An Impact of the User and Time Parameters to Sequence Alignment Methods for Process Mining

Jakub Štolfa, Svatopluk Štolfa, Kateřina Slaninová, Jan Martinovič, Václav Snášel

An Impact of the User and Time Parameters to Sequence Alignment Methods for Process Mining

Jakub Štolfa[1], Svatopluk Štolfa[1], Kateřina Slaninová[1,2], Jan Martinovič[1,2] and Václav Snášel[1,2]

[1] Department of Computer Science, FEECS, VŠB - Technical University of Ostrava, 17. listopadu 15, 708 33, Ostrava-Poruba, Czech Republic
[2] IT4Innovations, VŠB - Technical University of Ostrava, 17. listopadu 15, 708 33, Ostrava-Poruba, Czech Republic
{jakub.stolfa, svatopluk.stolfa, katerina.slaninova, jan.martinovic, vaclav.snasel}@vsb.cz

Abstract. Process mining is relatively new domain that opens many opportunities for process control and improvement. Anyway, the basis of the process mining is the examination of bunch of data from processes. There are many methods that already had been used in this domain and many other are still waiting for the discovery of their benefits. One of the main issues is to find out whether the new method is useful or not. The main purpose of this paper is to present the usability of sequence alignment method in process mining especially from the user and time perspective.

Keywords: Sequence Alignment Methods, Process Mining

1 Introduction

Process mining is the examination of the data from processes. Data are taken especially from systems that support business processes. Processes are not sequential and the instances can follow different prescribed ways of process description, same activities are performed by different users, lasting of the activities varies etc. Therefore, there are many different types of information that can be logged by the supporting system. These logs are then used for the examination of process behavior, user behavior, entities behavior and many other ways. Another methods allows us to obtain result that can help to study the anomalies in process execution, social nets, timing aspects of activities and processes etc.

Our intention is to use and adopt sequence alignment methods as one of the tools that can be used to study the data from the processes. Since the sequence alignment methods work with the sequences, the issue is then to try preparing the proper sequence. The sequence must be prepared from the perspective of intended view of the process. Relatively easy task is to study the topology of the process. In that case, the sequence consists from the activities that are performed

by the process. In case that we would like to study other type of information, like to involve the time and user information, we have to prepare a sequence that obtains such type of data.

Ideas in this paper are based on our previous work that included process reconstruction and path analysis [15] and analysis of the process data that involved usage of the sequence alignment methods [7]. The approach was tested and used in other research areas, for example in e-learning area [9], in analysis of behavior of agents during the simulation [11] and in analysis of user behavior on the web [10].

The research follows our paper [6], where we discussed the usage of sequence alignment methods in the area of process mining. We have described four types of the sequences that allowed us to form four different points of view of the particular process. The aim of the current paper is to describe and explain the meaning of results of the similarities defined by the sequence alignment methods for the different types of sequences. We would like to prove that the proper creation of the sequences with this type of information - topology of the process, duration of the process, users involved in the process can be useful for the study of process instances and bring us new way how to study the process instances data.

The paper is organized as follows: Section 2 introduces the state of the art; Section 3 describes the analyzed process that was used for the experiments, Section 4 depicts the preparation and data structure, Section 5 presents the experiment that we have performed, shows the usage of our process mining method and explains obtained results; concluding Section 6 provides a summary and discusses the planned future research.

2 State of the art

Business process definitions are sometimes quite complex and allow many variations. All of these variations are then implemented to supportive systems. If you want to follow some business process in a system, you have many decisions and process is sometimes lost in variations. Modeling and simulations can help you to adjust the process, find weaknesses and bottlenecks during the design phase of the process.

The idea of process mining was introduced by Aalst in 2004 [13, 12]. This area of the research has been developing during the years, lot of methods were introduced to this topic. In 2005, ProM tool was introduced [14]. ProM aggregates methods and approaches in this area of study. There are a lot of papers that describe new ways or improvements of methods, techniques and algorithms used in the process mining, but only several papers are focused on the case studies [1].

In the area of process mining, the methods of the sequence alignment were introduced by Esign and Karagoz [2] in 2013. Focus of their work was quantitative approach for performing process diagnostics. The approach uses sequence alignment methods for delta analysis. It is comparison of actually performed process and prescriptive reference model [13]. Our paper provides another usage

of the sequence alignment methods. We use these methods for comparison of extracted processes to find similarity in the process executions, i.e. some patterns of the process.

The basic approach to the comparison of two sequences, where the order of elements is important, is The longest common substring method (LCS). This is used in exact matching problems [4]. It is obvious from the name of the method that its main principle is to find the length of the common longest substring. The LCS method respects the order of elements within a sequence. However, the main disadvantage of this method is that it can only find the identical subsequences, which meet the characteristics of substrings.

Unlike substrings, the objects in a subsequence might be intermingled with other objects that are not in the sequence. The longest common subsequence method (LCSS) allows us to find the common subsequence [5]. Contrary to the LCS method, the LCSS method allows (or ignores) these extra elements in the sequence and, therefore, it is immune to slight distortions.

The important method is The time-warped longest common subsequence (TWLCS) [3]. This method combines the advantages of the LCSS method with dynamic time warping [8]. Dynamic time warping is used for finding the optimal visualization of elements in two sequences to match them as much as possible. This method is immune to minor distortions and to time non-linearity. It is able to compare sequences, which are for standard metrics, evidently not comparable.

The methods LCS and LCSS used for the comparison of sequences find the longest common subsequence z of compared sequences x and y, where $(z \subseteq x) \wedge (z \subseteq y)$. The relation weight $w_{seq}(x, y)$ between the sequences x and y was counted by Equation 1:

$$w_{seq}(x, y) = \frac{l(z)^2}{l(x)l(y)} \frac{Min(l(x), l(y))^2}{Max(l(x), l(y))^2}, \tag{1}$$

where $l(x)$ and $l(y)$ are lengths of the compared sequences x and y, and $l(z)$ is a length of a subsequence z. Equation 1 takes account of the possible difference between $l(x)$ and $l(y)$. Due to this reason, z is adapted so that $w_{seq}(x, y)$ is strengthened in the case of similar lengths of sequences x and y, and analogically weakened in the case of higher difference of $l(x)$ and $l(y)$. For the methods LCS and LCSS, w_{seq} meets all the similarity conditions: $w_{seq} \geq 0$, $w_{seq}(x, x) = 1$, $w_{seq}(x, x) > w_{seq}(x, y)$ and $w_{seq}(x, y) = w_{seq}(y, x)$.

The output z is only the sequence which characterizes the relation between the sequences x and y for T-WLCS method. Therefore, $w_{seq}(x, y)$ does not meet all the similarity conditions due to its characteristics. Respectively, it is possible that $w_{seq}(x, y) > w_{seq}(x, x)$. Although we know that $w_{seq}(x, y)$ is not a similarity for T-WLCS method, due to a simplification, the 'sequence similarity' will be used as a relation weight $w_{seq}(x, y)$ between the sequences x and y for all the methods of sequence comparison in the following text.

3 Process Context

We used data logs of the SAP system for running of testing examples. Current SAP system runs in the company that operates in five European countries. We chose business process of the invoice verification that is implemented in SAP system, user activities are controlled by SAP workflow system. Users participate in the invoice verification workflow in several different roles (creator, accountant completion, approver, and accountant decision and posting). Generally, it is process in which the accountant should create the invoice, verify it, send to the approvers and finally, when he gets it back he does invoice posting.

We have loaded the log of the process between 1/1/2012 and 6/30/2012, totally we loaded 70,855 records for adjusting. Detailed description of the obtaining log and data preprocessing is described in our previous work [15]. We know that the log contains data from the three factories of the company. That means in the results we can also focus on that if the people, or users, cooperate or if the factories are separated.

We know the architecture of the process model because user activities in the SAP are controlled by SAP business workflow. It means that process execution should follow the process model. On the other hand, we can find out some deviations. This model is depictured in the Fig. 3. Process starts with event Creation. Next one is Verification. These two events Creation and Verification can be done repeatedly. Approval event can be done repeatedly too. If the invoice is not acknowledged, then process goes from the beginning. Last events are Posting.

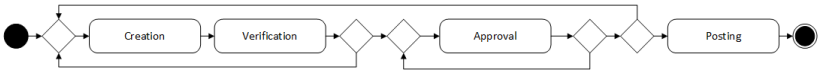


Fig. 1. Process Model

4 Data Preparation and Sequences Types

The main purpose of this paper is to prove that sequence alignment methods are useful for finding the similarity between the sequences with user and time parameters. We would like to prove that adding the parameters to the sequences has an appropriate impact to the results of these methods and can be used for the further analysis.

Like it was written in the introduction, our paper extends the finding in our previous paper in this area [6]. We have set up four sequence types. The sequence types represent four points of view that we are able to see according to the data in the log. These four types are:

– Type A - Events without Time and Users,

– Type B - Events with Time and without Users,
– Type C - Events without Time and with Users,
– Type D - Events with Time and Users.

We have solved the task about event duration. We have set up three categories of the duration of the events:

1. Category 1 - If the event lasts less than 32 hours it fits to the first category
2. Category 2 - If the event lasts more than 32 hours and less than 168 hours it fits to the second category
3. Category 3 - If the event lasts more than 168 hours it fits to the third category

We set up aliases for the type of the events in the sequence. It means that Verification event is in the sequence like V, Creation event is C, Approval is A, and Posting is P. Sequence type A focuses on the topological structure of the process only. Sequence type B focuses on the topological structure of the process and combines it with the information about the duration of the events. Tagging of the duration of the events is made by repeating the symbol of the event in the sequence. We use string comparison methods and thus we need to transform the duration to the strings. Repetition of the symbol depends on the duration category of the particular event. First category is represented by one single symbol, second category by two same symbols, and third category by three same symbols. Sequence type C combines the topological structure of the process and meta-information about the users. Sequence type D combines all possible views to the process - topological, time view and users view.

5 Experiments and Results

This section presents performed experiments and obtained results. Experiments were run on the data from the process that was described in the Section 3. The main purpose of these experiments is to analyze if there is any impact of adding user, time or user and time parameter to the sequence. We have made experiments for each sequence type - A, B, C and D.

Consideration of the useability of particular sequence alignment method for analyzed data collection was discussed in [6]. After several tests we have found that each sequence type requires different sequence alignment method. This finding follows the mentioned paper. We have more data in the experiments described in this paper. Therefore, we have decided that LCS, or LCSS method is more usable than T-WLCS method for this type of business process. The LCS method do not accept any differences in the middle of the sequence, that means that really similar sequences fits together only. This might be more usable for the business process described by sequence type A in our experiments. If we add other attributes like user and time (or both) then LCSS method is more suitable for the definition of sequence similarity.

Table 1. Sequence Type A: Result Ordered by Occurrence

Case Type	Occurrence	Weighted Degree
C;V;A;A;S;	3788	9.79
C;V;A;A;A;S;	2952	27.03
C;V;A;S;	2007	6.51
C;V;A;A;A;A;S;	587	16.50
C;V;A;A;A;A;A;S;	209	18.57
C;V;C;V;A;A;A;S;	147	19.01
C;V;A;S;A;A;A;S;	142	18.93
C;V;C;V;A;A;S;	116	15.47
C;V;A;A;S;A;S;	114	15.78
C;V;A;S;A;A;S;	90	14.88

5.1 Sequence Type A

We can see the most used case types in the examined process for sequence type A in the Table 1. Case type CVAAS has the most occurrences. It means that the most invoices in the reality went approved twice. The third one shows us that 2007 cases went through only one approving, so the invoices might not be properly checked. The four eyes company rule is broken here. The four eyes rule means that the invoice is checked at least by two persons.

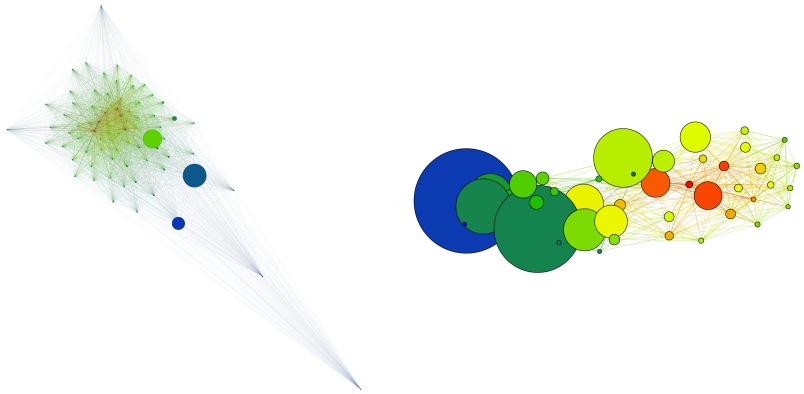
Other information is about weighted degree. This information is obtained using network analysis. The network (weighted undirected graph $G(V, E, w)$) was created by case types as nodes V, while the relation weight w represents the similarity between the case types. Weighted degree is the sum of the weights of the edges for the particular node. That tells us how much the current node is similar to the other nodes. If the weighted degree is higher then the node is more similar to at least some of its neighbor nodes.

The case type CVAAS had the most occurrences, CVAAAS had the second biggest number, and CVAS had the third biggest number of occurrences. We can see that other nodes that have deviation in meaning of number of the occurrences have a relatively high weighted degree. They have edges between each other with the high weight and create one cluster, which can be seen in Figure 2(a). We can see that these nodes are deviant from the biggest nodes (most used case types) in meaning of the number of the occurrences and also in meaning of the similarity and distance from the most used case types.

Weighted degree data range was from 0.62 to 62.42 for the LCS method. We can see that the case types with the most occurrences have relatively small weighted degree according the range of the weighted degree. It tells us that the most used case types are relatively out of the other nodes. It means that the other nodes, we can say deviation nodes in meaning of number of the occurrences, are quite a lot different from the nodes with the most occurrences (according to the chosen sequence alignment method and its settings). From the business process view, this information says that if some case type is deviation then the difference is quite big from the most used case types of the process. On the other hand, we

have to look at it in a complex way in correlation with the particular graph or in correlation with the deep data examination. We can see whether the examined case type is really out of the other nodes, or if the examined case type has a lot of edges with small weight, or that it has only one or small number of edges with high weight. For example, if the node can have weighted degree 9 then we can think that this node is out of the other nodes. But but we have to examine if that node has 90 edges with weight 1, or only 5 edges with weight 18. The result can tell us completely other information. There are two issues that have two different meanings and have to be examined. In the first way, it is node out of the other nodes, and in the second way, it is a node that has some quite similar nodes.

Figure 2 shows the examples of undirected weighted sequence graph $G(V, E, w)$. The nodes V represent sequences (cases), the ties E represent relations between them. Weight w is determined by similarity between the sequences set by method for sequence alignment, LCS. As we can see, Figure 2(a) for the sequence type A shows that the most used case types are really out of the other case types which create one evident cluster. The graph was made by force atlas decomposition. The smallest node represents the lowest occurrence of the case type, the biggest one represents the higher occurrence. Node with minimal weighted degree has the blue color, then it continues through green, yellow and the maximal degree is colored by the red color.



(a) Sequence Type A (b) Sequence Type B

Fig. 2. Sequence Graphs

Next experiments with case types B, C, and D deal with the addition of time and user parameters to the sequence. We have performed the experiments for all case types from the previous experiment, but the results were not clearly visible for their visualization and study them from the overall overview. Therefore, we show the impact of the time and user parameters addition on the selected case

type only. We have selected case type CVAAS from the previous example; this case type has the highest occurrence in the log.

5.2 Sequence Type B

Table 2. Sequence Type B: Result Ordered by Occurrence

Case Type	Occurrence	Weighted Degree
C;V;A;A;S;	573	2.99
C;V;A;A;A;S;	474	4.66
C;V;A;A;A;A;S;	312	9.25
C;V;A;A;S;S;	291	4.70
C;C;V;A;A;S;	222	5.14
C;V;A;A;A;S;S;	217	8.00
C;C;V;A;A;A;S;	203	10.52
C;C;V;A;A;A;S;S;	164	10.45
C;V;A;A;A;A;A;S;	150	10.08
C;C;V;A;A;A;A;S;	141	13.02
...	...	...

Table 2 shows case types ordered according to their occurrence for sequence type B. These case types were created by addition of time parameter and were derived from the selected case type CVAAS in the sequence type A. Addition of this parameter caused a decomposition of the case type CVAAS to other 43 case types. Case type with the most occurrences is the CVAAS. The time parameter was added with following rules: if the event lasts less then 32 hours then it is repeated in the sequence once, 30-168 twice, more then 168 three times. Taking into account the rules for the time parameter we can see that the most used case type was the one with the shortest execution time. The second one was the sequence type CVAAAS that had 474 occurrences. This case type shows us that 474 cases were executed by the same way. This type of the sequence may be the case with the three approvals, and also it could be the case type with one approval that takes a lot of time. Anyway, according to the time parameter, these cases are signed as the same case type. This information can be easily obtained form the statistical analysis as well, but our methods can be seen as a different view of this data with possibility to easily set up adjusting parameters.

Figure 2(b) shows the distribution of the sequences from the time-sequence view. We can see several main nodes according to their occurrence. This graph shows mainly the distribution of the sequences according to their duration.

5.3 Sequence Type C

Table 3 depicts case types ordered by occurrence for the sequence type C. This case type reflects the topology of the process and user parameter. Case type

Table 3. Sequence Type C: Result Ordered by Occurrence

Case Type	Occurrence	Weighted Degree
C_U260;V_U260;A_U074;A_U202;S_U260;	761	3.36
C_U068;V_U068;A_U074;A_U202;S_U068;	298	2.64
C_U068;V_U068;A_U249;A_U192;S_U068;	254	3.00
C_U162;V_U162;A_U074;A_U202;S_U162;	174	0.64
C_U068;V_U068;A_U227;A_U202;S_U068;	145	0.00
C_U040;V_U040;A_U102;A_U030;S_U040;	124	1.08
C_U087;V_U087;A_U126;A_U124;S_U087;	117	2.08
C_U110;V_U110;A_U114;A_U043;S_U200;	91	2.44
C_U260;V_U260;A_U249;A_U192;S_U260;	89	1.72
C_U178;V_U178;A_U249;A_U192;S_U178;	82	2.36
...	...	...

CVAAS of the sequence type A by the addition of the user parameter has been augmented to 217 case types for the sequence type C. We can see that the most used case type is that one, where the user260 starts the process and ends the process as well. Mostly used user in the role approval is user074 followed by user202. In the perspective with the user parameter, we are able to see what user is the most frequented, as well as the process connection between the users. From this point of view, we can study for example whether the users do mostly only simple invoices where the process is easy, whether the user that creates the invoice occurs in more deviations (this can mean that there can be problem with the knowledge to whom the invoice should be send to approve it), etc.

Figure 3(a) shows the distribution of the sequences from the user-sequence view. We can see several main nodes. These nodes are made around the same user for C, V and S activities. These activities are mainly performed by the same user. For example the node (case type) in the graph down right has the main user U087, the node on the top has main user U068, and we can see there the case types with the second and the third highest occurrence. Case types around the user U260 are in the middle of the graph. We also can see that one cluster of nodes is separated (top right). We have analyzed the sequences in this cluster and had found out that this cluster contains the sequences from one factory only. Since we have used data from three factories of the current company, we can see that other two factories probably cooperate on the user level and the third one is separated.

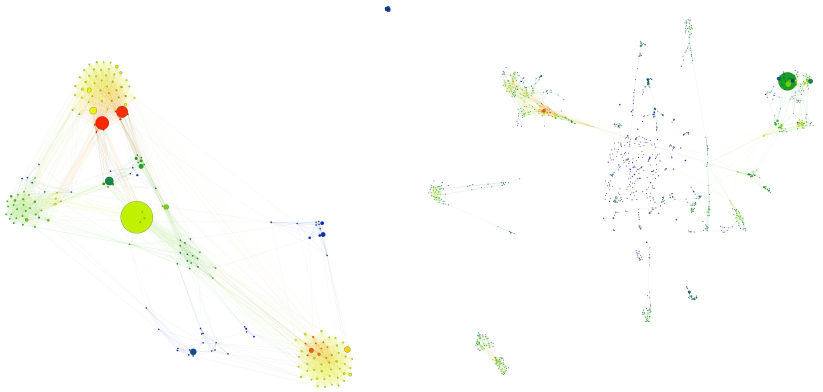
5.4 Sequence Type D

The results for the sequence type D are shown in Table 4. Sequence type D combines a topology of the process, user and time parameter. We can see that the most used case type is a case type in which user U260 created the invoice in less then 32 hours, then verified the invoice in less then 32 hours, user U074 followed by user U260 approved the invoice in less then 32 hours, and finally, user U260 sent the invoice in less then 32 hours.

Table 4. Sequence Type D: Result Ordered by Occurrence

Case Type	Occ.	W. Deg.
C_U260;V_U260;A_U074;A_U202;S_U260;	309	4.09
C_U260;V_U260;A_U074;A_U202;S_U260;S_U260;	97	6.48
C_U260;V_U260;A_U074;A_U074;A_U202;S_U260;	72	3.42
C_U260;C_U260;V_U260;A_U074;A_U202;S_U260;	57	6.03
C_U162;V_U162;A_U074;A_U202;S_U162;	51	2.32
C_U260;V_U260;A_U074;A_U202;S_U260;S_U260;S_U260;	46	7.84
C_U162;V_U162;A_U074;A_U074;A_U202;S_U162;	46	1.49
C_U068;C_U068;V_U068;A_U249;A_U192;A_U192;A_U192;S_U068;S_U068;	45	13.04
C_U068;C_U068;V_U068;A_U074;A_U074;A_U202;S_U068;S_U068;	38	8.59
C_U087;V_U087;A_U126;A_U126;A_U124;A_U124;S_U087;	37	2.73
...	...	...

Figure 3(b) shows even more distribution of the same topological sequences based on the time and user view. There are more interesting findings. For example the cluster on the right side of the graph is made around the sequences performed by the user U260 (C, V, S activities), but even the sequences with different user for C, V and S activities are clustered here. The reason is that when the A activity took too long one user, his sequences where put together. The reason might be e.g. that one particular user worked on difficult invoices that took long time to approve.



(a) Sequence Type C (b) Sequence Type D

Fig. 3. Sequence Graphs

6 Conclusion

The paper was focused on various points of view to process mining, especially from the user and time perspective. The process instances were compared and similarity between them analyzed using methods for sequence alignment. We can relatively easily reach many different types of findings that have to be analyzed.

Our approach involves time and user metadata to the examination which enables us to find possible ways to see interesting results. For example, the particular person behavior can be showed and analyzed what are her/his process instances. The behavior patterns from the time perspective can be observed as well. The creation of different communities, user participation on deviations, etc. can be also examined using the proposed approach.

We would like to continue with the extension of this approach in the future. We intent to find out what exact types of results can be obtained by the usage of different methods, examine the methods and accustom them for the usage on different real examples. The detailed interpretation of different case studies will help us then to determine which method, adjustment and what views will be used then for data mining and real process examination in general. The idea is to have a very good control of the process by the usage of the data about already performed process instances.

Acknowledgments

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and the national budget of the Czech Republic via the Research and Development for Innovations Operational Programme, and supported by the project New creative teams in priorities of scientific research, reg. no. CZ.1.07/2.3.00/30.0055, supported by Operational Programme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic, and co-financed by SGS, VB - Technical University of Ostrava, Czech Republic, under the grants No. SP2014/154 'Complex network analysis and prediction of network object behavior' and No. SP2014/157 'Knowledge modeling, simulation and design of processes'.

References

1. J. de Weerdt, A. Schupp, A. Vanderloock, and B. Baesens. Process mining for the multi-faceted analysis of business processes - a case study in a financial services organization. *Computers in Industry*, 64(1):57–67, 2013.
2. E. Esgin and P. Karagoz. Sequence alignment adaptation for process diagnostics and delta analysis. In *Proceedings of 8th International Conference HAIS 2013*, pages 191–201, 2013.
3. A. Guo and H. Siegelmann. Time-warped longest common subsequence algorithm for music retrieval. In C. L. Buyoli and R. Loureiro, editors, *Proceedings of 5th*

International Conference on Music Information Retrieval ISMIR 2004, pages 258–261. Universitat Pompeu Fabra, 2004.

4. D. Gusfield. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology.* Cambridge University Press, 2008.
5. D. S. Hirschberg. Algorithms for the longest common subsequence problem. *J. ACM*, 24:664–675, October 1977.
6. S. J. Štolfa, S. Štolfa, K. Slaninová, and J. Martinovič. Searching time series based on pattern extraction using dynamic time warping. In *Proceedings of the Dateso 2014 Annual International Workshop on DAtabases, TExts, Specifications and Objects.CEUR Workshop Proceedings*, pages 81–90, 2014.
7. T. Kocyan, J. Martinovič, P. Dráždilová, and K. Slaninová. Searching time series based on pattern extraction using dynamic time warping. In *Proceedings of the Dateso 2013 Annual International Workshop on DAtabases, TExts, Specifications and Objects.CEUR Workshop Proceedings*, pages 129–138, 2013.
8. M. Müller. *Information Retrieval for Music and Motion.* Springer, 2007.
9. K. Slaninová, T. Kocyan, J. Martinovič, P. Dráždilová, and V. Snášel. Dynamic time warping in analysis of student behavioral patterns. In *Proceedings of the Dateso 2012 Annual International Workshop on DAtabases, TExts, Specifications and Objects. CEUR Workshop Proceedings.*, pages 49–59, 2012.
10. K. Slaninová, J. Martinovič, T. Novosád, P. Dráždilová, L. Vojáček, and V. Snášel. Web site community analysis based on suffix tree and clustering algorithm. In *Proceedings - 2011 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology - Workshops, WI-IAT 2011*, pages 110–113, 2011.
11. K. Slaninová, J. Martinovič, Šperka R., and P. Dráždilová. Extraction of agent groups with similar behaviour based on agent profiles. In C. et al., editor, *12th IFIP TC8 International Conference on Computer Information Systems and Industrial Management, CISIM 2013*, volume 8104 of *Lecture Notes of Computer Science*, pages 348–357, 2013.
12. W. van der Aalst and A. Weijters. Process mining: A research agenda. *Computers in Industry*, 53(3):231–244, 2004.
13. W. van der Aalst, A. Weijters, and L. Maruster. Workflow mining: Discovering process models from event logs. *Transaction on Knowledge and Data Engineering*, 16(9):1128–1142, 2004.
14. B. F. van Dongen, A. K. A. de Medeiros, H. M. W. Verbeek, A. J. M. M. Weijters, and W. M. P. van der Aalst. The prom framework: A new era in process mining tool support. In *Applications and Theory of Petri Nets 2005*, volume 3536 of *Lecture Notes in Computer Science*, pages 444–454. Springer Berlin Heidelberg, 2005.
15. J. Štolfa, M. Kopka, S. Štolfa, O. Koberský, and V. Snášel. An application of process mining to invoice verification process in sap. In *Proceedings of 4th International Conference on Innovations in Bio-Inspired Computing and Applications, IBICA 2013*, pages 61–/74, 2014.