DOI: 10.15514/ISPRAS-2024-36(3)-7

Automated Extraction of Facts from Tabular Data based on Semantic Table Annotation

N.O. Dorodnykh, ORCID: 0000-0001-7794-4462 <*nikidorny@icc.ru*>
A.Yu. Yurin, ORCID: 0000-0001-9089-5730 <*iskander@icc.ru*>

Matrosov Institute for System Dynamics and Control Theory of the Russian Academy of Sciences,
134, Lermontov st., Irkutsk, 664033, Russia.

Abstract. The use of knowledge graphs in the construction of intelligent information and analytical systems provides to effectively structure and analyze knowledge, process large volumes of data, improve the quality of systems, and apply them in various domains such as medicine, manufacturing, trade, and finance. However, domain-specific knowledge graph engineering continues to be a difficult task, requiring the creation of specialized methods and software. One of the main trends in this area is the use of various information sources, in particular tables, which can significantly improve the efficiency of this process. This paper proposes an approach and a tool for automated extraction of specific entities (facts) from tabular data and populating them with a target knowledge graph based on the semantic interpretation (annotation) of tables. The proposed approach is implemented in the form of a special processor included in the Talisman framework. We also present an experimental evaluation of our approach and a demo of domain knowledge graph development for the Talisman framework.

Keywords: knowledge engineering; knowledge graph; knowledge graph population; tabular data; semantic table interpretation; fact extraction.

For citation: Dorodnykh N.O., Yurin A.Yu. Automated Extraction of Facts from Tabular Data based on Semantic Table Annotation. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 3, 2024. pp. 93-104. DOI: 10.15514/ISPRAS-2024-36(3)-7.

Acknowledgements. The research was carried out within the state assignment of Ministry of Science and Higher Education of the Russian Federation (theme No. 1023110300006-9). Dorodnykh N.O. is a scholarship holder of the Council for Grants of the President of Russia (grant No. SP-978.2022.5).

Автоматизированное извлечение фактов из табличных данных на основе семантического аннотирования таблиц

Н.О. Дородных, ORCID: 0000-0001-7794-4462 <*nikidorny@icc.ru*>
А.Ю. Юрин, ORCID: 0000-0001-9089-5730 <*iskander@icc.ru*>

Институт динамики систем и теории управления имени В.М. Матросова РАН,
Россия, 664033, г. Иркутск, ул. Лермонтова, д. 134.

Аннотация. Использование графов знаний при построении интеллектуальных информационно-аналитических систем позволяет эффективно структурировать и анализировать знания, обрабатывать большие объемы данных, повышать качество систем и применять их в различных областях, таких как медицина, производство, торговля и финансы. Однако создание графов знаний для конкретной предметной области по-прежнему остается сложной задачей, требующей создания специализированных методов и программного обеспечения. Одной из основных тенденций в этой области является использование различных источников информации, в частности таблиц, что позволяет существенно повысить эффективность этого процесса. В данной статье предложен подход и программное средство для автоматического извлечения конкретных сущностей (фактов) из табличных данных и пополнения ими целевого графа знаний на основе семантической интерпретации (аннотирования) таблиц. Предложенный подход реализован в виде специализированного обработчика, входящего в состав платформы Talisman. В статье также представлена экспериментальная оценка предлагаемого подхода и демонстрация разработки предметного графа знаний для платформы Talisman.

Ключевые слова: инженерия знаний; граф знаний; пополнение графа знаний; табличные данные; семантическая интерпретация таблиц; извлечение фактов.

Для цитирования: Дородных Н.О., Юрин А.Ю. Автоматизированное извлечение фактов из табличных данных на основе семантического аннотирования таблиц. Труды ИСП РАН, том 36, вып. 3, 2024 г., стр. 93–104 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(3)–7.

Благодарности. Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (тема № 1023110300006-9). Автор Дородных Н.О. является стипендиатом Совета по грантам Президента России (проект СП-978.2022.5).

1. Introduction

Currently, the development of intelligent information and analytical systems aimed at solving complex practical problems remains a relevant area of scientific research. Such systems are actively used in the fields of corporate information retrieval (e.g., Microsoft SharePoint, Oracle Secure Enterprise Search, Elasticsearch), knowledge bases and text analysis (e.g., Palantir Gotham, IQPlatform, Semantic Archive (ANBR), Aiteko “X-files 2.0”), media and social network monitoring (e.g., LexisNexis, Medialogy, Kribrum, BrandAnalytics, Scan Interfax), competitive intelligence (e.g., Maltego, Hensoldt Analytics, Vitok-OSINT), forecasting, and data analytics (e.g., SAS Analytics, IBM Watson Studio, PolyAnalyst Megaputer). Knowledge graphs can be used to build such systems. They are designed to accumulate and transfer knowledge about the real world, the nodes of which represent objects of interest, and the edges represent relationships between these objects [1]. Knowledge graphs represent complex knowledge in a structured form, which facilitates the analysis and use. They can be scaled to any size, allowing them to process large volumes of data. In general, the use of knowledge graphs in the construction of intelligent systems makes it possible to effectively structure knowledge and identify hidden associations and dependencies between various concepts, which can be useful for decision-making or forecasting [2]. However, knowledge graph engineering is a rather time-consuming task and may require processing huge amounts of data obtained from various sources (e.g., databases, documents, web resources) [3]. Thus, our research aimed at automating knowledge graph construction and population them with new facts when solving practical and weakly formalized problems in various domains is relevant [4].

Over the past decade, the scientific community has proposed a wide range of different methods and software aimed at knowledge graph engineering. The main trend is the use of various information sources. In particular, such a source is tables that present information in the form of a set of rows and columns. In general, each row of a table represents a record, and each column represents an attribute or field. Tables allow one to store and process data efficiently because they provide a structured way to store information. They are widely used in various fields (e.g., finance, statistics, and programming) and are the basis for many applications and systems that work with data. According to some research [5], millions of useful facts can be extracted from tables contained both on the web and as part of different e-documents. All these factors make tables a valuable source of knowledge when constructing knowledge graphs for intelligent systems. However, tables are very heterogeneous in their structure and are not accompanied by explicit semantics necessary for automatic interpretation of their content as intended by their author. This fact prevents the active and practical use of tabular data.

In this paper, we propose an approach for automated extraction of facts from tables and filling in a target knowledge graph with them. The main feature of the proposed approach is the ability to automatically restore the semantics of tabular data based on a set of intuitive heuristics. This approach is implemented in the form of a special processor included in the Talisman (Tracking and Learning Insights from Social Media Analysis) framework [6]. The development of the processor and its subsequent testing were carried out as part of a research project with the Ivannikov Institute for System Programming of the Russian Academy of Sciences (ISP RAS).

The paper is organized as follows: Section 2 presents the current state of research in the field of semantic interpretation of tables. Section 3 describes the problem statement, the proposed approach, and the features of its software implementation. Section 4 provides a demo example with the experimental evaluation, while the Conclusions provide discussion of the results and concluding remarks.

2. Related works

Automation of knowledge graph engineering is an actual topic in the field of artificial intelligence and big data processing. One of the main trends in this direction is the use of automatic knowledge extraction from various information sources, in particular, presented in the form of tables. At the same time, automatic creation of domain knowledge graphs and their population with new facts is not possible without automatic understanding of the structure and content of tabular data. The restoration of this semantics is carried out by methods in such a scientific direction as Semantic Table Interpretation (STI) [6, 7]. The first works in this area [8, 9] appeared in 2010 and were aimed at comparing individual table elements with concepts from a knowledge graph, ontology, or other external dictionary (e.g., DBpedia, Wikidata, Yago, Freebase, WordNet). Traditionally, the semantic interpretation (annotation) of tables includes four main tasks [7]:

- *Cell Entity Annotation (CEA)* is the mapping cell values to entities (class instances) of a target knowledge graph;
- *Column Type Annotation (CTA)* is the mapping individual table columns to the semantic types (classes) of a target knowledge graph;
- *Column Property Annotation (CPA)* is the mapping relationships between columns with properties of a target knowledge graph;
- *Table Annotation (TA)* is the comparison of the entire table with some class of a target knowledge graph (table topic detection).

The following two main stages can be defined in STI research:

Stage 1 (*initial: 2010–2019*). At this stage, a formulation of the problem of semantic interpretation of tables was carried out, and the main goals and objectives were identified. The stage is also characterized by the gradual publication of works aimed mainly at analyzing the natural language content and context of tables using methods of ontology matching, entity lookup (both in global cross-domain knowledge graphs and in domain-specific ontologies), Wikification and knowledge graph embeddings [10-14]. Iterative approaches based on the use of probabilistic graphical models [15, 16] and machine learning methods [15, 17, 18] can also be highlighted.

Stage 2 (*modern: 2019–present*). This stage is characterized by the rapid growth of high-quality works, extensively studied projects, and decent results for separate STI tasks. The first commercial solutions that expand the functionality of data preparation and analysis systems, such as Microsoft Power BI, Trifacta and Google Looker Studio, in terms of the semantic type detection of table columns, are appearing. At this stage, approaches based on deep machine learning (e.g., JHSTabEL [19], Sato [20]) and especially using pre-trained language models (e.g., TURL [21], TaBERT [22], TABBIE [23], TUTA [24], Doduo [25]) have gained great popularity. Since 2019, the SemTab competition (Semantic Web Challenge on Tabular Data to Knowledge Graph Matching) [27] aimed at comparing tabular data with knowledge graphs takes place every year as part of the International Semantic Web Conference (ISWC). The main metrics and criteria for evaluating table annotation systems were formulated as part of the competition. In addition, many datasets (e.g., WebTables, WikiTables, GitTables, SOTAB) to test the performance of such systems have been released.

In recent years, significant progress has been made in research on automatic understanding of tabular information. However, there is a gap between the effectiveness of existing solutions in tests and their applicability in practice. First of all, this is due to the lack of high-quality labeled training data, coupled with the difficulty of customizing existing models and approaches for specific domains. It should also be noted that there is no stage associated with extracting new facts from semantically annotated tabular data and filling in a target knowledge graph with them in most approaches and tools. This determines the relevance of the development of new methods and software aimed at a comprehensive solution to problems of semantic interpretation of tables and fact extraction within specific domains.

3. The proposed approach

3.1 The problem statement

Vertical tables are considered input data for the proposed approach. A vertical table is an array of data arranged in the form of vertical columns. A column may contain a header. In such tables, each column can be divided into two types:

- *a named entity (categorical) column* contains entity mentions of some domain (e.g., persons, organizations, events);
- *a literal column* contains some values of simple datatypes (e.g., date, time, cardinal number).

Assumption 1. There are no merged cells in the tables being processed.

Assumption 2. The source tables are processed independently of each other.

A knowledge base of the Talisman framework is used as a target knowledge graph. Let’s formalize the description of this knowledge base:

$$KB = \{DM, F\},$$

where *KB* is a knowledge base of the Talisman framework; *DM* is a domain model that defines an ontological scheme with an abstract description of concepts and their relationships; *F* is a set of specific entities (facts) that are typified based on the domain model. At the same time:

$$DM = \{CT, PT, PVT, BVT, RT\},$$

where *CT* is a concept type (e.g., person, organization, product); *PT* is a property type (e.g., residential address, work phone, birthdate); *PVT* is a property value type (e.g., address, date, distance); *BVT* is a basic value type (e.g., geolocation (coordinates), date, date interval, string, string indicating language, number); *RT* is a relation type defined between two concept types (e.g., “*works in*”, “*studies in*”, “*is a*”).

$$F = \{C, P, AV, R, M\},$$

where *C* is a concept (e.g., a specific person, a specific organization, or product); *P* is a concept property that is of interest to end users. In this case, a property can be identifying (e.g., “*name*” that uniquely characterizes a specific object); *AV* is a specific atomic value of a property (e.g., a person’s age or mobile phone number); *R* is a relation between two concepts; and *M* is a mention, which is a text fragment that directly points to an object/event/concept of the real or virtual world, corresponding to some concept, property, or relation.

An example of using the Talisman knowledge graph is shown in Fig. 1.

The proposed approach implements semantic annotation of columns and relationships between them, which consists of matching certain property types to columns, finding the most suitable concept type based on them, as well as identifying relation types between certain concept types. Next, let’s take a closer look at the main stages of this approach.

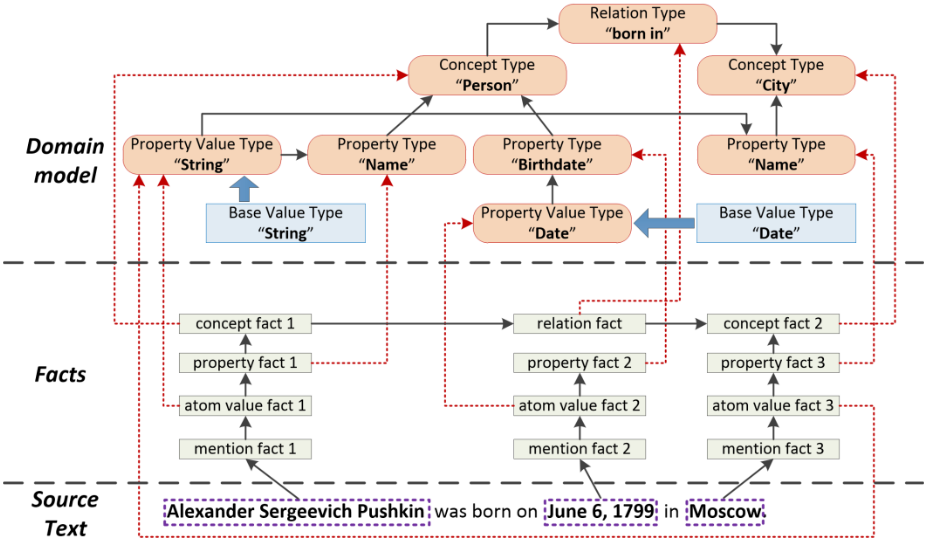


Fig. 1. An example of using the Talisman knowledge graph

3.2 The main stages

The proposed approach builds on our previous work, and aims to process Talisman documents presented in the Talisman Document Model (TDM) format, version 1.0. This model may contain a set of vertical tables, each of which needs to be processed. A Talisman document is an entity of the Talisman framework, containing data collected by the system in a unified form. The source of documents is files of various formats (e.g., PDF, DOCX, CSV, HTML) downloaded from file storages or web pages, including files posted on pages from the Web. In this case, a universal system for extracting content and logical structure from textual documents, namely, Dedoc [28] is used. A document is structured textual content and/or image along with extracted facts.

The main stages of the proposed approach are presented in Fig. 2.

Stage 1: Table Preprocessing. At this stage, Named Entity Recognition (NER) is performed for each cell in a source table. For this purpose, the pre-trained XLM-RoBERTa model [29] is used, which recognizes occurrences of some named entities (persons, companies, locations, etc.) in the text. This model was fine-tuned on the following datasets: CoNLL 2003 (English), OntoNotes (English), OntoNotes (Chineese), and DocRED (English). The corresponding NER labels of named entities are assigned to each cell in a source table, thus characterizing the data that it contains.

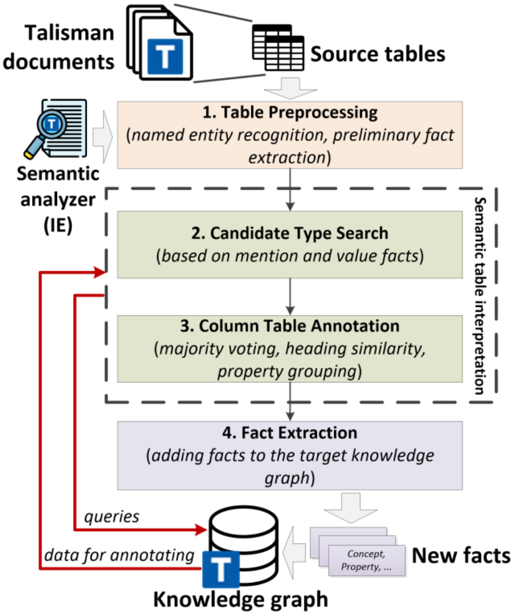

Talisman documents
Source tables
1. Table Preprocessing
(named entity recognition, preliminary fact extraction)
Semantic analyzer (IE)
2. Candidate Type Search
(based on mention and value facts)
3. Column Table Annotation
(majority voting, heading similarity, property grouping)
Semantic table interpretation
4. Fact Extraction
(adding facts to the target knowledge graph)
queries
data for annotating
Concept, Property, ...
New facts
Knowledge graph

Fig. 2. The main stages of the proposed approach

Depending on the assigned NER label, the relevant mention facts and atom value facts are automatically extracted from cells, which correspond to a specific property value type defined in a domain model. In addition, at this stage, preliminary property facts and concept facts can also be extracted from cells. This stage is performed using the semantic analyzer (IE), which is part of the Talisman framework.

Stage 2: Candidate Type Search. A set of candidate property types for each column obtained from a domain model is generated based on defined mention facts and atom value facts. It should be noted that columns for which facts were not extracted in the previous stage will be excluded from subsequent table processing.

Stage 3: Column Type Annotation. At this stage, the most suitable property type is selected from a set of candidates to assign to its column. For this purpose, a special aggregated method is used consisting of a combination of the following heuristics:

1) *Majority Voting.* This heuristic is a fairly simple baseline solution, which consists of the fact that the most suitable property type from a set of candidates is assigned to a column based on direct inference from those property facts (these facts have already been extracted from column cells using the semantic analyzer). Thus, there is a set of property types for each specific property fact to which it corresponds. Next, the number (frequency of occurrence) of each candidate type is calculated. The value of this frequency is a natural number, including zero. Therefore, the normalization method defined in [30] our previous work is used to represent this value in a score range from 0 to 1 (each type from a set of candidates will be determined by such a score);

2) *Heading Similarity.* A lexical matching of a column header with names (labels) of property types from a set of candidates is carried out based on the Levenshtein distance. Depending on this distance, a score is given to each candidate type. Moreover, if concept facts were previously identified in a column (at the table preprocessing stage), then a column header is compared not with names (labels) of property types from a set of candidates but with names (labels) of concept types that are associated with these candidate property types. The normalization method is also used to represent the obtained score in the range from 0 to 1;

3) *Property Grouping.* This heuristic is based on the assumption that a source table may have

one or more categorical columns in which the semantic analyzer has already extracted some concept facts with identifying property facts (e.g., "*name*" property for some organization concept). Next to such categorical columns, there are usually columns with their properties. In this case, columns with properties can be located anywhere in a source table and don't depend on the location of categorical columns. The number of possible properties for each categorical column that are located in other non-categorical (literal) columns and relate to this concept is calculated. Next, it is determined which categorical column corresponds to the maximum number of properties. For such a column and columns with properties, a score equal to one is set.

An aggregated score is determined based on all three heuristics and defines the overall probability that a certain property type from a set of candidates is the most suitable (relevant) for annotating a table column.

Stage 4: Fact Extraction. New concept facts, mention facts, atom value facts, and concept property facts are extracted from a source table using the established column annotations. In this case, the extracted mention facts include the value of the entire cell as a whole. The facts are extracted row by row, from left to right. Property facts are created only for the leftmost categorical column in a source table; if a table defines the same property type as an annotation for several categorical columns (e.g., if a table contains two columns holding persons and all other columns are defined as some properties for persons, then only for concept facts from the first column the corresponding properties will be created). In this case, identifying properties (e.g., names) will always be extracted. All possible relation facts between the extracted concept facts are also extracted row by row from a source table. All the facts extracted in this way enrich the target Talisman knowledge graph.

3.3 Implementation

The proposed approach is implemented in the form of a special processor called "*tables-annotator*". This processor is written in Python 3.10 and is part of the Talisman Information Extraction (Talisman-IE) subsystem. The processor is the REST server that transforms the input Talisman document. The processor also receives as input a configuration for document processing, which is a JSON object that specifies rules and/or restrictions for transforming input documents.

The configuration for the "*tables-annotator*" processor:

```
{
   "table_indices": "<table sequence numbers>",
   "column_indices": {
      "<table sequence numbers>": "<column sequence numbers>",
      ...
   },
   "header_numbers": [ <row number 1>, ..., <row number n> ]
}
```

Configuration parameters:

- "*table_indices*" is an optional parameter that specifies indexes for tables found in a source document that must be excluded from processing. For this purpose, both individual table indexes and range indexes can be indicated separated by commas, for example: "*1, 2, 3, 5-8, 10*". The special value "*end*" can be used in this range. In this case, tables will be counted automatically until the end of a document, for example: "*1, 3, 5-end*";
- "*column_indices*" is an optional parameter that specifies indexes for columns that need to be excluded from processing in the specified tables. For this purpose, a dictionary is specified, where a key is table indexes or range indexes, and a value is column indexes or range indexes related to the specified tables. These table and column indexes are text

values and are compiled according to the same principle as the “*table_indices*” parameter;

- “*header_numbers*” is an optional parameter that specifies a list of row indexes that is a table header. By default, the first row of a table is considered a header. Row indexes must be numeric values.

Thus, if it is necessary to process all tables from the Talisman document and extract facts from them, then the default configuration is not specified.

4. The usage example

The developed “*tables-annotator*” processor was used as part of a research project conducted for ISP RAS. The problem of an automated population of domain knowledge graphs in the Talisman framework with new facts extracted from tabular data was solved. The Talisman framework is a set of tools for the automation of typical tasks such as data processing (e.g., collection, integration, analysis, storage, and visualization). This framework provides rapid development of specialized multi-user analytical systems that combine information from internal databases and open web-sources.

Our processor was tested by analyzing test tables collected by categories: “*organization employees*”, “*open vacancies*”, “*car market*”, “*famous scientists*”, “*book sales*”. The following web resources were used to generate a test set of tabular data:

- web sites of scientific and educational institutions (e.g., the Matrosov Institute of System Dynamics and Control Theory named SB RAS, Irkutsk National Research Technical University);
- job bank of the Irkutsk region and web service of hh (Irkutsk);
- avito web service;
- russian-language part of Wikipedia;
- labyrinth web store.

Tabular data was collected manually from web tables and stored in the form of DOCX documents. The average number of columns in the collected tables is 5, and the average number of rows is 12.

A fragment of the domain model for a target knowledge graph of the Talisman framework was used in the process of semantic annotation of tables and at the stage of population with new facts extracted from tables. This fragment is presented in Fig. 3. A target knowledge graph is presented as a semantically labeled property graph, which is accessed using the GraphQL interface.

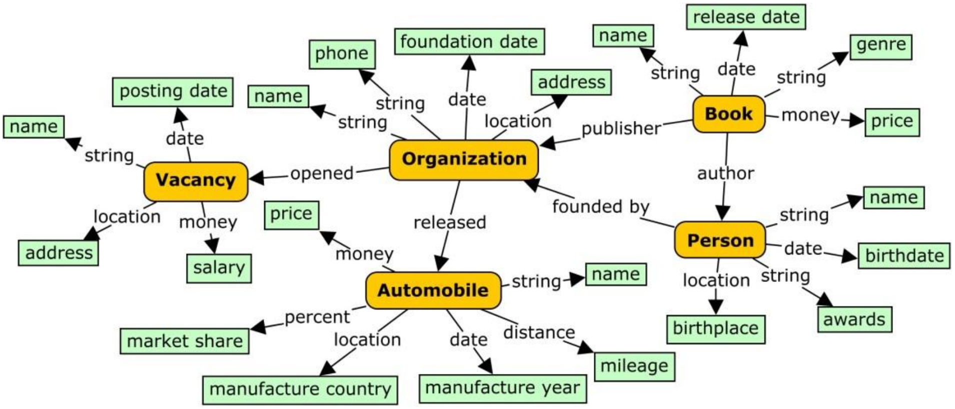


Fig. 3. A fragment of a domain model

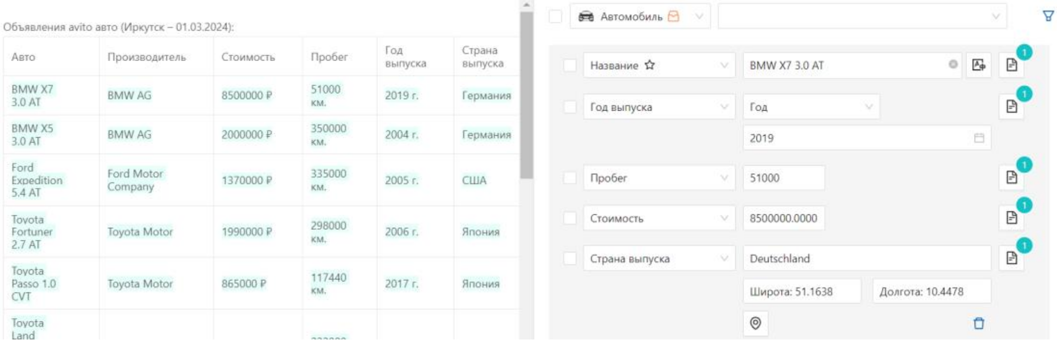
The domain model describes the main concepts such as “*Person*” (NER labels: PERSON, PER), “*Organization*” (NER labels: ORGANIZATION, ORG), “*Vacancy*” (there is no corresponding NER labels), “*Car*” (NER labels: PRODUCT) and “*Book*” (NER labels: WORK_OF_ART).

Fig. 4 shows an illustrative example of a processed source table from the “car market” category on the Talisman framework with a description of advertisements for the sale of used cars in the city of Irkutsk, as well as specific column annotations and extracted facts.

Well-known measures such as precision, recall, and F1 score were used to perform an experimental evaluation for the stage of the automated semantic annotation of columns by using the “*tables-annotator*” processor:

$$precision = \frac{P}{C}, \; recall = \frac{P}{CN}, \; F1 = \frac{2 \times percision \times recall}{percision + recall},$$

where *P* is a number of correctly annotated columns (perfect annotations); *C* is a number of annotated columns; and *CN* is the total number of columns in a source table.



Объявления avito авто (Иркутск – 01.03.2024):

Авто	Производитель	Стоимость	Пробег	Год выпуска	Страна выпуска
BMW X7 3.0 AT	BMW AG	8500000 ₽	51000 км.	2019 г.	Германия
BMW X5 3.0 AT	BMW AG	2000000 ₽	350000 км.	2004 г.	Германия
Ford Expedition 5.4 AT	Ford Motor Company	1370000 ₽	335000 км.	2005 г.	США
Toyota Fortuner 2.7 AT	Toyota Motor	1990000 ₽	298000 км.	2006 г.	Япония
Toyota Passo 1.0 CVT	Toyota Motor	865000 ₽	117440 км.	2017 г.	Япония
Toyota Land					

Автомобиль

Название ☆	BMW X7 3.0 AT
Год выпуска	Год 2019
Пробег	51000
Стоимость	8500000.0000
Страна выпуска	Deutschland Широта: 51.1638 Долгота: 10.4478

Fig. 4. A fragment of the processed source table from the “car market” category on the Talisman framework

The resulting accuracy score for test tables from various categories is presented in Table 1.

Table 1. The experimental evaluation for test tables

Table category	**Precision**	**Recall**	**F1**
Organization employees	1,00	0,80	0,89
Open vacancies	0,20	0,16	0,18
Car market	1,00	0,83	0,91
Famous scientists	0,75	0,75	0,75
Book sales	0,80	0,67	0,73

Our processor showed acceptable evaluation results. However, the current version of the processor is based entirely on named entity recognition results received at stage 1 of our approach. This fact does not allow us to involve columns for which NER labels have not been defined in the table processing (e.g., for a column with the name of an open vacancy for tables from the “*vacancies*” category). This is the main problem that influenced the current evaluation results (e.g., quite low scores for tables from the “*open vacancies*” category).

Other limitations of our approach are the following:

- only vertical tables are processed;
- values (mentions) are extracted entirely from cells (e.g., “*First Name*,” “*Last Name*,” and “*Patronymic Name*” are not extracted separately from cell with “*Full Name*”);
- a single value is not formed from the values of several cells;
- complex composite values for concept properties are not considered;
- relation properties are not extracted.

A comprehensive comparison of individual elements of our approach (e.g., majority voting and heading similarity methods) with some similar solutions is presented in [31]. In general, the results obtained show the promise of using the developed approach and processor to support the process of domain knowledge graph population based on semantically annotated tabular data.

5. Conclusions

In this paper, we present an approach to the STI process and extract specific entities (facts) from semantically annotated tabular data. The proposed approach includes a combination of heuristic solutions to automatically annotate table columns and relationships between columns. The approach uses Talisman documents as input data and a knowledge base of the Talisman framework as a target knowledge graph. The approach is implemented in the form of a special "*tables-annotator*" processor, which is part of the Talisman-IE subsystem.

The evaluation results for the developed software are considered in a research project conducted with ISP RAS. The results obtained have shown the applicability of our approach to practical tasks. In the future, we plan to improve the evaluation results by combining the proposed heuristic solutions with methods based on deep machine learning. In particular, we plan to use pre-trained language models to predict the relevant semantic type for table columns.

References

[1]. Hogan A., Blomqvist E., Cochez M., d'Amato C., De Melo G., Gutierrez C., Gayo J. E. L., Kirrane S., Neumaier S., Polleres A., Navigli R., Ngomo A.-C. N., Rashid S. M., Rula A., Schmelzeisen L., Sequeda J., Staab S., Zimmermann A. Knowledge Graphs, 2021.

[2]. Ji S., Pan S., Cambria E., Marttinen P., Yu P. S. A Survey on Knowledge Graphs: Representation, Acquisition and Applications. IEEE Transcations on Neural Networks and Learning Systems, vol. 33, no. 2, 2021, pp. 494-514. DOI: 10.1109/TNNLS.2021.3070843.

[3]. Martinez-Rodriguez J. L., Hogan A., Lopez-Arevalo I. Information Extraction meets the Semantic Web: A Survey. Semantic Web, vol. 11, 2020, pp. 255-335. DOI: 10.3233/SW-180333.

[4]. Villazon-Terrazas B., Garcia-Santa N., Ren Y., Srinivas K., Rodriguez-Muro M., Alexopoulos P., Pan J. Z. Construction of Enterprise Knowledge Graphs (I). Exploiting Linked Data and Knowledge Graphs in Large Organisations, Springer, Cham, 2017.

[5]. Lehmberg O., Ritze D., Meusel R., Bizer C. A large public corpus of web tables containing time and context metadata. Proc. 25th International Conference Companion on World Wide Web, 2016, pp. 75-76. DOI: 10.1145/2872518.2889386.

[6]. Talisman framework, Available at: http://talisman.ispras.ru, accessed 06.05.2024.

[7]. Bonfitto S., Casiraghi E., Mesiti M. Table understanding approaches for extracting knowledge from heterogeneous tables. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 11, no. 4, 2021, e1407. DOI: 10.1002/widm.1407.

[8]. Liu J., Chabot Y., Troncy R. From tabular data to knowledge graphs: A survey of semantic table interpretation tasks and methods. Journal of Web Semantics, vol. 76, 2023, 100761. DOI: 10.1016/j.websem.2022.100761.

[9]. Limaye G., Sarawagi S., Chakrabarti S. Annotating and Searching Web Tables Using Entities, Types and Relationships. Proc. VLDB Endowment, vol. 3, 2010, pp. 1338-1347. DOI: 10.14778/1920841.1921005.

[10]. Mulwad V., Finin T., Syed Z., Joshi A. Using linked data to interpret tables. Proc. the First International Conference on Consuming Linked Data (COLD'10), vol. 665, 2010, pp. 109-120.

[11]. Bhagavatula C. S., Noraset T., Downey D. TabEL: Entity Linking in Web Tables. Proc. the 14th International Semantic Web Conference (ISWC'2015), 2015, pp. 425-441. DOI: 10.1007/978-3-319-25007-6_25.

[12]. Efthymiou V., Hassanzadeh O., Rodriguez-Muro M., Christophides V. Matching web tables with knowledge base entities: From entity lookups to entity embeddings. Proc. 16th International Semantic Web Conference (ISWC'2017), 2017, pp. 260-277. DOI: 10.1007/978-3-319-68288-4_16.

[13]. Ritze D., Bizer C. Matching web tables to DBpedia - A feature utility study. Proc. 20th International Conference on Extending Database Technology (EDBT'17), 2017, pp. 210-221. DOI: 10.5441/002/EDBT.2017.20.

[14]. Zhang Z. Effective and efficient semantic table interpretation using TableMiner+. Semantic Web, vol. 8, no. 6, 2017, pp. 921-957. DOI: 10.3233/SW-160242.

[15]. De Vos M., Wielemaker J., Rijgersberg H., Schreiber G., Wielinga B., Top J. Combining information on structure and content to automatically annotate natural science spreadsheets. International Journal of Human-Computer Studies, vol. 103, 2017, pp. 63-76. DOI: 10.1016/j.ijhcs.2017.02.006.

[16]. Takeoka K., Oyamada M., Nakadai S., Okadome T. Meimei: An Efficient Probabilistic Approach for Semantically Annotating Tables. Proc. the AAAI Conference on Artificial Intelligence, vol. 33, no. 01. 2019, pp. 281-288. DOI: 10.1609/aaai.v33i01.3301281.

[17]. Kruit B., Boncz P., Urbani J. Extracting Novel Facts from Tables for Knowledge Graph Completion. Proc. the 18th International Semantic Web Conference (ISWC’2019), 2019, pp. 364-381. DOI: 10.1007/978-3-030-30793-6_21.

[18]. Chen J., Jimenez-Ruiz E., Horrocks I., Sutton C. ColNet: Embedding the Semantics of Web Tables for Column Type Prediction. Proc. the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, 2019, pp. 29-36. DOI: 10.1609/aaai.v33i01.330129.

[19]. Hulsebos M., Hu K., Bakker M., Zgraggen E., Satyanarayan A., Kraska T., Demiralp Ç., Hidalgo C. Sherlock: A Deep Learning Approach to Semantic Data Type Detection. Proc. the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD’19), 2019, pp. 1500-1508. DOI: 10.1145/3292500.3330993.

[20]. Xie J., Lu Y., Cao C., Li Z., Guan Y., Liu Y. Joint Entity Linking for Web Tables with Hybrid Semantic Matching. Proc. the International Conference on Computational Science, 2020, pp. 618-631. DOI: 10.1007/978-3-030-50417-5_46.

[21]. Zhang D., Suhara Y., Li J., Hulsebos M., Demiralp C., Tan W.-C. Sato: Contextual semantic type detection in tables. Proc. the VLDB Endowment, vol. 13, no. 11, 2020, pp. 1835-1848. DOI: 10.14778/3407790.3407793.

[22]. Deng X., Sun H., Lees A., Wu Y., Yu C. TURL: Table Understanding through Representation Learning. Proc. the VLDB Endowment, vol. 14, no. 3, 2020, pp. 307-319. DOI: 10.14778/3430915.3430921.

[23]. Yin P., Neubig G., Yih W. TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data. Proc. the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8413-8426. DOI: 10.18653/v1/2020.acl-main.745.

[24]. Iida H., Thai D., Manjunatha V., Iyyer M. TABBIE: Pretrained Representations of Tabular Data. Proc. the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 3446-3456. DOI: 10.18653/v1/2021.naacl-main.270.

[25]. Wang Z., Dong H., Jia R., Li J., Fu Z., Han S., Zhang D. TUTA: Tree-based Transformers for Generally Structured Table Pre-training. Proc. the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD’21), 2021, pp. 1780-1790. DOI: 10.1145/3447548.3467434.

[26]. Suhara Y., Li J., Li Y. Annotating Columns with Pre-trained Language Models. Proc. the 2022 International Conference on Management of Data (SIGMOD’22), 2022, pp. 1493-1503. DOI: 10.1145/3514221.3517906.

[27]. SemTab challenge, Available at: http://www.cs.ox.ac.uk/isg/challenges/sem-tab/, accessed 06.05.2024.

[28]. Belyaeva O., Bogatenkova A., Turdakov D. Dedoc: A Universal System for Extracting Content and Logical Structure From Textual Documents. 2023 Ivannikov Ispras Open Conference (ISPRAS), IEEE, 2023, pp. 20-25.

[29]. Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L., Stoyanov V. Unsupervised Cross-lingual Representation Learning at Scale. Proc. the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8440-8451. DOI: 10.18653/v1/2020.acl-main.747.

[30]. Dorodnykh N. O., Yurin A. Yu. Extraction of Facts from Web-Tables based on Semantic Interpretation Tabular Data. In Proc. the 2022 Ivannikov Memorial Workshop (IVMEM’2022), 2022, pp. 7-17. DOI: 10.1109/IVMEM57067.2022.9983959.

[31]. Dorodnykh N. O., Yurin A. Yu. Knowledge Graph Engineering Based on Semantic Annotation of Tables. Computation, vol. 11, no. 9, 2023, 175. DOI: 10.3390/computation11090175.

Информация об авторах / Information about authors

Никита Олегович ДОРОДНЫХ – кандидат технических наук, старший научный сотрудник Института динамики систем и теории управления им. В.М. Матросова Сибирского отделения РАН (ИДСТУ СО РАН) с 2021 года. Сфера научных интересов: автоматизация создания интеллектуальных систем и баз знаний, получение знаний на основе преобразования концептуальных моделей и электронных таблиц.

Nikita Olegovych DORODNYKH – Cand. Sci (Tech.), senior associate researcher at Matrosov Institute of System Dynamics and Control Theory named SB RAS (ISDCT SB RAS) since 2021. Research interests: computer-aided development of intelligent systems and knowledge bases, knowledge acquisition based on the transformation of conceptual models and tables.

Александр Юрьевич ЮРИН – доктор технических наук, заведующий лабораторией Информационно-телекоммуникационных технологий исследования природной и техногенной безопасности ИДСТУ СО РАН, доцент Института информационных технологий и анализа данных Иркутского научно-исследовательского технического университета (ИрНИТУ). Его научные интересы включают разработку систем поддержки принятия решений, экспертных систем и баз знаний, использование прецедентного подхода и семантических технологий при проектировании интеллектуальных диагностических систем.

Alexander Yurievich YURIN – Dr. Sci. (Tech.), Head of a laboratory “Information and telecommunication technologies for investigation of natural and technogenic safety” at ISDCT SB RAS and associate professor of the Institute of information technologies and data analysis of Irkutsk National Research Technical University (INRTU). His research interests include development of decision support systems, expert systems and knowledge bases, application of the case-based reasoning and semantic technologies in the design of diagnostic intelligent systems, maintenance of reliability and safety of complex technical systems.