baid, A., Chai, X., Chen, F., Chen, T., Chu, E., DeRose, P., Gao, B., Gokhale, C., Huang, J., Shen, W., & Vuong, B.-Q. (2009). Information extraction challenges in managing unstructured data. *SIGMOD Rec.*, *37*, 14–20. doi:10.1145/1519103.1519106.

Douglas, S., Hurst, M., & Quinn, D. (1995). Using natural language processing for identifying and interpreting tables in plain text. In *Proc. of the 4th Annual Symposium on Document Analysis and Information Retrieval* (pp. 535–546). Las Vegas.

e Silva, A., Jorge, A., & Torgo, L. (2006). Design of an end-to-end method to extract information from tables. *Int. J. on Document Analysis and Recognition*, *8*, 144–171. doi:10.1007/s10032-005-0001-x.

Embley, D., Hurst, M., Lopresti, D., & Nagy, G. (2006a). Table-processing paradigms: a research survey. *Int. J. on Document Analysis and Recognition*, *8*, 66–86. doi:10.1007/s10032-006-0017-x.

Embley, D., Lopresti, D., & Nagy, G. (2006b). Notes on contemporary table recognition. In *Proc. of the 7th Int. Workshop on Document Analysis Systems* (pp. 164–175). Nelson, New Zealand. doi:10.1007/11669487_15.

Embley, D., Tao, C., & Liddle, S. (2005). Automating the extraction of data from html tables with unknown structure. *Data & Knowledge Engineering*, *54*, 3–28. doi:10.1016/j.datak.2004.10.004.

Feldman, R., & Sanger, J. (2006). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press.

Ferrucci, D., & Lally, A. (2004). Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, *10*, 327–348. doi:10.1017/S1351324904003523.

Gatterbauer, W., Bohunsky, P., Herzog, M., Krpl, B., & Pollak, B. (2007). Towards domain-independent information extraction from web tables. In *Proc. of the 16th Int. Conf. on World Wide Web* (pp. 71–80). New York, US. doi:10.1145/1242572.1242583.

Hu, J., Kashi, R. S., Lopresti, D. P., & Wilfong, G. (2000). Table structure recognition and its evaluation. In *Proc. SPIE 4307, Document Recognition and Retrieval VIII* (pp. 44–55). San Jose, CA, USA. doi:10.1117/12.410859.

Hurst, M. (2000). *The Interpretation of Tables in Texts*. Ph.D. thesis University of Edinburgh UK.

Hurst, M. (2001). Layout and language: Challenges for table understanding on the web. In *Proc. of the 1st Int. Workshop on Web Document Analysis* (pp. 27–30).

Inmon, W., & Nesavich, A. (2007). *Tapping into Unstructured Data: Integrating Unstructured Data and Textual Analytics into Business Intelligence*. (1st ed.). Prentice Hall PTR.

Kim, Y.-S., & Lee, K.-H. (2008). Extracting logical structures from html tables. *Computer Standards & Interfaces*, *30*, 296–308. doi:10.1016/j.csi.2007.08.006.

Lopresti, D., & Nagy, G. (2000). A tabular survey of automated table processing. *Lecture Notes in Computer Science*, *1941*, 93–120. doi:10.1007/3-540-40953-X_9.

Pinto, D., McCallum, A., Wei, X., & Croft, W. B. (2003). Table extraction using conditional random fields. In *Proc. of the 26th Annual Int. ACM SIGIR Conf. on Research and Development in Informaion Retrieval* SIGIR '03 (pp. 235–242). New York, NY, USA: ACM. doi:10.1145/860435.860479.

Pivk, A. (2006). Thesis: Automatic ontology generation from web tabular structures. *AI Communications*, *19*, 83–85.

Pivk, A., Cimiano, P., Sure, Y., Gams, M., Rajkovič, V., & Studer, R. (2007). Transforming arbitrary tables into logical form with tartar. *Data & Knowledge Engineering*, *60*, 567–595. doi:10.1016/j.datak.2006.04.002.

Tijerino, Y., Embley, D., Lonsdale, D., Ding, Y., & Nagy, G. (2005). Towards ontology generation from tables. *World Wide Web: Internet and Web Information Systems*, *8*, 261–285. doi:10.1007/s11280-005-0360-8.

Wang, J., Wang, H., Wang, Z., & Zhu, K. Q. (2012). Understanding tables on the web. In *Proc. of the 31st Int. Conf. on Conceptual Modeling* ER'12 (pp. 141–155). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-34002-4_11.

Wang, X. (1996). *Tabular Abstraction, Editing, and Formatting*. Ph.D. thesis University of Waterloo Waterloo, Ontario, Canada.

Yoshida, M., Torisawa, K., & Tsujii, J. (2001). A method to integrate tables of the world wide web. In *Proc. of the Int. Workshop on Web Document Analysis* (pp. 31–34).

Zanibbi, R., Blostein, D., & Cordy, J. (2004). A survey of table recognition: Models, observations, transformations, and inferences. *Int. J. on Document Analysis and Recognition*, *7*, 1–16. doi:10.1007/s10032-004-0120-9.

Zanibbi, R., Blostein, D., & Cordy, J. (2008). Decision-based specification and comparison of table recognition algorithms. In S. Marinai, & H. Fujisawa (Eds.), *Machine Learning in Document Analysis and Recognition. Series: Studies in Computational Intelligence* (pp. 71–103). Springer Berlin Heidelberg volume 90. doi:10.1007/978-3-540-76280-5.