

Creating a Scholarly Knowledge Graph from Survey Article Tables

Allard Oelen^{1,2}[0000-0001-9924-9153], Markus Stocker²[0000-0001-5492-3212], and
Sören Auer^{2,1}[0000-0002-0698-2864]

¹ L3S Research Center, Leibniz University of Hannover, Germany
oelen@l3s.de

² TIB Leibniz Information Centre for Science and Technology, Germany
{markus.stocker,auer}@tib.eu

Abstract. Due to the lack of structure, scholarly knowledge remains hardly accessible for machines. Scholarly knowledge graphs have been proposed as a solution. Creating such a knowledge graph requires manual effort and domain experts, and is therefore time-consuming and cumbersome. In this work, we present a human-in-the-loop methodology used to build a scholarly knowledge graph leveraging literature survey articles. Survey articles often contain manually curated and high-quality tabular information that summarizes findings published in the scientific literature. Consequently, survey articles are an excellent resource for generating a scholarly knowledge graph. The presented methodology consists of five steps, in which tables and references are extracted from PDF articles, tables are formatted and finally ingested into the knowledge graph. To evaluate the methodology, 92 survey articles, containing 160 survey tables, have been imported in the graph. In total, 2626 papers have been added to the knowledge graph using the presented methodology. The results demonstrate the feasibility of our approach, but also indicate that manual effort is required and thus underscore the important role of human experts.

Keywords: Scholarly Communication · Scholarly Knowledge Graphs · Tabular Data Extraction

1 Introduction

Scholarly communication is mainly document-based and the communicated scholarly knowledge therefore hardly machine-actionable [21]. Scholarly knowledge graphs have the potential to solve these issues by making knowledge structured and thus more machine processable. Existing initiatives for scholarly information systems, e.g., the Microsoft Academic Graph [8] or Crossref [15] mainly focus on bibliographic metadata and not on the actual research contributions. The Open Research Knowledge Graph (ORKG) [10] aims to build a knowledge graph infrastructure that publishes the research contributions of scholarly publications rather than only the metadata. The approach is to crowdsource structured paper descriptions by including paper authors and domain experts. ORKG

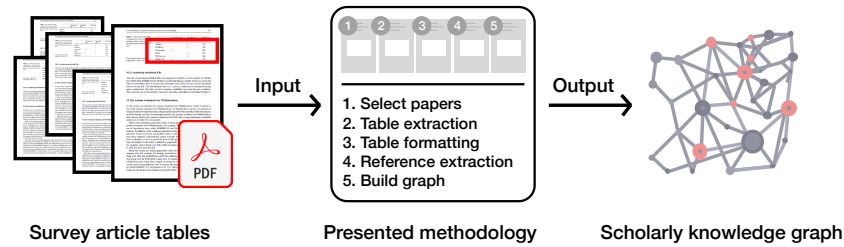


Fig. 1: Systematic workflow in which survey articles are used to build a scholarly knowledge graph. The input of our methodology is survey articles in PDF format and the output is a scholarly knowledge graph.

primarily relies on synergistically combining crowdsourcing and automated extraction rather than, as other systems such as Semantic Scholar³, exclusively on automated techniques to extract knowledge from scholarly articles. Mainly because automated extraction methods, for example Natural Language Processing (NLP), do not have sufficient accuracy to generate the high-quality knowledge graph needed to obtain suitable state-of-the-art overviews for researchers.

In this work, we present a human-in-the-loop methodology to create a scholarly knowledge graph by extracting knowledge from survey tables. We leverage survey tables from literature review papers, specifically. Tables in survey papers generally consist of high-quality research data that has been manually curated by domain experts. Conducting a literature review is a labour-intensive task and writing a review article is often more time-consuming than writing a research article [32]. Compared to natural text, tables present information in a semi-structured manner, making the creation of a structured graph from such data less complicated. Additionally, survey tables present relevant information which is why the survey was conducted and published in the first place. We present a supervised approach to firstly extract data from survey articles and afterwards build a knowledge graph from this data. Compared to sole crowdsourcing, the approach of extracting knowledge is more efficient because the review has already been conducted by the authors of the survey paper. Taking into account the previously mentioned considerations, our work addresses the following research question: *How to efficiently populate a scholarly knowledge graph with high-quality knowledge?* We propose a methodology for extracting tabular survey data. This methodology is used to create a scholarly knowledge graph from survey articles. An overview of the systematic workflow is depicted in Figure 1.

The rest of this paper is structured as follows. Section 2 discusses the related work. Section 3 introduces the proposed five-step methodology for building the knowledge graph from survey articles. Section 4 presents the results. Section 5 discusses the present and future work. Finally, Section 6 concludes the presented work.

³ <https://www.semanticscholar.org>

2 Related Work

Survey articles provide well-structured overviews of the literature [33]. The terms “literature review” and “literature survey” are sometimes used interchangeably in the literature, but we make the following distinction. We refer to the tables within review articles as *literature surveys*. Together with a (textual) analysis and explanation, they form the *literature review*. Among other things, literature reviews are helpful in delimiting the research problem, avoiding fruitless approaches [5] and to discover new research directions [6]. Conducting a literature review is a complicated and time-consuming activity [33]. When literature reviews are not available for certain fields, its development could be weakened [32]. Because of the importance of literature surveys to scientific research, leveraging surveys to build a graph results in a high-quality and relevant scholarly knowledge graph. Some existing work with respect to semantifying literature surveys exists [4,29,23]. However, those approaches are not (semi-)automated and are therefore not scaling well to larger amounts of survey articles.

One aspect of the proposed methodology is table extraction from survey articles. Portable Document Format (PDF) is the most common format for scientific articles [12]. Extracting tables from PDF documents is a cumbersome process since the tabular structure is not stored within the file itself [11]. This means that regular PDF extraction tools are only able to extract the text within a table, but losing the tabular structure. Tools that specifically focus on table extraction from PDF files use segmentation techniques to estimate the position of rows and columns [7]. Corrêa et al. did a literature survey on table extraction tools [3]. They concluded that Tabula⁴ is the most suitable open-source tool. Based of these findings, we decided to use Tabula. Tabula is criticized because of the lack of documentation [26], but for our use case this is not considered problematic. Another aspect of the proposed methodology is reference extraction from PDFs. Since every individual article referenced within a survey table is imported, meta-data from this article should be collected. This is done by parsing the references that are used within a table. For this, we use the state-of-the-art PDF extraction tool GROBID [13]. GROBID focuses specifically on extracting bibliographic data from scholarly articles [19]. Lipinski et al. [17] compared GROBID to other PDF metadata extraction tools, and found out that GROBID performed best.

Publishing data as structured or semantic data is a well researched topic among various domains. For example, challenges related to publishing semantic open government data are similar to the challenges in our research. This includes extracting data from legacy documents, often in PDF format [2,3]. Furthermore, in the literature use cases are described on publishing unstructured data as semantic data (e.g., [9,20,27]). These existing approaches differ from our approach since they generally aim to semantify a homogeneous set of documents. This enables them to create data specific ontologies. In our case, this is not feasible since we work with a highly heterogeneous set of survey tables coming from different domains and comparing different aspects of papers. Table 1 provides

⁴ <https://tabula.technology>

Table 1: Related work compared to the method presented in our study. The full comparison is available via the ORKG.⁵

Study	Name	Method automation	Scope	Input format	Output format	KG ^a creation	Reference extraction	User interface
This study	ORKG	Semi-Automatic	Survey tables	PDF	JSON, RDF ^b	✓	✓	✓
[28]	SemAnn	Semi-Automatic	Scholarly articles	PDF	RDF ^b	✗	✗	✓
[16]	Web Tables	Automatic	Web tables	HTML	JSON	✗	✗	✗
[18]	TableSeer	Automatic	Scholarly articles	PDF	Relational database	✗	✗	✓
[25]	TEXUS	Automatic	Documents (application agnostic)	PDF	Abstract table representation	✗	✗	✗
[1]	<i>None</i>	Automatic	Web tables	HTML, Spreadsheet	Relational schema	✗	✗	✗

^a Knowledge Graph; ^b Resource Description Framework

a related work overview. In this overview, our proposed method is compared to other related approaches. To the best of our knowledge, this work is the first to build a knowledge graph at scale from survey tables.

Use Case: Open Research Knowledge Graph. Extracted survey data can be imported in a variety of different (scholarly) knowledge graphs, such as the Microsoft Academic Graph, Wikidata [14] or ORKG. We chose ORKG as our use case for the following reasons. The ORKG provides tools that specifically focus on building paper comparisons (i.e., literature surveys), making it the most suitable infrastructure for this study. By using the extracted survey data, the ORKG automatically generates a similar tabular survey view as was originally presented in the review paper [22]. Additionally, the literature surveys within ORKG are compliant [23] with the FAIR data principles [34] thus making them Findable, Accessible, Interoperable and Reusable. The imported survey tables are FAIR in contrast to the originally presented ones in the non-FAIR PDF article. This has several benefits, among others:

- Comparisons can evolve over time, are not static and do not become stale after publication.
- Comparisons do represent a broader community consensus, since many researchers and curators can revise, discuss and annotate.

⁵ <https://www.orkg.org/orkg/comparison/R36099>

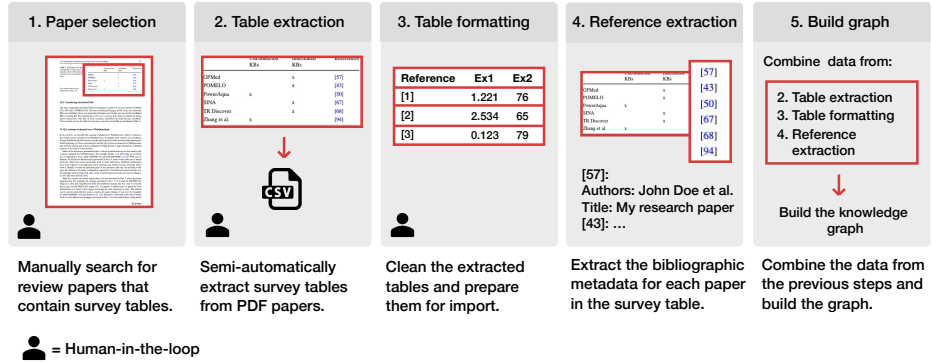


Fig. 2: Methodology for importing survey tables into the scholarly knowledge graph.

- Via the ORKG search interface it is possible to search for specific comparisons and to create dynamic custom comparison views.
- Survey data can be reused by other researchers more easily because of its machine readable export formats (e.g., export as CSV or RDF).

3 Methodology

We now present a five-step methodology for the creation of a scholarly knowledge graph from survey tables. In order to reach sufficient quality, the methodology takes a human-in-the-loop approach in which multiple steps require human interaction. Data quality improves with human evaluation and, if needed, correction of the extracted data. The methodology is displayed in Figure 2. The scripts required to perform the steps are available online.⁶

3.1 Paper Selection

In the first step, suitable survey papers are selected based on multiple criteria. The purpose is to find survey papers from a diverse range of domains. Therefore, a protocol has been designed to determine which papers are suitable for data extraction. The structured nature of the selection process is needed to be able to make conclusions about the percentage of survey papers that present the information in such a way that extracting data is relatively straightforward.

Search Strategy. Table 2 lists the search engines used to find survey articles. Google Scholar is chosen to ensure that survey papers from various fields are searched. Additionally, ACM Digital Library has been selected because the ORKG currently focuses mainly on the Computer Science domain. The search is

⁶ <https://doi.org/10.5281/zenodo.3739427>

Table 2: Search engines used to find survey articles.

Search engine	Field	Evaluated papers
Google Scholar	All	335
ACM Digital Library	Computer Science	80

limited to 100 papers that are suitable for import. The following search criteria are used:

- Google Scholar: the article title contains the term “literature survey”.
- ACM Digital Library: queries “literature review” and “literature survey”.
- The survey article has been published after 2002.
- The results are sorted by relevance.

The rationale for selecting papers published after the year 2002 is because in general more recent papers are more interesting for research and should therefore have more priority in the scholarly knowledge graph. In the end, articles published before 2002 can still be part of the graph, since this criterion only applies to the survey articles themselves, and not to the papers being reviewed in those articles.

Selection Criteria. Papers that satisfy the inclusion criteria are selected for the import process. The inclusion criteria are defined as follows:

1. The article contains at least one table that lists scientific literature (i.e., the literature is presented in a semi-structured manner).
2. The article compares literature based on published results and does not solely textually summarize the content of original papers.
3. The survey table should be in markup format and not included as raster image.
4. The table structure should be suitable for import (e.g., one table row should provide information about one publication).
5. The article is written in English.

Inclusion criterion 1 ensures that a survey article does not only textually summarize the literature, but does also provide a semi-structured comparison (in tabular form). Although papers that are textually reviewing scientific literature are interesting for importing as well, it is out of scope for this work. Criterion 2 ensures only surveys that compare actual paper results are included. This excludes surveys researching, for instance, the growth of a field. Criterion 3 excludes tables in image format. This is because of the tabular extraction method we use, which is based on character extraction and does not use Optical Character Recognition (OCR) needed to support image extraction [30]. Criterion 4 only selects tables that are suitable for import. Our methodology does only support paper import when one row in a table represents one paper. Although minor

changes can be made manually (e.g. merging multiple tables), in case the structure of the table deviates significantly from the required format, the table is excluded. Finally, criterion 5 ensures a homogeneous semantic integration into the currently English monolingual knowledge graph. The result of this step is a set of the selected papers in PDF format.

3.2 Table Extraction

This step focuses on extracting the tables from the PDF files collected in the previous step. Not only the text within the table should be extracted, but the tabular structure should be preserved as well. As explained in the related work section, we use Tabula to perform the table extraction. Each PDF article is uploaded via the Tabula user interface. Afterwards, the regions of the tables are manually selected within the interface. Although Tabula provides a functionality to automatically detect tables, the accuracy is not sufficient for our use case. The performance is especially low for articles with a two-column layout. Additionally, not all tables within an article have to be extracted since not all of them are listing and comparing literature. Arguably, the manual selection method is most useful in this methodology since human judgment is needed in the selection process. Part of the extraction step is quality assurance after the extraction. When needed, extraction errors are manually fixed. Tabula supports two types of extraction, namely “Stream” and “Lattice”. The Stream extraction method is based on white space between columns while Lattice is based on boundary lines between columns. During the extraction it is possible to switch between the different methods, which allows for selecting the best method for a particular table. The result of this step is a set of CSV files, in which each file represents one survey table from a review article.

3.3 Table Formatting

The CSV files containing the extracted tables from the review articles should be formatted in a structure that is suitable for building a graph. Since the data from the CSV file is extracted automatically, all tables should have the same format. In this step, the formatting of the tables is changed when necessary. For some tables, a considerable amount of changes is required while for other tables only minor changes are needed. Changes could include merging, splitting, adding and removing both columns and rows. We use OpenRefine [31] to perform bulk operations on tables. A table is formatted in such a way that it adheres to the following rules:

1. The first row of the table is the header.
2. Each row represents one reviewed paper.
3. Each row has a column called: “Reference”.
4. The reference cell should contain the citation key for a paper .
5. Non-literal values are prefixed with “[R]” in the column header.
6. When needed, abbreviations are replaced by the full value from the legend.

For rule 2, in some cases a multidimensional table has to be flattened. This can often be accomplished by adding additional columns to the table. Also, in some cases a table has to be transposed to ensure that each row contains one paper. Rules 3 and 4 ensure that bibliographic metadata can be fetched for each paper in the next step. Rule 5 makes a distinction between literal values and resources. The default cell type is a literal, and when [R] is prefixed to a header label, the cells are considered as resources. Finally, rule 6 makes the content of the table readable without requiring the original text from the legend. Often table legends are used to condense information to improve user readability.

3.4 Extracting References

As mentioned earlier, each table row represents one paper. For each row, there is a value that contains the reference key from the original paper. The reference key is often a numerical reference, in the form of [n], where n represents the reference number. In another frequently used citation style, the author names combined with their publication year is used as a reference key. The citation key is used to automatically capture the bibliographic metadata for an article. In order to extract references from article, we use the PDF extraction tool GROBID. GROBID processes the full PDF article. In the first place to extract all citations from the paper's reference list and then to connect the citation keys used in the text to their respective citation string. In case a reference key cannot be extracted from the paper's text, a reference key is generated automatically based on the author's name and publication year.

When the citation is extracted and parsed, five additional columns are appended to the table: paper title, authors, publication month, publication year and the DOI⁷. In case a citation key could not be automatically mapped to an actual citation, a citation can be provided manually. The full citation text can be copied directly from the paper (including paper title, authors etc.) and is then parsed by GROBID to get structured bibliographic metadata. To perform the process of adding references, we created a Python script.⁸ This script first tries to automatically fetch the metadata. In case the reference is not found, a command line input field is displayed to enter the citation manually.

3.5 Build Graph

The final step is to build a knowledge graph from the previously created CSV files. An example of the resulting graph for a single paper is depicted in Figure 3. Firstly, a settings file is created which lists the table numbers, a suitable title for the table and a reference to the original survey article. The reference is required to attribute the work done by the authors of the survey article. The table title is manually created based on the original table caption. In case no suitable caption is available, a more suitable title is written.

⁷ Digital Object Identifier

⁸ File `4_reference_extraction.py` from <https://doi.org/10.5281/zenodo.3739427>

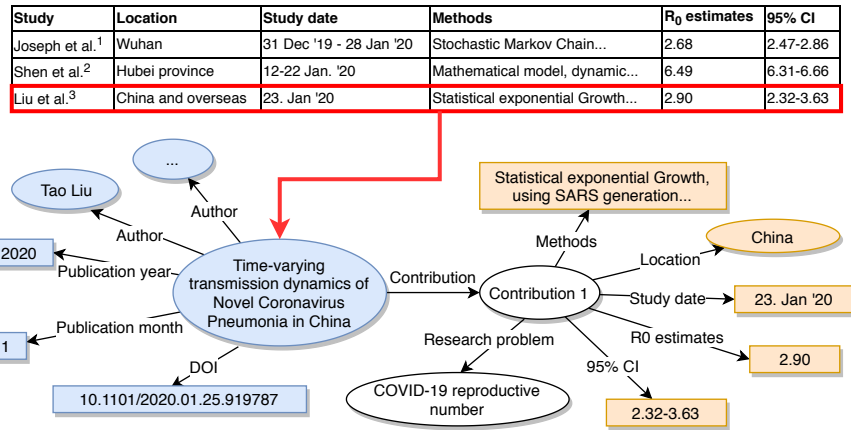


Fig. 3: Example of the resulting subgraph for importing a single paper from a survey table. Metadata captured by reference extraction is displayed in blue. Data coming from the survey table is displayed in orange and ORKG specific data is displayed in white.

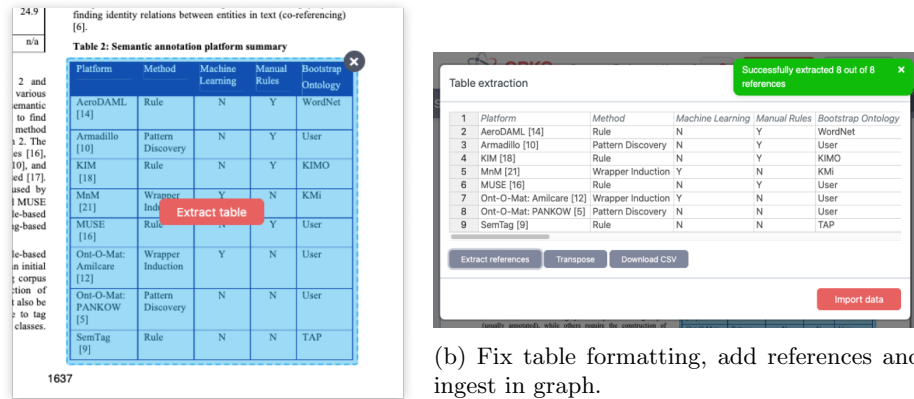
Next, a Python script⁹ is used to select all rows from the tables. For each row, a paper is added to the graph via the ORKG API. For each table, a comparison is created in ORKG. The title and reference from the previously generated settings file are attached to this comparison. The comparison can be used later in ORKG to generate the same tabular literature overview as originally presented in the survey paper.

3.6 User Interface

Based on the steps from our methodology, a web User Interface (UI) is created that integrates all steps into a single interface. The interface provides a streamlined process for importing survey tables as depicted in Figure 4. The UI is specifically designed to make importing a table an effortless task without the need of downloading any tools or the need to be able to operate these tools. In the background, the same tools from the methodology are used to extract tables (Tabula) and extract references (GROBID). The first step is to upload a PDF file and select the survey table within this file. Afterwards, the table is extracted and the formatting can be fixed with an integrated spreadsheet editor. Then, for each row the respective paper reference is extracted. Finally, the data is ingested in the knowledge graph.

The UI is not used to import the surveys tables presented in Section 4. The interface is designed to import individual survey tables rather than importing large amounts of tables at once. In the UI, all steps required to import a single table should be performed consecutively. To increase efficiency when importing

⁹ File 5_build_graph.py from <https://doi.org/10.5281/zenodo.3739427>



(a) Select and extract table from PDF.

Fig. 4: Survey table import User Interface integrating all steps from the methodology.

large amounts of tables, it helps to first finish a step for all papers before moving to the next step. The UI provides a method to extend the graph beyond the extracted surveys from this work. In the future, this interface will therefore be integrated in the ORKG.

4 Results

In this section, we report the results of the import process for each step of the methodology. Table 3 summarizes the results for all steps.

4.1 Paper Selection

The dataset of the results are published online [24]. This set contains the selected papers, the ORKG comparisons and the ingested papers. The selected papers file lists IDs, paper titles, table references, sources and references. The IDs are used to record any additional information about the import process for this specific paper. IDs are missing for papers that were selected in the first place, but were excluded after revising the inclusion criteria. Additionally, table references refer to the original table references used in the survey article.

In total, 335 papers from Google Scholar were evaluated against the selection criteria described in Section 3.1. Out of these papers, 78 met the criteria and have therefore been selected for importing. From the ACM Digital Library 80 papers were evaluated and 14 papers have been selected. In total 22% of the evaluated review papers are suitable to be imported with the presented approach.

Table 3: Summary of the results of all steps.

Description	Amount
<i>Paper selection</i>	
Amount of evaluated papers	415
Amount of selected papers	92
<i>Table extraction</i>	
Total amount of extractions (partial tables)	265
Amount of extracted complete tables	160
<i>Reference extraction</i>	
Found references	2 069
Not found references	1 137
<i>Build graph</i>	
Individual amount of imported papers	2 626
Imported data cells (with metadata)	40 584
Imported data cells (without metadata)	21 240

4.2 Table Extraction

We extracted 160 tables from the 92 survey articles. In 22 cases, tables stretched across multiple pages, which results in a total of 265 extractions performed with Tabula. Table 4 lists the most frequently occurred issues with the extraction. Issue 1 and 2 occur mostly when no boundary lines are present between table columns. In this case, the Stream extraction method has to be used, which often results in rows that are not correctly merged (e.g., multi-line sentences are put in separate rows while in the original table they are in the same row). Also, issue 3 is mostly present when using the Stream method. When the Lattice method can be used for the extraction, the result is generally of higher quality. When no table borders (or boundary lines) are present, this method does not work and the Stream method has to be used. Issue 4 is caused by general extraction errors,

Table 4: Issues that occurred during the extraction of tables from the survey articles. Issues are counted per article.

#	Issue	Percentage %
1	Columns are not extracted correctly	26
2	Rows are not extracted correctly	14
3	Empty columns in the extracted table	14
4	Text not correctly recognized (e.g., missing letters or formulas)	12
5	Issue with table header text	12
6	Vertical text not imported correctly	4
7	Cell value not supported (e.g., use of image instead of text check marks)	3
8	Table within table not extracted correctly	3

which can result in tables with wrongly extracted text. Additionally, formulas and other text styling are not supported, which compounds this issue. Issues 7 and 8 result in tables that are not, or only partially, imported. The other issues are self-explanatory.

4.3 Reference Extraction

In total, we extracted unique 2 626 papers from 3 206 rows. For each paper, the respective citation was retrieved. In 2 069 cases the citation could be extracted automatically from the row (65% of the cases). In 1 137 cases it was not possible to automatically extract the reference (35% of the cases). For those cases, the citation is manually copied from the paper. There were multiple reasons why automatic reference extraction was not successful. Most issues occurred for references that used a numeric citation key. GROBID’s performance for extracting numeric references from tables was low, oftentimes numeric table references were not recognized. The amount of rows is higher than the amount of extracted papers because multiple rows could refer to the same paper. Each paper only has one graph entry and any additional data is added to the existing paper.

In case a reference is only used in a table and not somewhere else in the article, automatic reference extraction was oftentimes not possible. When an author name was used as citation key, problems occurred mostly because of the different citation styles. While some citation formats only use the last name of the first author, suffixed by *et al.*, other formats could list all author names. When a format was used that deviates from the standard implementation, automatic extraction was not possible.

4.4 Build Graph

In total, we added 2 626 papers to the knowledge graph. These papers are used in 160 different comparisons. A complete list of the generated ORKG comparisons and a list of all ingested papers is available via [24]. In total, 21 240 table cells have been imported, excluding the bibliographic metadata. Including metadata, the total is 40 584 data cells.

5 Discussion and Future Work

5.1 Time Performance

The presented methodology takes a human-in-the-loop approach as opposed to a fully automated approach. Compared to a fully manual approach, the proposed approach saves considerable time. In previous work [23], we manually imported only four survey articles. On average, this process took 4 hours per article. For each of the papers, a Python script was created specifically to import the survey table with its references and data. An example of such a script for one paper can

be found online.¹⁰ For the methodology used in this paper, the time to import one survey article was on average 15 minutes. Compared to the 4 hours of the manual approach, this is considerably faster (i.e., 16 fold increase in speed). The minimum amount of time needed to import a relatively small table was 2 minutes. The table could be extracted without any issues. The maximum amount of required time was approximately 60 minutes. This was for a table with a complex layout, stretched across multiple pages. Also, this table did not have boundary lines. Most time was spend on fixing extraction issues. To further improve time performance, we identified two tasks that are time-consuming and can potentially be improved. The first task relates to fixing errors occurred during the table extraction by Tabula. Most errors occurred when tables did not have boundary lines between columns and rows. A potential solution, and possible future research direction, is to create an interface that supports manually drawing boundary lines between rows and columns. The second task is related to adding missing references, which have to be manually copied from the PDF article. In total, 65% of the references were extracted automatically. By applying more advanced heuristics to match reference keys with their respective reference, this percentage can be improved.

5.2 Impact of Methodology

The impact of the methodology relates to the amount of survey papers that are suitable for our approach (i.e., surveys representing information in tabular format). To order to provide insights on the impact, a structured search protocol has been employed in the paper selection step. As the results show, out of the 415 evaluated papers, 92 of them are suitable to be imported. This indicates that since 2002, 22% of the published survey papers contain comparison tables. Therefore, arguably our methodology can have considerable impact when applied more broadly. In the paper selection, non-survey papers were excluded. However, it is not uncommon for research articles to also contain tables with related work (e.g., Table 1 in this article). Thus the paper selection step could be extended to also include other articles to have a broader impact.

5.3 Semantics of Data

The extracted knowledge graph consists of structured scholarly data. The quality of the knowledge graph could be further improved by providing more semantics to the data. Currently, a primitive method is used to map existing properties and resources. This is based on a lookup by resource label, in case a result is found, the resource is mapped. If not, a new resource is created. A more advanced mapping of resources and properties to existing ontologies improves the machine readability of the data. Tables containing large amounts of natural text (e.g., textually describing a methodology) could be further processed using

¹⁰ <https://gitlab.com/TIBHannover/orkg/orkg-papers/-/blob/master/question-answering-import.py>

named entity recognition and linking. This results in more structured data and therefore a higher quality knowledge graph. Approaches to improve the overall quality of the graph are part of future work.

5.4 Future Research Directions

In total, we extracted 92 survey articles from a variety of domains. In the future, more survey articles will be ingested in ORKG. This will be done for multiple domains. The User Interface (UI) presented in Section 3.6 can be used to support users to import survey tables. The UI will be further improved to make to process more efficient. Due to the dynamic nature of the interface (especially compared to a regular spreadsheet editor), mapping properties and resources to existing concepts is better supported. In the end, we aim to import as many surveys from a specific domain as possible. There are several reasons why such an approach is useful. In the first place, ORKG can serve as a digital library for literature surveys. As discussed in the related work, the platform provides tools to better find and organize surveys. Additionally, when all existing reviews for a domain are imported, the ORKG can be used as a source to find literature surveys. In case a survey is not present in the ORKG, it means that it does not (yet) exist. This can be used as a basis to start working on new literature surveys.

6 Conclusions

Knowledge graphs are useful to make scholarly knowledge more machine actionable. Manually building such a knowledge graph is time-consuming and requires the expertise of paper authors and domain experts. In order to efficiently build a high-quality scholarly knowledge graph, we leverage survey tables from review articles. Generally, survey tables contain high-quality, relevant, semi-structured and manually curated data, and are therefore an excellent source for building a scholarly knowledge graph. We presented a methodology used to extract 2 626 papers from 92 survey articles. The methodology adopts a human-in-the-loop approach to ensure the quality and usefulness of the extracted data. Compared to manually reviewing and entering research data, or to manually importing literature surveys, the methodology is considerably more efficient. In conclusion, the presented methodology provides a full pipeline that can be used to extract knowledge from PDF documents and represent the extracted knowledge in a knowledge graph. The corresponding evaluation with survey articles demonstrates the effectiveness and efficiency of the proposed methodology.

Acknowledgements. This work was co-funded by the European Research Council for the project ScienceGRAPH (Grant agreement ID: 819536) and the TIB Leibniz Information Centre for Science and Technology. We want to thank our colleagues Mohamad Yaser Jaradeh and Kheir Eddine Farfar for their contributions to this work.

References

1. Adelfio, M.D., Samet, H.: Schema extraction for tabular data on the web. *Proceedings of the VLDB Endowment* **6**, 421–432 (2013). <https://doi.org/10.14778/2536336.2536343>
2. Corrêa, A.S., Corrêa, P.L.P., Da Silva, F.S.C.: Transparency portals versus open government data. An assessment of openness in Brazilian municipalities. *ACM International Conference Proceeding Series* pp. 178–185 (2014). <https://doi.org/10.1145/2612733.2612760>
3. Corrêa, A.S., Zander, P.O.: Unleashing tabular content to open data: A survey on PDF table extraction methods and tools. *ACM International Conference Proceeding Series* pp. 54–63 (2017). <https://doi.org/10.1145/3085228.3085278>
4. Fathalla, S., Vahdati, S., Auer, S., Lange, C.: Towards a Knowledge Graph Representing Research Findings by Semantifying Survey Articles. In: *Research and Advanced Technology for Digital Libraries*. pp. 315–327 (2017)
5. Gall, M.D., Borg, W.R.: *Educational Research: An introduction* (sixth edition). White Plains, NY: Longman Publishers USA (1996)
6. Hart, C.: *Doing a Literature Review: Releasing the Social Science Research Imagination*. Sage (1998)
7. Hassan, T., Baumgartner, R.: Table recognition and understanding from PDF files. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR* pp. 1143–1147 (2007). <https://doi.org/10.1109/ICDAR.2007.4377094>
8. Herrmannova, D., Knoth, P.: An analysis of the microsoft academic graph. *D-lib Magazine* **22**(9/10) (2016). <https://doi.org/10.1045/september2016-herrmannova>
9. Hyvönen, E.: Publishing and using cultural heritage linked data on the semantic web. *Synthesis Lectures on the Semantic Web: Theory and Technology* **2**(1), 1–159 (2012). <https://doi.org/10.2200/S00452ED1V01Y201210WBE003>
10. Jaradeh, M.Y., Oelen, A., Farfar, K.E., Prinz, M., D’Souza, J., Kismihók, G., Stocker, M., Auer, S.: Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge. *K-CAP 2019 - Proceedings of the 10th International Conference on Knowledge Capture* pp. 243–246 (2019). <https://doi.org/10.1145/3360901.3364435>
11. Jiang, D., Yang, X.: Converting PDF to HTML approach based on text detection. *Proceedings of the 2nd international conference on interaction sciences: Information technology, culture and human* **403**, 982–985 (2009). <https://doi.org/10.1145/1655925.1656103>
12. Klampfl, S., Granitzer, M., Jack, K., Kern, R.: Unsupervised document structure analysis of digital scientific articles. *International Journal on Digital Libraries* **14**(3-4), 83–99 (2014). <https://doi.org/10.1007/s00799-014-0115-1>
13. Körner, M., Ghavimi, B., Mayr, P., Hartmann, H., Staab, S.: Evaluating Reference String Extraction Using Line-Based Conditional Random Fields: A Case Study with German Language Publications. In: *New Trends in Databases and Information Systems*. pp. 137–145. Cham (2017)
14. Krotzsch, M., Vrandečić, D.: Wikidata : A Free Collaborative Knowledge Base. *Communications of the ACM* **57**(10), 78–85 (2014). <https://doi.org/10.1145/2629489>
15. Lammey, R.: CrossRef text and data mining services. *Insights: the UKSG Journal* **28**(2), 62–68 (2015). <https://doi.org/10.1629/uksg.233>

16. Lehmberg, O., Ritze, D., Meusel, R., Bizer, C.: A large public corpus of web tables containing time and context metadata. Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion (2016). <https://doi.org/10.1145/2872518.2889386>
17. Lipinski, M., Yao, K., Breitinger, C., Beel, J., Gipp, B.: Evaluation of header metadata extraction approaches and tools for scientific PDF documents. Proceedings of the ACM/IEEE Joint Conference on Digital Libraries pp. 385–386 (2013). <https://doi.org/10.1145/2467696.2467753>
18. Liu, Y., Bai, K., Mitra, P., Giles, C.L.: Tableseer: automatic table metadata extraction and searching in digital libraries. Proceedings of the 2007 conference on Digital libraries - JCDL '07 (2007). <https://doi.org/10.1145/1255175.1255193>
19. Lopez, P.: GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications. International conference on theory and practice of digital libraries **5714**, 473–474 (2009). https://doi.org/10.1007/978-3-642-04346-8_62
20. Mäkelä, E., Hyvönen, E., Ruotsalo, T.: How to deal with massively heterogeneous cultural heritage data - Lessons learned in CultureSampo. Semantic Web **3**(1), 85–109 (2012). <https://doi.org/10.3233/sw-2012-0049>
21. Mons, B., Velterop, J.: Nano-publication in the e-science era. In: Workshop on Semantic Web Applications in Scientific Discourse (SWASD 2009). pp. 14–15 (2009)
22. Oelen, A., Jaradeh, M.Y., Farfar, K.E., Stocker, M., Auer, S.: Comparing Research Contributions in a Scholarly Knowledge Graph. In: Proceedings of the Third International Workshop on Capturing Scientific Knowledge (SciKnow19). pp. 21–26 (2019)
23. Oelen, A., Jaradeh, M.Y., Stocker, M., Auer, S.: Generate FAIR Literature Surveys with Scholarly Knowledge Graphs. JCDL '20: The 20th ACM/IEEE Joint Conference on Digital Libraries (In Press) (2020). <https://doi.org/10.1145/3383583.3398520>
24. Oelen, A., Stocker, M., Auer, S.: Dataset for Creating a Scholarly Knowledge Graph from Survey Article Tables (2020). <https://doi.org/10.5281/ZENODO.3735152>
25. Rastan, R., Paik, H.Y., Shepherd, J.: Texus. Proceedings of the 2015 ACM Symposium on Document Engineering - DocEng '15 (2015). <https://doi.org/10.1145/2682571.2797069>
26. Ros, G.: Analysis of Tabula : A PDF-Table extraction tool (2019)
27. Skjæveland, M.G., Lian, E.H., Horrocks, I.: Publishing the Norwegian Petroleum Directorate's FactPages as semantic web data. International Semantic Web Conference **8219**, 162–177 (2013). https://doi.org/10.1007/978-3-642-41338-4_11
28. Takis, J., Islam, A.S., Lange, C., Auer, S.: Crowdsourced semantic annotation of scientific publications and tabular data in pdf. SEMANTICS '15 Proceedings of the 11th International Conference on Semantic Systems (2015). <https://doi.org/10.1145/2814864.2814887>
29. Vahdati, S., Fathalla, S., Auer, S., Lange, C., Vidal, M.E.: Semantic Representation of Scientific Publications. International Conference on Theory and Practice of Digital Libraries **11799**, 375–379 (2019). https://doi.org/10.1007/978-3-030-30760-8_37
30. Vasileiadis, M., Kaklanis, N., Votis, K., Tzovaras, D.: Extraction of tabular data from document images. Proceedings of the 14th Web for All Conference, W4A (2017). <https://doi.org/10.1145/3058555.3058581>
31. Verborgh, R., De Wilde, M.: Using OpenRefine. Packt Publishing Ltd (2013)
32. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Quarterly **26**(2), xiii – xxiii (2002)

33. Wee, B.V., Banister, D.: How to Write a Literature Review Paper? *Transport Reviews* **36**(2), 278–288 (2016). <https://doi.org/10.1080/01441647.2015.1065456>
34. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., Bouwman, J., Brookes, A.J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C.T., Finkers, R., Gonzalez-Beltran, A., Gray, A.J., Groth, P., Goble, C., Grethe, J.S., Heringa, J., t Hoen, P.A., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S.J., Martone, M.E., Mons, A., Packer, A.L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M.A., Thompson, M., Van Der Lei, J., Van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., Mons, B.: Comment: The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* **3**, 1–9 (2016). <https://doi.org/10.1038/sdata.2016.18>