Acta Informatica Pragensia, 2020, 9(2), 170–183
DOI: 10.18267/j.aip.139
Original Article

The Process of Unit Price Extraction from Public Sector Contracts

Tomáš Bruckner [1], **Filip Vencovský** [1]

Abstract

Czech government institutions commissioned a research on extracting usual unit prices from public IT contracts to aid future public tender sizing. The goal of the project is to obtain millions of contracts from the public register, convert them to full text, extract unit prices from the text and publish a pricelist of IT industry manday prices. This paper designs the process and method of price extraction, demonstrates and evaluates the result on five iterations of extraction and discusses the experience of two years of project performance. The process is designed as a set of repeatable workflows and specified activity and role description. The method is designed as a combination of automated and manual actions. Due to the format and content variability of involved documents and the low mistake tolerance, the possibility of automated extraction of unit prices from full text contract is limited, and human workforce for validation is crucial.

Keywords: Usual price, Contracting, Full text analytics, Information technology, Business process.

1 Introduction

Although the major discussion of pre-tender estimation in academic literature is linked with building tenders (Aibinu & Pasco, 2008; Skitmore & Picken, 2000), the same issue applies to tenders in the Information technology (IT or ICT) area. IT or Information Systems contracts tendered by public institutions are sensitive with regard to correct sizing (Ochrana & Pavel, 2013), and public institutions often struggle with the need of usual price determination before any public tender.

A public institution usually sets the price with the help of a counsellor, who uses several demonstrably negotiated contracts and proves prices negotiated in a given time and place, as stated in (Czechia, 1990). In IT and Information Systems contracts, such findings can vary considerably and hence are not very useful for contract sizing.

This lead 13 public institutions to join in an attempt to create a broad overview of usual prices related to IT services, based on all contracts demonstrably negotiated by public institutions in the Czech Republic available in the public contract register (Czechia, 2015; Ministry of Internal Affairs, 2019b).

The public contract register (Czechia, 2015; Ministry of Internal Affairs, 2019b) contains specific meta-data such as contracting parties, date of conclusion, or contract value. However, the unit price remains latent in contract content. Hence, the unit price has to be extracted using text mining techniques.

[1] Department of Information Technologies, Faculty of Informatics and Statistics,
Prague University of Economics and Business, W. Churchill Sq. 1938/4, 130 67 Prague, Czech Republic
✉ bruckner@vse.cz

The result of this attempt should not only be an overview at a certain time but mainly the process that allows public institutions to recreate the overview on a regular basis. Therefore, we chose process design as the main subject of this paper. In order to design the process of price extraction, we reviewed academic literature, and defined solution objectives and method.

1.1 Related work

There are examples of mining data from contracts in academic literature. A complex tool called Contract Miner (Gao et al., 2012) parses contract sentences and identifies sentences matching a pattern. The tool was designed for service exception extraction and was demonstrated on IT services contracts. The approach is not suitable for unit price extraction because unit price is stated not only in sentences but also in tables or blocks. A very recent paper on information extraction from contracts (Kim et al., 2020) uses dependency parsing and capturing the subject-verb-object pattern, and thus is also inadequate due to the need for natural language input.

The next closest research area focuses on information extraction from invoices. The area is connected with optical character recognition (OCR) and copy machine producers (Ha et al., 2018). Besides textual data, other layout and visual features are extracted from a document (Ha, 2017; Ha et al., 2018; Palm et al., 2017). There are studies aimed at invoice detection among other business documents (Ha, 2017), or the detection of invoice type (handwritten, printed, receipt) (Tarawneh et al., 2019). The identification of the right blocks on an invoice that include the price field by a combination of visual and text features has reached an accuracy of 80% (Ha et al., 2018), and F1 of 84% with the use of recurrent neural networks (Palm et al., 2017).

Nevertheless, the advances in invoice processing area are promising. Contracts in the public register (Czechia, 2015; Ministry of Internal Affairs, 2019b) are mostly in PDF format, but also in doc, docx, plain text, and rarely in image formats. Therefore, it is not possible to depend on visual layout features. Moreover, unit prices are in different forms and different sections, such as in appendices in different files, or are not present at all. Contracts that contain a unit price may also contain a clause that limits the price to specific conditions only, or cancels the contract altogether. This led us to conclusion that the solution cannot be a computer algorithm but rather a complex process that is reliable, involves different specialisations and accumulates domain knowledge for future use.

1.2 Solution objective

The goal of the solution is to design a process, as a set of prescribed activities, the following of will successfully generate the pricelist of usual manday prices in Czech ICT industry for a given period.

The process should fullfill the purpose of the pricelist, should be able to produce a pricelist of the expected quality with minimum errors, every unit price taken into processing should be back-traced to the original contract, and the majority of public ICT contracts from the public register should be extracted.

The process should be suitable for use, it should be applicable consistently in various iterations, with changing staff and within a deadline.

2 Method

As the main topic of this paper is the process design, we chose Design Science Research (DSR) as the main methodological framework. DSR (Hevner & Chatterjee, 2010) is a research paradigm widely used in Information Systems area that is suitable for designing artifacts, such as processes (Hevner et al., 2019).

Hevner and Chatterjee (Hevner & Chatterjee, 2010) propose DSR that consists of six steps:

1. Problem identification
2. Defining the objectives for a solution
3. Design and development
4. Demonstration
5. Evaluation
6. Communication

The selected methodological framework of DSR with these six steps determines the structure of this paper. This section describes how we followed these steps and which chapters contain the results of each step.

2.1 Problem identification and definition of the objectives for a solution

We identified the problem, conducted a literature review, and set the objectives for a solution in the Introduction.

2.2 Design and development

We chose BPMN as a notation for process modelling (OMG, 2014). The initial events, outputs, actors, and activities were defined by textual description. The design of particular activities consists of the goal, inputs, outputs, execution directions and rules, and limitations. We investigated the character of the input sources, the ISRS(Ministry of Internal Affairs, 2019b), initially downloaded several contracts, extracted prices, and set the statistical method of pricelist assembly. Then we created the initial set of activities for the first full-range pricelist generation. Non-trivial activities, which are not easily performed as a single task, were decomposed into separate tasks. On an ongoing basis we prepared infrastructure and created or configured supporting software(Bruckner, 2019). The design of the processes proceeded during the pilot iteration. The purpose was discussed with the stakeholders. The final deployable process design was the result of the pilot project. Finally, the design was ready for improvement in the following iterations of pricelist generation.

The results are described in chapter 3.

2.3 Demonstration

We demonstrated the process design by using it for its purpose, i.e. running the process instance on real data for customers. The process ran repeatedly, twice a year. Five iterations have been completed by now.

The results are described in chapter 4.

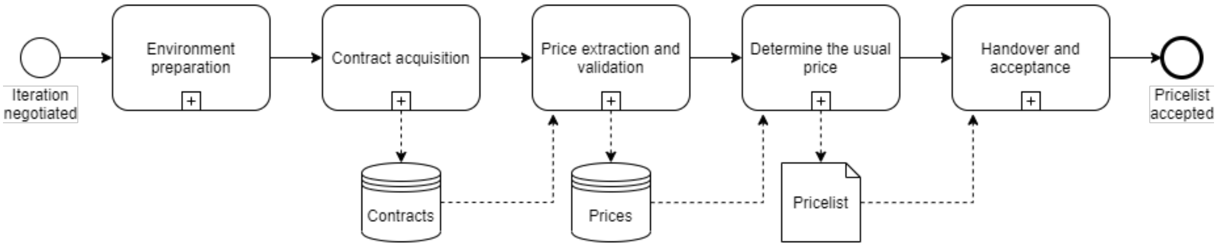

Fig. 1: *Pricelist generation process*

2.4 Evaluation

We evaluated the process according to two criteria, fit for purpose and fit for use. Fit for purpose and fit for use are inseparable components of Value composition (Taylor et al., 2007).

The first criterion, fit for purpose, is based on a formal acceptance of the output of the process by the research sponsor, according to the research contract, and a formal acceptance of the process design, again by the research sponsor. The formal acceptance of the output, the pricelist, takes place every six months and every iteration is performed to meet the deadline of the acceptance. The formal acceptance of the process design happened once, after the first iteration, while the design was consulted with the sponsor during the design. The measure of the evaluation is acceptance verdict (accepted, not accepted, accepted with reservations) and complaints. The acceptation team of the sponsor consists of 13 people, representatives of every entity interested in the research.

The second criterion, fit for use, is an iterative evaluation of the problems which occurred during the iteration and of suggestions for improvement made by the actors in the processes. During the pilot iteration, the problems and suggestions were continuously implemented into the processes. After the pilot iteration, the problems and suggestions are evaluated after every iteration and processes are improved before the next iteration.

The results are described in chapter 5.

2.5 Communication

Apart from communication with the customer and research sponsor, communication is carried out via a series of conference and journal publications, including this paper.

3 Process design

In this chapter, the resulting process as an artefact is described as it is designed after the iterations. First, the overall process is described on top level and then parts are decomposed and described in detail.

3.1 Pricelist generation process

The process defines a complete pricelist generation. It defines a complete set of activities necessary for recurring price acquisition from new data for further time periods.

The process may run routinely, using the current method, or an innovation or improvement may be applied, both overall/structural or partial, in parts of the process. The process is defined as a routine process after every improvement. The current version published here is the process as it is at present, up to date at the last iteration. The Pricelist generation process consists of the following steps (see Fig. 1):

Environment preparation. The environment is a matter of technology more than a matter of process perspective, though it is beyond the scope of this paper. The complete technological architecture and environment is described in a separate paper (Bruckner, 2019). The configuration files and all script codes are part of the solution as well.

Contract acquisition. The contract acquisition step is described in depth below.

Price extraction and validation. The price extraction and validation step is described in depth below.

Determine the usual price. The price determination step is a matter of the statistical method of determining the usual price from the price data from individual contracts. It is beyond the scope of this text, published as separate paper (Bruckner & Vencovsky, 2020).

Handover and acceptance. The handover and acceptance of the Pricelist is defined in the research contract and it is not more widely defined as a process. However, part of the handover and acceptance is the process itself, as part of the method of the next iteration of the pricelist generation. From the point of view of this text, this is part of the evaluation of the process design, described below.

3.2 Contract acquisition

The process of Contract acquisition is included as one step of the overall Pricelist generation process. The aim of this process is to acquire full texts of ICT contracts from institutions.

Initial conditions. The designed process respects the following initial conditions: The data source for the pricelist is primarily the Czech Public Register of Contracts (ISRS) (Ministry of Internal Affairs, 2019b), specifically its public part. In case of a change in ISRS data structures, the scripts are appropriately adjusted. The update of the pricelist will not be continuous, there will be a delay between iterations. The process is designed as repeatable, but it is not designed to run in parallel. The degree of process automation expects the role of an administrator who is familiar with the solution architecture and infrastructure. We consider as ICT contracts only those contracts, in which one of the parties is a company with a registered activity belonging to chosen ICT activities of NACE (Eurostat, 2008). Other companies' contracts are included only if the company is positively listed by the research sponsor.

The model of the process of Contract acquisition is in Fig. 2. In the following text we describe the starting events, the output, the roles and the activities of the process.

Events. The process is initiated by two events, one is a time event and one factual. The time event is established as two months before the expected final results of the Pricelist generation, as we found that two months allow sufficient time to perform the complete generation process. The factual event represents the infrastructure is ready for the process to be performed, including appropriate code adjustments induced by the ISRS data structures or API changes. The process of contract acquisition starts after both the events occurred.

Output. The output of the process is all full texts of all relevant contracts within the iteration period indexed and prepared for further processing.

Roles. The whole process does not need to specify different roles. The only overall role can be specified as a Researcher or an Administrator. The Administrator is sufficient for routine process performance, while the maximum number of operations is automated. If an innovation is expected, a Researcher is the more appropriate name of the actor.

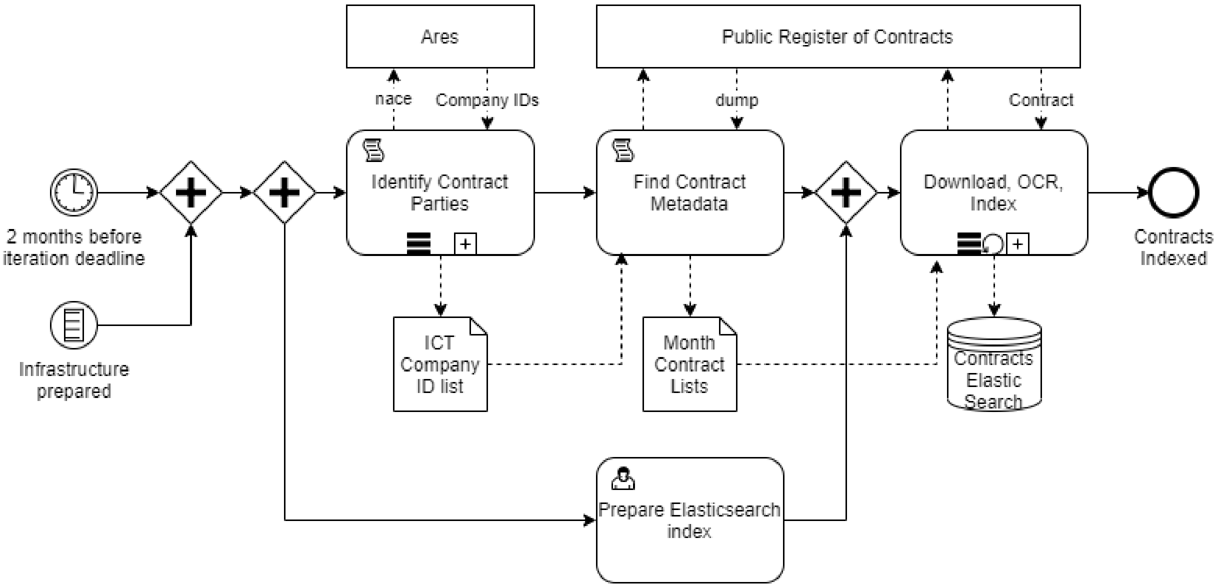

Fig. 2: *Contract acquisition process*

3.2.1 Activity Identify contract parties

The goal of the activity is to provide a list of all companies (or subjects) which provide ICT-related business activities. The input of the process is a NACE code list, and a public register of business subjects of the Czech Republic called ARES (Administrative Register of Economic Subjects) (Ministry of Finance, 2019). The output of the process is a list of all unique registered ID numbers of companies that at the time of the pricelist iteration provided ICT-related services.

Limitations. In order to reduce the amount of data downloaded and indexed and thus reduce the time and other resources needed, contracts outside the ICT industry are not acquired. The decision about whether the contract is related to ICT industry must be made before downloading the contract, thus it is based on contract metadata. The industry is not a special metadata item and the contract title did not seem to provide reliable data for this decision. Therefore, the NACE classification of the contract parties' activity was considered as the criterion. The expectation is that there will be no contract that is related to the ICT industry and has no contract party registered as providing ICT-related business activities. And if there is one, it will be lost. This method reduces the number of downloaded contracts significantly, even if not completely, because all non-ICT contracts of relevant parties are also acquired. In the pilot iteration, the NACE code J (Information and Communication), activities 58, 59, 60, 61, 62 and 63 (Eurostat, 2008) were used. All companies in the Czech Republic should have their business activities registered in the public register ARES.

Execution. A set of queries to the API of ARES was designed. Due to the restrictions of the public register API, the queries were divided according to regions in order not to exceed the limit of returned records. The queries are performed sequentially with delays caused by possible access blocking due to intensive API use. For the case of blocking and a necessary repeated run, the tools for concatenating results are coded.

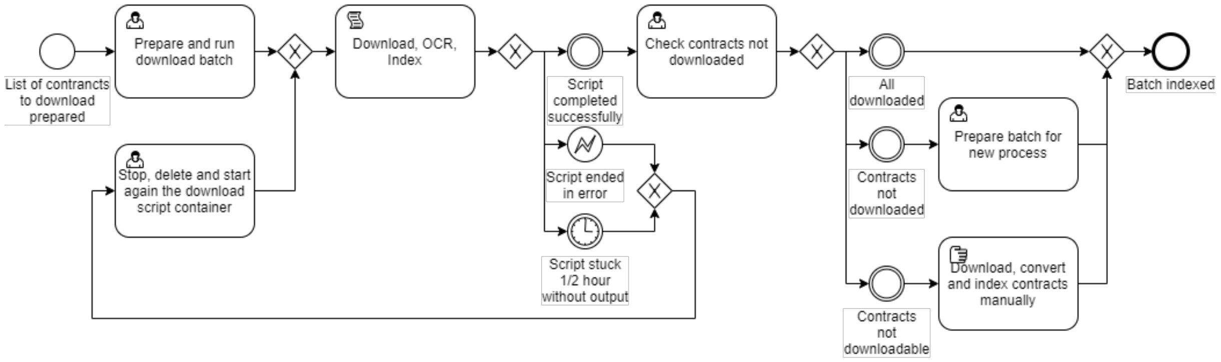
List of contrancts to download prepared
Prepare and run download batch
Stop, delete and start again the download script container
Download, OCR, Index
Script completed successfully
Script ended in error
Script stuck 1/2 hour without output
Check contracts not downloaded
All downloaded
Contracts not downloaded
Contracts not downloadable
Prepare batch for new process
Download, convert and index contracts manually
Batch indexed

Fig. 3: *Download, OCR, Index sub-process*

3.2.2 Activity Find contract metadata

The goal of the activity is to provide a list of ICT contracts, which should be downloaded for a specified period, which is one month. The input for the activity is the list of all company IDs meeting the NACE condition and the month period. The output are batches of contract numbers to be downloaded from ISRS, sorted by months.

Execution. The ISRS provides contract metadata dump files containing all contracts added to the register or modified within the given months. The files may vary as contract versions are updated or contracts removed due to various reasons. The execution script asks for the dump files and reduces the file by identifying the contracts in which one of the parties meets the NACE condition. Moreover, from different versions of the same contract the latest one is selected. The contract numbers to be downloaded are stored in a file for every month. A contract number (or, more precisely, the number of contract version in ISRS) is enough, as the contract download URI can be derived from the number.

3.2.3 Activity Prepare Elasticsearch index

The goal of the activity is to prepare a clean index in Elasticsearch software for the iteration's contract full text data, and backup the data of the previous iteration. The input is a file of mapping metadata and the output is a prepared index for the given period, including access privileges.

Execution. For any of the metadata items, the mapping sets if, how, and in which language the data is analyzed (incl. lemmatization, stemming etc.) or indexed as is without analysis. Every iteration is put into a new index. The index of the last iteration is moved to archive index using Elasticsearch re-indexation, and contracts from the current index are deleted. The index is created so that ID of the contract in Elasticsearch is the number of the contract in ISRS, not the number of contract version. Therefore, if the same contract is updated by a new version in a later month, the old version is overwritten in the index.

3.2.4 Activity Download, OCR, Index

This activity is designed as a sub-process sequentially repeated for particular month data. The goal of the sub-process is to ensure that the full texts of relevant contracts for the pricelist iteration are reverse indexed and ready for price extraction. The input is the batch of contract numbers to be downloaded. The output is the contracts indexed in Elasticsearch. The model of sub-process decomposition is shown in Fig. 3

Execution. For every month included in the pricelist iteration the sub-process is run once. The sub-process iterations should be run in the chronological order of the included months. The process logic is designed for fault tolerance when downloading and reading contracts. Even when there are problems on the network, on the side of ISRS, or in OCR software, the process ensures the assembly of batches for repeated attempts of downloading, and for system restoration if the software stops operating. The restorations were performed manually in the pilot iteration; in subsequent iterations it was automated.

3.3 Price extraction and validation

The process of Price extraction and validation is included as one step of the overall Pricelist generation process. The aim of the process is to identify ICT contracts with manhour or manday prices, extract the prices, verify correct extraction, make corrections and prepare the price data for further statistical processing.

3.3.1 Event

The process is started after all contracts of the current pricelist iteration are indexed in Elasticsearch and ready for price extraction. The process directly follows the previous step, Contract acquisition.

3.3.2 Output

The output of the process is a database of validated prices extracted from contracts, each with full metadata history for backward validity evidence – especially the link to the source contract, names of the expert validators who validated the prices and dates of validation. The output is in a form suitable for statistical processing.

3.3.3 Roles

Within this process we define four roles: Expert validator, Extraction coordinator, Query maker and Script coder.

Expert validator. extracts price data from the contracts and checks the correctness of machine extracted price data from contracts, or price data extracted or validated by other Expert validatros. The qualification of Expert validator is the knowledge of price construction in ICT industry, especially the ability to quickly understand the contract subject and price structure.

Extraction coordinator. runs the price extraction scripts, allocates parts of the machine-extracted data to Expert validators for validation, and consolidates the validated data.

Query maker. creates queries for relevant contract search and for price extraction from contract index. In case of routine extraction without the need for improvement, the ready-made queries from previous iterations can be run with no need for this role. The qualification for this role is the knowledge of query languages for Elasticsearch, the ability to use user interface Kibana, the knowledge of JSON document structure, ant the skills of full text and structured search in documents.

Script coder. creates and debugs scripts for price extraction and consolidation, adjusts the structure of validation sheets, creates extraction functions and regular expressions for extraction. In case of routine extraction without the need for improvement, the ready-made scripts can be used with no need for this role. The qualification is the knowledge of technologies of search in unstructured data, the ability of functional programming in ECMA Script 6 in node.js environment and the knowledge of regular expressions.

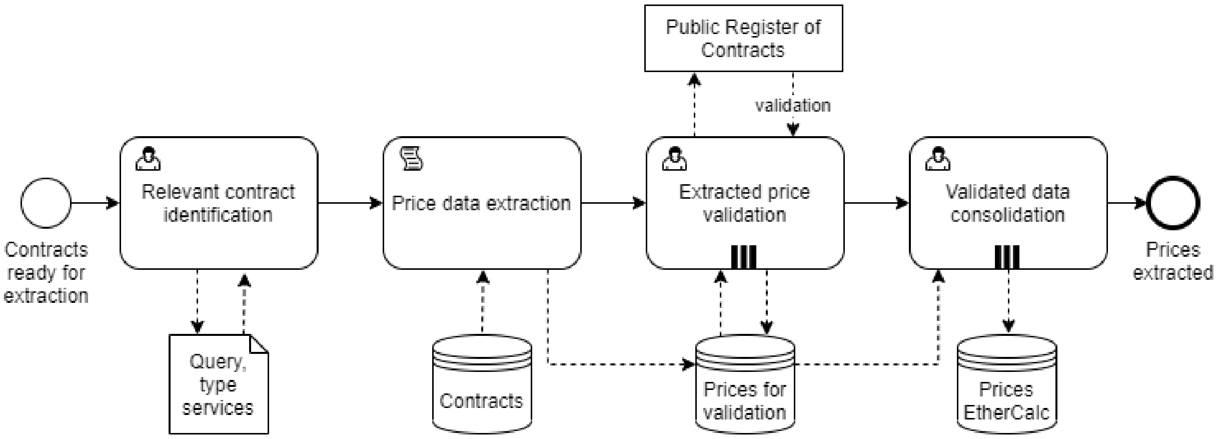
Public Register of Contracts
validation
Contracts ready for extraction
Relevant contract identification
Price data extraction
Extracted price validation
Validated data consolidation
Prices extracted
Query, type services
Contracts
Prices for validation
Prices EtherCalc

Fig. 4: *Price extraction and validation process*

The model of the process of Price extraction and validation is in Fig. 4.

3.3.4 Activity Relevant contract identification

The goal of the activity is to identify contracts related to the ICT industry which contain manday or manhour prices within the indexed contracts in Elasticsearch. Relevant contracts are identified by Elasticsearch query language, which uses Apache Lucene library. During the pricelist iterations several queries for various service types were created and tested. Here we describe the construction of the queries that have produced the best results so far.

For the determination of relevant contracts, a queries containing keywords related to services are used, by a code list of service types which includes synonyms. Keywords can be in the basic form, as the full texts are indexed including lemmatization and stemming. Besides that, every query also searches expressions typical for ICT industry contracts, in order to exclude or penalize contracts from other industries.

To distinguish the manday and manhour prices for automated extraction in next the step, special queries are created for each case.

Every query is created with the following logic: Find a specific amount of contracts, sorted by relevance (search score), which contain words identifying an ICT industry contract and which contain the description of the service type (incl. synonyms) in defined proximity to the manday or manhour expression (incl. synonyms). The defined proximity and chosen words tune the relevance of the results. The queries were tuned by direct Elasticsearch querying via Kibana application Dev Tools console.

3.3.5 Activity Price data extraction

The goal of Price data extraction is to extract price data, namely the service name, context and price value excluding VAT, from given contracts, sorted by relevance. This activity is designed as highly automated; however, expert validators are allowed to voluntarily extract data from contracts manually, in small amounts.

The principle of automated extraction is finding the pattern that looks like relevant price data in the full text of the contract. The patterns are defined as extraction functions, which could be any algorithm working with text. Based on various experiments, the extraction function with the best results was standardized. It is based on the regular expression of the following construction: Take name of the type service or its synonyms (modified for various text forms), and from this point max 200 characters on the right (incl. line ends) find 5-cipher or 4-cipher number, or a number compound of zero to two digits, a dot or a space and three digits.

A script is designed for the mass automated extraction, which takes all query results from the previous step for the service type and type of price (manday or manhour) and applies all extraction functions defined for the case.

It is possible that different queries and different extraction functions find the same contract or even the same price data. For this reason, the results are clustered by contracts and then the contracts are sorted by the most relevant price within each contract.

Then script saves the extracted prices, fulltext context, and other metadata to a table for manual validation and, by default, every price is marked as invalid. If the extraction function does not identify the price, the contract is included for manual extraction. The price data for validation are formatted to an online spreadsheet for the collaborative validation in the next step.

3.3.6 Activity Extracted price validation

The goal of the activity is to validate all extracted price data to ensure only valid prices will be processed in the pricelist.

The extracted prices are validated manually. Every expert validator goes through individual price data in the validation spreadsheet and, using the extracted full text context, evaluates whether it is certain and undoubted that the price is extracted correctly. If the expert is not sure, she/he can see the full text of the contract as indexed in Elasticsearch via the Kibana application, or the PDF files of the contract as they are stored in the public register of contracts ISRS, by clicking on attached links. In the case of incorrect data, the expert corrects it (service type, amount, manday/manhour) and, if necessary, adds tags, words which better specify the service.

If the price was not extracted, the expert validator assesses whether it is possible to extract the price manually from the indexed full text or source files. For a collaborative or parallel validation, a spreadsheet based on EtherCalc technology is used. Many expert validators can simultaneously edit the table online in a web browser.

Every validated line of the price data is marked by the expert as valid or invalid. As the experts make mistakes, it is desirable that every validated line be re-validated by one or more other experts. However, this activity is too expensive and therefore at least the prices at both the low and the high extremes are checked more than once.

Every expert validator is trained, and detailed instructions are provided with the validation spreadsheet. The experts also have a platform where they can chat about the solution of unclear cases. The price data are sorted by relevance, therefore the lower in the table, the less valid the prices are found.

3.3.7 Activity Validated data consolidation

The goal of this activity is to prepare the validated data for statistical processing. The main activity is to consolidate tags added by expert validators into sets and remove duplicate or contradictory values.

4 Demonstration

The process design was demonstrated on five real-life iterations. The first, pilot iteration, was performed from March to June 2017. The following 4 iterations were run every six months with the duration of two months. The outputs were planned to the end of May and November, in accordance with the research contract.

In the last iteration, the complete set of all contracts in the public contract register comprised millions of documents. After the NACE restrictions, the scraped amount was 566,315 contracts. After the price extraction step, the output was a set of 10,051 candidate price data items. After manual validation and tag consolidation, the data preparation finished with 6,408 price items from 2,336 unique contracts from 776 public contracting authorities and 884 vendors. The quantities were similar in all other iterations.

All process roles were performed by one person except the role of Expert validator, which included about ten workers. The Extracted price validation was the most time-consuming activity, which took three weeks in every iteration.

The detailed statistical results of the iterations are published as a conference paper (Bruckner & Vencovsky, 2020) and as reports for each iteration (Ministry of Internal Affairs, 2019a).

5 Evaluation

According to the method, there were two criteria of evaluation: formal acceptance (fit for purpose) and iterative improvement (fit for use).

5.1 Fit for purpose

Formal acceptance was successful in all iterations, the decision was "accepted" with no formal reservations. Every acceptance, contrary to the research contract, took about two months, as each of the 13 institutions involved checked the results. Although the formal result was successful, there were many of questions and explanation requests, especially related to non-intuitive or unexpected numerical results. The pilot consultations with the sponsor stakeholders prior to acceptance were intensive and led to some process changes, e.g to removing steps with API queries to paid services.

Within the pilot acceptance, the sponsor formally decided that the extraction method based on regular expressions gives satisfactory results, and commissioned focus on routine iterations instead of investing in new methods of extraction. While the regex extraction functions gave the best results of the methods investigated, there is a potential for future research with further testing of AI methods of semantic document extraction.

Two complaints were made after acceptation, each concerning data discrepancy in the unit price dataset, in which the stakeholders verified selected price data in source contracts. Both complaints were justified, the problem was not in the data itself but in the dataset exported for acceptance.

5.2 Fit for use

Problems and innovation potentials were collected after each iteration, and processes were adjusted so they ran better before the next iteration. The processes presented in this paper are in the state after the last iteration.

The aim of the innovations was mainly to maximize process automation. The maximum number of routine tasks were automated (coded using node.js and linux bash scripts, and docker compose configuration). The logic of the sub-process Download, OCR and Index of the Contract acquisition process had to be adjusted due to the instability of the used opensource software for OCR of image pdf documents.

However, the automation of Extracted price validation has not been reached. Due to the high precision requirements, the verification had to be performed by human experts.

The manual validation has been improved as well. First iterations expected every Expert validator to validate in their own assigned validation tables. When finished, the tables from validators were consolidated and passed to re-validation. This practice appeared as inefficient and the validation work gradually passed to one common validation spreadsheet with tens of thousands of lines for collaborative work.

The next process design evaluation possibility is its use by another entity. The processes are passed to the research sponsor and could be used as requirements for a potential new pricelist processor.

ORCID

Tomáš Bruckner https://orcid.org/0000-0001-9120-556X
Filip Vencovský https://orcid.org/0000-0002-4963-3912

References

Aibinu, A. A. & Pasco, T. (2008). The accuracy of pre-tender building cost estimates in Australia. *Construction Management and Economics*, *26*(12), 1257–1269. https://doi.org/10.1080/01446190802527514.

Bruckner, T. (2019). Design of the technological architecture for PUMPIT project. *Journal of Systems Integration*, *10*(2), 34–40. http://www.si-journal.org/index.php/JSI/article/view/370.

Bruckner, T. & Vencovsky, F. (2020). Extracting usual service prices from public contracts. In *3rd International Conference on Advanced Research Methods and Analytics (CARMA 2020)*, (pp. 259–268). Editorial Universitat Politècnica de València. https://doi.org/10.4995/CARMA2020.2020.11645.

Czechia (1990). The Act No. 536/1990 Coll., on prices.

Czechia (2015). The Act No. 340/2015 Coll., on special conditions for the effectiveness of some contracts.

Eurostat (2008). *NACE Rev. 2 - Statistical classification of economic activities*. European Communities. https://ec.europa.eu/eurostat/web/nace-rev2.

Gao, X., Singh, M. P., & Mehra, P. (2012). Mining Business Contracts for Service Exceptions. *IEEE Transactions on Services Computing*, *5*(3), 333–344. https://doi.org/10.1109/TSC.2011.1.

Ha, H. T. (2017). Recognition of Invoices from Scanned Documents. In *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2017*, (pp. 71–78). Tribun EU.

Ha, H. T., Medved, M., Neverilova, Z., & Horak, A. (2018). Recognition of OCR Invoice Metadata Block Types. In *Text, Speech, and Dialogue. TSD 2018. Lecture Notes in Computer Science*, (pp. 304–312). Springer, Cham. https://doi.org/10.1007/978-3-030-00794-2_33.

Hevner, A. & Chatterjee, S. (2010). *Design Research in Information Systems*. Springer US. https://doi.org/10.1007/978-1-4419-5653-8.

Hevner, A., vom Brocke, J., & Maedche, A. (2019). Roles of Digital Innovation in Design Science Research. *Business & Information Systems Engineering*, *61*(1), 3–8. https://doi.org/10.1007/s12599-018-0571-z.

Kim, Y., Lee, J., Lee, E.-B., & Lee, J.-H. (2020). Application of Natural Language Processing (NLP) and Text-Mining of Big-Data to Engineering-Procurement-Construction (EPC) Bid and Contract Documents. In *2020 6th Conference on Data Science and Machine Learning Applications (CDMA)*, (pp. 123–128). IEEE. https://doi.org/10.1109/CDMA47397.2020.00027.

Ministry of Finance (2019). Register of business subjects. Ministry of Finance of the Czech Republic. https://wwwinfo.mfcr.cz/ares/ares_es.html.cz.

Ministry of Internal Affairs (2019a). Pricelist of it industry unit prices. Ministry of Internal Affairs of the Czech Republic. https://www.mvcr.cz/clanek/prehled-obvyklych-cen-ict-praci.aspx.

Ministry of Internal Affairs (2019b). Public register of contracts. Ministry of Internal Affairs of the Czech Republic. https://smlouvy.gov.cz/.

Ochrana, F. & Pavel, J. (2013). Analysis of the impact of transparency, corruption, openness in competition and tender procedures on public procurement in the Czech Republic. *Central European Journal of Public Policy*, *7*(2), 114–134.

OMG (2014). Business Process Model and Notation (BPMN), Version 2.0.2. Standard, Object Management Group. https://www.omg.org/spec/BPMN/2.0.2.

Palm, R. B., Winther, O., & Laws, F. (2017). CloudScan - A Configuration-Free Invoice Analysis System Using Recurrent Neural Networks. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, (pp. 406–413). IEEE. https://doi.org/10.1109/ICDAR.2017.74.

Skitmore, M. & Picken, D. H. (2000). The accuracy of pre-tender building price forecasts: An analysis of USA data. *Australian Institute of Quantity Surveyors Refereed Journal*, *4*(1), 33–39.

Tarawneh, A. S., Hassanat, A. B., Chetverikov, D., Lendak, I., & Verma, C. (2019). Invoice Classification Using Deep Features and Machine Learning Techniques. In *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology, JEEIT 2019 - Proceedings*, (pp. 855–859). IEEE. https://doi.org/10.1109/JEEIT.2019.8717504.

Taylor, S., Iqbal, M., & Nieves, M. (2007). *ITIL Service strategy*. Stationery Office.

The article has been reviewed.
Editorial record: First submission received on 24 August 2020. Revisions received on 1 October 2020. Accepted for publication on 1 October 2020. The editor in charge coordinating the review of this manuscript and approving it for publication was Miloslav Hub.