

Extracting Business Objects and Activities from Labels of German Process Models

Philip Hake^{1,2(✉)}, Peter Fettke^{1,2}, Günter Neumann²,
and Peter Loos^{1,2}

¹ Institute for Information Systems, Saarland University, Saarbrücken, Germany
{philip.hake, peter.fettke, peter.loos}@dfki.de

² German Research Center for Artificial Intelligence, Saarbrücken, Germany
neumann@dfki.de

Abstract. To automatically analyze and compare elements of process models, investigating the natural language contained in the labels of the process models is inevitable. Therefore, the adaption of well-established techniques from the field of natural language processing to Business Process Management has recently experienced a growth. Our work contributes to the field of natural language processing in business process models by providing a word dependency-based technique for the extraction of business objects and activities from German labeled process models. Furthermore, we evaluate our approach by implementing it in the RefMod-Miner toolset and measuring the quality of the information extraction in business process models. In three different evaluation scenarios, we show the strengths of the dependency-based approach and give an outlook on how further research could benefit from the approach.

Keywords: Business process modeling · Information extraction · Language processing · Business process management

1 Introduction

Beside the structure and process semantics contained in business process models, the language contained in the labels represents an important factor when it comes to describing the business activities. To automatically process these models in a way that the underlying process semantics is considered, i.e. analyzing, comparing, matching or even refactoring models, the contained natural language should be investigated. The automatic processing of natural language in shape of textual representations is dedicated to the field of computational linguistics. Over the past decade, a remarkable set of processing techniques for natural language texts have been successfully applied to various problems such as Optical Character Recognition (OCR), named entity recognition and sentiment analysis [1]. However, recent BPM (Business Process Management)-driven approaches have revealed shortcomings in the applicability of well-established natural language processing (NLP) techniques in the context of automatic language processing in business process models. These shortcomings are caused by the different languages that are used to describe processes in terms of process models and textual descriptions. While the latter contains what is known as natural language, the

former can be considered a slightly controlled language, which is less complex regarding the sentence structure. Due to the mismatch of the two languages, the applicability of the techniques that are based on natural language models cannot be ensured. Therefore, a variety of approaches faces the challenge of processing the language contained in process model labels by successfully extending existing NLP techniques to process model-specific language characteristics. Among these are techniques dedicated to the detection of business objects and activities [2], which is fundamental regarding the comparison [3] and matching of process models [4]. While most of the BPM NLP approaches focus on processing the English language, [5] discusses characteristics of other natural languages. Since the language contained in the process models usually depends on a company's location, either a translation of the models or an adaption of existing techniques must be considered to automatically process these models.

Therefore, we aim at providing a German language-based approach for detecting business objects and activities in business process models. Our approach answers the following research question:

(RQ) How and to what extent can business objects and activities be extracted from German process models applying state of the art NLP techniques?

We address the research question by developing and evaluating a detection approach based on the insights from English extraction approaches and state-of-the-art NLP techniques. We contribute to the field of natural language processing in business process models by providing and assessing an approach for detecting business objects and activities of German process models. Our work follows a design science-oriented methodology. We aim at extending existing knowledge about the extraction of information from business process models. Furthermore, we propose a novel artifact, which will be beneficial for research fields relying on language processing in process models. We evaluate the artifact by providing an implementation and empirical evaluation using three different scenarios.

The remainder of the paper is structured as follows: In Sect. 2 we introduce BPM NLP foundations, the problem statement and our methodological approach. In Sect. 3 we present our approach for detecting business objects and activities. In Sect. 4 we present our evaluation, consisting of an implementation and an empirical investigation in three different scenarios. Section 5 analyzes and discusses the evaluation result. Finally, we conclude our work in Sect. 6 and outline impacts on future research.

2 Foundations

2.1 Natural Language Processing

To automatically detect business objects and activities in labels of process models, the language contained in the labels needs to be investigated. There exists a variety of modeling guidelines [6, 7] and artificial languages used for business process modeling, which significantly influence the natural language used in business process models. Moreover, there are fundamental methods and frameworks known to the field of NLP

that are suitable for processing natural language contained in any process model. An extensive overview of approaches to processing language in business process management is provided in [8].

These approaches are based on manifold linguistic techniques and resources. Approaches investigating the language contained in labels of process models depend on filtering, parsing and chunking the labels, to segment the contained language. The segmentation [9] covers simple techniques, which obtain a list of words by splitting labels at whitespace characters, but also sophisticated machine learning-based approaches, which are trained on extensive linguistic datasets.

The segmentation of a label is crucial to automatically examine the semantic relation between labels. Approaches investigating the similarity of labels require information about semantic relations between single words. The synonym relation between two words is a widely-used relation type and enables approaches to identify similar labels, e.g. *create invoice* and *generate bill* [10]. However, looking up the relation in a database requires the words contained in the label to be available in its basic form known as *lemma*. Depending on the language and provided context, the automatic derivation of a lemma remains a challenging task.

Moreover, there are approaches investigating the syntactic structure of labels. Based on an identified word category, e.g. noun, verb, adjective, a category-based comparison of labels is conducted [9]. The derivation of these word categories reaches from simple techniques depending on dictionary lookups to complex heuristic techniques using pre-trained language models. Determining the syntactic word-category of the words contained in a sentence is called *part-of-speech (POS) tagging*.

In [11] the authors propose a framework for processing natural language in process models. This framework also covers the techniques presented in [8]. The components of this framework describe a toolchain of specific techniques and resources addressing several mostly high-level challenges, e.g. Named-Entity Recognition. However, this framework is not applicable to our approach since we use low-level NLP techniques, which are omitted in the framework. Given a corpus of process models in a machine-readable format, information extraction requires several chained NLP techniques including *parsing* and *tagging* the labels, as well as analyzing the contained words and their dependencies within the label. Each of the chained techniques influences the results of consecutive processing steps.

2.2 Information Extraction in Business Process Models

The goal of information extraction is to transform unstructured information contained in natural language texts into structured data. Information extraction refers to manifold linguistic problems such as named entity recognition, temporal analysis or relation detection and classification [1]. The approach proposed in this work is considered as classification and relation detection problem since we aim at extracting embedded objects and activities and reveal their relation. While [12] relies on a manual processing of labels to extract a business vocabulary, we focus on an automated information

extraction approach. We denote a business object as a tangible or intangible artifact, which is described in a node label of a process model. An activity describes the manipulation, usage or generation of a business object within a label. An activity is always associated with a business object and vice versa.

Before we introduce our novel approach, we will revisit an approach for detecting process model labeling styles [8] since we rely on the same linguistic patterns identified in German business process models. The approach was recently used to detect labeling style violations in business process models and is also known for its applicability to extracting information from English business process models. The approach investigates the syntax of the labels. Based on the detected labeling style, segments of the label can be declared as objects and related activities. Since they aimed at measuring the style detection performance, the evaluation of the additionally provided techniques for extracting business objects and activities were neglected. The proposed labeling styles represent predefined syntactic patterns. Table 1 presents the patterns and the respective labeling styles for German and English labels. Based on the patterns, the authors propose a heuristic matching of labels to the proposed patterns. In case that a label starts with a verb in infinitive form, the label is assigned to the verb object style. The identification of the respective verb form is conducted via a lookup in a pre-tagged corpus. Since the syntactic function of a word might be ambiguous, the authors propose a disambiguation technique. Thus, the precision of the label style detection depends on the syntactic disambiguation and on whether a word can be looked up. After matching the label to a pattern, the business object and activities are determined. Given a label of verb object style, the identified verb can be denoted as the label’s activity. The next object that follows the identified verb is denoted as business object.

Table 1. Activity labeling styles [8].

Style	Pattern	Language	Example
Verb object	verb (imperative) + noun	en, ger	Create invoice
			Erstelle Rechnung
Infinitive style	verb (infinitive) + noun	ger	Erstellen Rechnung
Objective-infinitive	noun + verb (infinitive)	ger	Rechnung erstellen
Action-noun (NP)	noun + noun	en	Invoice creation
Action-noun (of)	noun + ‘of’ + noun	en	Creation of invoice
Action-noun (gerund)	verb (gerund/nominalization) + [article] + noun	en, ger	Creating invoice
			Erstellung der Rechnung
Action-noun (irregular)	Anomalous	en, ger	–
			–
Descriptive	[noun] + verb (3P) + noun	en, ger	Mitarbeiter erstellt Rechnung
			Clerk creates invoice
No-action	Anomalous	en, ger	–

2.3 Problem Statement

The proposed derivation of business objects and activities in [13] is designed for the English language and covers action-noun and descriptive labels, as long as they do not contain a coordinate conjunction. We consider the evaluation of this derivation approach implicit, since the authors explicitly measure the label refactoring quality, which relies on the derivation results, but does not focus on the derivation results. In the following, we will examine German language characteristics that influence the adaption of the proposed conceptual considerations regarding an extraction of business objects and activities.

Contrary to the English language, the German language does not suffer as much from syntactic ambiguity. Capitalized nouns make it easier to determine the syntactic function of a word within a label. However, investigating the techniques and resources provided in [2, 8] reveals further challenges for the German language regarding the determination of the syntactic function, and consequently, for the extraction of business objects and activities:

- (1) The resolution of verb forms to their respective infinitive form is based on pre-tagged English and German corpora. While they allow a resolution of English gerunds to their infinitive form (*creating* \rightarrow *create*), this relation is not maintained for the German language (*Erstellung* \rightarrow *erstellen*) regarding nominalizations.
- (2) Furthermore, an investigation on the general applicability of lookup approaches revealed that the used corpora are likely to miss domain-specific knowledge. For instance, the widely-used TIGER corpus is based on texts published in the newspaper *Frankfurter Rundschau*. Considering the German language, a lookup approach highly depends on the maintained compound words within the corpus. Therefore, providing a domain independent technique is necessary to ensure an accurate detection of business objects and activities.
- (3) Beside the German characteristics, the techniques introduced in [2] are not able to determine multiple objects and activities in more complex sentence structures such as conjunctive sentences. Furthermore, they rely on the part of speech information of each single word but do not investigate the dependencies of the words within the label.

Finally, the detection of business objects and activities has only been implicitly evaluated regarding the *action-noun* and the *descriptive* labeling styles. Hence, we aim at developing and evaluating an information extraction approach for German business process models to identify business objects and activities.

3 Business Object and Activity Extraction

Our approach is based on the insights concerning the different linguistic patterns used in German business process models [8]. The approach is subdivided into two processing steps. The first step consists of state of the art language processing techniques to investigate the syntactic structure of a label, i.e. the POS tags and the syntactic dependencies.

Based on the gathered syntactic information, we derive the business objects and activities in the second step. Since the language models and techniques are trained on natural language, the correct detection of the *POS tag* and the *dependencies* in short labels (less than three words) are difficult. Therefore, we also apply a rule-based detection for short labels using pre-tagged linguistic corpora.

3.1 Step 1: Language Processing Pipeline

Step 1 includes the extraction of POS information as well as the syntactic semantics between the words. Figure 1 shows the information we obtain by applying the POS tagger and dependency parser to the label *Es liegt eine digitale Rechnung vor* (*A digital invoice was received*). We obtain a list of identified words called *tokens*, the respective part-of-speech tags and the dependencies between these words. Using the derived POS information, we know that the label contains a word tagged as noun (NN) *Rechnung* and a finite verb (VVFIN) *liegt*. The directed edges depicted in Fig. 1 describe the dependencies between the identified words. Additionally, we receive the type of dependency, e.g. direct object or genitive object. Hence, we know that the noun *Rechnung* is a direct object that depends on the finite verb *liegt*.

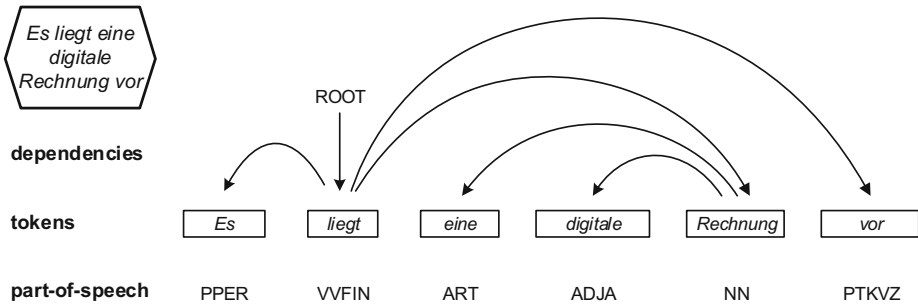


Fig. 1. Derived tokens, dependencies and part-of-speech information

In order to achieve the necessary degree of efficiency and robustness, our label analysis is based on DFKI's (German Research Center for Artificial Intelligence) multilingual statistical-based dependency analysis framework, called *MDParser* [14]. *MDParser* consists of a complete pipeline of tools for *text segmentation*, *tokenization*, *POS tagging*, morphological tagging, named entity (NE) recognition, and syntactic dependency analysis. There exist only a few other available similar complete NLP pipelines, notably, the Stanford dependency parser, which has also been applied to the detection of process model labeling styles [13]. In a recent comparison, *MDParser* was shown to be more than 5-times faster than the latest version of the Stanford dependency parser [15] and achieved a better performance on the tested universal dependency treebanks [16]. *MDParser* is a lightweight easy to train and applicable system, e.g., the POS tagger and parser component have been trained and tested on more than 52

languages (using the Universal Dependency treebanks v1.3.¹ Therefore, the conceptual considerations of our approach are transferable to an extensive number of languages.

The core of MDparser consists of a general name tagger (*GNT*) and the dependency parser *MDP*. Both are based on the same statistical-based Machine Learning engine (LIBLINEAR, [17]) and share a common data and annotation schema for training and application, i.e. the CONLL data format.² *GNT* is used for training and applying models for *POS tagging*. In contrast to comparable NLP applications in BPM, which also focus on an automated language processing, the applied parser is capable of handling information in brackets, punctuations and other special characters. Due to the observed difficulties using NLP parsers in [13], we employ the MDParser because of the performance measured in several benchmarks. The MDparser achieved 97.30% on the TIGER 2.2 test set (compared to 97.73% for the currently best result reported by [18]). Similarly, we achieve an F1 score of 73.91% on the GermEval 2014 dataset, which would have been the 3rd best result according to Metric3 [19].

The MDP dependency parser is currently one of the fastest statistical-based dependency parsers available - more than 5,000 sentences/second [16], which also enables a real-time application of our information extraction approach.

3.2 Step 2: Business Object and Activity Derivation

To derive the business objects and activities, we rely on the dependencies and the POS tags gathered in the previous step. At first, we investigate the label regarding the contained activity. Based on the activity, we determine the business object of a label. Based on the introduced German labeling styles (Sect. 2.2), we assume that the activity is contained as a verb *prüfen* (*examine*), a simple nominalization *Prüfung* (*examination*) or a compound nominalization *Rechnungsprüfung* (*invoice verification*). In case that a label contains different types of activities, we define an order based on which we apply the selection of the activity.

A verb dominates a simple nominalization and a simple nominalization dominates a compound nominalization. Our approach extracts *beenden* (*close*) as activity in the label *Rechnungsprüfung beenden* (*close invoice verification*), since the compound nominalization *Rechnungsprüfung* is dominated by the verb *beenden*. If multiple activities of the same type occur and they are not dominated by another activity type, they are denoted as activities. In the label *Rechnungsprüfung und Qualitätskontrolle* (*invoice verification and quality check*) both compound nominalizations are identified as activities.

Since simple nominalizations and compound nominalizations are tagged as nouns, the distinction of nouns not describing an activity is difficult. A verb, however always describes an activity and only a few ambiguities considering the syntactical function exist, e.g. *Pflege* (*care*) as a noun and as an imperative form at the beginning of a sentence.

Therefore, we apply different dependency-based extraction approaches for both activity cases. At first, we obtain the tokens, POS tags, word dependencies and

¹ cf. <http://universaldependencies.org/>.

² cf. <http://ufal.mff.cuni.cz/conll2009-st/task-description.html>.

dependency types from the MDparser (Fig. 1). Then, we execute the dependency-based label analysis (DLA). If no activity could be determined, we apply an additional heuristic approach. Algorithm 1 describes the first dependency-based extraction. We determine the verbs contained in the description using the POS tag information. Our analysis not only covers the activity detection in verb object labels, but also the activities contained in infinitive style, objective infinitive style as well as the descriptive style and irregular labels. To identify a verb in the label, we use the function *getNextVerbPos*, which returns the position of the next verb in the list of tokens. If a verb is identified, the verb is added to the set of activities. Afterwards, we investigate the dependencies of the activity to nouns contained in the label to derive the business objects. If the identified verb and a noun are related regarding the derived dependencies, the noun is denoted as business object. Before adding the obtained business object to the result set, we call *getConj* to extract further related nouns based on a conjunction dependency.

Algorithm 1: DLA

Input: tokens, tags, depends, depTypes

Output: ({objects}, {activities})

```

activities:={};
objects:={};
while verbExists(tags)
    verbPos:=getNextVerbPos(tags);
    conjA:=getConj(verbPos,depends,depTypes);
    activities=activities U {tokens[verbPos]};
    while nounExists(tags)
        nounPos:=getNextNounPos(tags);
        object;
        if rootChildren(verbPos,nounPos,depends)
            object=tokens[nounPos];
        else if childParent(verbPos,nounPos,depends)
            object=tokens[nounPos];
        else if childParent(nounPos,verbPos,depends)
            object=tokens[nounPos];
        conjO:=getConj(nounPos,depends,depTypes,tokens,tags);
        objects= objects U conj U {object};
while nounExists(tags)
    nounPos:=getNextNounPos(tags);
    if genitive(depTypes[nounPos])
        depPos:=depends[nounPos];
        conjA:=getConj(depPos,depends,depTypes,tokens,tags);
        conjO:=getConj(nounPos,depends,depTypes,tokens,tags);
        activities=activities U {tokens[depPos]} U conjA;
        objects=objects U {tokens[nounPos]} U conjO;
return ({objects},{activities});

```


Hence, in the description *Rechnung und Qualität prüfen* (*check invoice and quality*) not only *Qualität* but also *Rechnung* is identified as business object, as both words exhibit a conjunction dependency. Our approach is also capable of identifying *Rechnung* and *Ware* in the label *Rechnung/Ware auf Vollständigkeit prüfen*.

Since German labels contain activities masked as nouns, we also examine the contained nouns if no verbs could be identified. Analogously, in [13] a label is assigned the action noun style if no verb could be identified. We then check the dependency of each noun contained in the label. The *genitive* function checks whether the noun holds a genitive relation to any other token in the label. If the genitive rule applies, the respective noun is denoted as business object. The related token of the noun is declared as business activity. Thus, in the label ‘Erstellung der Rechnung’ the activity ‘Erstellung’ and the object ‘Rechnung’ are identified. Furthermore, using *getConj*, we identify further objects and activities with a conjunction relation. If no activity is identified using Algorithm 1, we apply the heuristic label analysis (HLA). This algorithm covers the case that the linguistic processing does not identify a contained verb form. This mainly applies for short labels containing only two words. Thus, the identification of the business objects and activities requires lookups in a pre-tagged corpus. To minimize errors due to missing entries in the corpus, we apply multiple investigations of the label.

Given a label with two words, we assume that the second word either represents a noun referring to an activity or object, or a verb representing an activity. The first word is either an adverb, an adjective, an activity masked as verb or noun or an object represented by a noun. Identifying the position of the verb is the best way to also correctly identify the object. If the label contains two words, we look up the POS tags of the second token in a corpus. The function *isVerb* checks whether the token appears as a verb in the corpus. Using *isAd* we look up whether the first token is contained as adjective or adverb in the corpus. If the first word is an adverb or an adjective, the second token is denoted as activity. Otherwise, we additionally declare the first token an object. If the first token of the label is a verb according to the corpus look-up, the first word becomes the activity and the second word the object. If neither the first nor the second token could be identified as a verb, we declare the first token as an object if it is not contained as an adjective or adverb in the corpus and declare the second token the corresponding activity. If the label contains more than 2 words, we consider the first verb in the tokens as activity and the first noun as object. If still no verb is identified, we pick the first noun as activity and the second noun as object.

Algorithm 2: HLA

Input: tokens, tags, depends, depTypes

Output: ({objects}, {activities})

```

activities:={};
objects:={};
if length(tokens)==1
    activities=activities U {tokens[0]};
else if length(tokens)==2
    # check if second token is a verb
    if isVerb(tokens[1])&&!isAd(tokens[0])
        activities=activities U {tokens[1]};
        objects=objects U {tokens[0]};
    else if isVerb(tokens[0])
        activities=activities U {tokens[0]};
        objects=objects U {tokens[1]};
    else
        if isAd(tokens[0])
            activities=activities U {tokens[1]};
        else
            activities=activities U {tokens[0]};
            objects=objects U {tokens[1]};
    else
        if hasNextVerbCorpus(tokens)
            act:=getNextVerbCorpus(tokens);
            activities=activities U {activity};
            if hasNextNounCorpus(tokens)
                object:=getNextNounCorpus(tokens);
                objects=objects U {object};
            else if hasNextNoun(tokens)
                activity=getNextNounCorpus(tokens);
                object=getNextNounCorpus(tokens);
                activities=activities U {activity};
                objects=objects U {object}
        return ({objects},{activities});

```

4 Evaluation

4.1 Evaluation Setup

We evaluate our information extraction approach for German business process models using three different evaluation scenarios with different sets of German process models and two different corpora containing natural language. To evaluate our approach, we

implement the dependency-based and the heuristic extraction in the RefMod-Miner toolset.³ We annotate each activity node of a process model, to provide a gold standard for information extraction approaches. Therefore, we tag each word describing a business object or business activity. The creation of the gold standard is an iterative procedure and involves two process modeling experts. At first, the experts are provided the definitions of business objects and activities as given in Sect. 2.2. Then, the experts are told to independently tag the process model nodes and indicate the words representing business objects and activities. Afterwards, the experts discuss all labeling decisions. If there is a consensus regarding the labeling decisions, the gold standard is achieved. Otherwise, the next iteration starts and the experts relabel all the nodes based on the insights acquired from the discussion.

We evaluate our approach, comparing the achieved results with the derived gold standard. To measure the performance of our approach, we use Precision, Recall and F-measure, which have already been applied in the field of process matching and label style detection. Hence, we denote an extracted information, i.e. object or activity, from a label as *true positive* if the respective information is contained in the gold standard of this label. An extracted information of a label is denoted a *false positive* extraction, if the respective gold standard does not contain the extracted information. In case the gold standard of a label contains an information, which is not identified by our approach, the neglected information is considered a *false negative* extraction.

Since we aim at measuring scenarios containing several models, we define *TP* the number of *true positive* extractions, *FP* the number of *false positive* extractions and *FN* the number of *false negative* extractions within a given set of process models. We define Precision, Recall and F-measure as follows:

$$Precision = TP / (TP + FP) \quad (1)$$

$$Recall = TP / (TP + FN) \quad (2)$$

$$F\text{-measure} = 2 * (Precision * Recall) / (Precision + Recall) \quad (3)$$

In our experimental evaluation, the processes are modeled as event-driven process chains containing functions, events and connectors. We extract the business objects and activities from the function nodes since events tend to describe states rather than activities. Each model set describes domain-specific business knowledge. The first scenario (S1) contains 56 models of the SAP R3 reference model [20]. The second scenario (S2) consists of 60 models of the Retail-H reference model [21]. While S1 and S2 represent real process models, the third set contains 25 artificial process models, each representing the same business process. These models were derived by graduate students in an exam.

Based on a provided textual description of the process, the students were told to model the information contained in the text using the event-driven process chain. Figure 2 shows the distribution of different label sizes. The label size is denoted as the number of words which are acquired by splitting up the label at each whitespace. Since

³ <http://refmod-miner.dfki.de>.

Table 2. Evaluation of the scenarios S1, S2 and S3.

	S1 (SAP R3)	S2 (Handels H)	S3
Domain	Industry	Retail	Artificial
# models	56	60	25
# function labels	608	557	283
AVG words per label	2.79	3.87	2.77
SD words per label	1.45	1.85	1.25
# objects in gold standard	501	564	253
# max. objects per label	3	3	2
# activities in gold standard	619	643	289
# max. activities per label	2	3	3

the scenarios do not contain labels of size 11 and 13, we omitted the respective bars due to space constraints.

We give an overview of the label size distribution to analyze the influence of the different label sizes on the performance of our approach. Across the three scenarios, we observe a decreasing number of labels with four words and more. Moreover, in S1–S3 the number of labels of size two is higher than the number of labels of size one. While in S2 and S3 labels of size one represent 1% and 8% of all function labels, in scenario S3 16% of all function labels contain only one word. In S1 and S3 more than 50% of the labels (56% and 54%) are of size one or two, while S3 contains 32% of labels of the respective sizes. Table 2 shows the characteristics of the model sets as well as the derived gold standards. We chose three different scenarios to cover different maturity levels of process models as well as the specialization of the process.

We chose S1 and S2, since they represent reference models of two different domains, i.e. industry and retail. Furthermore, they hold a high maturity level, which also affects a consistent labeling style. The process of S3 describes the customer handling in a mileage bonus program and is chosen because of its narrow domain of application and low maturity level. Contrary to S1 and S2, the labeling does not follow a consistent style and the models contain a rather specific vocabulary and spelling errors. Therefore, we expect S3 to be the most challenging scenario, especially for our heuristic analysis. We use GermaNet 11.0 [22] and the TIGER corpus (v 2.2) [23] to derive POS information for the heuristic label analysis. Both have been successfully applied to the derivation of syntactic functions in labeling style detection and process matching.

GermaNet covers more than 140,000 lexical units. The TIGER corpus contains 50,000 distinct sentences consisting of approximately 900,000 tokens that are pre-tagged. Both corpora were created and verified by humans. In contrast, the correctness of the gathered POS tags and dependencies of our approach cannot be ensured. We evaluate the extraction of business objects and activities individually. Furthermore, we also investigate the impact of the applied algorithms in our approach. We compare the performance of the dependency-based label analysis (DLA) and the combination of DLA and the heuristic label analysis (HLA), which we denote as +HLA.

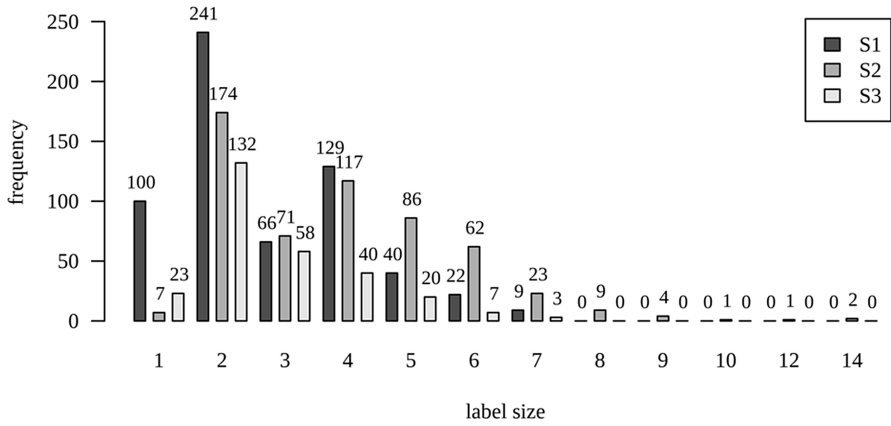


Fig. 2. Distribution of different label sizes across the scenarios.

4.2 Results Analysis

In the following, we report the results of our experimental evaluation. The evaluation of the extraction techniques of [13] applied to our scenario proves to be difficult, especially since they are not designed to handle *verb-object* style labels. By design, these algorithms are only capable of processing *action-noun* and *descriptive* labels. Moreover, the described dictionary-based concepts [8] to process the labels are only applicable for action-noun labels not containing nominalizations. This is not a conceptual problem of the approach per se, but rather the lack of German linguistic resources containing the relation between a nominalized verb and the verb itself. Only a small portion of action-noun labels could be processed correctly. Thus, a fair comparison of both approaches is not possible.

Table 3 summarizes the results of the business object and activity extraction step and shows the impact of our two configurations on the quality of the results. Across all scenarios, the +HLA configuration of the approach achieves the best results regarding the F-measure, ranging from 66% up to 92%. Table 3 also shows that DLA performs well in scenarios S1 and S3, while it only achieves moderate results in scenario S2.

Table 3. Results of the business object and activity extraction step.

	Algorithm	Objects			Activities		
		P	R	F	P	R	F
S1	DLA	0.91	0.90	0.91	0.91	0.74	0.82
	+HLA	0.90	0.91	0.91	0.92	0.91	0.91
S2	DLA	0.42	0.23	0.30	0.19	0.08	0.12
	+HLA	0.75	0.81	0.78	0.69	0.62	0.66
S3	DLA	0.94	0.89	0.91	0.88	0.84	0.86
	+HLA	0.93	0.90	0.91	0.85	0.88	0.87

P = Precision, R = Recall, F = F-measure

A significant gain in performance can be observed through the application of the heuristic analysis in S2. The additional heuristic analysis increases the F-measure by 48% regarding the objects and 54% regarding the activities. In the other scenarios, a gain of at most 9% can be observed. Scenario S2 appears to be the most challenging scenario regarding the dependency-based extraction. Furthermore, we analyze the increased F-measure of the +HLA configuration for the activity detection of S1 and S2, as well as for the object detection of S2.

Applying the +HLA in scenario S1 increases the F-measure for labels of size one from 0% to 98%. For labels of size three and five, we observe only minor improvements of 2%. Investigating the false positives and false negatives revealed that these improvements are mostly ascribed to the false negatives in the tagger's verb identification. Applying the DLA to labels of size two, even results in an F-measure of 96% in the activity detection, which is one percentage point above the +HLA configuration. This difference originates from additional false positive activities identified by the heuristic algorithm. The performance within the other label sizes remains unchanged. Therefore, the increased overall F-measure in the activity extraction of S1 is mainly attributed to the handling of labels of size one.

Scenario S2, however exhibits a stronger deviation between the performance of DLA and +HLA. Figure 3 describes the effect of +HLA on the activity detection in S2. Contrary to S1, the quality of the DLA is considerably low, especially for small labels. A further investigation of the false positives and false negatives revealed that our tagger is not capable of identifying imperative verb forms. While *prüfen* is recognized as a verb in the label *Rechnung prüfen*, *Prüfe* in *Prüfe Rechnung* is not classified as a verb. However, most of the labels of size two exhibit this imperative object style. Since we apply +HLA to avoid false negatives in the verb detection, the F-measure increases significantly. Nevertheless, Fig. 3 also shows that this effect diminishes with increasing label size. This is related to the characteristics of the labels contained in S2.

The usage of the German imperative might result in splitting up the verb into components; e.g. *durchführen* is split up in *führe* and *durch*. Even though the heuristic

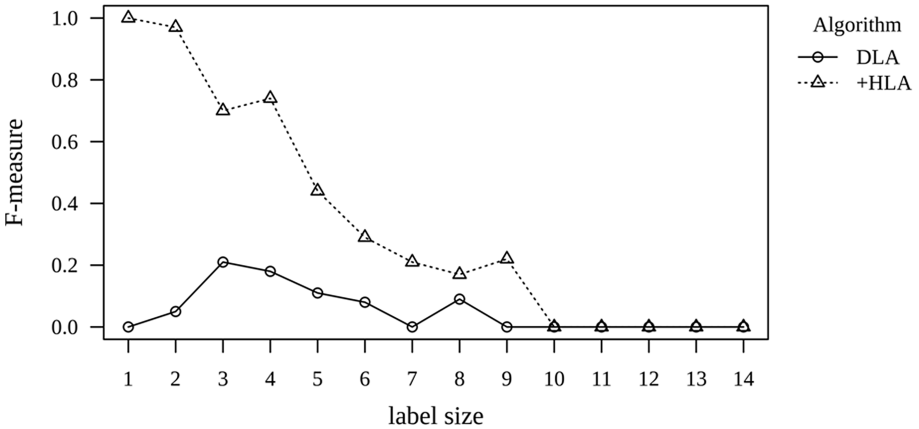


Fig. 3. Scenario S2: evaluation of the activity extraction using F-Measure.

analysis recognizes *führe* as a verb form, it is not capable of linking *durch* to the verb. The missing identification decreases the Recall and explains the drop of the F-measure regarding labels of size three. Moreover, with increasing label size the probability of a second verb occurring within the label grows. The tagger recognizes *prüfen* (*check*), but not *Entscheide* (*decide*) in the label *Entscheide, ob Qualität zu prüfen ist* (*decide if the quality should be checked*).

Figure 4 depicts the increase in performance by applying +HLA on the object extraction in scenario S2. Since the object detection relies on correct verb detection, the previously described label characteristics also affect the object extraction.

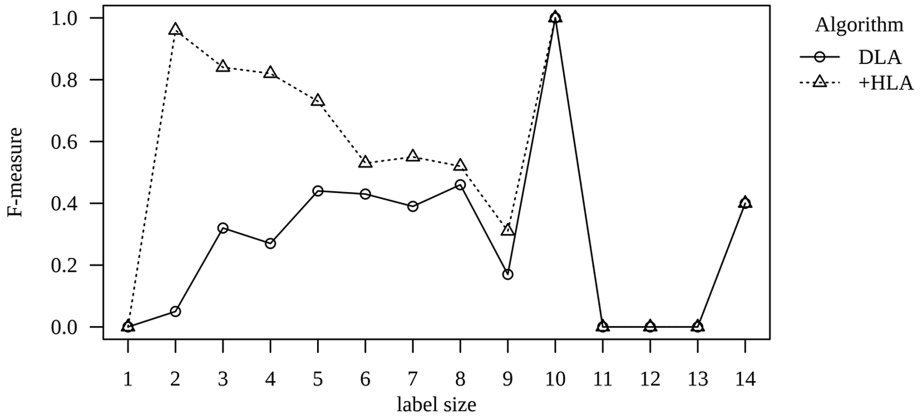


Fig. 4. Scenario S2: evaluation of the object extraction using F-Measure.

5 Discussion

The proposed dependency-based extraction achieves outstanding results in scenarios S1 and S3. The additional heuristic analysis did not yield a significant increase of quality in these scenarios. Moreover, the proposed technique does not rely on a linguistic corpus or a dictionary as the heuristic approach does. Thus, the correctness and completeness of a linguistic resource do not influence the quality of our approach. This is especially beneficial for German process models as the language exhibits a frequent usage of compounds, which often are not contained in dictionaries.

Furthermore, the dependency-based extraction can handle labels containing multiple activities and objects as well as conditional and conjunctive sentences. Up to a certain degree, we are also able to handle labels containing spelling errors, since the dependency parser not only relies on characteristics of the word itself but also on the context contained in the label.

Our approach requires either a trained language model or a tagged language corpus following the UD taxonomy, which enables us to automatically train a language model.

In contrast to scenarios S1 and S3, in scenario S2 DLA achieved only poor results for both, the object (30% F-measure) and activity (12% F-measure) detection. Here

+HLA led to a significant performance increase compared to DLA. The performance was good for the object extraction (78% F-measure), but only moderate concerning the activity extraction (66% F-Measure). Even though imperative verb forms rarely appear in linguistic corpora, due to morphological ambiguities to other verb forms, a corpus lookup could handle the imperative detection. Nonetheless, the missing dependency information about a potential direct object makes it hard to identify the related business object. Moreover, a lookup would not resolve the observed difficulties concerning split up imperative verbs.

Our dependency-based concept is also transferable to other natural languages, since we rely on the taxonomy of *universal dependencies* (UD) [24]. Independent from a specific language, the taxonomy describes semantic dependencies between words. The taxonomy consists of 37 dependencies and covers a plethora of languages.⁴ Thus, employing our approach for other languages does not require redesigning the dependency-based algorithm. However, scenario S2 also revealed shortcomings of the dependency-based approach when imperative verb forms are involved. Other languages might also reveal further shortcomings, which require a different heuristic approach to increase the extraction quality. Therefore, the overall performance of our approach regarding manifold languages is hard to estimate and needs further evaluation. The required linguistic resources to train a language model can be obtained from [26]. Moreover, there are trained models available for popular languages, e.g. *Arabic*, *Chinese*, *French*, *German*, *English* and *Spanish*.⁵

We are also aware that evaluating our approach in further scenarios might reveal additional shortcomings of the dependency-based extraction. However, we carefully selected the models to cover an extensive range of labeling styles. On the one hand, we ensured to include the proposed labeling styles depicted in Table 1, on the other hand, we increased the diversity of natural language contained in labels by investigating models of high and low maturity levels.

6 Conclusion

We answered the research question by proposing, implementing and evaluating a novel artifact. We presented and evaluated an approach for deriving business objects and activities from node labels of business process models. Contrary to existing BPM approaches, which investigate the language contained in the labels, we focus on exploiting not only the POS tag of a word but also the syntactic dependencies between the words. Moreover, by relying on the derived word dependencies, we laid the foundations for the extraction of further information. Based on the dependencies, related adverbs, adjectives and participles could be derived. This additional information allows a precise disambiguation of business objects and activities. Therefore, our information extraction approach is beneficial to manifold research fields. Process model matching often relies on bag of word comparisons between two labels, which means

⁴ The available languages are maintained at <http://universaldependencies.org>.

⁵ cf. <https://nlp.stanford.edu/software/lex-parser.shtml>.

each word of the first label is compared to each word of the second label. Using our approach, a semantic and directed comparison of relevant information could be achieved. Consequently, comparisons based on the extracted business objects and activities would also support model comparisons [3]. Furthermore, an inductive mining of reference models [25] could benefit from our approach, since it provides techniques for extracting and relating essential linguistic components of business processes. Nevertheless, this implies a high accuracy concerning the applied dependency detection. Since the current configuration of our approach uses models trained on natural language texts, further research needs to investigate training language on process model data. Considering the recent flowering of neural networks and deep learning, the potential of these techniques should be investigated.

Acknowledgement. This research was funded in part by the German Federal Ministry of Education and Research under grant number 01IS12050 (project SemGo). The responsibility for this publication lies with the authors.

References

1. Jurafsky, D., Martin, J.H.: *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Pearson Education International, Upper Saddle River (2009)
2. Leopold, H., Smirnov, S., Mendling, J.: Recognising activity labeling styles in business process models. *Enterp. Model. Inf. Syst. Architect. (EMISA)* **6**, 16–29 (2011)
3. Becker, M., Laue, R.: A comparative survey of business process similarity measures. *Comput. Ind.* **63**, 148–167 (2012)
4. Antunes, G., Bakhshandeh, M., Borbinha, J., Cardoso, J., Dadashnia, S., Di Francescomarino, C., Dragoni, M., Fettke, P., Gal, A., Ghidini, C., Hake, P., Khat, A., Klinkmüller, C., Kuss, E., Leopold, H., Loos, P., Meilicke, C., Niesen, T., Pesquita, C., Péus, T., Schoknecht, A., Sheetrit, E., Sonntag, A., Stuckenschmidt, H., Thaler, T., Weber, I., Weidlich, M.: The process model matching contest 2015. In: Kolb, J. (ed.) *Enterprise Modelling and Information Systems Architectures*, pp. 127–155. Gesellschaft für Informatik, Bonn (2015)
5. Leopold, H.: *Natural Language in Business Process Models*. Springer, Berlin (2013)
6. Mendling, J., Reijers, H.A., Recker, J.: Activity labeling in process modeling: empirical insights and recommendations. *Inf. Syst.* **35**, 467–482 (2010)
7. Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG). *Inf. Softw. Technol.* **52**, 127–136 (2010)
8. Leopold, H., Eid-Sabbagh, R.-H., Mendling, J., Guerreiro Azevedo, L., Araujo Baião, F.: Detection of naming convention violations in process models for different languages. *Decis. Support Syst.* **56**, 310–325 (2013)
9. Sonntag, A., Hake, P., Fettke, P., Loos, P.: An Approach For Semantic Business Process Model Matching Using Supervised Machine Learning. *Association for Information Systems (AIS)* (2016)
10. Dijkman, R., Dumas, M., van Dongen, B., Käärik, R., Mendling, J.: Similarity of business process models: metrics and evaluation. *Inf. Syst.* **36**, 498–516 (2011)

11. Bergner, M., Fill, H.-G., Johannsen, F.: Supporting business process improvement with natural language processing: a model-based approach. In: Mayr, H.C., Pinzger, M. (eds.) *GI Informatik 2016*, Klagenfurt, pp. 717–730 (2016)
12. Skersys, T., Butleris, R., Kapocius, K., Vileiniskis, T.: An approach for extracting business vocabularies from business process models. *Inf. Technol. Control* **42**, 150–158 (2013)
13. Leopold, H., Smirnov, S., Mendling, J.: On the refactoring of activity labels in business process models. *Inf. Syst.* **37**, 443–459 (2012)
14. Volokh, A., Neumann, G.: Dependency parsing with efficient feature extraction. In: Glimm, B., Krüger, A. (eds.) *KI 2012. LNCS*, vol. 7526, pp. 253–256. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-33347-7_26](https://doi.org/10.1007/978-3-642-33347-7_26)
15. Chen, D., Manning, C.D.: A fast and accurate dependency parser using neural networks. In: *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 740–750. Association for Computational Linguistics (2014)
16. Weichselbraun, A., Süssstrunk, N.: Optimizing dependency parsing throughput. In: *2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*, vol. 1, pp. 511–516 (2015)
17. Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., Lin, C.-J.: LIBLINEAR: a library for large linear classification. *J. Mach. Learn. Res.* **9**, 1871–1874 (2008)
18. Müller, T., Schütze, H.: Robust morphological tagging with word representations. In: *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics (2015)
19. Benikova, D., Biemann, C., Kisselew, M., Padó, S.: Germeval 2014 named entity recognition shared task: companion paper. In: *GermEval 2014 Named Entity Recognition Shared Task*, vol. 7, p. 10 (2014)
20. Keller, G., Teufel, T.: *SAP R/3 prozeßorientiert anwenden – Iteratives Prozeß-Prototyping zur Bildung von Wertschöpfungsketten*. Addison-Wesley, Bonn (1998)
21. Becker, J., Schütte, R.: *Handelsinformationssysteme*. MI Wirtschaftsbuch (2004)
22. Hamp, B., Feldweg, H.: GermaNet - a lexical-semantic net for German. In: Vossen, P., Adriaens, G., Calzolari, N., Sanfilippo, A., Wilks, Y. (eds.) *ACL/EACL Workshop*, pp. 9–15. Association for Computer Linguistics, Madrid (1997)
23. Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., König, E., Lezius, W., Rohrer, C., Smith, G., Uszkoreit, H.: TIGER: linguistic interpretation of a German corpus. *Res. Lang. Comput.* **2**, 597–620 (2004)
24. De Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., Manning, C.D.: Universal stanford dependencies: a cross-linguistic typology, pp. 4585–4592 (2014)
25. Rehse, J.-R., Hake, P., Fettke, P., Loos, P.: Inductive reference model development: recent results and current challenges. In: Mayr, H.C., Pinzger, M. (eds.) *INFORMATIK 2016. Jahrestagung der Gesellschaft für Informatik (INFORMATIK-2016)*, vol. P-259. GI, Bonn/Klagenfurt (2016)
26. Universal Dependencies. <http://universaldependencies.org>

Designing the Digital Transformation
12th International Conference, DESRIST 2017,
Karlsruhe, Germany, May 30 – June 1, 2017,
Proceedings
Maedche, A.; vom Brocke, J.; Hevner, A. (Eds.)
2017, XVI, 492 p. 106 illus., Softcover
ISBN: 978-3-319-59143-8