Assisted Data Annotation for Business Process Information Extraction from Textual Documents

Julian Neuberger[1], Han van der Aa[2], Lars Ackermann[1], Daniel Buschek[1], Jannic Herrmann[1], and Stefan Jablonski[1]

[1] University of Bayreuth, Bayreuth, Germany
firstname.lastname@uni-bayreuth.de
[2] University of Vienna, Vienna, Austria
han.van.der.aa@univie.ac.at

Abstract. Machine-learning based generation of process models from natural language text process descriptions provides a solution for the time-intensive and expensive process discovery phase. Many organizations have to carry out this phase, before they can utilize business process management and its benefits. Yet, research towards this is severely restrained by an apparent lack of large and high-quality datasets. This lack of data can be attributed to, among other things, an absence of proper tool assistance for business process information extraction dataset creation, resulting in high workloads and inferior data quality. We explore two assistance features to support dataset creation, a recommendation system for identifying process information in the text and visualization of the current state of already identified process information as a graphical business process model. A controlled user study with 31 participants shows that assisting dataset creators with recommendations lowers all aspects of workload, up to -51.0%, and significantly improves annotation quality, up to $+38.9\%$ in F_1 score. We make all data and code available to encourage further research on additional novel assistance strategies.

Keywords: Business Process Management · Process Information Extraction · Natural Language Processing · Human Computer Interaction

1 Introduction

Business process management (BPM) can provide organizations with many benefits by improving their regular operating procedures. Organizations looking to utilize these benefits first need to discover and model their business processes, which is a very time consuming, and therefore expensive task [13]. To alleviate this, researchers in the BPM community use the information contained in natural language process descriptions from sources like quality management handbooks, documentation of standard operating procedures, or employee notes to automatically generate formal process models [4]. While this area is actively researched [1, 5, 8, 11, 13, 28], new and innovative approaches are quite rare. One reason is the limited availability of data to develop, train, and assess approaches in BPMN contexts [18]. Recent initiatives aim to mitigate this issue, providing a gold-standard dataset for the process information extraction task—PET [9]. With this dataset, systems for extracting process information can be developed, e.g., machine learning models for extraction are trained, and subsequently evaluated. Extracted information is then the basis for automated model generation methods, allowing fully automated process model generation from process descriptions. Still, PET contains only 45 process descriptions, which is not enough to train deep neural networks [28], although they have been shown to be well suited for similar tasks in other areas [6]. Even techniques based on pretrained large language models are affected by the lack of data, as rigorous evaluation on many different data sources is essential to assess their practicality, especially in light of the large variation in terms of the structure, style, and contents of textual documents that contain process information [2].

The lack of suitable data for process information extraction tasks can in part be attributed to the effort required to establish gold standard annotations. Such annotations are a critical requirement for both training and evaluation of information extraction approaches. However, manually annotating process information in textual process descriptions involves elaborate guidelines [9] and considerable ambiguity [3, 12], making it time consuming and mentally taxing. Fig. 1 shows two sentences of a process description from the PET dataset, fully annotated with the gold-standard process information. Note, that annotating these sentences requires identifying 14 process-relevant elements, and 16 dependencies between them, in just these two sentences, where the average description in PET has 9.27 sentences [9]. We discuss the task in detail in Sect. 3.1 and how to circumvent this complexity in Sect. 3.2. Additionally, depending on the annotation schema, some of these annotations are not intuitive, e.g., "*decides*", which would intuitively be annotated as an *activity*, underlining the need for annotation guidelines mentioned above.

Recognizing this issue, we explore how dataset creators (*annotators*) can be assisted in their data annotation task, so that their workload is lightened, while simultaneously improving the quality of their extractions. Therefore, in this paper, we propose and evaluate the benefits of two assistance features that can support human annotators: (1) AI-based recommendations, which allow annotators to quickly tackle trivial parts of the annotation task—as well as receive suggestions for less trivial aspects—and (2) the use of a visualizations of the currently annotated information through a graphical process model, which allows annotators to observe the process that they have so far captured. Note that, although the task of text annotation is generic, the assistance features are tailored to the specifics of text annotations for process information extraction.

We implement both assistance features in a prototypical annotation tool that we use as a basis for a rigorous user study with 31 participants, ranging from modeling novices to experts, to assess the effectiveness and efficiency of the proposed features. Code[3] and data[4] are made publicly available to the re-

[3] see https://github.com/JulianNeuberger/assisted-process-annotation
[4] see https://zenodo.org/doi/10.5281/zenodo.12770686

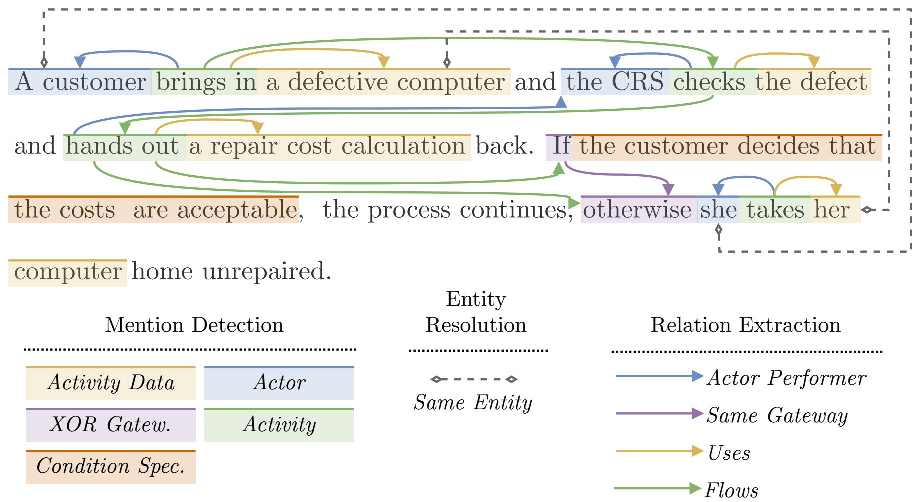


Fig. 1. The first two sentences of document *doc-1.2* in the PET dataset, fully annotated with entity mentions, entity references, and relations.

search community, to allow others to efficiently and effectively annotate their own datasets, but also encourages exploration of additional assistance features. The main insights from this study are as follows.

1. Assisting annotators with suggestions made by artificial intelligence systems is observed to make annotating process relevant information in textual process descriptions significantly easier. This results in a significant reduction of key workload metrics by more than one half (−51.0%). At the same time, assistance improves the quality of extracted information measured in F_1 score by up to +0.224 (+38.9%).
2. The use of assistance features is recognized to considerably reduce the gap between novice and experienced process modelers in annotation tasks. Specifically, complete beginners can reach annotation quality comparable to expert annotators, speeding up the training process for new data annotators considerably. This insight shows that annotation features can help assembling larger data annotation teams, and speeds up the creation of new datasets in the space of business process model extraction from natural language text.

The rest of this paper is structured as follows. In Sect. 2, we discuss related work on process information extraction, relevant user studies, and annotation tools. In Sect. 3, we present our concept behind a tool built specially for annotating textual process descriptions, its implementation in a research prototype, and the assistance features. In Sect. 4 we describe the design and execution of the user study. We present results for this study in Sect. 5. We conclude the paper in Sect. 6, summarizing, discussing limitations, and describing future work.

2 Related Work

Work related to this paper can be roughly categorized into three sections.

Business process information extraction. The last decades have seen various approaches to the task of extracting process relevant information from natural language text, including systems based on expert-defined rules [1, 8, 11, 30], data-driven ones [5, 9, 28], and systems based on pretrained generative large language models [8,21,27]. We use approaches from [28] and [9] in our work to implement annotation recommendations. Many of the works mentioned also propose data annotation schemata tailored towards specific modeling languages, such as [1, 30] for declarative process modeling, towards different task descriptions, such as [29] for information extraction from process relevant sentence fragments, or towards other stages in the process life cycle, e.g., process redesign [25].

We focus on PET, as it is heavily biased towards the current industry standard, BPMN, and the to date largest available dataset. PET was extended with the notion of entity identities [28], i.e., the task of resolving multiple mentions of the same process element across the textual description to a single one. This is important for properly modeling business objects and process participants, which would otherwise be duplicated in the generated model. In this paper we use this extended version of PET.

User studies. Schützenmeier et al. [32] present a user study on the cognitive effort of understanding declarative process models, though they do not consider data annotation, but process simulation and verification. Rosa et al. [31] develop and evaluate a tool for business process modelling which assists users by identifying core BPMN 2.0[5] elements and highlighting them in the process description. Our work, in contrast, aims to be a step towards alleviating the data scarcity problem in business process model generation from text, by making data annotation easier. In a study of similar size to ours, the authors evaluate the usefulness of the *BPMN Sketch Miner* for process modelling based on textual descriptions with visual representations of process elements [16]. While their study mainly focuses on usability and subjective values, ours also considers objective measures.

Annotation tools. Both our concept and implementation for assisted process-relevant information annotation are related to a number of annotation tools. These can usually be used to annotate text for use in Named Entity Recognition (NER), Entity Matching and Resolution (ER), or Relation Extraction (RE), tasks which are similar to business process information extraction (compare definition in Sect. 3). Still, these tools are not designed with characteristics of business process descriptions in mind, including but not limited to, the high information density present in such descriptions, its inherent ambiguity (see Sect. 1), and the target down-stream task, i.e., generation of a formal and graphical process model. Additionally, unlike in the NLP community, data annotators for process information extraction are often times experts in BPM, but not in NLP, and therefore can benefit from purposeful simplifications in the annotation tool. In the following we will describe several notable examples of multi-purpose

[5] https://www.omg.org/bpmn/, accessed July 4, 2024.

Natural Language Processing (NLP) data annotation tools, from which we drew inspiration and how our proposed concept differs from them.

Doccano [26] provides features useful for collaborative data annotation, creating datasets in multiple languages, and comparing annotations between annotators. Label Studio [34] supports more machine learning domains, e.g., computer vision, and audio processing. This makes the tool even more multi-purpose and less bespoke, compared to our research prototype. The authors already have experience annotating textual business process descriptions using Label Studio [5], which is integrated into our concept for assisted annotation (Sect. 3). Finally, INCEpTION [20] uses *recommenders* to make suggestions for new annotations, which would fit our requirement for AI-based annotation recommendations, but to the best of the author's knowledge can not be extended to show the current state of annotation as a BPMN model. The authors of [20] did not investigate the effectiveness of recommendations for text annotation, and while a positive effect seems plausible, we are interested in proving and quantifying this effect.

3 Concept for Assisted Annotation

This section outlines our concept for assisted data annotation. First, we define the task of annotators in Sect. 3.1. Based on this we motivate the need for more efficient and effective data annotation and derive assistance features in Sect. 3.3. Finally, we describe our research prototype implementation in Sect. 3.4.

3.1 The Process Information Extraction Task

Ultimately, human annotators have to complete the process information extraction task to annotate process descriptions with process-relevant information. Therefore, we define this task in the following. Consider, for example, document *doc-1.2* from the PET dataset describing the process of a computer repair. Fig. 1 shows this document fully annotated with all process relevant information, which consists of three major categories. First, ***Mentions*** of process relevant entities in PET are continuous sequences of text with a given type, for example, *Actors* (process participants, "a customer"), *Activity Data* (business objects, "a computer"), or *XOR Gateways* (decision points, often indicated by "if", "otherwise"). The last example illustrates, why we call detecting and extracting such mentions *Entity Mention Detection* (MD) and not *Named Entity Recognition* (NER). Named Entities are defined by either proper names (e.g., persons, locations) or natural kind terms (e.g., enzymes, species) [22]. "If" or "otherwise" do not fall into this definition, which is why we use the more relaxed definition of (non-named) entities and the detection of their mentions within the text [35]. Mentions are then resolved to ***Entities***, i.e., clustered, allowing subsequent model generation steps to only render a single process element, instead of multiple (one for each of its mentions). This task is called *Entity Resolution* (ER) and is closely related with co-reference and anaphora resolution [33]. ***Relations*** between mentions define how these elements interact with each other.

PET defines, for example, *Flow* (order of task execution), *Uses* (association between a task and the business object it uses), or *Actor Performer* (assigning a process participant as executor of a given task).

3.2 Annotation Workflow

As we discussed in Sect. 1, annotating the process relevant information contained in textual process descriptions is a complex task and very demanding for the human performing the annotation, as it requires attention to three sub-tasks, as outlined in the previous section 3.1. We therefore split the task into its sub-tasks MD, ER, and RE. While this partially alleviates the issue of complexity, it will also allow us to assist annotators in these sub-tasks differently, and analyze how assistance features help during a specific sub-task. Fig. 2 depicts the resulting workflow. After the annotator submits a natural language process description, they are then asked to select mentions (MD), resolve entities (ER), and define relations between mentions (RE), in three separate steps. Finally, all information is shown again, so that the annotator may reconcile any errors.

While this workflow reduces the complexity of process information annotation by splitting it up into smaller tasks, the overall complexity remains high. High density of information makes annotating very confusing, especially for beginners. The example in Fig. 1 contains a total of 40 words, of which only eight are not part of one of the 14 entity mentions (20%), while also containing 14 relations between them. From previous annotation experience in other tools (see Sect. 2), we know that this can be partially mitigated by splitting the task into smaller sub-tasks, e.g., focusing on a subset of entity and relation types, or by annotating the categories from above one after the other. Based on this experience we defined a *workflow*, which we describe in Sect. 3.2. High information density and the resulting complexity of displaying this information also motivates us to find ways to visualize the information better, and help the user focus on information they potentially would miss otherwise. This results in two assistance features, *visualization* and *recommendation*, which we describe in Sect. 3.3.

3.3 Assistance Features

In one of our preliminary studies two assistance features were identified as promising candidates for improving the efficiency, quality, and user experience of annotation documents for the process information extraction task.

AI-based annotation recommendations. Building on the progress that has already been made in the development of automated information extraction approaches for mentions, entities, and relations, we can present the user with recommendations for these elements. Interviews with BPM experts during the preliminary study and our review of related work (Sect. 2) suggested that recommendations can be a powerful tool for speeding up annotation in trivial cases and provide useful ideas in non-trivial ones. We used an approach based on conditional random fields for extracting mentions, as presented by Bellan et al. [9], with code from [28], a pretrained neural co-reference resolver for entities [28], and a relation extraction approach based on gradient boosting on decision trees [28]. All trainable approaches were trained with 80% of the available data (36 documents) and the rest was held out for use during the user study. Recommendations are shown during the appropriate workflow steps and can be confirmed, discarded, edited, or marked for later review. The extraction approaches we use are limited in their understanding of business processes, and as such have no real world knowledge that could help during extracting process information from unseen descriptions. This means recommendations can be flawed, but are enough to effectively support data annotation (Sect. 5).

Visual result representation. Second, the information that a human annotator marks in a textual process description is always process relevant, i.e., a perfect annotation results in a model that perfectly reflects the process description. This shows how a human annotator may benefit from a graphical process model as a visualization of the currently annotated information, as any missing information is reflected in the (therefore incomplete) graphical process model. Visualizing the current state of annotation involves three major stages. First, in the ***Consolidation*** stage, we assign conditions to their respective paths in the process, merge mentions of entities, and find the closest actor in the text left of activities that are not explicitly assigned one. In the second stage, the ***Vertex*** stage, we create process elements for all mentions, e.g., *Tasks*, *Data Objects*, *Swimlanes*, etc. The final ***Linking*** stage connects related elements, e.g., successive tasks and gateways with *Sequence Flows*, if they are located in the same Pool, or *Message Flows* between them. We also create *Data Associations* between Data Objects and Tasks, adding the label of the Data Object to the label of the Task, for labels like "*send a mortgage offer*". In this way the graphical process model is generated and layouted automatically, and as such has limitations that might affect its usefulness, which we discuss in Sect. 6.

3.4 Implementation

We have implemented our concept in a usable research prototype. It consists of a user-facing web application, implemented in JavaScript, using React[6].

A backend server provides NLP pre-processing functionality, such as tokenization and the information extraction approaches for the recommendation assistance feature. It is implemented in Python 3.11, using spaCy[7] for pre-processing. When the user inputs a textual process description, it is first sent to this server to pre-process the text. The result is then displayed in the web application. In each of the annotation sub-tasks defined in Sect. 3.2 the relevant information is extracted by the backend server and presented to the annotator as recommendations. The backend server is also responsible for storing annotation results. A second backend server is used for visualizing the current annotations by generating a formal process model in BPMN and its graphical representation. We

[6] https://react.dev/, last accessed July 11, 2024.
[7] https://spacy.io/, last accessed July 11, 2024.

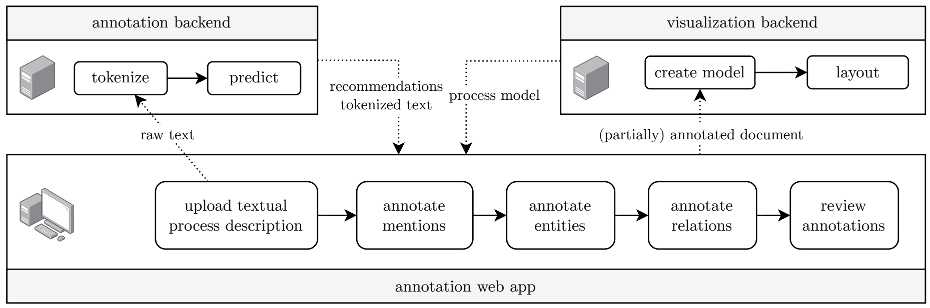


Fig. 2. Visualization of the workflow and general architecture of our implementation.

implement this using Java 17 and the Camunda Model API[8]. Both back-end servers expose any functionality using REST interfaces, which the user-facing web application can query. Fig. 2 shows the workflow in the web application, as well as the communication between servers.

4 Study Design

We conducted a user study with 31 participants to assess the effectiveness of the two assistance features, based on several measures in scenarios of varying assistance. Both measures and scenarios are defined later in this section. We focus our efforts on answering the following three key research questions.

RQ1 Which annotation assistance features or combinations lower the workload of annotating process information?

RQ2 Which annotation assistance features or combinations improve the quality of annotations?

RQ3 Which annotation assistance bridge the gap in annotation quality between beginner annotators and those with BPMN experience?

The general setup of this study is as follows. All supplementary material, such as the questionnaires and resulting data can be found online, see Sect. 1.

Study procedure. Each participation within our user study, is structured into three blocks. First, general demographic information is collected, and the task and annotation tool are explained to the participant, which involved giving a brief tutorial and a small guided annotation task, without any assistance features enabled. This task is only done for training purposes and is not evaluated later. Next, the participant has to complete four annotation scenarios, which we describe later in this section. After each scenario a short questionnaire is conducted, aimed at collecting user opinion, sentiment, and feedback concerning the

[8] https://docs.camunda.org/manual/7.21/user-guide/model-api/bpmn-model-api/, last accessed July 11, 2024.

scenario they just completed. The last block involves a questionnaire to gather general feedback and data regarding overall user preferences.

Measures. We measure the effectiveness of assistance features using several metrics aimed at *objective* and *subjective* values. For objective values we measure the time a user takes to annotate a document and the quality of mention, entity, and relation annotations, each measured with the F_1 score. Subjective values are derived from the NASA Task Load Index (TLX), which is widely used for measuring the workload during or right after performing a task [15]. The NASA-TLX can be used in many different contexts, and was also already used to evaluate information systems [7]. It defines a total of six dimensions that measure different aspects of workload. We used the four relevant to our study.

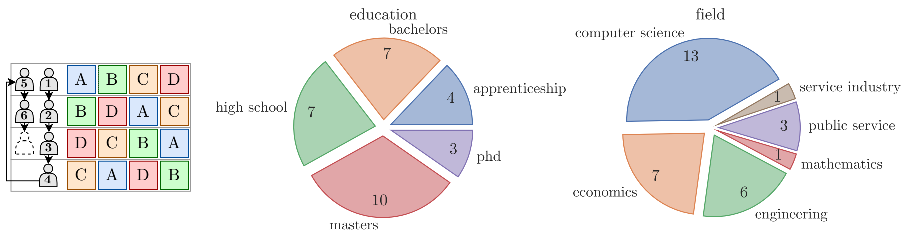
mental demand: How much the annotator has to focus on the task
uncertainty: How uncertain the annotator is of their annotations
effort: How much work is needed to complete the task
frustration: How frustrated the annotator is with the task

We excluded physical and temporal demands, to focus on the subjective metrics most relevant to our research questions. While *physical* demand is not completely irrelevant (think of mouse movements), it is far less informative than the other measures. Regarding *temporal* demands we refer to our objective measure of task completion time. Note that compared to the original definition of the NASA-TLX, we rephrase *performance* to measure the *uncertainty* of an annotator with their annotation results. Additionally to the NASA-TLX, we also asked users to share their experiences with the tool and assistance features in a questionnaire using 5-point Likert items [17].

Annotation scenarios. We assess the efficiency and effectiveness of annotators in four scenarios. Scenario *(A)* entails no assistance, besides the workflow defined in Sect. 3.2 and serves as a baseline. Scenario *(B)* visualizes the current state of annotation, and *(C)* gives recommendations for annotations made by an artificial intelligence system. Finally, scenario *(D)* combines both assistance features.

Documents in PET contain 168 words on average, which took experts in a preliminary experiment as much as 25 minutes to annotate. We therefore decided to instead only use fragments of documents, containing two sentences. These fragments were carefully selected by measuring the number of mentions, relations, as well as their types. We selected fragments from documents *doc-1.2*, *doc-3.6*, *doc-8.3*, and *doc-9.2*. Document fragments are part of the supplementary material for this paper and available in the repositories mentioned in Sect. 1.

To avoid carry-over effects, i.e., confounding variables such as familiarity with the task after completing a scenario and therefore performing better in the next one, we use the Balanced Latin Square method [19]. This method systematically produces sequences of the scenarios described above, so that each scenario appears as the first one in the sequence equally often, as well as two scenarios preceding or succeeding one another equally often. Users are assigned a sequence of scenarios in a round-robin fashion, compare Fig. 3a. This setup minimizes the number of scenarios each user has to perform while addressing carry-over effects between scenarios, such as increasing familiarization with the annotation task.



(a) (b)

Fig. 3. Assigning annotators to a sequence of scenarios based on a balanced Latin square (left), and demographic information about user study participants (right).

5 Results

In this section we will describe our observations during the experiments described in Sect. 4, starting with an overview of study participants. We had respondents of various age, education, and field of work. 39% have not obtained a university degree, or did not pursue higher education, while 39% completed either Masters or PhD studies. The majority (71%) of participants work in a technical field, i.e., computer science, engineering, or mathematics. Fig. 3b shows a detailed break-down of demographic characteristics of participants.

5.1 Subjective Measures

As described in Sect. 4, we measure four subjective sub-metrics of the NASA-TLX — mental demand, uncertainty, effort, and frustration — across four different assistance scenarios. We then used a repeated measure ANOVA [10] to find if there are statistically significant differences in the four assistance scenarios defined in Sec. 4. A repeated measure ANOVA can be used to test if two ore more non-independent samples (measurements) are from the same distribution, measured by $p \in [0, 1]$. In our case, we test for differences in workload between annotation assistance scenarios. We reject the Null hypothesis (no difference) and accept the alternative one (difference exists) when $p < 0.05$.

Repeated measures ANOVA assumes sphericity in data, i.e., the difference in metrics for all combinations of two scenarios have the same variance. This assumption can be tested with Mauchly's test for sphericity [24]. Data for three out of four metrics violated the assumption of sphericity ($p < 0.05$). We use the Greenhouse-Geisser correction [14] to account for this. Even then, our observations show that each one of the four workload metrics are affected by changing how annotators are assisted by our annotation tool and the differences are statistically significant with $p < 0.001$.

Since the repeated measures ANOVA indicated a difference in the NASA-TLX metrics when using different assistance features, we ran six post-hoc tests,

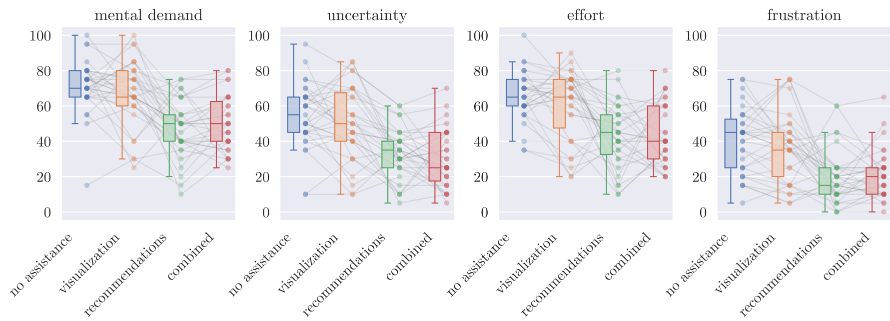


Fig. 4. Subjective measures for each of the four scenarios from Sect. 4.

looking for the differences between each combination of two features, e.g., measurements for non-assisted annotation (scenario A in contrast to only recommendations (scenario C). We corrected all p values using Bonferroni's method [23] for running multiple tests. Intuitively, running many tests increases the likelihood of finding statistically significant differences in one of them, even though there is none. This correction multiplies the P-value with the number of tests, to account for this increased likelihood. Tab. 4 in Sect. 6 reports details.

In summary, no assistance feature at all (scenario A) is statistically significantly worse than either only recommendations (scenario C) or both assistance features combined D). Surprisingly, assisting annotators with a visualization of the information they found in the text (i.e., the generated graphical process model, scenario B) was not found to help with reducing the workload.

Compared to no assistance, assisting the annotator with recommendations reduced mental demand by 24.7 (−34.6%), effort by 22.4 (−34.2%), and frustration by 20.5 (−51.0%). Uncertainty is best lowered by combining recommendations with visualizations, which reduces it by 24.4 (−44.8%), according to our observations. Note, that we found no statistically significant effect on any sub-metric when comparing recommendations (scenario C) to the combination of recommendations and visualizations (scenario D). Similarly, we could not observe a difference between non-assisted annotation (scenario A) and just visualization of annotated information (scenario B). This indicates that only visualizing the currently annotated process-relevant information is not enough to reduce the workload of the annotation task. Contrary, recommendations are a way to reduce it by up to to nearly 50%. Some limitations apply to our findings concerning the quality of the graphical process representation, which we discuss in Sect. 6.

Fig. 4 aggregates our data for each sub-metric into plots, showing values for each participant and scenario as strip plots, where the values of a given participant are connected by lines. Additionally the data points are aggregated into box plots, showing the data mean, 25^{th} and 75^{th} percentiles as box, and the rest of the distribution as whiskers, excluding outliers. The plots mirror the general observations we drew from Tab. 4, and shows that recommendations and the combination of both assistance features help best with reducing the workload of data annotators. Overall, the ordering of assistance features in terms of reducing the workload is obvious from the plots. No assistance (scenario A) and visualizations only (scenario B) share the spot for least useful, while recommendations and the combination of features seem to be equally useful in lowering the workload of process information annotation, thus answering research question **RQ1**.

5.2 Objective Measures

As discussed in Sect. 1, our goal with assisting data annotators is twofold. The previous section 5.1 discussed metrics that are subjective, i.e., are based on the experiences of a data annotator. On the other hand, assisting annotators also affects the quality of annotations. We measured a total of four objective metrics, which we presented in detail in Sect. 4. These are the F_1 scores for annotated mentions, entities, and relations, as well as the total time a given annotator needed to complete annotating a document fragment. An aggregate of the data we obtained is shown in Fig. 5.2 as a plot, similar to the one we showed and explained in Sect. 5.1. Detailed results are listed in Tab. 2 in Appendix 6.

Again, using a repeated measure ANOVA we found significant effects on the annotation quality measured in F_1 when using different assistance features during the annotation of mentions ($p < 0.001$) and relations ($p < 0.001$). The annotation quality of entities was not affected ($p = 0.450$), which may be caused by the low number of entities[9], as well as the fact that we count an entity only as correct, if contains all expected mentions. This means errors by the annotator during MD propagate to the ER task. Furthermore, we did observe a statistically significant difference in the time an annotator needs to annotate a document ($p = 0.011$), but during post hoc tests we could only explain this with a statistically significant difference between the assistance features *visualization* (B) and *recommendations* (scenario C), where the latter speeds up completion times by about 1.5 minutes (see Tab. 3 in the Appendix).

Several participants remarked during the user study, that identifying relations and classifying them correctly is a very challenging task. These participants were mostly inexperienced with BPM and BPMN and thus greatly benefited from the recommendation assistance feature. We can observe this across all participants, as the post hoc tests for relation annotation quality show. Comparing scenario A (no assistance) against scenario C (recommendations only), we see a statistically significant ($p = 0.001$) increase in F_1 of 0.224 (+38.9%). Using the visualization does not seem to have a significant effect compared to no assistance at all ($p = 1.000$), but is significantly worse than using recommendations (-0.259, $p < 0.001$) or using both assistance features (-0.204, $p < 0.001$).

The same analysis can be made for the task of mention detection. Participants of our user study seem to benefit most from recommendations, when compared

[9] On average, fragments used in the user study only contained one entity that needed resolution, i.e., there are at least two entity mentions referring to the same entity.

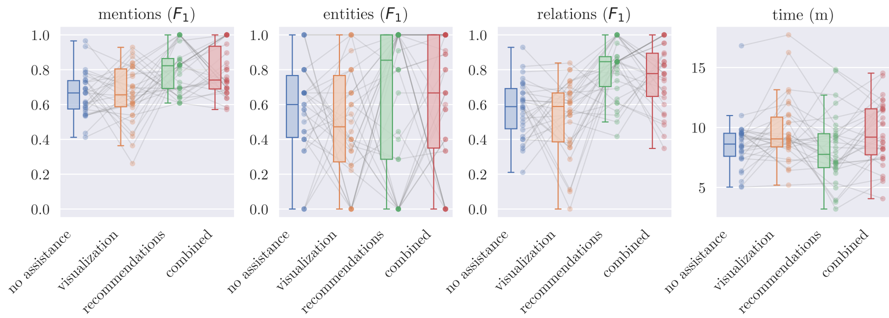


Fig. 5. Objective measures for each of the four scenarios from Sect. 4.

to no assistance at all ($p < 0.001$), with an improvement of 0.141 (+21.4%). Visualization has no statistically significant effect ($p = 1.000$) compared to no assistance at all. Using only visualization has an adverse effect compared to just recommendations ($p < 0.001$) with a decrease in F_1 of 0.141. Similar to Sect. 5.1, this effect can be attributed to limitations in our graphical process model, which we discuss in Sect. 6, or a user's familiarity with BPMN (Sect. 5.3).

This also answers research question **RQ2**, as recommendations seem to be the best choice for improving the quality of annotations. Notably, for all three tasks the annotation recommendations themselves are of lower quality than the average annotations by a human annotator assisted by recommendations (scenario C). Human review improved the F_1 score of annotations by +0.100 for MD, +0.181 for ER, and +0.151% for RE, showing how humans assisted by AI-based systems can perform better than each part in isolation.

5.3 Effects of Annotator Experience

We asked participants for their experience with BPMN, measured in years. With this information, we now investigate if a user's experience with BPMN influences how much they can benefit from assistance. To this end we split the data into two groups — *experts*, which we define for the purposes of this analysis as participants with at least one year of BPMN modeling experience, and *novices*, which are the remaining study participants. This split results in 10 experts and 18 novices. We hypothesize that annotations by *novices* are worse in terms of F_1 score compared to those by *experts*. Tab. 1 lists results for an independent samples T-Test.

We can confirm our assumption that BPMN experience improves the quality of mention (MD) and relation annotations (RE), for un-assisted annotation (scenario A). In all assisted scenarios (B, C, D) we have to reject our hypothesis, i.e., *novices* no longer produce worse annotations than *experts*, from which we infer that the two assistance features can indeed bridge the gap in annotation quality

Table 1. Independent Samples T-Test for the hypothesis that objective scores for annotations by *novices* are lower than those by *experts*.

		t	df	p^a
mentions	no assistance	−1.950	20.691	0.032 *
	recommendations	0.023	23.041	0.509
	visualization	−1.166	23.984	0.128
	combined	−1.463	24.491	0.078
relations	no assistance	−1.800	25.072	0.042 *
	recommendations	0.664	21.132	0.743
	visualization	−1.182	28.000	0.124
	combined	−0.590	20.711	0.281

*, **, *** statistically significant results of increasing degrees.
[a]P-value following Welch's test.

caused by differences in experience with BPMN. We therefore answer **RQ3** with *annotation recommendations*, *visualizations*, and *combined assistance*.

6 Conclusion

In this section we will reiterate the core contribution of this paper, the limitations of the user study we conducted, and we describe our plans for future work.

Core contributions. This paper presents an in-depth exploration on the usefulness of two features for assisting data annotators in the domain of business process information extraction. A user study with 31 participants shows that annotation recommendations reduce certain workload aspects by up to −51.0% (**RQ1**). We find that recommendations obtained by a system based on machine learning improve annotation quality as much as +38.9% (**RQ2**). The same recommendations bridge the gap in annotation quality between beginner and expert annotators, promising easier assembly of annotation teams by means of shorter training times (**RQ3**). We make all data and code publicly available.

Limitations. First, we focused our study on two assistance features, to ensure its feasibility, while also guaranteeing methodological correctness. Investigating more assistance features would either increase participation times, or limit each participation to a subset of scenarios. Next, while we could not observe statistically significant effects of any assistance features on the quality of entity resolution annotations, we cannot eliminate the possibility that this caused by errors propagated from the MD task. Our automated method used for generating and layouting a graphical process model from the process information (annotations) used for the visualization assistance feature has limitations in terms of structure, accuracy stemming from the employed heuristics, and clarity of generated labels. This may affect its usefulness, as these limitations may make the graphical model harder to understand, especially for untrained annotators. Finally, we only present and analyze a sub-set of the data collected during the user study. For example, we recorded all user interaction with the tool, such as when a recommended annotation is discarded, or a new annotation is created. These logs constitute valuable data for improving the workflow for annotating process relevant data in textual process descriptions.

Future work. Our future work is mainly concerned with eliminating the limitations we discussed in the previous section (6). As such we plan to improve the implementation of our annotation tool, e.g., improve the way relations are displayed. We also plan to extend our analysis of the data we already obtained during this user study, e.g., by evaluating the interaction logs. This data can be very valuable to learn how annotators interact with the annotation tool, and give indications on how to improve the workflow, or which parts of the interface are still unintuitive. Furthermore, we want to explore better annotation recommendation methods, as this feature seems to have a consistently positive effect. We plan to evaluate integrating incremental training, as soon as annotators have submitted a document. Finally we would like to extend the user study to new assistance features, in addition to comparing different workflows and user interface options. The initial findings regarding how experts benefit in different ways from assistance features, compared to novice users, motivate us to conduct a targeted study to find ways to properly assist users of different experience levels.

Appendix

Table 2. Post hoc comparisons of assistance features on objective metrics. Largest statistically significant absolute difference to unassisted annotation is set in **bold**. We abbreviate mean difference with MD and standard error with SE.

			MD	*SE*	t	p_{bonf} [a]
mentions	no assistance	recommendations	**-0.141**	0.037	−3.774	0.002 **
		visualization	0.000	0.037	−0.008	1.000
		both	−0.132	0.037	−3.529	0.014 **
	recommendations	visualization	0.141	0.037	3.766	0.005 **
		both	0.009	0.037	0.244	1.000
	visualization	both	−0.132	0.037	−3.522	0.004 **
entities	no assistance	recommendations	−0.068	0.097	−0.706	1.000
		visualization	0.080	0.097	0.825	1.000
		both	−0.038	0.097	−0.398	1.000
	recommendations	visualization	0.148	0.097	1.532	0.775
		both	0.030	0.097	0.309	1.000
	visualization	both	−0.118	0.097	−1.223	1.000
relations	no assistance	recommendations	**-0.224**	0.049	−4.524	< .001 ***
		visualization	0.025	0.049	0.842	1.000
		both	−0.179	0.049	−3.691	0.002 **
	recommendations	visualization	0.259	0.049	5.367	< .001 ***
		both	0.055	0.049	0.834	1.000
	visualization	both	−0.204	0.049	−4.533	< .001 ***

*, **, *** statistically significant results of increasing degrees.
[a] P-value adjusted for comparing a family of six using Bonferroni correction.

Table 3. Post hoc comparisons of assistance features on completion time. We abbreviate mean difference with MD and standard error with SE.

			MD	*SE*	*t*	$p_{bonf}{}^{a}$
time (s)	no assistance	recommendations	23.778	30.528	0.779	1.000
		visualization	−67.621	30.528	−2.215	0.176
		both	−56.082	30.528	−1.837	0.418
	recommendations	visualization	−91.399	30.528	−2.994	0.022 *
		both	−79.860	30.528	−2.616	0.063
	visualization	both	11.539	30.528	0.378	1.000

*, **, *** statistically significant results of increasing degrees.
[a]P-value adjusted for comparing a family of six using Bonferroni correction.

Table 4. Post hoc comparisons of assistance features on subjective metrics. Largest statistical significant absolute difference to unassisted annotation for a given metric is set in **bold**. We abbreviate mean difference with MD and standard error with SE.

			MD	*SE*	*t*	$p_{bonf}{}^{a}$
mental demand	no assistance	recommendations	**24.677**	3.521	7.009	< .001 ***
		visualization	4.194	3.521	1.191	1.000
		both	21.290	3.521	6.047	< .001 ***
	recommendations	visualization	−20.484	3.521	−5.818	< .001 ***
		both	−3.387	3.521	−0.962	1.000
	visualization	both	17.097	3.521	4.856	< .001 ***
uncertainty	no assistance	recommendations	21.129	3.558	5.938	< .001 ***
		visualization	4.677	3.558	1.314	1.000
		both	**24.355**	3.558	6.844	< .001 ***
	recommendations	visualization	−16.452	3.558	−4.623	< 0.001 ***
		both	3.226	3.558	0.907	1.000
	visualization	both	19.677	3.558	5.530	< .001 ***
effort	no assistance	recommendations	**22.419**	3.710	6.043	< .001 ***
		visualization	5.000	3.710	1.348	1.000
		both	21.129	3.710	5.695	< .001 ***
	recommendations	visualization	−17.419	3.710	−4.695	< .001 ***
		both	−1.290	3.710	−0.348	1.000
	visualization	both	16.129	3.710	4.347	< .001 ***
frustration	no assistance	recommendations	**20.484**	3.655	5.604	< .001 ***
		visualization	3.548	3.655	0.971	1.000
		both	18.548	3.655	5.074	< .001 ***
	recommendations	visualization	−16.935	3.655	−4.633	< .001 ***
		both	−1.935	3.655	−0.530	1.000
	visualization	both	15.000	3.655	4.104	< .001 ***

*, **, *** statistically significant results of increasing degrees.
[a]P-value adjusted for comparing a family of six using Bonferroni correction.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. van der Aa, H., Di Ciccio, C., Leopold, H., Reijers, H.A.: Extracting declarative process models from natural language. In: CAiSE (2019)
2. van der Aa, H., Leopold, H., Mannhardt, F., Reijers, H.A.: On the fragmentation of process information: Challenges, solutions, and outlook. In: International Workshop on Business Process Modeling, Development and Support (2015)
3. Van der Aa, H., Leopold, H., Reijers, H.A.: Checking process compliance against natural language specifications using behavioral spaces. IS (2018)
4. Ackermann, L., Käppel, M., Marcus, L., Moder, L., Dunzer, S., Hornsteiner, M., Liessmann, A., Zisgen, Y., Empl, P., Herm, L.V., Neis, N., Neuberger, J., Poss, L., Schaschek, M., Weinzierl, S., Wördehoff, N., Jablonski, S., Koschmider, A., Kratsch, W., Matzner, M., Rinderle-Ma, S., Röglinger, M., Schönig, S., Winkelmann, A.: Recent advances in data-driven business process management (2024)
5. Ackermann, L., Neuberger, J., Jablonski, S.: Data-driven annotation of textual process descriptions based on formal meaning representations. In: CAiSE (2021)
6. Ackermann, L., Neuberger, J., Käppel, M., Jablonski, S.: Bridging research fields: An empirical study on joint, neural relation extraction techniques. In: CAiSE (2023)
7. Bagozi, A., Bianchini, D., De Antonellis, V., Garda, M., Melchiori, M.: Personalised exploration graphs on semantic data lakes. In: On the Move to Meaningful Internet Systems: OTM 2019 Conferences: Confederated International Conferences: CoopIS, ODBASE, C&TC 2019, Rhodes, Greece, October 21–25, 2019, Proceedings (2019)
8. Bellan, P., Dragoni, M., Ghidini, C.: Extracting business process entities and relations from text using pre-trained language models and in-context learning. In: EDOC (2022)
9. Bellan, P., Ghidini, C., Dragoni, M., Ponzetto, S.P., van der Aa, H.: Process extraction from natural language text: the PET dataset and annotation guidelines. In: NL4AI (2022)
10. Bergh, D.D.: Problems with repeated measures analysis: Demonstration with a study of the diversification and performance relationship. Academy of Management Journal (1995)
11. Ferreira., R.C.B., Thom., L.H., Fantinato., M.: A semi-automatic approach to identify business process elements in natural language texts. In: ICEIS (2017)
12. Franceschetti, M., Seiger, R., López, H.A., Burattin, A., García-Bañuelos, L., Weber, B.: A characterisation of ambiguity in bpm. In: International Conference on Conceptual Modeling (2023)
13. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: CAiSE (2011)
14. Greenhouse, S.W., Geisser, S.: On methods in the analysis of profile data. Psychometrika (1959)
15. Hart, S.G.: Nasa-task load index (nasa-tlx); 20 years later. In: Proceedings of the human factors and ergonomics society annual meeting (2006)
16. Ivanchikj, A., Serbout, S., Pautasso, C.: From text to visual bpmn process models: Design and evaluation. In: Proceedings of the 23rd ACM/IEEE international conference on model driven engineering languages and systems (2020)

17. Joshi, A., Kale, S., Chandel, S., Pal, D.K.: Likert scale: Explored and explained. British journal of applied science & technology (2015)
18. Käppel, M., Schönig, S., Jablonski, S.: Leveraging small sample learning for business process management. Information and Software Technology (2021)
19. Kim, B.G., Stein, H.H.: A spreadsheet program for making a balanced latin square design. Revista Colombiana de Ciencias Pecuarias (2009)
20. Klie, J.C., Bugert, M., Boullosa, B., de Castilho, R.E., Gurevych, I.: The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In: Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations (2018)
21. Kourani, H., Berti, A., Schuster, D., van der Aalst, W.M.: Process modeling with large language models. arXiv preprint arXiv:2403.07541 (2024)
22. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering (2020)
23. Ludbrook, J.: Multiple comparison procedures updated. Clinical and Experimental Pharmacology and Physiology (1998)
24. Mauchly, J.W.: Significance test for sphericity of a normal n-variate distribution. The Annals of Mathematical Statistics (1940)
25. Mustansir, A., Shahzad, K., Malik, M.K.: Autoeprs-20: extracting business process redesign suggestions from natural language text. In: ASE (2020)
26. Nakayama, H., Kubo, T., Kamura, J., Taniguchi, Y., Liang, X.: doccano: Text annotation tool for human (2018), https://github.com/doccano/doccano, software available from https://github.com/doccano/doccano
27. Neuberger, J., Ackermann, L., van der Aa, H., Jablonski, S.: A universal prompting strategy for extracting process model information from natural language text using large language models (2024), https://arxiv.org/abs/2407.18540
28. Neuberger, J., Ackermann, L., Jablonski, S.: Beyond rule-based named entity recognition and relation extraction for process model generation from natural language text. In: CoopIS (2023)
29. Qian, C., Wen, L., Kumar, A., Lin, L., Lin, L., Zong, Z., Li, S., Wang, J.: An approach for process model extraction by multi-grained text classification. In: CAiSE (2020)
30. Quishpi, L., Carmona, J., Padró, L.: Extracting annotations from textual descriptions of processes. In: BPM 2020 (2020)
31. Rosa, L.S., Silva, T.S., Fantinato, M., Thom, L.H.: A visual approach for identification and annotation of business process elements in process descriptions. Computer Standards & Interfaces (2022)
32. Schützenmeier, N., Käppel, M., Fichtner, M., Jablonski, S.: Scenario-based model checking of declarative process models. In: ICEIS (2023)
33. Sukthanker, R., Poria, S., Cambria, E., Thirunavukarasu, R.: Anaphora and coreference resolution: A review. Information Fusion (2020)
34. Tkachenko, M., Malyuk, M., Holmanyuk, A., Liubimov, N.: Label Studio: Data labeling software (2020-2022), https://github.com/heartexlabs/label-studio, open source software available from https://github.com/heartexlabs/label-studio
35. Xu, M., Jiang, H., Watcharawittayakul, S.: A local detection approach for named entity recognition and mention detection. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (2017)