Using a Semantic-Based Support System for Merging Knowledge from Process Participants

Krzysztof Kluza, Mateusz Kagan, Piotr Wiśniewski, Weronika T. Adrian, Pawel Jemiolo, Anna Suchenia, Antoni Ligęza

Using a semantic-based support system for merging knowledge from process participants

Krzysztof Kluza[1] (✉), Mateusz Kagan[1], Piotr Wiśniewski[1],
Weronika T. Adrian[1], Paweł Jemioło[1], Anna Suchenia[2], Antoni Ligęza[1]

[1] AGH University of Science and Technology
al. A. Mickiewicza 30, 30-059 Krakow, Poland
kluza@agh.edu.pl
[2] Cracow University of Technology
ul. Warszawska 24, 31-155 Kraków, Poland

Abstract. High complexity of business processes is a continually growing problem in real-life organisations. Hence, modelling a workflow poses a challenge for different participants. Many methods have been proposed for automatic generation of process models. This paper aims at presenting an approach for uniting knowledge from a number of stakeholders. In essence, a collection of tabular tasks definitions are combined into one classification of unordered activities. Later, semantic analysis of the input definitions is suggested in order to fuse them based on similarity of parameters. Using a set of predefined constraints and a dedicated construction algorithm, the resulting spreadsheet-based structure can then be converted into a model of a process.

1 Introduction

Business Process Management (BPM) is an area of research that deals with processes. Such processes describe how work is carried out in an enterprise to boost overall performance [1]. The exact definition of such an improvement depends on the nature and the scope of the specific organization, but usually it involves reduction of cost, execution times or error levels. One can accomplish this by redesigning or eliminating unproductive structures, reducing redundancies and transforming the processes. Business Process Models are thus used widely in BPM strategies aiming at making the work in a company less complicated.

Every business process can be depicted in various ways, i.e. workflows or diagrams [2]. Thanks to the Business Process Model, formalization of business processes is also possible. Business Process Models can help with process enhancement, or process verification, or process stimulation [3]. Nevertheless, its main aim is to enable better understanding of the business processes for involved people. While it does not seem complicated on the surface, modelling a process requires a great deal of data about a process. Acquiring such data is a challenging task, which can be tackled using Knowledge Engineering methods.

The main goal of Knowledge Engineering (KE) is providing solutions for acquisition and representation of knowledge in the unified form [4]. Its core field of study is called Knowledge Acquisition (KA). It focuses on developing tools and methods facilitating the collection of knowledge from domain experts, as well as its later validation [5]. Knowledge Engineers often collaborate with Software Engineers in order to develop sophisticated Information Technology (IT) solutions.

Software Engineering (SE), in turn, is a discipline that refers to the production of software, beginning from the specification of the system to its maintenance [6], including all the intermediate steps. Opposite to software development, SE is concerned with solutions that are way more complex. It also tackle additional phases during the process of creation of the product comparing to software development. The holy grail of SE is to deliver products which are characterized by high quality.

Because business process can affect costs and delivery time of both services and goods, they remain a critical part of any enterprise. What is more, they can assess an organization's ability to adjust or adapt to new circumstances. It is thus essential for an organization to be aware of the mentioned dependencies. However, business processes have not long been properly understood and only recently gained attention they deserved. Henceforth, initiatives aiming at improving companies' processes have been started worldwide. One must understand that what matters most is the necessity of thorough comprehension of what exactly specific process depicts. Therefore, improving a process has to be preceded by active data collection, then, modelling it to the form of workflows and graphs. Afterwards, process analyst using methods developed by software engineers is able to shift time constraints and simplify the process. It can be achieved thanks to providing standards for interoperability [7] or generating models from text and documents [8] and from spreadsheets [9]. Because spreadsheets are used by most of the companies, this last approach remains particularly interesting. Such a method draws from the concept of Constraint Solving Problem. After transforming spreadsheet with tasks that were run within a process to a proper format, the solution is generated. Then, a Process Model is formed using methods called Model Construction or Process Mining. Such a model is encoded using XML and therefore can be easily improved in the future. Despite taking care of generating a model from previously acquired data, this concept does not tackle the need for gathering information and possible inconsistencies, i.e. contradictory or duplicated items. Such elements have to be addressed manually by business process analysts.

This paper aims at presenting an approach for business analysts which can offer assistance with the laborious task of consolidating gathered pieces of information about business processes into a uniform spreadsheet. Using such a spreadsheet, it is later possible to generate a process model. We thus reckon that our tool would enable users to save time that is needed to organize information about business processes. This paper extends our research presented and discussed during AI4KM 2019 workshop [10].

The method discussed in this paper constitutes a phase in a more complex general approach that automates process and decision model generation. This

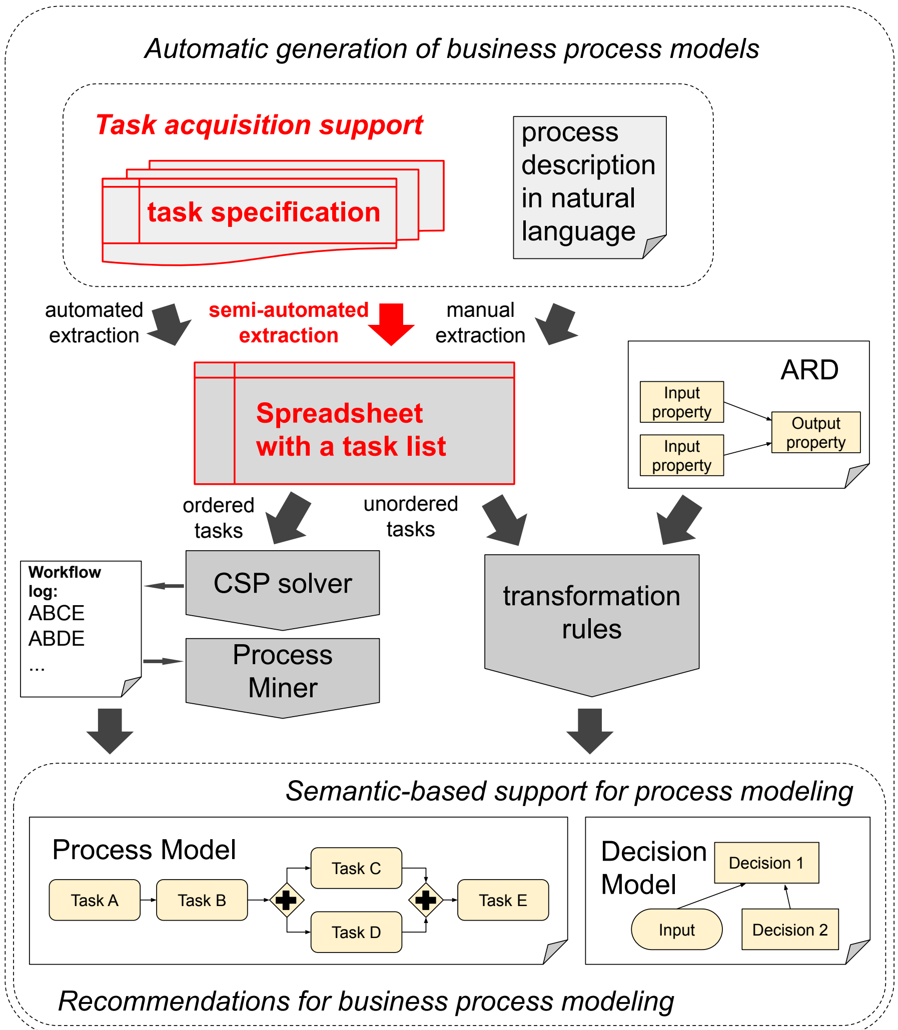


Fig. 1: Outline of the methodology proposed for the automated generation of business processes and business decisions.

general approach is presented in Figure 1, where the method discussed in this paper is highlighted (in red).

This general approach is based on a concept of process modelling with spreadsheets [11], in which a process model can be generated from a spreadsheet. In the approach, another option is to use an existing knowledge specification, like Attribute Relationship Diagrams (ARD) [12], which is a knowledge representation method for structured specification of a system. From such a specification, it is possible to generate a business process model integrated with a decision model [13].

The rest of the paper is organized as follows. In Section 2, we discuss the origins and the key concepts regarding Knowledge Acquisition. We also provide an overview of tools used to gather knowledge about business processes. Section 3 focuses on the details of the proposed method for integrating knowledge based on users specifications. Then, in Section 5, we investigate the results of the test carried out on an exemplary process data. The paper ends with section 6, where we summarize our work and outline our future ideas.

2 Knowledge Acquisition for Business Process Management

Knowledge Acquisition is concerned with study and development of methodologies and methods aiming at automating the process of knowledge collecting. It has been established as a separate research area in order to meet the demand for theories and tools supporting the development of knowledge and expert-based solutions [14]. What was stressed when the area of Knowledge Acquisition was established, was the necessity of gathering expert knowledge. Such knowledge was considered to be the basis in the fields that were hard to mimic by traditional systems. Later, it became clear that the knowledge retrieved from data structures and text compared to expertise cannot be considered the same. A deeper analysis of this inconsistency led to a conclusion that the knowledge acquired from the experts ought to be perceived differently. The extracted knowledge is thus a model of expert behaviour, not necessarily the exact knowledge which they possess.

According to the literature, the process of decision-making [15] as well as business processes [16] can be enhanced by Knowledge Acquisition. Over the years, several techniques for knowledge acquisition have emerged [17,18]. These methods include:

- **Observation.** Observation in its assumptions is probably the most simple technique for acquiring knowledge from experts. A knowledge engineer's task is to observe the behaviour of the expert and then infer what exactly they know. Although this may seem a simple way of extracting knowledge, it is also very time-consuming. What is more, it also depends on the way how the knowledge engineer perceives specific behaviours. Nevertheless, it can also be a source of diverse information that is used along with other techniques [19,20].
- **Questionnaires.** In contrast to observation, questionnaires seem to be a very efficient method in terms of time and acquired information. The way they are constructed enables experts to fill them in their spare time. Most importantly, questionnaires allow for uncovering classes of elements and its internal relations. What is more, they can sometimes be used for automatic knowledge elicitation. This approach requires structuring questionnaires in a proper way. Moreover, questionnaires may also serve as a useful tool for variability modeling for system configuration [21,22].

- **Interviews.** Interviews are another basic method enabling knowledge discovery. A knowledge engineer poses several questions about the process and an expert's task is to pass the knowledge to their investigator. Such a conversation may reveal details about the process or a particular task. Like observation, this method is very naive and it can also be time-consuming for all people involved, especially when numerous iterations of interviews are envisioned. Development of models by business analysts may be supported by using the existing domain patterns [23,24,25] or recommendation tools [26,27,28].
- **Automatic Knowledge Acquisition.** Contrary to the previously enumerated methods, this one is meant for detecting knowledge from a digital source, namely documents, reports and logs. One of its field of application is risk identification and management [29]. This way of handling data is very domain-specific. It also depends on the amount of available information. Although it might provide a neat overview of a particular domain, it may also fail to acquire knowledge when relevant data is missing. Additionally, it might take some time for a system to gather enough information in order to start knowledge elicitation. If there are existing event logs, various process mining techniques may be used [30]. However, not always complete logs are available, and sometimes even process mining requires acquiring knowledge from multiple process participants [31].

When talking about BPM purposes, Knowledge Acquisition is highly useful at an early stage of model creation. It can be applied when one wants to acquire information about the process before its modelling. Among methods incorporated by business analysts, there are all those mentioned above. Plenty of other strategies are also developed particularly for BPM initiatives, e.g. document analysis and workshops. The first one focuses on using information about processes derived from existing documentation within a company. According to the state-of-the-art paper [32], there is a variety of solutions in mining process models from natural language description, from these based on form of structured text (use cases [33], group stories [34]), to the general descriptions in natural language [35,36].

On the other hand, workshops gather knowledge engineers, process analysts, together with experts. Then, by discussion and other forms of collaboration, they develop a common Business Process Model. It can thus be considered as an extended version of interviews. What is more, triangulation techniques are often applied. It means that several methods are used in order to acquire knowledge, i.e. filling out questionnaires in between interview sessions. There are various ways of supporting collective decision making [37]. Using interactive technologies during workshops may improve participant involvement as well as improve the quality of outcomes [38]. Another important issue is when to stop the meetings as at some point the models may not be improved any further i.e. more workshops do not necessarily lead to better models [39]. To sum up, Knowledge Acquisition is key when talking about modelling of a process or other BPM tasks in general.

3 Similarity Assessment of Collected Specifications

Our method is designed to facilitate knowledge acquired from domain experts by examining the similarities between different tasks within the process. We distinguished two main requirements. First of all, the system ought to allow for the smart integration of data provided by various users. Mediation in regards to specific data is also necessary. Secondly, data provided by different users should enable the system to generate a spreadsheet with a declarative process specification. The business analyst could then use such an output for creating BPMN process model.

To meet specified requirements, we introduced a comparing tool to our systems. It enables comparison of tasks due to statistical algorithms. The algorithms calculate the rate of similarity between data entities and tasks. Additionally, they base on the semantically-oriented dictionary of English, WordNet [40]. In WordNet, concepts and their synonyms are called WordNet synsets. The definitions are connected and form a hierarchy. Some of the concepts, called root synsets or unique beginners, are very general, i.e. State, Entity and Event. Moreover, other concepts may not even have a synonym in English dictionary. NLTK[3] contains the English WordNet, with 155,287 words and 117,659 synonym sets [40].

Each part of the task is compared individually because of differences between roles and standardization of a particular task. We thus compare:

- **Data Created and Data Required**. Data Attributes and Data Names are handled separately. Firstly, to calculate the similarity between the names of the Data, synonyms of such names are acquired through WordNet. Then an algorithm finds the intersection between these two sets. Afterwards, following formula is applied:

$$1 - \frac{1.4}{(0.4x + 1.14)^{2.8}}$$

  where x represents the size of mentioned intersection. The output of this formula is a number between 0 and 1. Such a function has a shape similar to a logarithmic one. When the name is not present in the WordNet, the Levenshtein distance is applied:

$$1 - \frac{m}{l}$$

  where m is the length of the longest of the two words and the l represents the Levenshtein distance between these words.
- **Similarity between individual attributes of data** is handled using the same method that was applied to measure similarity between Data names.

[3] The Natural Language Toolkit, see `https://www.nltk.org/`.

– **Similarity between sets of entities**, i.e. data or attributes is determined as follows:

1. First, a similarity matrix is formed (see Table 1). Its columns consist of one set of entities, and rows represent the other set. Every intersecting cell comprises information about the similarity between these two units.
2. Next, the greatest value is found in such a matrix.
3. The greatest value is then added to the total similarity value.
4. Next, row and column comprising this value are removed from the matrix.
5. Steps 2-4 are repeated till the similarity matrix is empty.
6. In the end, total similarity value is divided by the size of the greater of the sets in order to calculate the similarity between the sets of entities.

	purchase	bill	funds
invoice	0.28	0.74	0.17
order	0.93	0.21	0.43
assets	0.14	0.21	0.71

Table 1: Similarity matrix. Each cell represents the ratio of similarity between the words in the column and the row.

The workflow of the described algorithm is depicted in Figure 2.

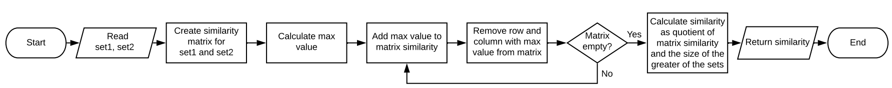


Fig. 2: Flowchart of the algorithm used for measuring the similarity between two sets of units.

– **Description**. Since it is the least structured aspect of a task, the exact way of comparing it has proved to be difficult to come up with. We thus use the naive approach, with the Cosine similarity as a metric. Firstly, each sentence is represented as a vector comprising numbers that correspond to the occurrences of each word from both sentences. Then, the angle between these vectors is calculated.

Once all partial similarity scores are determined for a pair of tasks, the result is the mean of all of those similarity scores estimated.

4 Implementation of a Merger System

The main goal of the system presented in this paper is to help a process analyst that facilitates knowledge gathered from domain experts with an analysis of similarity between different tasks within the process. The system should enable smart merging of the data provided by different users, as well as mediation with respect to specific data. The result should be a declarative process specification in a form of a spreadsheet based on data gathered from different users that could be used as a base for BPMN process model generation. Figure 3 presents a set of use cases for the designed system.

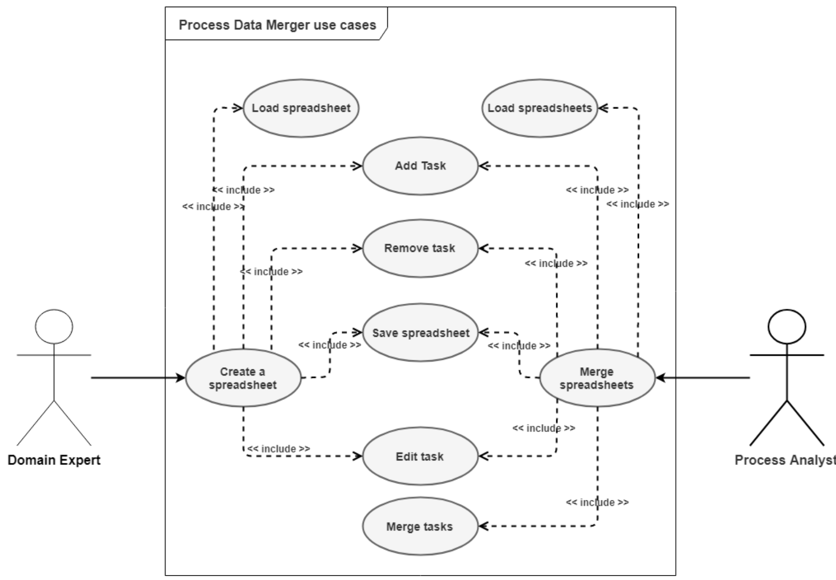


Fig. 3: Use case diagram for the merger system.

4.1 Merger system architecture

The system is constructed in a client-server architecture where client is designed to operate independently relying on server only for advanced computing. Figure 4 shows an overview of the system and its main components:.

- Spreadsheet Creator is used for creating a spreadsheet.
- Spreadsheet Merger is used for merging previously created spreadsheets.
- Spreadsheet Validator offers service for spreadsheet validation in terms of task duplications and data entities misuse.
- Spreadsheet Merger Assistance offers a web service that searches for duplications of a specific task.
- Task Comparator is used to calculate similarity index between two tasks.
- Wordnet API is a module that queries information from the Wordnet.

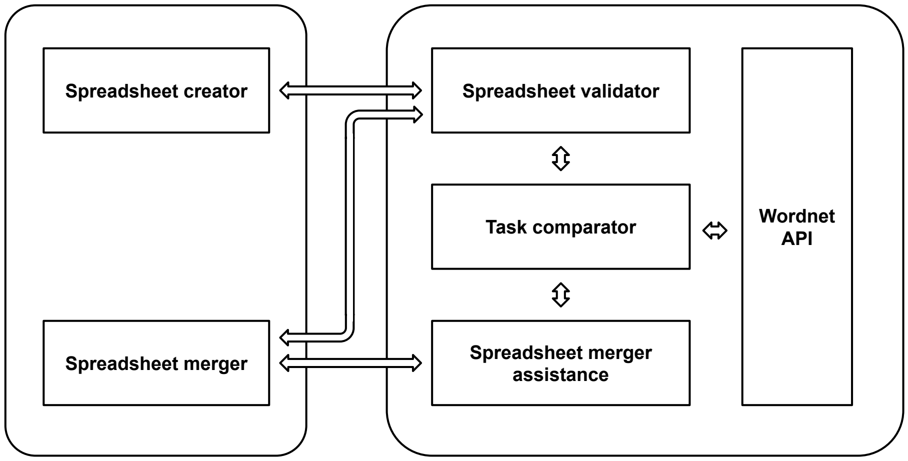
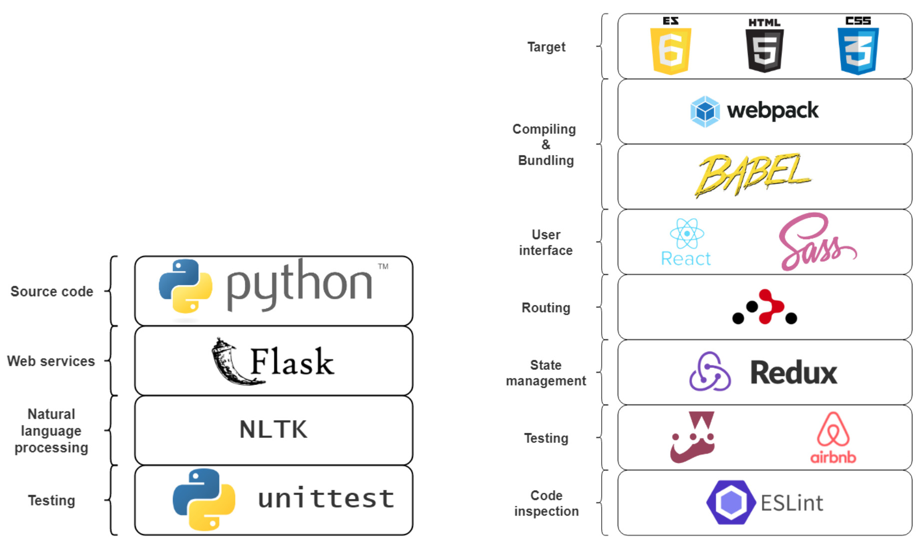


Fig. 4: Overview of the systems architecture.



(a) Server side technology stack.

(b) Client side technology stack.

Fig. 5: Technology stack.

4.2 Technology stack

The overview of the technology stack is presented in Figure 5. The technology stack for the client application consists mostly of JavaScript libraries in the Node.js environment as well as as tools for code inspection and code compiling. The technology stack of the server-side of the system consists of several modules in Python environment that create API for more advanced calculations.

4.3 User interface

The application contains a web user interface which can work in two modes. In the "Creator Mode" (as seen in Figure 6), the application contains only a list of tasks which can be edited and validated. In the "Merger Mode" (as presented in Figure 7), the interface is split into two parts, each consisting of a single list of tasks. The list on the left-hand side is the main list of tasks, and it can be saved into a `.csv` file. The secondary list, on the right hand side, serves as a buffer for all tasks collected from domain experts. The tasks from both lists can be moved to the other one, highlighted, deleted or checked for duplicates in both lists. A user can also edit any task from both lists.

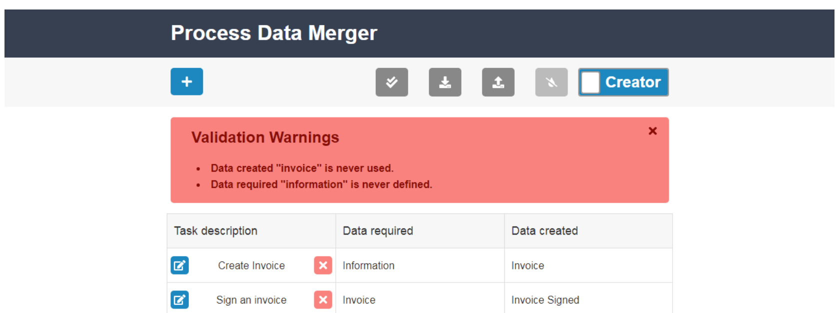


Fig. 6: Application View (Creator Mode).

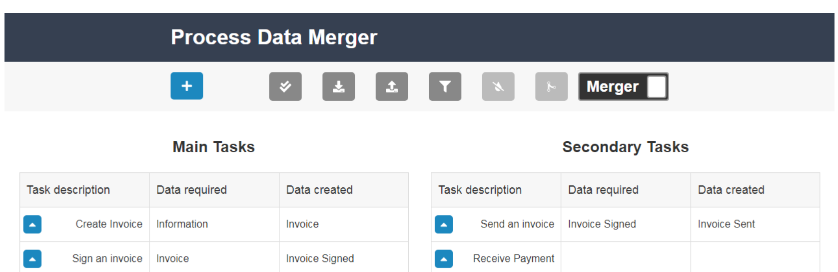


Fig. 7: Application View (Merger Mode).

5 Case Study

To evaluate our approach, we take into account a supply process case study. Such a process is realized by four persons, namely: purchasing specialist, warehouse operator, purchasing manager and accounts payable specialist. In this example, we consider two different accounts payable specialists and therefore have five spreadsheets as input.

Task description	**Data required**	**Data created**
Reserve funds	Order Reviewed	Funds Reserved
Receive Invoice	Invoice	Invoice
Record Invoice	Invoice	Invoice Recorded
Release funds	Invoice Recorded, Packing slip Recorded	Funds Released

Table 2: Activities performed by the first accounts payable specialist.

Analysis of the first of the accounts payable specialists (see Table 2) with similarity threshold set to 65% showed that some of the data created and required are used only once. What is more, the tasks "Receive Invoice" and "Record Invoice" are probably too similar. Data required and data created are most often identical. Only slight differences can be observed sometimes. Being in such a situation process analyst might contact a domain expert to establish why is the spreadsheet constructed in this way. Additionally, it can be also profitable to inspect other spreadsheets to find a solution.

Task description	**Data required**	**Data created**
Issue payment	Funds Discharged	Order Completed
Report Invoice	Invoice	Invoice Reported
Discharge funds	Invoice Reported, Packing-slip Reported	Funds Discharged
Receive invoice	Order Sent	Invoice

Table 3: Activities performed by the second accounts payable specialist.

Following this, we can use the second accounts payable specialist's spreadsheet (see Table 3) to resolve the mentioned issues. First, we can start by looking for duplicates of tasks "Receive invoice" in both spreadsheets. Again we set the similarity threshold to 65%. As an output, we get "Report Invoice" and "Receive Invoice". Because the tasks duplicate themselves, we can now merge them into one. By doing so, we remove redundancy. "Invoice" is set for data created and "Order created" for data required. Then we look for unique tasks within both spreadsheets. Such a task is "Issue Payment", and we move it to the main table. Next, we check if some tasks are similar to the task "Report invoice" from the

Task description	Data required	Data created
Review Order	Order Created	Order Reviewed
Send Order	Funds Reserved	Order Sent

Table 4: Activities performed by the purchasing manager.

Task description	Data required	Data created
Create Order	Inventory Checked	Order Created
Reprocess Order	Order Reviewed	Order Reprocessed

Table 5: Activities performed by the purchasing specialist.

second spreadsheet. In such a case, there is one task that can be considered similar. It is called "Record Invoice" in the first spreadsheet. These tasks are very likely to be the same, different names being the only difference. An analogous situation takes places when talking about tasks "Discharge funds" and "Release funds". However, we still do not know which version of the tasks is used by other people in their spreadsheets. We thus leave them all and will decide about it later.

Task description	Data required	Data created
Check Inventory	Goods-request	Inventory Checked
Receive Packing Slip	Order Sent	Packing Slip
Record Packing Slip	Packing-slip	Packing Slip

Table 6: Activities performed by the warehouse operator.

Afterwards, we take into account the spreadsheet created by the purchasing specialist (see Table 5). All of the tasks are unique. We can thus move them into the main table. Next, we consider tasks in a spreadsheet from a purchasing manager (see Table 4). Once again, we can move all of them into the main table because they are all unique.

Eventually, we analyse a spreadsheet generated by the warehouse operator (see Table 6). Firstly, we establish that tasks "Receive Packing-slip" and "Check inventory" are unique. We thus move them into the main table. The task "Record Packing-slip" is unique too. However, both data required and created are the same. Therefore, it seems that something is missing in the spreadsheet. To tackle this issue, we have to look at specific tasks. We establish that two tasks use "Packing-slip" as their input, namely "Discharge funds" and "Release funds". In our example, data created by the "Discharge funds" is used whereas the output from "Release funds" is not. We can, therefore, move "Discharge funds" to the main table and remove the other task. Now, we can perform two fixes. Firstly, the tasks "Record packing-slip" are set to create data entity "Packing-slip reported". Secondly, we rename "Packing-slip reported" to "Report packing-slip" and move

Task description	Data required	Data created
Reserve funds	Order Reviewed	Funds Reserved
Issue payment	Funds Discharged	Order Completed
Report invoice	Invoice	Invoice Reported
Discharge funds	Invoice Reported, Packing-slip Reported	Funds Discharged
Receive invoice	Order Sent	Invoice
Review Order	Order Created	Order Reviewed
Send Order	Funds Reserved	Order Sent
Create Order	Inventory Checked	Order Created
Reprocess Order	Order Reviewed	Order Reprocessed
Check inventory	Goods-request	Inventory Checked
Receive packing slip	Order Sent	Packing-slip
Report packing slip	Packing-slip	Packing-slip Reported

Fig. 8: Final version of the spreadsheet.

it to the main table. Additionally, we move the task "Report invoice" into the main table.

As a general note, one can observe that three units are not used, namely: "Order Completed", "Goods-request" and "Order Reprocessed". First two of them are most probably data received and data created of the entire process. On the other hand, the third entity ought to be consulted with an expert to establish its connection to the whole process. Final version of merged spreadsheets is shown in Figure 8.

At the very end, we present a BPMN model generated using this final spreadsheet (see Figure 9). We base on the method proposed in [41]. The output of the "Reprocess Order" task should probably be connected back via the exclusive gateway to the "Review Order" task. Though it must be mentioned here, the issue is relatively minor. It can be easily fixed using automated repairing methods [42] or directly by process analyst.

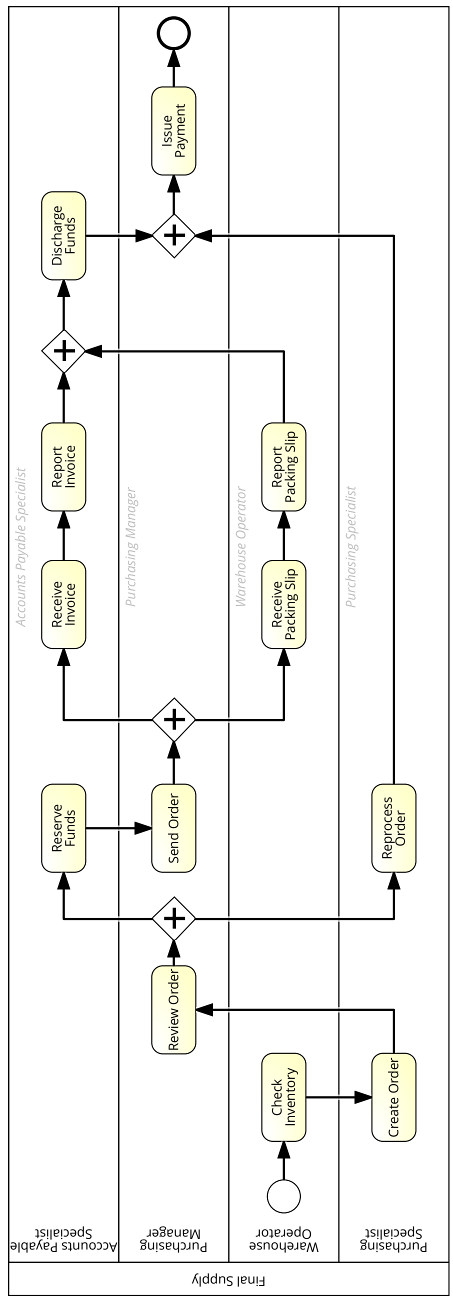


Fig. 9: Model of the supply case generated using our approach.

6 Concluding Remarks and Future Works

In this paper, we presented a system which can support the work of a business analyst. We believe that our method can be considered highly beneficial thanks to the ability to generate spreadsheets containing process knowledge. We presented results using case study. We also pointed out gaps in this intelligent tool as it sometimes lacks the insight of the domain expert and a process analysts. Though the structure of the proposed model is not fully accurate, it can be used as a basis for generating the final solution.

In the future, we want to focus on improving similarity calculation by using machine learning models. Such an approach requires preparing training sets based on data collected from the systems.

We also intend to extend the number of BPMN elements that are supported. Besides, support for real-time collaboration on one single instance of the spreadsheet is required for practical application. It could also be beneficial in terms of time needed to resolve inconsistencies [43] about the input and output of tasks specified by various business experts.

References

1. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A., et al.: Fundamentals of Business Process Management. Volume 1. Springer (2013)
2. Vaisman, A.: An introduction to business process modeling. In: European Business Intelligence Summer School, Springer (2012) 29–61
3. Indulska, M., Green, P., Recker, J., Rosemann, M.: Business process modeling: Perceived benefits. In Laender, A.H.F., Castano, S., Dayal, U., Casati, F., de Oliveira, J.P.M., eds.: Conceptual Modeling - ER 2009, Berlin, Heidelberg, Springer Berlin Heidelberg (2009) 458–471
4. Darai, D., Singh, S., Biswas, S.: Knowledge engineering-an overview. International Journal of Computer Science and Information Technologies **1**(4) (2010) 230–234
5. Kendal, S., Creen, M.: An introduction to knowledge engineering. Springer (2007)
6. Sommerville, I.: Software Engineering. International Computer Science Series. Pearson (2011)
7. Ganzha, M., Paprzycki, M., Pawłowski, W., Szmeja, P., Wasielewska, K.: Towards semantic interoperability between internet of things platforms. In: Integration, interconnection, and interoperability of IoT systems. Springer (2018) 103–127
8. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: International Conference on Advanced Information Systems Engineering, Springer (2011) 482–496
9. Kluza, K., Wiśniewski, P.: Spreadsheet-based business process modeling. In: 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE (2016) 1355–1358
10. Kluza, K., Kagan, M., Wisniewski, P., Adrian, W.T., Ligeza, A.: Semantic-based support system for merging process knowledge from users. In Mercier-Laurent, E., Owoc, M.L., Ritter, W., eds.: AI4KM 2019: 7th International Workshop on Artificial Intelligence for Knowledge Management: August 11, 2019, Macao, China. (2019) 56–61

11. Wiśniewski, P., Kluza, K., Kucharska, E., Ligęza, A.: Spreadsheets as interoperability solution for business process representation. Applied Sciences **9**(2) (2019) 345
12. Nalepa, G.J., Wojnicki, I.: Towards formalization of ARD+ conceptual design and refinement method. In Wilson, D.C., Lane, H.C., eds.: FLAIRS-21: Proceedings of the twenty-first international Florida Artificial Intelligence Research Society conference: 15–17 May 2008, Coconut Grove, Florida, USA, Menlo Park, California, AAAI Press (2008) 353–358
13. Kluza, K., Wiśniewski, P., Adrian, W.T., Ligęza, A.: From attribute relationship diagrams to process (BPMN) and decision (DMN) models. In: International Conference on Knowledge Science, Engineering and Management, Springer (2019) 615–627
14. Shaw, M.L.G., Gaines, B.R.: Requirements acquisition. Software Engineering Journal **11** (1996) 149–165
15. Owoc, M.L., et al.: Benefits of knowledge acquisition systems for management. an empirical study. In: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE (2015) 1691–1698
16. Honkisz, K., Kluza, K., Wiśniewski, P.: A concept for generating business process models from natural language description. In: International Conference on Knowledge Science, Engineering and Management, Springer (2018) 91–103
17. Welbank, M.: An overview of knowledge acquisition methods. Interacting with computers **2**(1) (1990) 83–91
18. Olson, J.R., Rueter, H.H.: Extracting expertise from experts: Methods for knowledge acquisition. Expert systems **4**(3) (1987) 152–168
19. White, G.R., Cicmil, S.: Knowledge acquisition through process mapping. International Journal of Productivity and Performance Management (2016)
20. Silva, I., Sousa, P., Guerreiro, S.: Business processes compliance in partially observable environments. Technical report, EasyChair (2019)
21. La Rosa, M., Dumas, M., Ter Hofstede, A.H., Mendling, J.: Configurable multi-perspective business process models. Information Systems **36**(2) (2011) 313–340
22. La Rosa, M., van der Aalst, W.M., Dumas, M., Ter Hofstede, A.H.: Questionnaire-based variability modeling for system configuration. Software & Systems Modeling **8**(2) (2009) 251–274
23. Koschmider, A., Reijers, H.A.: Improving the process of process modelling by the use of domain process patterns. Enterprise Information Systems **9**(1) (2015) 29–57
24. Makni, L., Haddar, N.Z., Ben-Abdallah, H.: An automated method for the construction of semantic business process patterns. International Journal of Process Management and Benchmarking **8**(3) (2018) 263–290
25. Fellmann, M., Delfmann, P., Koschmider, A., Laue, R., Leopold, H., Schoknecht, A.: Semantic technology in business process modeling and analysis. part 2: Domain patterns and (semantic) process model elicitation. In: EMISA Forum. Volume 35. (2015) 12–23
26. Koschmider, A., Hornung, T., Oberweis, A.: Recommendation-based editor for business process modeling. Data & Knowledge Engineering **70**(6) (2011) 483–503
27. Deng, S., Wang, D., Li, Y., Cao, B., Yin, J., Wu, Z., Zhou, M.: A recommendation system to facilitate business process modeling. IEEE transactions on cybernetics **47**(6) (2016) 1380–1394
28. Bobek, S., Baran, M., Kluza, K., Nalepa, G.J.: Application of bayesian networks to recommendations in business process modeling. In: AIBP@ AI* IA. (2013) 41–50

29. Mercier-Laurent, E.: Knowledge management &risk management. In: 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE (2016) 1369–1373
30. van der Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer Berlin Heidelberg (2011)
31. Ivanchikj, A., Pautasso, C.: Sketching process models by mining participant stories. In: International Conference on Business Process Management, Springer (2019) 3–19
32. Riefer, M., Ternis, S.F., Thaler, T.: Mining process models from natural language text: A state-of-the-art analysis. Multikonferenz Wirtschaftsinformatik (MKWI-16), March (2016) 9–11
33. Sinha, A., Paradkar, A.: Use cases to process specifications in business process modeling notation. In: 2010 IEEE International Conference on Web Services, IEEE (2010) 473–480
34. de AR Goncalves, J.C., Santoro, F.M., Baiao, F.A.: Business process mining from group stories. In: 2009 13th International Conference on Computer Supported Cooperative Work in Design, IEEE (2009) 161–166
35. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: International Conference on Advanced Information Systems Engineering, Springer (2011) 482–496
36. van der Aa, H., Di Ciccio, C., Leopold, H., Reijers, H.A.: Extracting declarative process models from natural language. In: International Conference on Advanced Information Systems Engineering, Springer (2019) 365–382
37. Kucharska, E., Grobler-Dębska, K., Klimek, R.: Collective decision making in dynamic vehicle routing problem. In: MATEC Web of Conferences. Volume 252., EDP Sciences (2019) 03003
38. Nolte, A., Brown, R., Anslow, C., Wiechers, M., Polyvyanyy, A., Herrmann, T.: Collaborative business process modeling in multi-surface environments. In: Collaboration Meets Interactive Spaces. Springer (2016) 259–286
39. Chounta, I.A., Nolte, A., Hecking, T., Farzan, R., Herrmann, T.: When to say" enough is enough!" a study on the evolution of collaboratively created process models. Proceedings of the ACM on Human-Computer Interaction **1**(CSCW) (2017) 1–21
40. Bird, S.: Nltk-lite: Efficient scripting for natural language processing. In: Proceedings of the 4th International Conference on Natural Language Processing (ICON). (2005) 11–18
41. Wiśniewski, P., Kluza, K., Ligęza, A.: An approach to participatory business process modeling: BPMN model generation using Constraint Programming and graph composition. Applied Sciences **8**(9) (2018) 1428
42. Cervantes, A.A., van Beest, N.R., La Rosa, M., Dumas, M., García-Bañuelos, L.: Interactive and incremental business process model repair. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer (2017) 53–74
43. Adrian, W.T., Ligęza, A., Nalepa, G.J.: Inconsistency handling in collaborative knowledge management. In: 2013 Federated Conference on Computer Science and Information Systems, IEEE (2013) 1233–1238