Narrowing the Business-IT Gap
in Process Performance Measurement

Han van der Aa[1], Adela del-Río-Ortega[2], Manuel Resinas[2], Henrik Leopold[1],
Antonio Ruiz-Cortés[2], Jan Mendling[3], Hajo A. Reijers[1,4]⋆

[1] Department of Computer Sciences, VU University Amsterdam, The Netherlands
[2] Dpto. de Lenguajes y Sistemas Informticos, University of Seville, Spain
[3] Institute for Information Business, WU Vienna, Austria
[4] Department of Mathematics and Computer Science, Eindhoven University of
Technology, The Netherlands

Abstract. To determine whether strategic goals are met, organizations must monitor how their business processes perform. Process Performance Indicators (PPIs) are used to specify relevant performance requirements. The formulation of PPIs is typically a managerial concern. Therefore, considerable effort has to be invested to relate PPIs, described by management, to the exact operational and technical characteristics of business processes. This work presents an approach to support this task, which would otherwise be a laborious and time-consuming endeavor. The presented approach can automatically establish links between PPIs, as formulated in natural language, with operational details, as described in process models. To do so, we employ machine learning and natural language processing techniques. A quantitative evaluation on the basis of a collection of 173 real-world PPIs demonstrates that the proposed approach works well.

Keywords: performance measurement, process performance indicators, model alignment, natural language processing

1 Introduction

Process Performance indicators (PPIs) play an important role in monitoring the performance of a process [12]. Defining and measuring suitable PPIs are key tasks for aligning strategic business objectives with the operational implementation of a process [22]. A major problem in this regard is that the formulation of PPIs is typically a *managerial* concern, while the monitoring of PPIs requires a *technical* perspective on a process [26]. The resultant gap, representative of the well-known *Business-IT-Gap* (cf. [11,14]), leads to a mismatch between the

definitions of PPIs on the one hand and the process models that describe the actual implementation of processes on the other. This mismatch can result in PPI descriptions that refer to concepts of managerial interests that do not appear in the technical process definition.

The monitoring of process performance is furthermore hindered by the fact that managers frequently start out to provide relevant indicators in the form of unstructured natural language descriptions [24,26]. In order to compute values for these PPIs, the concepts contained in these textual PPI descriptions must be linked to their corresponding process model elements [23]. Currently, the only way to obtain these links is through manual identification. The effort associated with such a manual identification is considerable and, in many cases, hardly manageable due to the vast number of process models and accompanying PPIs that exist in organizations. Specifically, manual alignment actions do not scale with business process model repositories that contain hundreds or even thousands of process models [25], each of which may be accompanied by up to a dozen PPIs. These observations call for an effective and efficient means of automated support.

The goal of the presented research is to provide the necessary support for the establishment of links between textual PPI descriptions and process model elements. To this end, we introduce an approach that automatically relates a textual PPI description to the relevant parts of a process model. We shall refer to this relation as an *alignment*, following the terminology used to describe relations between concepts from different artifacts in contexts such as schema matching [5] and process model matching [4]. An alignment consists of a number of pair-wise *correspondences* between the PPI and process model elements. To obtain this alignment, we combine machine learning and natural language processing techniques in a novel manner. A quantitative evaluation with a set of 173 PPIs obtained from industry and industrial reference frameworks, demonstrates that our automated approach produces satisfying results. The vast majority of the automatically identified correspondences is in line with how people would manually align them. The approach thereby successfully supports what would otherwise be a laborious manual endeavor.

The remainder of this paper is structured as follows. Section 2 illustrates the problem of aligning unstructured textual PPI descriptions to process model elements. Section 3 describes the proposed approach to automatically generate alignments. The quality of the generated alignments is evaluated in Section 4. Section 5 discusses related work on both the problem and solution domains. Finally, we conclude the paper and discuss future research directions in Section 6.

2 Problem Illustration

We illustrate the challenges associated with the automated alignment of process model elements to a PPI using the process model depicted in Figure 1. The process model describes the *request for change* process as implemented by the

IT Department of the Andalusian Health Service.[5] The process starts when a requester submits a request for change (RFC). Then, the planning & quality manager analyzes the request in order to make a decision on its approval. Based on several factors, including the availability of required resources, expected costs, and the nature of the requested changes, the RFC will be either approved, canceled, or the decision will be elevated to further analysis by a committee. In the latter case, the RFC will return for a final decision to the planning & quality manager, after an in-depth consideration by the committee.
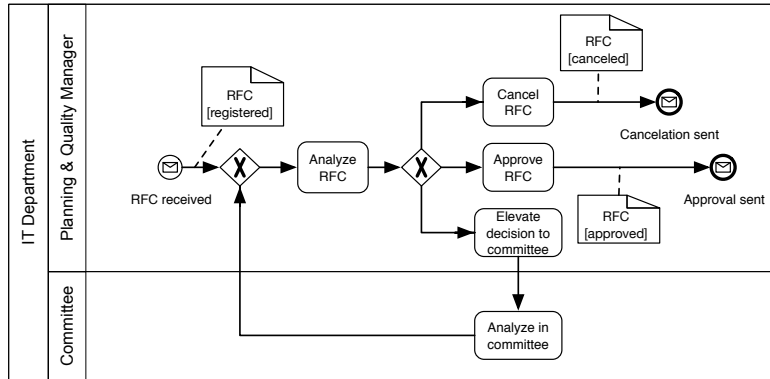

Fig. 1. Process model for the request for change example (simplified)

Table 1 presents six exemplary PPIs related to the request for change process. We will use the examples to illustrate that PPIs can have different measure types. Based on the classification from [24], we distinguish four such types: time, count, data, and derived measures. *Time* measures consider the duration between two instants during the execution of process instances. For instance, PPI1 measures the average time between the receipt of an RFC and its approval. The start and end points of time measures can also relate to the same activity, as can be seen for PPI3. PPI3, namely, measures the time between the start and end of the "*Analyze in committee*" activity. A *count* PPI measures the number of times something happens, for instance the number of times an RFC is registered in the process. *Data* measures consider the attribute values of data objects. PPI5, for example, sums the "*cost*" attribute of all approved RFCs. Lastly, we consider *derived* measures, which involve mathematical functions over one or more other measures. Because *fraction* measures represent the most common kind of derived measures, we consider these as an explicit sub-class of derived measures. Fraction measures divide the value of one measure by another, as seen for PPI6: it divides the number of canceled RFCs by the number of registered RFCs.

---

[5] The interested reader is referred to [24] for an extensive description of the process

Table 1. PPIs for the request for change example

ID	Description	Measure	Model elements
PPI1	Average time between receipt and approval of an RFC	Time	RFC received, Approve RFC
PPI2	Average lifetime of approved RFCs	Time	RFC received, Approve RFC
PPI3	Average duration of a committee decision	Time	Analyze in committee
PPI4	Number of registered RFCs	Count	RFC [registered]
PPI5	Estimated costs of approved RFCs	Data	RFC [approved]
PPI6	% of rejected RFCs from all registered RFCs	Fraction	RFC [canceled], RFC [registered]

The exemplary PPIs and their related model elements specified in Table 1 illustrate that the type of a PPI affects the *kind* and *number* of process elements to be included in an alignment. For instance, though most measure types can relate to activities, events, and data objects, data-based measures exclusively relate to the latter. Furthermore, count and data-based measures, by definition, relate to a single process model element, whereas a fraction requires at least one element as a numerator and one as a denominator. Due to the differences that exist among the various measure types, the first challenge is, therefore, to ensure that generated alignments are well-defined, i.e. in accordance with the semantics of a PPI's measure type. To create an alignment, an automated approach must furthermore deal with the inherently ambiguous nature of natural language. In particular, a second challenge to overcome is the ability of natural language to express the same semantic concepts through a variety of syntactic patterns [3]. PPI1 and PPI2, for instance, both refer to the time duration between the "*RFC received*" event and the completion of the "*Approve RFC*" activity. However, the two descriptions are clearly distinct. PPI2 just refers to "*the lifetime of approved RFCs*", whereas PPI1 explicitly specifies start and end points of the measure. To overcome this challenge, an automated approach must be able to deal with the flexible and informal language preferred by human users [15]. Third, an alignment approach must handle differences between the terminology used to define PPIs and those used for the process model. For instance, PPI6 refers to "*rejected* RFCs", whereas the process model describes these as "*cancelled*". Such differences are particularly relevant because PPIs and process model are generally defined by different organizational stakeholders, with different perspectives (i.e. managerial versus operational). The alignment approach presented in Section 3 addresses these challenges in order to automatically generate alignments between PPIs and process model elements.

3 Alignment Approach

Figure 2 presents an overview of the proposed alignment approach. The approach takes a textual PPI description and a process model to which the PPI relates as input. Given this input, the approach generates an alignment in three steps. In the *type classification* step, we determine the measure type of a PPI based on its textual contents. In the *PPI parsing* step, we parse the textual PPI description in order to extract a set of phrases that specifically relate to parts of the considered process. Both of these steps build on a *decision tree classifier*. For the former step, this classifier provides the classification of a PPI's measure type. For the latter, we use a set of *type indicators* $\mathcal{T}$, automatically learned during the training of the decision tree, to support the parsing of a PPI's description. In the third and final step, we combine the results of the previous steps to generate an alignment between the extracted phrases and elements of the process model. In the following sections, we describe each step in detail.
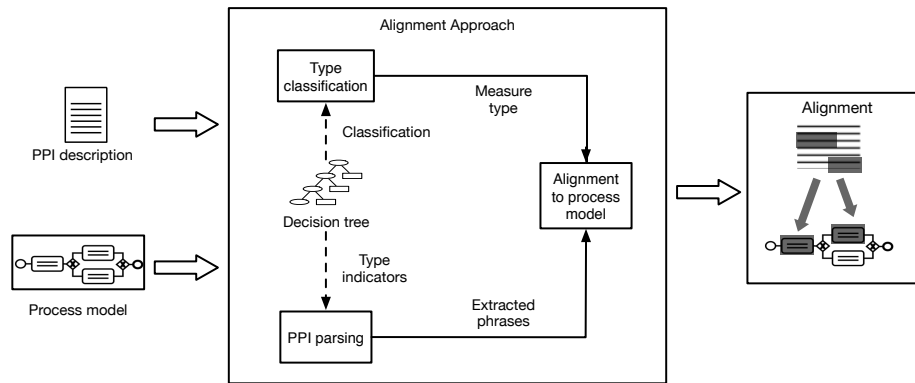

Fig. 2. Outline of the approach

3.1 Type Classification

The measure type of a PPI affects the number and kind of process model elements that such a PPI can or should be aligned to. It is, therefore, important to correctly determine the type of a given PPI. Without a correct type identification, an approach can yield nonsensical alignments, such as a data-based measure aligned to an activity, or a fraction without a denominator. To avoid such issues, we infer the type of a PPI based on the terms in its textual description. We achieve this by employing a *decision tree classifier.*

Classifiers are means to determine to which category of a pre-defined set a previously unseen data point most likely belongs. In the context of our approach, we specifically employ a decision tree classifier to determine if a PPI has a *time,*

count, *data*, *derived*, or *fraction* type of measure. A decision tree is a type of classifier which models the classification process as a series of data-based choices, represented as the nodes of a tree. The choice for a decision tree is driven by their particular suitability to identify keywords that discriminate among the different measure types. For example, the occurrence of the term *"percentage"* in a PPI description is a good indicator that this PPI describes a fractional measure. We identify these discriminatory terms, which in the remainder we shall refer to as *type indicators*, by training a decision tree on the bag-of-words representations of previously categorized PPI descriptions. Figure 3 presents a fragment of a decision tree obtained in this manner. At each node, the presence or absence of a given term in the description is checked. Based on the outcome of this check, a branch is chosen from several alternatives. The process continues alongside this branch until a leaf node is reached. This node then represents the measure type predicted for the PPI.
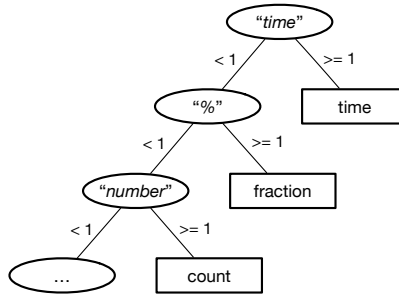

Fig. 3. Fragment of a decision tree

The purpose of the decision tree classifier is two-fold. First, we obtain a classifier as a means to classify the measure type of PPI in order to improve the quality of the alignments our approach generates. Second, we obtain a collection of type indicators $\mathcal{T}$, which are those terms that are used as nodes in the decision tree to distinguish between different measure types. We use these indicators to support the parsing of PPI descriptions, as described in Section 3.2

3.2 PPI Parsing

In order to align a PPI to a process model's elements, we extract the phrases of a PPI description that relate to specific parts of a process. To achieve this, we first split a PPI description into a number of phrases. Afterwards, we filter out those phrases that relate to the computation of a PPI's value rather than to elements of the process itself. In this section, we will use PPI6, *"% of rejected RFCs from all registered RFCs* as a running example.

Phrase Extraction We first divide a PPI description into constituent groups of words or *phrases*. To achieve this, we make use of the *Stanford Parser* [8], a widely employed natural language processing tool. The parser generates a *parse tree*, which captures the syntactic structure of a text in a hierarchical manner. Figure 4 provides an example of this for PPI6. A parse tree contains different types of phrases, e.g. prepositional phrases (denoted as PP), and noun phrases (NP), in a hierarchical structure. For the purposes of our alignment approach, we extract phrases that contain at most one (nested) noun phrase in its main clause. These phrases have a level of granularity similar to the granularity most commonly used in process models, where elements also generally contain a single noun [16]. For instance, most activity labels have a single noun in the form of a business object (e.g. an "RFC") on which an action (e.g. "approve") is performed. We augment the extracted main clauses with dependent clauses, if any, in order to capture information on resources that perform activities or on execution conditions. The latter is, for example, important if the computation of a PPI should only consider RFCs that have been rejected for a specific reason. For PPI6, the extraction step results in the following set of phrases P: {"%", "rejected RFCs", "from all registered RFCs"}.
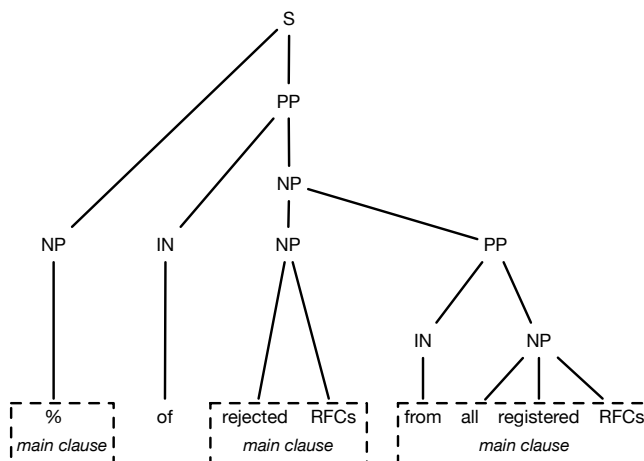

Fig. 4. Simplified parse tree for of PPI6

Phrase Filtering Next, we filter out those phrases in P that relate to the calculation of a PPI's value rather than to parts of the process. These, for example, include the phrase "*average lifetime*" in PPI2 and "*%*" in PPI6. We identify these phrases by considering the type indicators $\mathcal{T}$ obtained while training the decision tree used in the previously described step. These indicators represents keywords that exclusively relate to the computation of PPI values for a certain

measure type. Therefore, we identify a phrase that contains one or more of the terms in $\mathcal{T}$ as a phrase that relates to the calculation of a PPI. We thereby recognize that phrases such as "%" or "*average time*" do not relate to the process itself and as such should be excluded from consideration when creating an alignment. This approach has the great advantage that we filter phrases based on the automatically learned set of indicators $\mathcal{T}$, rather than depending on a manually defined catalog of keywords. For PPI6, this leaves the filtered set of phrases P_F as the outcome of this step: {"rejected RFCs", "from all registered RFCs"}.

3.3   Alignment to Process Model

In the final step of our approach, we generate an alignment between the extracted phrases P_F and the set of process model elements M. An alignment σ consists of a number of pair-wise correspondences, each between a phrase $p \in P_F$ and a process model element $m \in M$, denoted as $p \sim m$. Our approach sets out to find an optimal alignment $\hat{\sigma}$ between P_F and M, which we define as the alignment which (i) has the highest semantic similarity for its correspondences, and (ii) abides to constraints imposed based on the semantics of a PPI's measure type.

Semantic Similarity To quantify the semantic similarity between a phrase p and a model element m, we compare the *bag-of-words* representations of p and the textual label of m. To obtain this representation, we first apply a *tokenization* function on the plain texts. This function splits a text into its individual terms, filters out stop words like "*the*", "*if*", "*from*', and lemmatizes the remaining terms. This last step transforms all terms to their grammatical base form or *lemma*, e.g. "*is*" and "*been*" are both transformed into "*be*". We next compare the resultant bags-of-words, ω_m and ω_p, using a semantic similarity measure.

The usage of specific terminology from business settings, commonly contained in PPI descriptions and process models, poses an important challenge here. To overcome this challenge, we make use of a similarity method called *second order similarity* [7]. This method is based on the statistical analysis of co-occurrences in large text collections. It therefore has the great advantage that it can deal with context-specific terms, often not fully captured by other natural language processing tools suchs as WordNet [18]. To compute the similarity score between ω_p and ω_m, we make use of a metric introduced in [17], which combines second order similarity scores and the inverse document frequency (idf) of terms. By incorporating idf, the metric assigns higher scores to terms that have a high discriminatory power in a given process context. For instance, in the context of the request for change example, the rarely occurring term "*registered*" has a much higher discriminatory power than the frequently occurring term "*RFC*".

Alignment Constraints To generate an alignment in line with the semantics of PPIs, we impose constraints on the correspondences included in the alignment through a constraint function Γ. Specifically, we use Γ to capture constraints on

three characteristics: (i) the classes of process model elements to be included in σ, (ii) the number of correspondences or *cardinality* of σ, and (iii) the possible overlap among correspondences in σ.

Table 2. Constraints imposed on alignments per measure type

Measure Type	Model Elements	Cardinality	Overlap
Count	Flow elements, data objects	1	n/a
Data	Data objects	1	n/a
Time	Flow elements, data objects	2	depends on $|P_F|$
Derived	Flow elements, data objects	$|P_F|$	yes
Fraction	Flow elements, data objects	2	no

We instantiate the constraints for each type in accordance with their semantics, as specified in [24]. Table 2 provides an overview of the specific constraints imposed per measure type. The alignments generated for *count* and *data* measures are the least complex. Alignments of these types contain only a single correspondence between a phrase and model element in σ. For data measures, these elements can only include data objects, because these measures exclusively relate to attribute values of data objects. All other measure types can also relate to *flow elements*. These elements depict the steps executed in a process. For the BPMN notation, the most common flow elements include *activities* and *events*. The alignments for *time* and *derived* measures are more complex, because they can include multiple correspondences.

Time measures require start and end points. These two points may refer to the same model element, e.g. an activity, in order to describe a measure that computes the duration between the start and end of an activity. This is for instance seen for PPI3: "*The average duration of a committee decision*". We can identify these cases through the number of phrases extracted from the PPI description, i.e. $|P_F|$. If the description contains only a single phrase related to the elements of a process, we expect that the start and end points of a time measure refer to the same element. In those cases, we allow overlap between the correspondences. Otherwise, we generate an alignment that contains distinct correspondences for the start and end points.

Finally, derived measures allow for the widest variety in alignments, because these measures can describe any function over other measures. To capture this, we do not impose specific restrictions on the size of their alignments. Rather, we align each extracted phrase $p \in P_F$ to its most similar process model element. This allows the approach to generate a broad variety of alignments, in line with the semantics of derived measures. For *fraction* measures, a specific sub-class of derived measures, we do impose restrictions on the size of their alignments. A fraction measure requires distinct process model elements to reflect its *numerator* and *denominator*. Therefore, the cardinality of these measures always equals 2.

To obtain the optimal alignment $\hat{\sigma}$ for a PPI, we construct the alignment that has the maximum sum of similarity scores for its correspondences, while it still abides to the alignment constraints imposed by Γ. This alignment then represents the final outcome of our approach.

4 Evaluation

To demonstrate the strength of our alignment approach, we conduct a quantitative evaluation that compares the generated alignments to a manually created *gold standard*. The goal of the evaluation is to learn how well the automated approach approximates manual alignments. Section 4.1 introduces the data set used for the evaluation. Section 4.2 describes the details of the evaluation approach. Finally, we present and discuss the results in Section 4.3.

4.1 Test Collection

To evaluate our approach, we use a collection of process models and accompanying natural language PPI descriptions from practice. To allow for a high external validity of the evaluation results, the data in the test collection has been obtained from various sources. Part of the test collection consists of an industrial data set stemming from prior research on the formalization of PPI definitions and service level agreements [23,24]. The request for change example, used throughout this paper, provides a fragment of one of the models included in this collection. The test collection furthermore includes a number of process models and PPIs from the SCOR (Supply Chain Operations Reference) and ITIL (Information Infrastructure Technology Library) reference frameworks. From these frameworks we selected processes from various application contexts and with a high number of associated performance indicators. The resulting test collection consists of 15 different process models and a total of 173 PPIs. The PPIs in the collection comprise 65 count, 28 data, 47 time, and 33 derived measures. Table 3 presents an overview of the characteristics of the collection per source, including the average number of elements per process model, and the total number of correspondences between the PPIs and model elements.

Table 3. Overview of the test collection

ID	Source	Process models	Elements/model	PPIs	Correspondences
1	Industry	9	11.2	47	65
2	SCOR	3	8.3	86	138
3	ITIL	3	13.7	40	48
Total		15	11.1	173	251

Aside from the broad variety of domains it covers, the heterogeneity of the test collection mainly manifests itself in terms of granularity. The SCOR and ITIL process models represent reference models that are intended as templates for implementations in organizations. The models from the reference collection are more abstract and, thus, provide less fine-granular process descriptions.

4.2   Setup

To conduct the evaluation, we implemented the alignment approach in the form of a prototype. The Java prototype uses the Stanford Parser [8] to assist in the PPI parsing, the semantic similarity implementation DISCO [10], and the WEKA toolkit for classification [6]. Specifically, we apply the C4.5 algorithm [21] to generate a decision tree, one of the most commonly used implementations for decision tree learning. We train the decision tree on a collection of 300 PPIs from the SCOR and ITIL frameworks, for which we manually defined the measure types. To avoid any bias, the PPIs in this training collection are distinct from those used in the test collection. Furthermore, because the training collection is obtained from different processes and does not include PPIs from the industrial sources, the PPIs in the training and test sets differ considerably in terms of their domain, terminology, and structure.

We use our prototype to automatically generate alignments for the PPIs in the test collection. To assess the quality of the generated alignments, we compare them to a manually created *gold standard*. We involved three researchers in the creation of the gold standard for the industrial and ITIL collections. Two of them independently created the alignments. The differences were discussed in detail, involving a third researcher to settle ties. For the SCOR framework, we directly obtained the gold standard from the relations that the framework itself specifies between performance indicators and activities. To perform the comparison between the correspondences contained in the generated alignments $\mathcal{A}$ and those contained in the gold standard $\mathcal{R}$, we computed *precision* and *recall* metrics as given by Equations (1) and (2).

$$pre(\mathcal{A}, \mathcal{M}) = \frac{|\mathcal{A} \cap \mathcal{R}|}{|\mathcal{A}|} \qquad (1) \qquad\qquad rec(\mathcal{A}, \mathcal{M}) = \frac{|\mathcal{A} \cap \mathcal{R}|}{|\mathcal{R}|} \qquad (2)$$

Precision here reflects the number of correct generated correspondences, i.e. the correspondences from $\mathcal{A}$ that are also included in the gold standard $\mathcal{R}$, divided by the total number of generated correspondences. *Recall* is the fraction of correspondences in the gold standard that are correctly identified by our approach, i.e. included in the generated alignments. We furthermore report the f_1-score as the harmonic mean of precision and recall.

As we are the first to present an automated approach for the alignment of PPIs to process models, there is no commonly accepted benchmark available. To demonstrate the performance of our approach, we therefore compare its results

to a *baseline configuration*. For this baseline, we align each PPI to the process model element with the highest semantic similarity to the entire PPI description. Through this comparison, we are able to illustrate the added value of classifying and parsing the PPI descriptions instead of this straightforward, rough approach.

4.3 Results

Table 4 summarizes the evaluation results. It shows that the baseline configuration achieves a considerable precision for the total collection (0.75), but lacks in recall (0.51). This high precision can be attributed to the use of semantic similarity measures specifically suited to deal with specific terminology used in business settings. The lack of recall follows from the low number of correspondences the baseline configuration generates (173). The full approach avoids this problem by classifying the measure types of the PPIs. Through this classification, the approach much better approximates the number of correspondences to be included in the alignments. It generates 255 correspondences versus 251 included in the gold standard. The slightly higher precision achieved by the full approach (0.76) is remarkable, because it generates a significantly higher number of correspondences than the baseline. This achievement can be attributed to the extraction and filtering of phrases in the PPI parsing step. Because the parsing step removes extraneous information from consideration, the generated similarity scores are more accurate. The full approach therefore manages to maintain a high predictive precision. The increased number of correspondences, together with the stable precision, results in considerable improvements in recall (0.75 versus 0.51) and F_1 (0.76 versus 0.60).

Table 4. Evaluation results

Configuration	Source	$|\mathcal{A}|$	Precision	Recall	F_1-score
Baseline	Industrial	47	0.79	0.52	0.63
	SCOR	86	0.74	0.46	0.57
	ITIL	40	0.73	0.60	0.66
	Total	173	0.75	0.51	0.60
Full approach	Industrial	70	0.73	0.72	0.72
	SCOR	139	0.77	0.78	0.77
	ITIL	40	0.78	0.75	0.77
	Total	255	0.76	0.75	0.76

The evaluation results suggest that the classification of measure types and the tailored technique for parsing PPI descriptions greatly improve the quality of the generated alignments. A post-hoc analysis of the results reveals that alignments which depend on context-specific information present the most important challenge to the automated approach. This challenge manifests itself in the form of

PPI descriptions that refer to process concepts that are only related in a specific context. For instance, the ITIL process on Service Design contains the "*Average duration of service interruptions*" PPI. This PPI relates to an "*availability monitoring and reporting*" activity in the accompanying process model. To identify the correct correspondence, it must be recognized that service *interruptions* affect the *availability* of services. Still, due to the usage of semantic similarity measures, our approach successfully identifies the vast majority of such cases, in which PPIs and process models do not refer to the same concepts.

5   Related Work

The work presented in this paper mainly relates to two research streams. One is focused on the problem domain and includes different models to define the relationships between PPIs and process models. The other is focused on the solution domain and includes techniques that have been developed to automatically align process information between different artifacts.

Concerning the former, there are a number of frameworks for modeling PPIs and their relationship with business processes. For instance, Popova et al. [20] present a framework for modeling PPIs within a general organization modeling framework. The framework provides an explicit mechanism to link PPIs with process models. Momm et al. [19] introduce an approach, based on the principles of Model-Driven Architecture, for the development of infrastructure necessary to instrument the monitoring of a set of PPIs in a Service-Oriented Architecture. Wetzstein et al. [26] introduce a Key Performance Indicators (KPIs) ontology to specify KPIs over semantic business processes as part of a framework for Business Activity Monitoring. Finally, PPINOT [24] presents a metamodel to define PPIs with a high degree of expressiveness and an explicit link with process model elements. Although these frameworks provide mechanisms to link PPIs with process models, it was found that in practice, managers often start out to describe relevant PPIs in an unstructured and ad-hoc manner [24,26]. Our approach is, therefore, complementary to these frameworks. Based on these existing, unstructured PPI definitions, the approach can generate the links that are necessary to define PPIs in accordance with the structured notations of the frameworks.

To the best of our knowledge, there are no earlier methods that generate alignments in this context. By contrast, numerous approaches, referred to as *process model matchers*, exist that create alignments between different process models, e.g. [4,9,13]. To create alignments these matchers exploit different process model features, including natural language [9], model structure [4], and behavior [13]. Process model matchers face challenges similar to our approach in the form of different levels of detail and the usage of different terminology [1]. Contrary to the unstructured natural language descriptions used as input in this work, these matchers work with explicitly structured input. An exception to this is an earlier proposed approach, which aligns textual process descriptions to process models for the purpose of inconsistency detection [2]. However, the nature

of the input considerably differs from the PPI descriptions used in the presented work. This results in distinct parsing and alignment challenges.

6   Conclusions

In this paper, we presented an approach to automatically align natural language descriptions of PPIs to process models. To achieve this, our approach combines machine learning and tailored natural language processing techniques to deal with the variability of natural language and the different measures types of PPIs. A quantitative evaluation, conducted using a test collection obtained from various industrial sources, demonstrated that the approach generates alignments of a high quality. These generated alignments show a high level of similarity to manually created ones. The approach thus accurately identifies relations between textually described PPIs and process models in practical settings. As such, it successfully supports the operationalization of process performance monitoring. Despite the promising results, we need to reflect on some limitations. First, the dataset we employed is not representative in a statistical sense. However, the obtained result quality is stable among the processes from different sources, which illustrates the approach's ability to deal with heterogeneous data. Second, the approach does not generate perfect alignments in all cases, especially not when the link between PPI and process model depends on a considerable amount of contextual knowledge. The approach, therefore, remains a means to support users. It does have the potential to greatly reduce the effort required to identify correct correspondences for a process collection.

In future work, we set out to further develop the alignment approach. A promising direction is to develop extraction techniques tailored to the different PPI measure types in order to further improve the results. Second, the approach can be extended by also parsing the information in a description that relates to the calculation of a PPI's value. As such, the generated alignments can be extended into fully formalized PPI definitions.

References

1. Van der Aa, H., Leopold, H., Mannhardt, F., Reijers, H.A.: On the fragmentation of process information: Challenges, solutions, and outlook. In: Enterprise, Business-Process and Information Systems Modeling, pp. 3–18. Springer (2015)
2. Van der Aa, H., Leopold, H., Reijers, H.A.: Detecting inconsistencies between process models and textual descriptions. In: Business Process Management, pp. 90–105. Springer (2015)
3. Achour, C.B.: Guiding scenario authoring1. Information Modelling and Knowledge Bases X 51, 152 (1999)
4. Dijkman, R.M., Dumas, M., Van Dongen, B., Käärik, R., Mendling, J.: Similarity of business process models: Metrics and evaluation. Information Systems 36(2), 498–516 (2011)
5. Gal, A.: Uncertain schema matching. Synthesis Lectures on Data Management 3(1), 1–97 (2011)

6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD 11(1), 10–18 (2009)
7. Islam, A., Inkpen, D.: Second order co-occurrence pmi for determining the semantic similarity of words. In: Proceedings of the International Conference on Language Resources and Evaluation, Genoa, Italy. pp. 1033–1038 (2006)
8. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting of the ACL-Volume 1. pp. 423–430. ACL (2003)
9. Klinkmüller, C., Weber, I., Mendling, J., Leopold, H., Ludwig, A.: Increasing recall of process model matching by improved activity label matching. In: Business Process Management, pp. 211–218. Springer (2013)
10. Kolb, P.: Disco: A multilingual database of distributionally similar words. Proceedings of KONVENS-2008, Berlin (2008)
11. Kovacic, A.: Business renovation: business rules (still) the missing link. Business process management journal 10(2), 158–170 (2004)
12. Kronz, A.: Managing of process key performance indicators as part of the aris methodology. In: Corporate performance management, pp. 31–44. Springer (2006)
13. Künze, M., Weidlich, M., Weske, M.: Behavioral similarity–a proper metric. In: Business Process Management, pp. 166–181. Springer (2011)
14. Luftman, J., Papp, R., Brier, T.: Enablers and inhibitors of business-it alignment. Communications of the AIS 1(3es), 1 (1999)
15. Marshall, B., Chen, H., Madhusudan, T.: Matching knowledge elements in concept maps using a similarity flooding algorithm. Decision Support Systems 42(3), 1290–1306 (2006)
16. Mendling, J., Reijers, H.A., Recker, J.: Activity labeling in process modeling: Empirical insights and recommendations. Information Systems 35(4), 467–482 (2010)
17. Mihalcea, R., Corley, C., Strapparava, C.: Corpus-based and knowledge-based measures of text semantic similarity. In: AAAI. vol. 6, pp. 775–780 (2006)
18. Miller, G.A.: WordNet: a lexical database for english. Communications of the ACM 38(11), 39–41 (1995)
19. Momm, C., Malec, R., Abeck, S.: Towards a model-driven development of monitored processes. In: Wirtschaftsinformatik (2). pp. 319–336 (2007)
20. Popova, V., Sharpanskykh, A.: Modeling organizational performance indicators. Information Systems 35(4), 505–527 (2010)
21. Quinlan, J.R.: C4. 5: programs for machine learning. Elsevier (2014)
22. del Río-Ortega, A., Cabanillas, C., Resinas, M., Ruiz-Cortés, A.: Ppinot tool suite: a performance management solution for process-oriented organisations. In: Service-Oriented Computing, pp. 675–678. Springer (2013)
23. del Río-Ortega, A., Gutiérrez, A.M., Durán, A., Resinas, M., Ruiz-Cortés, A.: Modelling service level agreements for business process outsourcing services. In: Advanced Information Systems Engineering. pp. 485–500. Springer (2015)
24. del Río-Ortega, A., Resinas, M., Cabanillas, C., Ruiz-Cortes, A.: On the definition and design-time analysis of process performance indicators. Information Systems 38(4), 470–490 (2013)
25. Rosemann, M.: Potential Pitfalls of Process Modeling: Part A. Business Process Management Journal 12(2), 249–254 (2006)
26. Wetzstein, B., Ma, Z., Leymann, F.: Towards measuring key performance indicators of semantic business processes. In: BIS. pp. 227–238 (2008)