The final authenticated version is available online at
https://doi.org/10.1007/978-3-319-98648-7_26

Assessing the Quality of Search Process Models

Marian Lux[1,2], Stefanie Rinderle-Ma[1], Andrei Preda[2]

[1]University of Vienna, Faculty of Computer Science, ds:UniVie, Vienna, Austria
marian.lux@univie.ac.at, stefanie.rinderle-ma@univie.ac.at
[2]LuxActive KG, Vienna, Austria, office@myoha.at

Abstract. Search processes are highly individual business processes reflecting the search behavior of users in search systems. The analysis of search processes is a promising instrument in order to improve customer journeys and experience. The quality of the analysis results depends on the underlying data, i.e., logs and the search process models. However, it is unclear what quality means with respect to search process logs and models. This paper defines search process models and revisits existing process model and log quality metrics. A metric for search process models is proposed that assesses their complexity and degree of common behavior. In order to compare metrics for search process models different logs and search processes are generated by using ontologies for user guidance during search process execution and for post processing of the logs. Based on an experiment with users in the tourism setting different logs and models are created and compared.

1 Introduction

It is a big asset for companies to know and understand their business processes. Process mining offers a bundle of promising techniques for discovering and analyzing business processes [1]. Business processes are ubiquitous and vary in their nature ranging from short-running and rather rigid administrative processes to highly individual processes such as patient treatment processes [12] and customer journeys describing the user interactions with the company [26]. Recent case studies show that process mining techniques can be successfully applied in order to derive customer journey processes from system logs, e.g., in the entertainment domain [25], in banking [3], and in the tourism domain [16]. Customer journey processes often imply search activities by the users, such as searching for activities when planning a trip. The search behavior can be captured as a *search process* [14] where each of the activities represents a search term a user has looked for through the search system provided by the company. Analyzing such search processes can provide valuable insight for companies [16] and answering the following **analysis questions (AQ)** through search processes:
• What is the typical customer search behavior?
• What are critical customer touch points?
These **AQ** can influence the customer satisfaction which helps companies to win in the market to increase their revenue [18].

The high variety in search terms, however, might lead to discovered search process models of paramount complexity (also referred to as spaghetti models [1]). Thus the promise of gaining valuable insight might be repealed by non-interpretable process models. Hence, assessing and improving the quality of discovered search process models is <u>the</u> prerequisite to reach analysis goals in a meaningful way. Further on, measures to reduce the complexity of the discovered search process models through pre- and post-processing of the analyzed search logs can be taken (cf. general data quality issues in process mining [1]). For search processes, operational ambiguities might be one major source of data quality issues as caused by description of process activities at different abstraction levels [11], but also by the usage of different languages or due to homonyms and synonyms [29]. Hence, this paper aims at assessing the quality of discovered search process models with respect to the **AQ**, considering how ontologies can be exploited for improving the quality of discovered search processes. In detail:

- Q1 How to measure the quality of search process models discovered by process mining techniques with respect to the **AQ**?
- Q2 Does the quality of search process models increase when users are supported by an ontology during the search process?
- Q3 Does the quality of search process models increase when using an ontology for log post processing?

This paper has an empirical focus with a concrete application setting, but also necessitates the creation of artifacts. As such the developed concepts are application-independent.
Research method and contribution: The paper follows design science research (cf. [34]). The relevance of the research problem is underpinned by practical applications from tourism [16], entertainment [25], and banking [3] as well as by literature, specifically on process model quality, e.g., [31] and process mining, e.g., [1]. The following artifacts are created to answer RQ 1–3. At first, a notion of search process models as well as a quality metric specifically tailored towards search process models with respect to **AQ** are proposed in Sect. 2 balancing complexity and clustering in search process models. Section3 introduces the concept of using ontologies during process execution and for post processing of logs. These artifacts are then evaluated based on an experiment with users in Sect. 4: it creates 4 types of logs for different modi operandi, i.e., for executing search processes in the tourism domain with and without using an ontology and combining these logs with or without post processing. The logs are compared statistically and different metrics are applied to assess the effectiveness of using ontologies on the quality of the resulting logs as well as the feasibility of the newly proposed metric. The paper continues with a discussion in Sect. 5 and a related work discussion in Sect. 6. It concludes in Sect. 7.

2 Search Process Models and Quality Metrics

Search Process Models: One type of human-driven and highly individual processes are search processes. The manifestation of real-world search processes are process logs as, e.g., stored by a tourism platform. Basically, process logs store events that refer to the execution of process activities together with the time stamp of execution and possibly further information such as the originator [1]. The events are grouped for the different process instances based on a case id. One can analyze the logs directly, but as we are particularly interested in typical customer behavior and touch points ($\rightarrow$ **AQ**) also the models behind these logs are of high interest. To the best of our knowledge no formal definition for search process models exists. In information science, informally, an (information) search process is defined as *"the user's constructive activity of finding meaning from information in order to extend his or her state of knowledge on a particular problem or topic. It incorporates a series of encounters with information within a space of time rather than a single reference incident."* [14]. In web (usage) mining, a log-based view is taken: a search or *"query trail qt comprises a user's query q (consisting of a sequence of terms* $\{t_1, t2, \ldots, t_{|q|}\}$*"* [2]. We converge and elaborate both definitions into a (graph-based) search process model:

Definition 1 (Search process model). *Let $\mathcal{S}$ be set of all search terms. A search process model is defined as directed graph $SP := (N, E, l)$ where*

- *N is a set of nodes*
- *$E \subseteq N \times N$ denotes the set of control edges*
- *$l : N \mapsto S$ denotes a function that maps each node to its label, i.e., $\forall n \in N$ n is a search activity, i.e., the node n represents the search for a certain search term and is labelled with this search term respectively.*

Figure 1a depicts a small example for a search process model in the tourism domain. The search terms label the process activities, e.g., *bicycling* or *sports shop*, meaning that – after searching for *active* – a user has searched for *bicycling* followed by searching for *sports shop*.

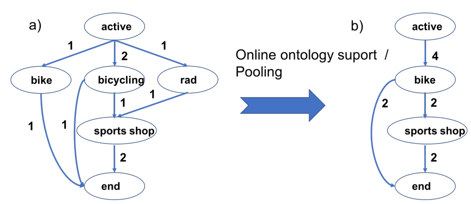


Fig. 1. Example search process model from tourism

One major difference to typical business processes is that a search process is not set out upfront to manage the behavior, but develops individually for each search during runtime. As we aim at ex post analysis of process data, we abstain from defining process instance states for search processes, but rather count the number of executions and annotate the control edges with this information as done for, for example, dependency graphs [33]. For the example shown in Figure 1a, 2 users searched for *bicycling* where one user followed up by searching for *sports shop*. Under the precondition that *active* is the starting point for all searches, overall 4 search processes (and instances) were conducted in this example. This perception of the number of executions is suitable for, e.g., process analysis regarding question such as "what are the main search paths taken by the users within the platform".

The example depicted in Figure 1 illustrates the tackled research problem. Even in this small example, 3 different search terms were used in order to describe the same concept, i.e., *bicycling*, *bike*, and *rad* (the latter being the German word for bike). If the goal is to analyze user behavior it could be more interesting to consolidate these terms into one term as depicted in Figure 1b where the 3 aforementioned terms have been pooled into search term *bike*. Here it can be seen more easily that 4 users were searching for a term related to concept *bike* and 2 users followed up looking for *sports shop*.

Quality Metrics: How can a metric assessing the quality of search process models with respect to the **AQ** be defined? We argue that one metric cannot assess all quality aspects of a process model at the same time as they might even be contradicting (e.g., showing all details vs. abstraction). The aim of the metric proposed in the following is to emphasize those properties of the model that relate to the **AQ**. Here, specifically, quality aspects refer to the comprehensibility of and the degree of common behavior in the search process models as well as the semantic enrichment of the process logs. Comprehensibility is tied to the complexity of the search process model by reducing the "spaghetti degree". The degree of common behavior is reflected by clustering of activities in the model and the log. The quality metric does not refer to other quality aspects such as how well the discovered search process models reflect the underlying process logs (cf. fitness for process conformance [27]). In the following, the metric is constructed by considering existing business process quality metrics for assessing the complexity and metrics for process log quality for assessing the clustering. In Sect. 4 the new metric is then evaluated against selected existing metrics.

Graph metrics can be applied to assess the complexity of a process model [20]. Transferring this to a search process model $SP := (N, E, l)$ one can consider size ($|N|$), diameter (length of the longest path in SP), structuredness (share of nodes in structured blocks), separability (share of cut vertices in SP), and cyclicity (number of nodes in cycles in SP). For measuring the relation or connection between activities in process models coupling and cohesion have been proposed by [30]. Coupling measures *"how strongly the activities in a workflow process are related, or connected, to each other"*. The connection is measured based on the information elements shared by the activities. The cross-connectivity metric assigns weights to nodes and edges to reflect their connectivity [31]: nodes are weighed based on the number of outgoing edges, edges by the product of the weights of source and target nodes. Cross-connectivity seems promising for the envisioned quality metrics in terms of expressing the role of a node in a network and indicating clusters in the process models. Contrary, for search processes, coupling and cohesion are not meaningful in the context of this paper as no information objects are currently considered for search processes. However, such metrics are promising for future analysis.

Process log and model quality is a major concern in process mining [1]. Different techniques have been proposed to deal with "spaghetti degree", including pre-processing of logs, process mining techniques, and post processing of logs. An example for pre-processing of logs is trace clustering [15] where logs can be clustered along certain criteria, e.g., for a certain process duration or where certain activities were executed. A process mining technique that aims at reducing the complexity of the mined models is the Fuzzy Miner [10]. It employs the principles of aggregation, abstraction, emphasis, and customization. Post processing as suggested by [6] also works with filtering, i.e., abstraction from details, in order to simplify the discovered models. From the principles of Fuzzy Miner and post processing aggregation and abstraction will be chosen for the assessment of search process models with respect to **AQ**. Moreover, the size of the logs will be considered in the proposed metric.

As an outcome of the above discussion, a quality metric for search processes shall incorporate ingredients of process model quality assessing the complexity and connection as well as the existence of clusters as used for process mining, i.e., the frequency of activity execution and the degree of the associate node as well as the number of overall activity executions in order to rate the frequency the activity of interest has been executed. Further on the overall number of events is incorporated to consider the overall diversity of the search process, formally:

Definition 2 (Search Process Quality Metric). *Let SP = (N, E, l) be a search process and let L be a log created by executing instances on SP. Let further* $|L|$ *be the number of all events contained in L and A be the set of distinct activities having been executed in L. Then the search process quality metric spm(n) for a node* $n \in N$ *is defined as*

$$spm(n) := 1 - \frac{degree(n) * |A|}{freq * |L|}$$

where freq denotes the number of executions of n.
Search process quality metric spm(SP) for SP turns out as:

$$spm(SP) := \frac{\sum_{n \in N} spm(n)}{|N|}$$

Search process quality metric spm(SP) of a path $p = < n_1, \ldots, n_k > n_i \in N(k \geq 2)$ *can be determined similarly, i.e., by*

$$spm(SP) := \frac{\sum_{n_i, i=1,\ldots,k} spm(n_i)}{k}$$

Note that the metric avoids isolated nodes being considered as paths. By construction, $spm(n) \in [-1; 1]$ holds as $|A| \leq |L|$ and $degree \leq 2 * freq.$.

3 Using Ontologies for User Support and Pooling

We consider the usage of ontologies to improve the process model quality of mined search processes. For that, we distinguish between two approaches where the same given ontology could be applied, a) during the search functionality, when users are entering search terms and b) when post processing event logs from search terms which were entered by users through the search process.

For a) – without the support by an ontology – the user has no guidance and in a broader sense no recommendations for entering search terms. When mining resulting search processes in the sequel, the discovered process models might get complex because of the possibly infinite options for search terms. For b), the search system has to possibly process synonyms and different languages which are known to pose challenges on later process mining [13], for example, resulting in different activities which have the same meaning and hence unnecessarily pump up the complexity of the search process models. In order to foster a) and b), we develop a meta concept which is responsible for recommendations when a user is searching and for post processing of event logs on search processes (cf. Fig. 2). Both approaches, a) and b) are implemented for a commercial tourism platform within the CustPro [35] project which aims to analyze the customer journey process of tourists where a) is already used in the live system by tourists and b) is implemented for evaluation purpose of this work but is also on the agenda to be implemented in the live system. The backend is written in *Java* as *RESTful Web-Service* and the ontology support with reasoning was conducted with the *Java* framework *Apache Jena*[1]. The frontend was developed in *HTML5* and *JavaScript* as single-page application[2] and talks to the backend with *AJAX*. Therefore the frontend was accessible in the web browser and the UI was optimized for mobile devices.

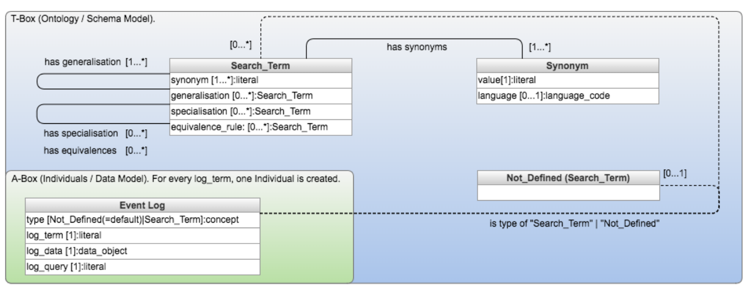


Fig. 2. Ontology with the capability of reasoning and possibilities for post processing

As set out in Fig. 2, a T-Box model [24] is defined which contains the knowledge base for possible search terms in a specific domain. There are only two concepts defined, which makes the ontology easy to implement and maintain. The first concept is called *Search_Terms*. It consists of the elements *synonyms*, *generalisations*, *specialisations*, and *equivalence rules*. Synonyms are literals in different languages which are referring to the possible search terms in the logs for the concept. Every *Search_Terms* concept has at least one *synonyms* element and every element describes the concept equally in contrast to e.g., SKOS [32] with property *skos:prefLabel*. Generalizations and specializations refer to the concept itself for defining relations between search terms and are optional. Also equivalence rules for defining relations, e.g., for combinations of search terms or

[1] `https://jena.apache.org/index.html`
[2] `http://itsnat.sourceforge.net/php/spim/spi_manifesto_en.php`

relations which cannot be defined in the outlined elements before, or for generating suggestions for search terms, e.g., based on combinations of search terms entered by a user before, are provided and also optional. The other concept, *Not_Defined* does not contain any knowledge. It acts as helper for post processing the event logs. Non-matching terms between the ontology and the logs are flagged with this concept.

Online Usage With Ontology: The T-Box model in Fig. 2 contains the ontology and thus a knowledge base for search terms in a specific domain. For every performed user search query, which can include multiple search terms, online suggestions are created during the search process for user support. Through rules in the ontology, search term specific combinations in a search query determine suggestions by using logical rule reasoning (e.g., a query containing "*mountain*" and "*sports*" results in a suggestion for "*hiking*"). Further, search term combinations like *"x, y"* in the same search query are separated into two search terms and for every search term based on its matching synonyms, specializations and generalizations are determined by using hierarchical reasoning (e.g., a specialization of "*bike*" is "*mountainbike*") for suggestions. As result, the user gets online suggestions based on the last performed search query with the given ontology.

Post Processing With Ontology: The A-Box [24] model in Fig. 2 contains the event logs which are individuals and defined with the concept *Event Log.* For every log query, which can include multiple search terms, a separate individual per search term is created. Hence, an event log with the search query *"x, y"* will be separated into two individuals, one for the term x and one for y. Each individual consists of the elements *log_term*, *log_data*, *log_query* and in the initial phase belongs to the concept *Not_Defined*. *log_term* refers to the search term, which will be processed by using logical rules with the T-Box model (ontology). *log_query* contains the original search query from the user and *log_data* acts as symbolic placeholder for further data from the origin log entry. We applied a logical rule which searches for matching synonyms between a given *log_term* and all *Search_Terms* from the ontology. If there is a match, the type "*Not_Defined*" from the individual is removed, if not already happened before, and the found *Search_Term* is added as type to the individual. After the rule was applied to all individuals, there is a lookup for individuals which do not contain the type "*Not_Defined*". For these individuals, the initial search strings of their corresponding logs are replaced with the class names from their *Search_Term* types. Therefore the logs contain pooled search queries for reducing the complexity of mined process models.

In this paper, we employ a controlled ontology (as defined by the analyst). Hence employing post processing does not lead to loss of information in the resulting models when compared to online usage. Contrary to online usage, in case of using an ontology that has not been defined by the analyst, post processing might not reflect the user's intention during search. In general, a user interface for the online ontology support through the search process could positively impact the frequency of its usage as well the quantity of observed logs.

4 Experiment

This section presents the design and execution of the experiment to evaluate the proposed artifacts, i.e., the metrics and the ontology support algorithms. The experiment bases on the implementation described in Section 3 and is conducted with subjects in a real world scenario.

4.1 Experimental Setup

The experiment was conducted with students of one course of the Bachelor Computer Science at the University of Vienna. This leads to a relatively homogeneous group of participants and can be regarded as sufficient with respect to knowledge on working with a tourism app. Overall, 93 students participated in the experiment. From an experimental point of view, 2 independent groups are required, i.e., one group working with ontology support and one without. Due to organizational reasons (the course is held in 4 groups), 4 independent groups were built where 2 worked with and 2 without ontology support.

Each of the 4 groups has the same scenario and task to accomplish: Every participant plans touristic activities on an imaginary three day stay from Friday to Monday in a hotel as tourist in the tourism region Mondsee in Austria. The subject writes down titles of activities on an empty schedule which was handed out [35]. The titles of the activities are searched in a search application which contains touristic activities. The experiments started with an introduction of 5 minutes explaining the task of the experiment. Then the subjects had a 10 minutes time frame to use their own mobile device (smartphone, tablet, notebook) to search for activities in the provided search application. Two of the four groups received ontology support in their search function. The search application and the ontology were provided in German. The search logs were recorded in the respective time frames and for every group. Different cases were recorded in the event log entries for distinguishing them. In Table 1, the groups of the subjects are depicted.

Table 1. Groups of subjects in experiment

group number	online usage with ontology	number of different devices (case device)	number of subjects
1	no	19	20
2	yes	24	24
3	no	27	24
4	yes	24	25

Note that there is a difference between subjects and devices. The reason for that is, that two subjects can share the same device and one subject can use multiple devices. In the following, only the used devices are further addressed because the number corresponds to the recorded event logs during the experiment. As explained before, the groups with and without ontology support are to be compared. For this purpose the 4 groups from Table 1 are merged into 2 logs, i.e., group number 1 and 3 and group number 2 and 4 (cf. Table 2)

Table 2. Merged groups of subjects in experiment

log name	group numbers	online usage with ontology	number of different devices (case device)	number of activities	number of events
log 1	1+3	no	46	117	246
log 3	2+4	yes	48	116	331

We also implemented a web service, according to Section 3, for post processing the obtained event logs, which uses the same ontology as in the experiment for online supporting the subjects on their search functionality in groups 2 and 4. Therefore *log 2* contains post processed event logs from *log 1* which means that both logs originate from the same log recording. The scheme is analogous between *log 4* and *log 3*. Overall, this results in the 4 logs shown in Figure 3. These logs with their designated log names (*log 1 - log 4*) build the basis for further analysis.

Post processing ⇨ Online usage ⇩	Without ontology	With ontology
Without ontology	log 1	log 2
With ontology	log 3	log 4

Fig. 3. Experimental log creation

4.2 Statistical Comparison of Logs

t-tests [21] are applied to compare the logs from Fig. 3 with respect to the mean of occurred events per *case device*. *Hypthesis I (HI): A higher mean in the logs results when providing online usage with ontology support*. As the subjects tend to perform more search queries because based on the recommendation for further search terms, the user does not have to think about formulating queries. Formulating *H1* is justified by the number of total events as in Table 2. *Hypthesis II (HII): A lower mean results when post processing the logs with ontology support because of pooling search queries.*

First we compared *log 1* with *log 3* to prove a statistical effect on online usage with ontology support. With a one tailed t-test and a 90% confidence interval, we obtained significance for *HI*. There was no evidence on a 95% confidence interval and as well on a two tailed test. Second, we compared *log 1* with *log 2* to prove a statistical effect on post processing logs with ontology support. HII cannot be accepted on a 90% confidence interval. Further log comparisons were not suitable because, as mentioned before, online usage with ontology support tends to increase and post processing with ontology support tends to decrease the mean. The complete t-test is shown in the supplemental material [35] (cf. folder "*T-Test*").

4.3 Process Model Quality Metrics

For assessing the process model quality of each log from Figure 3, we first applied selected process model quality metrics, i.e., *size* and *diameter* (cf. Sect. 2) as they provide an overview on the complexity of the models. Here, we also applied a filter, which counts only the 20% most frequent activities from each log (abstraction). We chose that percentage because of the *pareto principle*, which shows that in many cases, ranging from the economy to the nature behavior, 80% of causes are produced by 20% of activities [22]. Thus, depending on the given log with its distribution of activities and number of events, only activities with a specific frequency are included in the calculations. Table 3 shows the results which are explained in the following. For *log 1* and *log 2* the filtering has no effect, i.e., the least occurrence of activities remains 1. *log 1* had a *size* of 117 and the *size* of *log 2* was 110. The *diameter* was 19 for both logs. This shows that applying post processing has only a slight effect on the number of activities and no effect on the diameter. For *logs 3* and 4, filtering had an effect, i.e., *log 3* filtered on an activity occurrence ≥ 2 and *log 4* an activity occurrence ≥ 3. With respect to the metrics, this results in a notably reduced size, i.e., for *log 3* a reduction of the size from 116 to 41 and of the diameter from 19 to 15 and for *log 4* a reduction of the size from 109 to 26 and of the diameter from 19 to 13. Hence, it can be interpreted that online ontology support has a considerable effect on reducing the process model metrics size and diameter when using filtering. There is also to mention, that each of *log 1* and *log 2* contained 33 variants of paths where only 3 variants contained more than 1 *case device*. Nearly the same picture was discovered on *log 3* and as well *log 4*. Each of them had 39 variants and only 2 variants contained more than 1 *case device*. This is a good indicator, which shows how highly individual a user performed search process can be.

Table 3. Results of applying regular process model quality metrics

log name	size unfiltered	diameter unfiltered	size filtered	diameter filtered
log 1	117	19	117	19
log 2	110	19	110	19
log 3	116	19	41	15
log 4	109	19	26	13

As next step for assessing the process model quality of each log, we applied our defined *Search Process Quality Metric* from Section 2. Table 4 summarizes the results. In regard to the values of the path it is to be noted that there was as well a filter applied which includes only the 20% most frequent activities. But we also excluded paths which contain solely the search terms "*" or *FIRST_RUN_-_** in any combination. We did the latter because: the search term *FIRST_RUN_-_** signals, that after loading the application the first time on a device, an automatic "*"-search is performed. A "*"-search in the logs signals that the user just hit the search button without considering a search term as query input. Thus, such a path, where no search term was entered by a subject has no meaning in our case and was filtered out. As we can see, from *log 1* to *log 4* there was an increase in the quality of the logs. We can also see that the online usage of an ontology had a bigger impact than post processing with ontology. The difference between *log 1* and *log 2* is $0,058$ on *spm(SP) unfiltered* where the difference between *log 1* and *log 3* is $0,228$. The values of *spm(SP) filtered* have a greater impact through the online usage with ontology. In comparison to the regular process model quality metrics (size and diameter), we can see an improvement in terms of meaningfulness by using our *Search Process Quality Metric* for identifying clusters and quality comparisons of logs. All values in Table 4 increased when using ontology support. We can conclude, that the ontology usage improved the process model quality in general and on specific paths.

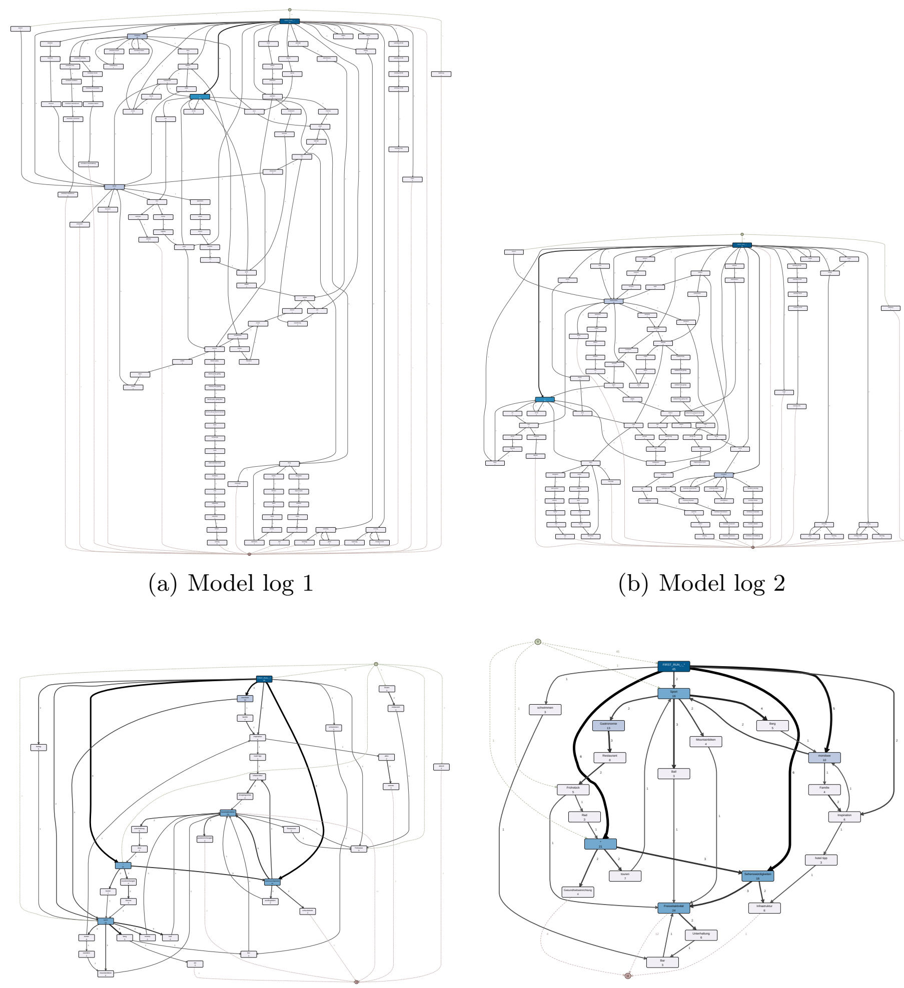
Table 4. Results of applying *Search Process Quality Metrics*

log name	spm(SP) unfiltered entire search process	spm(SP) filtered path highest value	spm(SP) filtered path lowest value
log 1	0,145	0,714	0,107
log 2	0,203	0,731	0,166
log 3	0,373	0,781	0,412
log 4	0,415	0,794	0,481

For visual inspection, process models were discovered for each of the 4 logs from Figure 3 using *Disco* (cf. Figure 4) which uses an adapted version of the Fuzzy Miner, called *Disco miner*, which is geared towards discovering clusters in process model [3]. The process models show only the most frequent activities and the most dominant paths in their process map. For this the *paths slider* was set to 0% to show only dominant connections between activities that have occurred and the *activities slider* was set to 20% to show only the 20% most frequent activities in the mined process map. Furthermore the absolute frequency of an activity is visualized using color strength. Only from visual inspection the resulting model for *log 4* seems to be less complex and to contain more clusters when compared to logs $1-3$.

Finally, activities are selected through visual inspection, i.e., those showing high clustering based on color strength, and analyzed using the *Search Process Quality Metric*. We started with the models for logs before and after post processing. Search term "*Kino*" (cinema), for example, is compared for *log 1* with value $0,049$ and *log 2* with value $0,285$ as well search term "*essen*" (eat) with value $0,239$ from *log 1* which was pooled to the term "*Gastronomie*" (gastronomy) with the value $0,292$. We also compared the model from *log 3* with the search terms "*Restaurant*" (restaurant) with value $0,299$ and "*Berg*" (mountain) with value $0,509$ with *log 4* and their scorings for "*Restaurant*" with value $0,506$ and "*Berg*" with value $0,539$. These results confirm that post processing with ontology usage has a positive impact on search process quality. Then selected activities are compared for the same process model. For the model of *log 4*, we chose two frequent search terms. The first is "*Restaurant*" with term frequency of

[3] `https://fluxicon.com/blog/2012/05/say-hello-to-disco/`



(a) Model log 1 (b) Model log 2

(c) Model log 3 (d) Model log 4

Fig. 4. Visual inspection of process discovery results based on logs 1 to 4 with enabled filter for showing 20% of most frequent activities and only most dominant paths

19 and a *spm(n)* of $0,567$. The second one is "*Freizeitaktivität*" (leisure activity) with term frequency of 24 and a *spm(n)* of $0,575$. Both terms had a lower value than the term "*Ball*" (ball) with value of $0,78$ despite the lower frequency of only 3 but with a clearer path (lower degree). For the terms "*Familie*" (family) with a term frequency of 4 and "*Restaurant*" with a term frequency of 8 and with the same value of *spm(n)*, which was $0,506$, nearly the same value was indicated than for the two very frequent terms "*Restaurant*" and "*Freizeitaktivität*". The reason for that is, that on "*Familie*" and "*Restaurant*" the incoming and

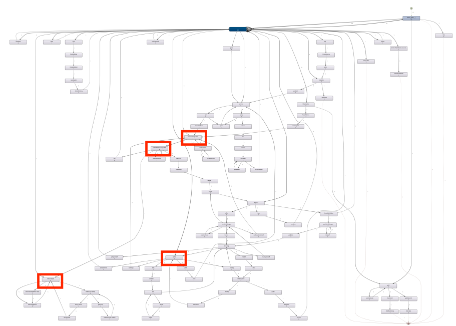
outgoing nodes were better clustered. We can conclude that the *Search Process Quality Metric* supports to discover and rank important paths and fragments in terms of clustering in search processes. The results, figures and calculations from the experiment can be found in the supplemental material [35] (cf. folder "*Experiment*").

5 Limitations and Threats to Validity

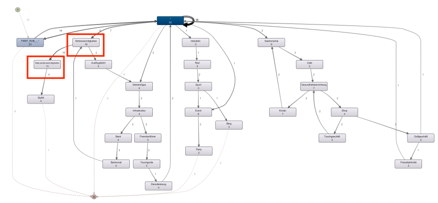
The results could be improved if the ontology increases in terms of its size and relevance to its domain where the search process is executed for. Moreover, the sample size of the experiment could have been too small for filtering and multiple languages. We will investigate the relation between filtering and sample size in future work. Because of the small sample size and therefore the small number of available subjects, we decided to run the experiment in one language, i.e., German, because the ontology contains a different number of classes and labels for describing them per language. Also rules for suggestions of combined search terms are not the same per language. This could be an explanation why post processing does not show a comparable impact to another study in the tourism domain on a live system that was conducted using German and English at the same time. Find the discovered models in Figure 5. Though there is not enough space to discuss this study in detail, the models give an impression that post processing using an ontology resolves ambiguities with respect to language. Another limitation of the proposed approach is the missing handling of compound nouns in search queries with mixed search terms that contain both, compound nouns and single nouns. In the tourism domain, we have had to deal with this problem and added hyphens between the search terms. The corresponding tourism ontology contains also hyphens for compound nouns, defined as synonyms. For example, if user enters search term *"nature sights"* it is modified to *"nature-sights"* and the label in the ontology is exact the same. But in the particular case of the tourism platform, we have to mention, that search queries with multiple nouns are very seldom. Nevertheless we are planning future research activities, to deal with compound nouns in search queries. A starting point would be the work presented in [28].

6 Related Work

Process mining algorithms have become mature and efficient [1]. There are various ways for simplifying discovered processes [6]. For improving the quality of process mining results, apart from implementing constantly improving algorithms [4], it seems obvious to improve also the event log quality. This can be performed in different kind of approaches [17,11]. One of them is the promising research field of semantic technologies [8], which is also quite proven very well. Process mining, combined with semantic technologies can improve the meaningfulness and therefore the quality of mined processes reasonably well [19,1]. Most approaches, for enhancing process mining with semantic technologies, are using



(a) Model without post processed log. 4 red squares show different search terms



(b) Model after post processing the log. 4 squares from (a) are pooled to 2 squares

Fig. 5. Mined search processes from logs which contain different languages. Without post processing the log (*left*) and after post processing the log (*right*) which reduces the complexity of the model by pooling the search terms from different languages (*English and German)*. The models can be found in the supplemental material [35] (cf. folder "*Figure5Detail*")

ontologies behind the scene for log preparation, e.g., by mapping process labels from event logs to hierarchical links in ontologies [5]. Ontologies are also used, to reduce the complexity of discovered process models by dealing with synonyms, hierarchies, reasoning or constraints [13]. But most ontologies, which are used for creating or enhancing semantic logs are either complex and thus burdensome [23] or too domain specific [7] for using them in different domains in industrial software solutions [29]. As opposed to all the aforementioned approaches this work addresses search processes where the search terms are defined by the users in an arbitrary manner, and not by the application in form of predefined labels as addressed mostly in literature and research [11]. Semantics, together with keyword search is also already covered in literature [9], but without log preparation and meta ontologies, that are easy to adapt with a minimum of effort in the widest possible range of domains and industries. Quality metrics from literature have been discussed in Section 2.

7 Conclusion

Case studies from different domains emphasize the potential of process mining for customer journey understanding and improvement. This work assesses and improves the quality of mined search processes as important brick in customer journeys. Regarding *RQ 1*, a newly proposed quality metric for search processes rates the complexity of the output combined with an assessment of the existence of clusters. The experiment evaluates the feasibility of the metric in comparison with results from visual inspection and existing metrics. Moreover, the experiment evaluates quality improvement when using ontologies for online user

support (*RQ 2*) as well as for post processing the resulting logs (*RQ 3*). The experiment is demonstrated by a case study in the tourism domain. In summary, *RQ 2* has been positively demonstrated in Sect. 4.2 by analyzing the mean of occurred events per *case device* and in Sect. 4.3 – where the results have been significantly reinforced by using filtering – by showing through existing metrics, visual inspection of mined process models, and the proposed quality metric, that the complexity of process models decreases and clusters improve. The same improvements could also be shown in Sect. 4.3 for *RQ 3*, but with slightly less evidence for the experiment because of the limitations as pointed out in Sect. 5. Overall, by answering *RQ 1–3*, it can be seen that complexity of and clustering in search processes improve, in particular, in combination with filtering. In future work, we will also consider time with respect to new metrics and experiment designs and measure the impact of an ontology on how accurate search results are for the users *"to get their jobs done"*. Moreover, the mined search processes will be further analyzed with respect to their differences in relation to context variables such as location or weather.
Acknowledgment: This work has been partly conducted within the CustPro project funded by the Vienna Business Agency.

References

1. van der Aalst, W.M.P.: Process Mining - Data Science in Action, Second Edition. Springer (2016)
2. Bailey, P., White, R.W., Liu, H., Kumaran, G.: Mining historic query trails to label long and rare search engine queries. TWEB 4(4), 15:1–15:27 (2010)
3. Celonis: Process mining story postfinance: Optimizing the customer journey in banking. `https://youtu.be/qJ2NcdZSxA4` (2018)
4. Dixit, P., Buijs, J.C., van der Aalst, W.M., Hompes, B., Buurman, H.: Enhancing process mining results using domain knowledge. In: SIMPDA. pp. 79–94 (2015)
5. Dunkl, R.: Data improvement to enable process mining on integrated non-log data sources. In: Computer Aided Systems Theory. pp. 491–498 (2013)
6. Fahland, D., van der Aalst, W.M.P.: Simplifying discovered process models in a controlled manner. Inf. Syst. 38(4), 585–605 (2013)
7. Fernandez, F.M.H., Ponnusamy, R.: Data preprocessing and cleansing in web log on ontology for enhanced decision making. Indian Journal of Science and Technology 9(10) (2016)
8. Fürber, C.: Data quality management with semantic technologies. Springer (2015)
9. Gulla, J.A.: Applied semantic web technologies. Auerbach Publications (2011)
10. Günther, C.W., van der Aalst, W.M.P.: Fuzzy mining - adaptive process simplification based on multi-perspective metrics. In: Business Process Management. pp. 328–343 (2007)
11. Ingvaldsen, J.E., Gulla, J.A.: Industrial application of semantic process mining. Enterprise Information Systems 6(2), 139–163 (2012)
12. Kaes, G., Rinderle-Ma, S.: Generating data from highly flexible and individual process settings through a game-based experimentation service. In: Datenbanksysteme für Business, Technologie und Web. pp. 331–350 (2017)
13. Koschmider, A., Oberweis, A.: Ontology based business process description. In: EMOI-INTEROP. pp. 321–333 (2005)

14. Kuhlthau, C.C.: Inside the search process: Information seeking from the user's perspective. American Society for Information Science 42(5), 361–371 (1991)
15. de Leoni, M., van der Aalst, W.M.P., Dees, M.: A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. Inf. Syst. 56, 235–257 (2016)
16. Lux, M., Rinderle-Ma, S.: Problems and challenges when implementing a best practice approach for process mining in a tourist information systeme. In: BPM 2017 Industry Track. ceur, vol. Vol-1985, pp. 1–12 (2017)
17. Ly, L.T., Indiono, C., Mangler, J., Rinderle-Ma, S.: Data transformation and semantic log purging for process mining. In: Advanced Information Systems Engineering. pp. 238–253 (2012)
18. Maechler, N., Neher, K., Park, R.: From touchpoints to journeys: Seeing the world as customers do (March 2016), `http://bit.ly/2AAzjcJ`
19. Alves de Medeiros, A.K., Pedrinaci, C., van der Aalst, W., Domingue, J., Song, M., Rozinat, A., Norton, B., Cabral, L.: An outlook on semantic business process mining and monitoring. In: On the Move to Meaningful Internet Systems. pp. 1244–1255 (2007)
20. Mendling, J., Strembeck, M.: Influence factors of understanding business process models. In: Business Information Systems. pp. 142–153 (2008)
21. Mertens, W., Pugliese, A., Recker, J.: Quantitative Data Analysis: A Companion for Accounting and Information Systems Research. Springer (2016)
22. Moore, H.: Cours d'économie politique. The ANNALS of the American Academy of Political and Social Science 9(3), 128–131 (1897), `http://bit.ly/2FGyANa`
23. Pedrinaci, C., Domingue, J., Alves de Medeiros, A.: A core ontology for business process analysis. The Semantic Web: Research and Applications pp. 49–64 (2008)
24. Petnga, L., Austin, M.: An ontological framework for knowledge modeling and decision support in cyber-physical systems. Advanced Engineering Informatics 30(1), 77–94 (2016)
25. Pmig, Y., Yongil, L.: Customer journey mining (2018)
26. Richardson, A.: Using customer journey maps to improve customer experience. Harvard Business Review 15(1), 2–5 (2010)
27. Rozinat, A., van der Aalst, W.M.P.: Conformance checking of processes based on monitoring real behavior. Inf. Syst. 33(1), 64–95 (2008)
28. Silverstein, C., Marais, H., Henzinger, M., Moricz, M.: Analysis of a very large web search engine query log. In: ACm SIGIR Forum. vol. 33, pp. 6–12. ACM (1999)
29. Thomas, O., Fellmann, M.: Semantic process modeling - design and implementation of an ontology-based representation of business processes. Business & Information Systems Engineering 1(6), 438–451 (2009)
30. Vanderfeesten, I.T.P., Reijers, H.A., van der Aalst, W.M.P.: Evaluating workflow process designs using cohesion and coupling metrics. Computers in Industry 59(5), 420–437 (2008)
31. Vanderfeesten, I.T.P., Reijers, H.A., Mendling, J., van der Aalst, W.M.P., Cardoso, J.S.: On a quest for good process models: The cross-connectivity metric. In: Advanced Information Systems Engineering. pp. 480–494 (2008)
32. W3C: Skos simple knowledge organization system reference (08 2009), `https://www.w3.org/TR/2009/REC-skos-reference-20090818/`
33. Weijters, A.J.M.M., Ribeiro, J.T.S.: Flexible heuristics miner (FHM). In: Computational Intelligence and Data Mining. pp. 310–317 (2011)
34. Wieringa, R.: Design Science Methodology for Information Systems and Software Engineering. Springer (2014)
35. WST: Supp. mat. (2018), `http://gruppe.wst.univie.ac.at/projects/CustPro/`