IRSTI 27.35.33 DOI: https://doi.org/10.26577/JMMCS.2022.v114.i2.08

A.B. Nugumanova[1], K.S. Apayev[2], Y.M. Baiburin[1], M. Mansurova[3*], A. Ospan[3]

[1]Sarsen Amanzholov East Kazakhstan University, Kazakhstan, Ust-Kamenogorsk
[2]D. Serikbayev East Kazakhstan Technical University, Kazakhstan, Ust-Kamenogorsk
[3]al-Farabi Kazakh National University, Kazakhstan, Almaty
*e-mail: madina.mansurova@kaznu.kz

QURMA: A TABLE EXTRACTION PIPELINE FOR KNOWLEDGE BASE POPULATION

This paper is proposed a pipeline aimed at automatically extracting tables from heterogeneous Web sources, such as HTML pages, pdf files and images. Table extraction is one of the actively developing areas of Information Extraction, for which many applications, libraries and frameworks are currently being developed. Nevertheless, most of these tools are focused on solving some specific tasks, for example, only on recognizing tables presented in the form of images. We propose to combine these tasks into a single pipeline that will support the full cycle of table processing – from the stages of their search, recognition and extraction to the stages of semantic analysis and interpretation, that is, the understanding of tables. Understanding tables and population of knowledge bases (knowledge graphs) with meaningful information contained in these tables is the ultimate goal of our design. The first part of the work presents methods for detecting tables on web pages, in pdf documents, as well as methods for automatically detecting attributes and values of objects. The second part presents the conceptual architecture of the Qurma system, designed to extract tables from heterogeneous sources on the Internet. The results section provides an example of a parser that parses the input resource type and passes control to one of the table lookup modules. Next, an operation is performed to determine the main column and link the entities contained in the main column with the corresponding categories in the external knowledge base.

Key words: web tables, table extraction, table recognition, table understanding, knowledge base population.

А.Б. Нугуманова[1], К.С. Апаев[2], Е.М. Байбурин[1], М. Мансурова[3*], Ә. Оспан[3]
[1]Сәрсен Аманжолов атындағытШығыс Қазақстан университеті., Қазақстан, Өскемен қ.
[2]Д. Серікбаева атындағы Шығыс Қазақстан техникалық университеті., Қазақстан, Өскемен қ.
[3]Әл-Фараби атындағы Казақ Ұлттық Университеті, Қазақстан, Алматы қ.
*e-mail: madina.mansurova@kaznu.kz

Qurma: білім базасын толтыруға арналған кестені шығарып алу құбыры

Бұл мақалада HTML парақтары, PDF файлдары және суреттер сияқты, веб парақтардың гетерогенді деректер көздерінен кестелерді автоматты түрде шығарып алу құбыры ұсынылады. Кестелерді шығару – қазіргі уақытта көптеген қосымшалар, кітапханалар мен жақтаулар құрастырылып жатқан ақпарат алудың белсенді дамып келе жатқан бағыттарының бірі. Алайда, бұл құралдардың көпшілігі кейбір нақты мәселелерді шешуге бағытталған, мысалы, тек суреттер түрінде ұсынылған кестелерді тануға бағытталған. Текстік кестелерді танитын, оқитын, оларды топтарға жіктейтін бағдарламалар кең танылмаған, және дайын кітапханалар неемесе құралдар жоқ. Біз бұл тапсырмаларды кестелерді өңдеудің толық циклын қолдайтын біртұтас құбырға біріктіруді ұсынамыз – оларды іздеу, тану және шығарып алу кезеңдерінен бастап, семантикалық талдау мен түсіндіру кезеңдерімен аяқтайды, яғни кестелерді түсінеді. Кестелерді түсіну және білім базасын (білім бағандары) осы кестелердегі маңызды ақпаратпен толтыру біздің жобамыздың түпкі мақсаты болып табылады. Жұмыстың бірінші бөлімінде веб-беттерде, pdf құжаттарында кестелерді анықтау, сонымен қатар атрибуттар мен объектілердің мәнін автоматты түрде анықтау әдістері берілген.

© 2022 Al-Farabi Kazakh National University

Екінші бөлімде ғаламтордың гетерогенді көздерінен кестелерді шығарып алуға арналған, Qurma жүйесінің концептуалды архитектурасы көрсетілген. Нәтижелер бөлімінде кіріс ресурстардың типін талдайтын және басқаруды кесте іздеудің бір модуліне тапсыратын парсердің жұмыс жасау мысалы келтірілген. Ары қарай, басты бағанды анықтау мен осы басты бағанда орналасқан мәндерді сыртқы білім базасындағы сәйкес категорияларымен байланыстыру операциясы орындалады.
Түйін сөздер: веб-кестелер, кестелер шығару, кестелерді тану, кестелерді түсіну, білім базасын толтыру.

А.Б. Нугуманова[1], К.С. Апаев[2], Е.М. Байбурин[1], М. Мансурова[3*], А. Оспан[3]
[1]Восточно-Казахстанский университет им. Сарсена Аманжолова, Казахстан, г. Усть-Каменогорск
[2]Восточно-Казахстанский технический университет им. Д. Серикбаева, Казахстан, г. Усть-Каменогорск
[3]Казахский Национальный Университет имени аль-Фараби, Казахстан, г. Алматы
*e-mail: madina.mansurova@kaznu.kz

Qurma: конвейер извлечения таблиц для пополнения баз знаний

В данной работе предлагается конвейер, направленный на автоматическое извлечение таблиц из гетерогенных источников Веба, так как HTML-страницы, pdf-файлы и изображения. Извлечение таблиц – одно из активно развивающихся направлений извлечения информации, для которого в настоящее время разрабатывается множество приложений, библиотек и фреймворков. Тем не менее, большинство этих инструментов ориентировано на решение каких-то конкретных задач, например, только на распознавание таблиц, представленных в виде изображений. Мы предлагаем объединить эти задачи в единый конвейер, который будет поддерживать полный цикл обработки таблиц – начиная с этапов их поиска, распознавания и извлечения и заканчивая этапами семантического анализа и интерпретации, то есть пониманием таблиц. Понимание таблиц и пополнение баз знаний (графов знаний) значимой информацией, содержащейся в этих таблицах, является конечной целью нашего проектирования. В первой части работы представлены методы обнаружения таблиц на веб-страницах, в pdf документах, также методы автоматического выявления атрибутов и значений объектов. Во второй части представлена концептуальная архитектура системы Qurma, предназначенной для извлечения таблиц из гетерогенных источников в сети Интернет. В разделе результатов представлен пример работы парсера, который анализирует тип входного ресурса и передает управление одному из модулей поиска таблиц. Далее выполняется операция по определению главного столбца и связыванию сущностей, содержащихся в главном столбце, с соответствующими категориями во внешней базе знаний.

Ключевые слова: веб-таблицы, извлечение таблиц, распознавание таблиц, понимание таблиц, пополнение базы знаний.

1 Introduction

Most of the significant and useful data available on the Internet is published in the form of tables. A person can easily identify, interpret and link the contents of these tables with the information available to him, while the methods of automatic analysis of web tables hardly cope with their task due to the wide variety of table presentation formats. In order to extract useful data from web tables, it is necessary to first correctly determine the boundaries and types of cells containing this data, and then match the identified cells to the corresponding headers. Thus, the process of automatic analysis of web tables is divided into 2 stages: 1) extracting tables, which implies defining the boundaries and structure of the cells of each table; 2) understanding tables, which implies linking the contents of cells with semantic information both inside and outside the tables. As a rule, the understanding of tables in

automatic streaming mode is used for the purpose of forming and filling the knowledge base population in any subject area.

Extracting tables involves two subtasks: 1) detecting a table on a web page or in a document; 2) directly extracting information from the detected table [1]. The subtask of detecting a table on a web page or in a document only looks trivial at first glance. Firstly, it is connected with the problem of classification, since tables are not only meaningful, but also layout tables. Mock-up tables do not contain meaningful information, but are used on a web page or in a document for formatting, for example, to align text or drawings. Secondly, some tables are not highlighted on the page or in the document with TABLE tags, i.e. other signs have to be used to search for them. Thirdly, long tables can be located on different pages of a website or document or hidden using special drop-down elements in order to save space, respectively, connecting individual fragments of the table into a single structure requires additional parsing operations. After detecting and verifying the table, it is necessary to correctly extract data from it for transmission to the next stage - the stage of understanding the table. The correct extraction of information involves such operations as the definition of headers (attribute names), the separation of combined data (when two different attributes are recorded in one cell, for example, address and phone number, or the cell contains list data, for example, several phones for one contact), etc.

In turn, understanding the table for the purposes of knowledge base population includes solving the following subtasks: 1) linking the contents of tables obtained from the Internet with the knowledge base; 2) building hypotheses about the structure and content of tables; 3) extracting new information from tables; 4) adding this new information to the knowledge base [2]. At the same time, a class of tables in which entities are described is of particular interest to the knowledge base population, i.e. tables in which one column, called the main or key, contains the name of the entity, and all their other attributes [3]. Such tables are easier to extract and interpret, so a large number of processing methods have been developed for them, unlike more complex tables expressing n-dimensional relationships, i.e. relationships between several entities.

The authors [3] call this four-step method of extracting data from tables in order to fill the knowledge base with the interpretation of tables. Interpretation concerns the rows and columns of the table, and allows you to determine which entities from the knowledge base are mentioned in the table, what are the types of these entities and what relationships are expressed in the columns. After the interpretation is completed, this information can be used to fill the knowledge base slots. In this paper, we are implementing a pipeline that includes a full cycle of table extraction plus the first stage of understanding tables, namely the identification of entities in the knowledge base.

2 Related works

As noted above, many applications and tools have been developed to extract tables from heterogeneous sources: from the contents of web pages, from PDF documents, from files representing images. In this section we will consider the following applications and tools: Tabula [4], TableSeer [5], TAO [6], TaKCO [7], TableLab [8], TableNet [9], TabbyPDF [10] and Camelot [11].

Among these applications, the oldest application is the TableSeer table search engine [5].

TableSeer scans scientific documents from electronic libraries, finds those that contain tables, then extracts information from each table, saving it to a table metadata file, indexes tables according to metadata, and provides the user with an interface for searching tables. The TableSeer architecture consists of five main components: 1) table crawler; 2) table metadata extractor; 3) table metadata indexer; 4) the TableRank algorithm for ranking tables according to the search query; 5) the interface for supporting search queries to tables. The extraction of tables is based on a statistical analysis of the templates for the design of articles used in the proceedings of the conference or in the journal, based on these templates, a set of heuristic rules is formed that compare different blocks of the document with various logical components (titles, list of authors, abstract, list of references), and physical components (tables, figures, etc.). The TableRank table ranking algorithm deserves special mention, which adapts the traditional model of the vector space <query, document> to the <query, table> pair, replacing the document vectors with table vectors. To determine the weight of each term in the vector space, the authors propose a new weighing scheme: The tabular frequency of the term is the inverted tabular frequency (TTF-ITTF).

Another Table Organization (WTO) table extraction system [6] generates an extended representation of data also extracted from tables in PDF documents. This representation includes the page number on which the table was found, the table number, and metadata for each cell: cell contents, column number, coordinates, font, size, data type, title, or data label. TAO transmits this data as an annotated document in JSON format. Directly to detect tables in a document and extract information about tables, TAO uses the k-nearest neighbors method and heuristic layout rules.

Another application for automatic detection and extraction of tables from PDF files Tabula [4] can both automatically detect tables and allows users to manually select them. The application uses two different algorithms to extract data from selected tables: the first algorithm (Lattice) is based on searching for control rows in the table and identifies cells in the table as separate if they are separated by a line, the second algorithm (Stream) processes text as separate cells if the text fragments are far from each other. The extracted data is output in several formats, including CSV format. Architecturally, Tabula consists of two separate modules: Tabula-Java and Tabula-Ruby. Tabula-Ruby is responsible for the graphical user interface for Tabula-Java, a module that, as the name suggests, is written in Java and is the server part of the application. Although it is intended to be used as a library for Tabula-Ruby, it can also be run separately as a command-line application.

[7] presents a new large-scale TAKCO platform designed to extract facts from tables that can be added to knowledge graphs (KG), such as, for example, WikiData. Takco works with both tables describing entities and tables describing n-dimensional relationships. For entity tables, the system first identifies a pool of candidate entities from the knowledge graph, then calculates an a priori probability distribution by comparing the attributes of the candidate object in the knowledge graph with other cell values in the same row, and then re-weights these matches by the significance of the relationships in the table. Then the system connects entities by constructing a probabilistic graphical model and collectively eliminating the ambiguity of all cells using Loopy Belief Propagation [11]. To interpret n-dimensional tables, the system applies several heuristic rules to transform the table into a "normal"form. Then the schema is compared and functional dependencies are detected to calculate the first elementary interpretation of the table. Finally, similar tables are grouped using schema and

mapping components to improve the quality of interpretation.

PyTabby [10] is another tool for extracting text from PDF tables with a text layer. The system uses a set of customizable special heuristics to detect tables and reconstruct the structure of cells based on the features of the text and lines presented in PDF documents. Most of them, such as horizontal and vertical distances, fonts and rulers, are well known and used in existing methods. Additionally, the system allows you to use the ability to display instructions for printing text in PDF files.

TableNet [8] is a system for extracting information from scanned tables via mobile phones or cameras. The system is based on deep learning models and allows you to accurately detect tabular areas in an image, and then extract information from the rows and columns of the detected table. The TableNet architecture includes neural networks working together to: 1) generate feature maps from low-level text rectangles (in fact, column names); 2) determine the border of the table, if it is in the image; 4) identify rows and identify columns and related canonical data (description, quantity, unit price, etc.).

The TableLab system proposed in [9] provides user interaction with data extraction models, which allows you to quickly train models on several labeled examples. Having received a collection of documents as input, TableLab first detects tables with a similar structure (templates) by clustering embeddings from the extraction model. Document collections often contain tables created using a limited set of templates or similar structures. The system then selects several representative examples of tables already extracted using a pre-trained basic deep learning model. Through an easy-to-use user interface, users provide feedback on these options without necessarily identifying every bug. TableLab then applies such feedback to fine-tune the pre-trained model and returns the fine-tuning results back to the user. The user can choose to repeat this process iteratively until a customized model with satisfactory performance is obtained.

The Camelot library [11] was created to offer users full control over table extraction. Despite the fact that there are both open source systems (for example, Tabula) and closed source systems (for example, PDFTables) that are widely used to extract tables from PDF files, they all have their strengths and weaknesses that do not allow us to talk about their versatility. The Camelot library contrasts versatility with flexibility of configuration due to which it achieves high accuracy and completeness of information extraction. Like Tabula, Camelot uses two methods of syntactic parsing when extracting tables: 1) Stream, which looks for spaces between words to identify the table; 2) Lattice, which looks for lines on the page to identify the table. Another interesting feature of Camelot is that it has a web interface called Excalibur for users who do not want to develop the code themselves, but at the same time want to use the library functions to extract data from tables.

Initially, work with the interpretation of web tables was presented in the work Annotating and Searching Web Tables Using Entities, Types and Relationships. Limaye et al. [12]. In this work, a system is developed that uses a probabilistic graphical model that makes controlled predictions based on a large number of attributes. Subsequent work approached the task-specific knowledge graph problem [13,14] and accelerated predictions by limiting the feature set [15] or using distributed processing [16].

In work [17] presents a semantic analysis for extracting attribute value pairs from product specifications on the Internet. Here are used HTML tables and HTML lists inside web page as product specification. Supervised learning is used to extract attribute-value pairs from

the HTML fragments identified by the specification detector columns as attribute column or value column.

Other successful feature extraction models based on named entity recognition have been developed in [18, 19, 20]. All approaches use similar models to extract attributes. In [18], an approach to annotating product descriptions based on NLP text fragmentation was proposed. Specifically, the authors train a linear chain conditional random field model on a hand-annotated training dataset to identify only eight generic term classes. However, this approach does not allow explicit attribute-value pairs to be extracted. Ristosky and Mika [20] corrected this shortcoming by applying a method with a full set of discrete features derived from the standard distribution of the NER3 mode. Ortona et al. [19] propose a triple approach that performs the following functions: checking the values of sentences, blocking to reduce the number of compared sentences, and evaluation of paired sentences. For verification, an annotator is used that performs NER extraction (places, locations, names, organizations) and an ontology that contains some domain-specific constraints. At the blocking stage, all pairs of products that violate some ontology constraints are grouped into different clusters.

3 Materials and methods

In this paper, we propose our own solution – a system called Qurma, which is based on the Camelot library. The Qurma system receives an input document from the user with its URL and then searches for the tables contained in this document. HTML pages, pdf documents, images can be used as a document. At the output, the system outputs a flat dataset, which is the result of extracting information from tables found in the document. Next, this data set can be exported in any way convenient for the user: to a CSV file, to a json format, or to an attribute-value format. The conceptual architecture of the Qurma system is based on the Clean Architecture concept [21]. The essence of the concept boils down to the fact that it is necessary to clearly understand the needs and limit the software interfaces in order not to lose control of the entire system. To do this, the system is divided into layers, and the interaction between layers is regulated by the boundary rule, which states that only data can be transferred between layers (see Figure 1). Layers are not equal, the main thing in the system is not the platform or technology used, but the layer containing the business logic or business model. Accordingly, two more rules are generated. The inner layer priority rule states that it is the inner layer that determines the interface through which it will interact with the user or the rest of the system. The dependency rule specifies that dependencies should be directed from the inner layer to the outer one.

As follows from the diagram, the core of the architecture are entity models, in this case, these are TableModel and User Model. TableModel is a model in which all data types from different table parsing packages are serialized, which allows you to have a single standard table object and process only this object. Socket Service provides the transfer of commands from the server to the client, while Table Service provides the processing of tables, searching for the main column, determining the types of input cells, etc.

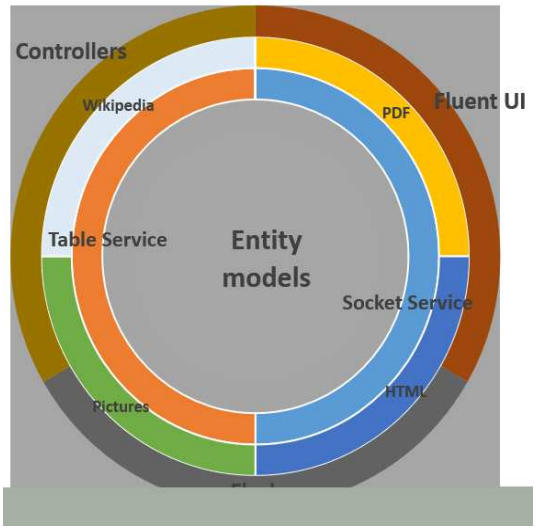


Figure 1: Software "clean" architecture of the Qurma system

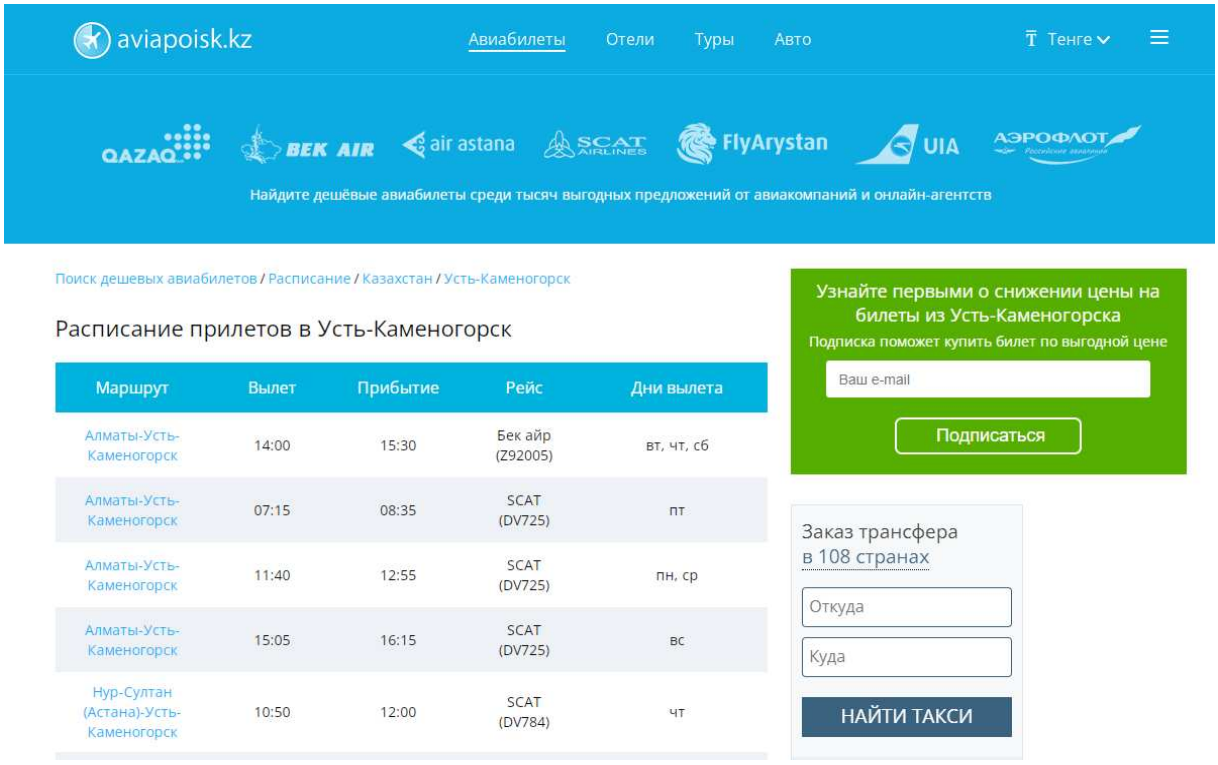
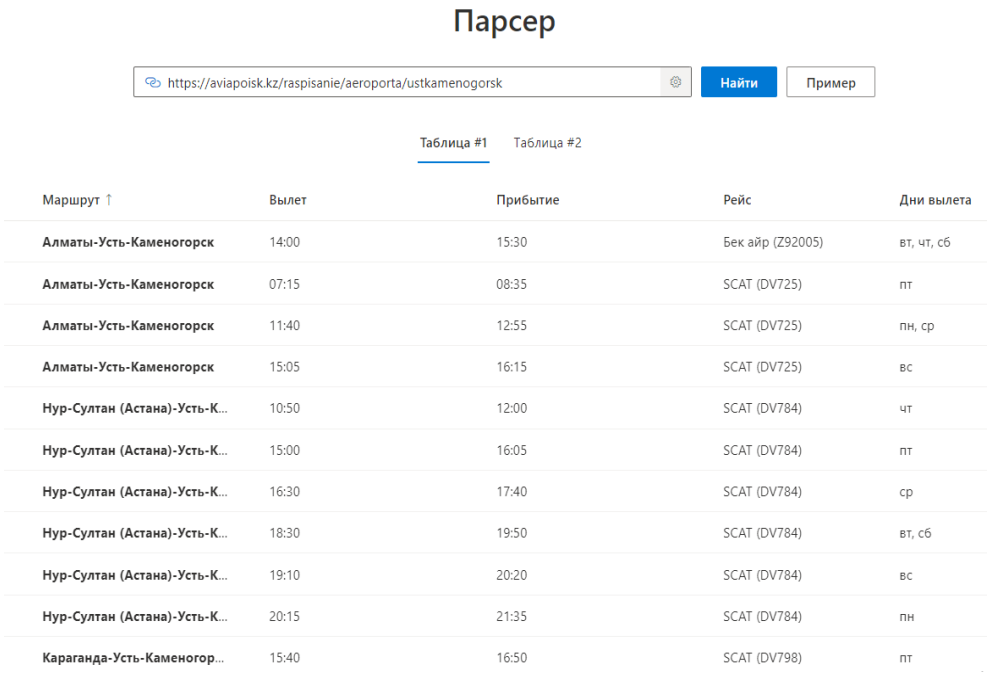


Figure 2: Web page of website aviapoisk.kz with tabular data

4 Results and discussions

The system interface is implemented using a set of Fluent Design elements from Microsoft. The user specifies the URL of the document from which the tables are extracted, for example,



Маршрут ↑	Вылет	Прибытие	Рейс	Дни вылета
Алматы-Усть-Каменогорск	14:00	15:30	Бек айр (Z92005)	вт, чт, сб
Алматы-Усть-Каменогорск	07:15	08:35	SCAT (DV725)	пт
Алматы-Усть-Каменогорск	11:40	12:55	SCAT (DV725)	пн, ср
Алматы-Усть-Каменогорск	15:05	16:15	SCAT (DV725)	вс
Нур-Султан (Астана)-Усть-К...	10:50	12:00	SCAT (DV784)	чт
Нур-Султан (Астана)-Усть-К...	15:00	16:05	SCAT (DV784)	пт
Нур-Султан (Астана)-Усть-К...	16:30	17:40	SCAT (DV784)	ср
Нур-Султан (Астана)-Усть-К...	18:30	19:50	SCAT (DV784)	вт, сб
Нур-Султан (Астана)-Усть-К...	19:10	20:20	SCAT (DV784)	вс
Нур-Султан (Астана)-Усть-К...	20:15	21:35	SCAT (DV784)	пн
Караганда-Усть-Каменогор...	15:40	16:50	SCAT (DV798)	пт

Figure 3: The result of tabular parsing in the source document, presented as a web page

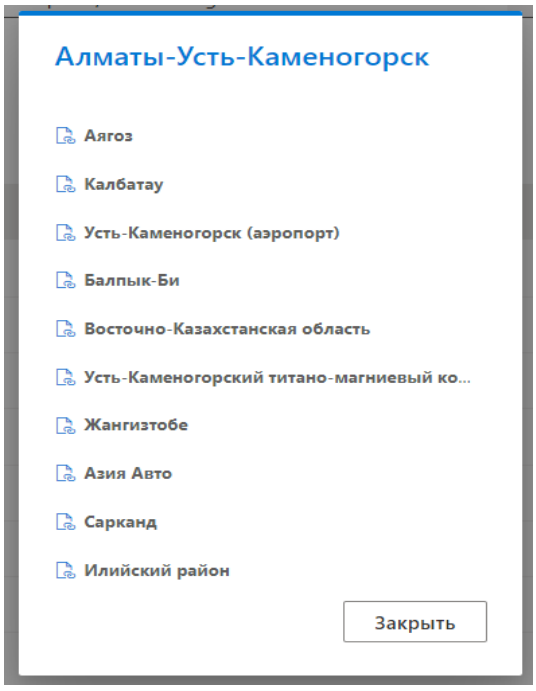


Figure 4: Comparison of the entity in the main column with the categories in the external knowledge base (Wikipedia)

a regular web page address can be used as the URL, as shown in Figure 2 [22]. The parser analyzes the type of input resource and transfers control to one of the three modules in which the table search is already implemented. In this case, control is transferred to the HTML parser, which finds two tables, passes them to the table parser, and the final result is returned as a data set, as shown in Figure 3. In addition to parsing, an operation is performed to determine the main column and associate the entities contained in the main column with the corresponding categories in the external knowledge base (see Figure 4).

5 Conclusion

In this paper, we presented our Qurma system, designed to extract tables from heterogeneous sources on the Internet. The system is a pipeline for searching, extracting and interpreting tables, the ultimate goal of which will be to replenish the knowledge graph on the subject area of interest to the user. Despite the fact that the subject area has not yet been selected and the basic principles of knowledge graph design have not been defined, the presented pipeline already allows solving the problems of semantic analysis of tables contained in web resources. The conceptual architecture of the proposed pipeline, based on the Clean Architecture metaphor, provides a hassle-free increase in the capacity of the designed system. Our future work involves fine-tuning the understanding of web tables using deep learning models. This will allow us to scale the proposed solution using comparatively small sets of training data. Accordingly, further work will be aimed at connecting data annotation modules and precision learning modules to the pipeline.

6 Acknowledgments

This work was carried out and sponsored within the framework of the scientific project AP09261344 "Development of methods for automatic extraction of spatial objects from heterogeneous sources for information support of geographic information systems".

References

[1] Embley D.W., Tao C., Liddle S.W., "Automating the extraction of data from HTML tables with unknown structure", *Data & Knowledge Engineering,* 54 (1) (2005): 3–28.

[2] Ell B., Hakimov S., Braukmann P., et al., "Towards a Large Corpus of Richly Annotated Web Tables for Knowledge Base Population", *15th International workshop on Linked Data for Information Extraction (LD4IE) at ISWC2017,* Vienna.

[3] Kruit B., Boncz P., Urbani J., "Extracting novel facts from tables for knowledge graph completion", *International Semantic Web conference. Springer. Cham.,* (2019): 364–381.

[4] Rosén G., *Analysis of Tabula: A PDF-Table extraction tool.* (2019).

[5] Liu Y., *TableSeer: automatic table extraction, search, and understanding.* (2009).

[6] Perez-Arriaga M.O., Estrada T., Abad-Mota S., "TAO: system for table detection and extraction from PDF documents", *The Twenty-Ninth International Flairs Conference,* (2016).

[7] Kruit B., Boncz P., Urbani J., "TAKCO: A Platform for Extracting Novel Facts from Tables", *Companion Proceedings of the Web Conference,* (2021): 705–707.

[8] Paliwal S.S. et al., "TableNet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images" , *International Conference on Document Analysis and Recognition (ICDAR). IEEE,* (2019): 128–133.

[9] Wang N.X.R., Burdick D., Li Y., "TableLab: An Interactive Table Extraction System with Adaptive Deep Learning" , *26th International Conference on Intelligent User Interfaces,* (2021): 87–89.

[10] Mikhailov A.A., Shigarov A., Kozlov I.S., *PyTabby: a Docreader's module for extracting text and tables from PDF with a text layer.* (2021).

[11] Camelot: PDF Table Extraction for Humans. Camelot 0.8.2 documentation (Jan. 29, 2021). URL: https://camelot-py.readthedocs.io/en/master/ (visited on 10.10.2021).

[12] Limaye G., Sarawagi S., Chakrabarti S., "Annotating and Searching Web Tables Using Entities, Types and Relationships" , *PVLDB,* 3 (1-2) (2010): 1338–1347.

[13] Venetis P., Halevy A., Madhavan J., Paca M., Shen W., Wu F., Miao G., Wu C., "Recovering Semantics of Tables on the Web" , *PVLDB,* 4 (2011): 528–538.

[14] Wang J., Shao B., Wang H., "Understanding Tables on the Web" , *In: ER,* 1 (2010): 141–155

[15] Mulwad V., Finin T., Joshi A., "Semantic Message Passing for Generating Linked Data from Tables" , *In: Proceedings of ISWC,* (2013): 363–378.

[16] Hassanzadeh O., Ward M.J., Rodriguez-Muro M., Srinivas K., "Understanding a Large Corpus of Web Tables Through Matching with Knowledge Bases: an Empirical Study" , *In: Proceedings of OM at ISWC,* (2015): 25–34.

[17] Petar Petrovski, Christian Bizer, "Extracting Atribute-Value Pairs from Product Specifications on theWeb" , *Web Intelligence (WI'17). Leipzig, Germany. ACM,* (2017) 978-1-4503-4951-2/17/08. DOI: 10.1145/3106426.3106449.

[18] Gabor Melli, "Shallow Semantic Parsing of Product Offering Titles (for better automatic hyperlink insertion)" , *In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM,* (2014): 1670–1678.

[19] Stefano Ortona, "An analysis of duplicate on web extracted objects" , *In Proceedings of the companion publication of the 23rd international conference on World wide web companion,* (2014): 1279–1284.

[20] Petar Ristoski and Peter Mika, "Enriching Product Ads with Metadata from HTML Annotations" , *In Proceedings of the 13th Extended SemanticWeb Conference (To Appear),* (2016).

[21] Ihler A.T. et al., "Loopy belief propagation: convergence and effects of message errors" , *Journal of Machine Learning Research,* 6 (5) (2005).

[22] https://aviapoisk.kz/raspisanie/aeroporta/ustkamenogorsk.