TeaPie — A Tool for Efficient Annotation of Process Information Extraction Data

Julian Neuberger[1,*], Jannic Herrmann[1], Martin Käppel[1], Han van der Aa[2] and Stefan Jablonski[1]

[1]University of Bayreuth, Universitätsstraße 30, 95444 Bayreuth, Germany
[2]University of Vienna, Universitätsring 1, 1010 Vienna, Austria

Abstract
Machine-learning based generation of process models from natural language text process descriptions is severely restrained by a lack of datasets. This lack of data can be attributed to, among other things, an absence of proper tool assistance for dataset creation, resulting in high workloads and inferior data quality. We address these shortcomings with a tool for annotating textual process descriptions. Compared to other, existing data annotation tools, ours implements a multi-step workflow specifically designed for extracting process information, including supporting features that have been shown to reduce workloads and improve data quality.

Keywords
Process Information Extraction, Text Annotation, Business Process Management

1. Introduction

Organizations looking to utilize the benefits of Business Process Management (BPM) initially have to model their internal business processes. These so-called as-is process models are expensive to create, as it is a time consuming task, usually performed by BPM experts together with process experts of the organization [1]. To accelerate this initial step, approaches using Natural Language Processing (NLP) have been proposed. These extract the process-relevant information contained in textual process descriptions of various sources, such as quality management handbooks, standard operating procedures, or employee notes [2]. In a subsequent step, this information is transformed into formal models, e.g., in the BPMN modeling standard (see https://www.omg.org/bpmn/).

While approaches based on machine learning became more common in recent years [3, 4], Process Information Extraction (PIE) still has not adopted the state-of-the-art machine learning techniques and architectures used in other fields of information extraction, even though the tasks share many similarities [5]. These approaches need vast amounts of annotated training data, which is not yet available in BPM in general [6], and especially for PIE [4], where the currently largest available dataset (PET [7]) contains just 45 process descriptions. Approaches based on Large Language Models (LLM) circumvent this issue, as they are pretrained on out-of-domain data, and only need marginal amounts of data for in-context learning [3]. However, they are hard to optimize for this task, cause considerable costs, and have a poor ecological footprint, making them a suboptimal solution. Accordingly, approaches specifically trained for PIE are preferable. To provide appropriate training data for this, it is necessary to annotate large amounts of natural language text. Although annotating text is a common task in machine learning research and is supported by various tools to enhance efficiency and productivity, most existing tools are not suited for annotating *process description* text. This is primarily due to three reasons: First, process descriptions have a very high density of information (cf. Fig. 1). As a result, identifying, annotating, and displaying information quickly becomes confusing, which hampers

Demo Track of CoopIS 2024 - CoopIS-DT 2024
*Corresponding author.
julian.neuberger@uni-bayreuth.de (J. Neuberger); jannic.herrmann@uni-bayreuth.de (J. Herrmann); martin.kaeppel@uni-bayreuth.de (M. Käppel); han.van.der.aa@univie.ac.at (H. v. d. Aa); stefan.jablonski@uni-bayreuth.de (S. Jablonski)
0009-0008-4244-7659 (J. Neuberger); 0009-0003-3420-8037 (M. Käppel); 0000-0002-4200-4937 (H. v. d. Aa)

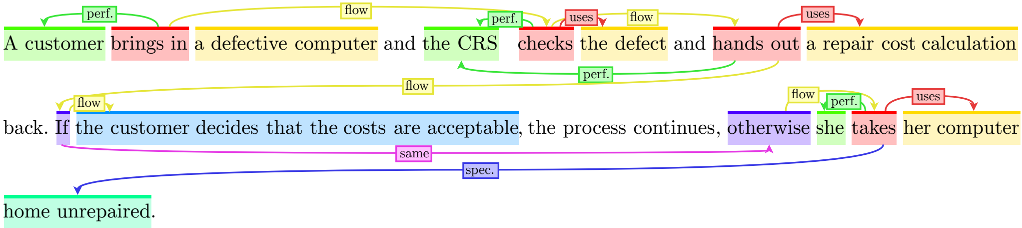


Figure 1: Example for the high information density of PIE data. Of 40 tokens total, only six (15%) are not directly relevant for the process. The text is a fragment of *doc-1.2* of the PET dataset.

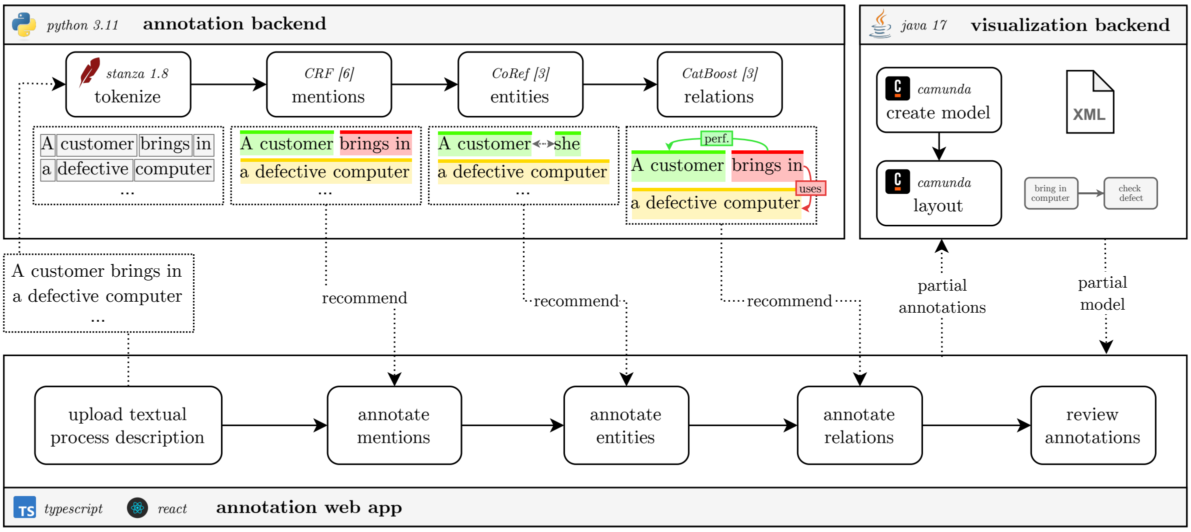


Figure 2: Overview of the modular architecture of TeaPie.

completeness and correctness. Second, annotation of process information is very susceptible to errors, which invalidate the resulting process entirely. Such errors include, but are not limited to accidentally reversing control-flow, disjointed process models, or missing decision points in the process (*XOR-Gates*). We argue, supporting the user with proper visualizations while they annotate yields more complete and correct process models. Third, annotation of process descriptions is often ambiguous. This means that there is more than one arguably correct set of annotations, which makes annotating process descriptions mentally demanding, as many possibilities have to be considered at any given time. The validity of these issues is underlined by other work in the same context, such as the tool Model Judge [8], which facilitates the training of novice modelers in the text-to-model task. To this end, the user's model is compared to a gold standard model and discrepancies are highlighted. While the means of Model Judge are very similar, the ends differ fundamentally — most notably, there is no gold standard during data annotation to which an annotation could be compared to. To address these issues, we present TeaPie, a tool for efficient process information annotation[1].

2. System Overview

To facilitate the extraction of process information from textual descriptions, we developed a modular annotation tool comprising three main components: the front-end web application, the annotation backend and the visualization server. Fig. 2 shows a high-level overview of the architecture of TeaPie.

[1]See https://github.com/JulianNeuberger/assisted-process-annotation for code, video, and live demo. Credentials: *coopis* (user), *processes2024* (password)

The front-end web application is implemented using TypeScript and React[2]. It serves as the primary interface for annotators to interact with the system, guiding them through the annotation workflow.
The annotation backend server is built with Python 3.11 and handles various NLP tasks required for generating annotation suggestions. After a user submits a process description, it is tokenized using the Stanza package[3]. The resulting tokens are then fed into three prediction models, to generate annotation recommendations for the user. First a Conditional Random Fields model from Bellan et al. [3] identifies and extracts mentions of process-relevant entities. Following this, the pre-trained neural co-reference resolution model presented in [4] clusters mentions referring to the same entity throughout the process description. Finally, we use the CatBoost model presented in [4] to extract relation. Models are trained on 80% of the PET dataset, while the remaining 9 documents were set aside for the user study.
The visualization server is developed in Java 17 and utilizes the Camunda Model API[4] to generate graphical models for the front-end. The graphical model updates whenever annotations are modified.

3. Key Innovations

One of TeaPie's three core innovation lies in its six-step workflow, which is specifically designed to reduce the complexity of PIE, both for expert, as well as for beginner annotators. As such, for each piece of process information (mentions, entities, relations) annotations are first recommended by TeaPie, reviewed by the annotator, and subsequently amended with missing annotations. During early prototyping iterations, we found that annotators often would recognize additional mentions during annotation of entities and relations, which is why we added a final review step, where all information is presented at once and annotations can be rectified before finalizing.

The second innovation that sets TeaPie apart from other annotation tools, is the visualization of the current state of annotated process information. This visualization is generated as soon as the first process relevant entity mention is annotated, and regenerated, whenever the annotations change. This gives users immediate feedback on the information they annotated so far, which helps them understand the impact of certain annotations on the over all process model. This was especially helpful for annotators familiar with BPMN, who compared the graphical process model with their expectations.

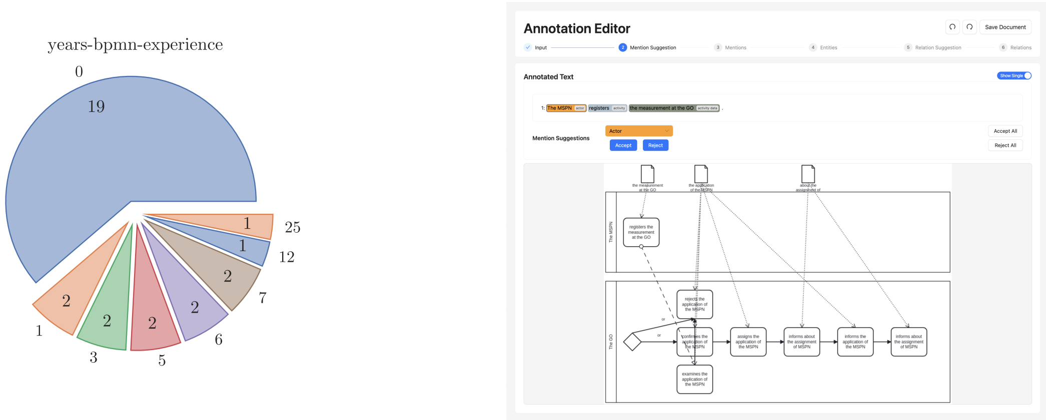
The third innovation of TeaPie are machine learning based recommendations. While other text annotation tools support recommendations of annotations, TeaPie generates recommendations for all three types of PIE data end-to-end. These recommendations have been shown to improve the quality of annotations beyond what either humans or recommendation system in isolation could achieve, while at the same time making the annotation process cognitively less taxing [9]. Furthermore, recommendations help to bridge the experience gap between annotators, which makes it easier to assemble teams [9].

4. Maturity

We evaluated TeaPie in a controlled user study where we asked 31 participants to annotate fragments of textual process descriptions and recorded their feedback regarding TeaPie's practicality. Note that 19 of the 31 participants (61.3%) had no prior experience in BPMN. These users are potential data annotators, currently unable to contribute to PIE annotation projects, due to their inherent complexity and ambiguity [10]. For this reason their feedback is particularly valuable to us. In the following we present a brief analysis focused on usability. A detailed analysis of metrics like annotation accuracy, mental workload, or time per document can be found in [9].

We found that the workflow we implemented was well suited to how most users extract process relevant information from text. Fig. 4 shows the how much users agreed with statements regarding certain aspects of the workflow implemented in TeaPie. Most users felt the speed with which they were able to complete their tasks was satisfactory (Fig. 4a), understood their task in each extraction

[2] See https://www.typescriptlang.org/ and https://react.dev/ respectively
[3] See https://stanfordnlp.github.io/stanza/pipeline.html
[4] See https://docs.camunda.org/manual/7.21/user-guide/model-api/



(a)

(b)

Figure 3: Years of experience with BPMN of user study participants (left), and a screenshot of TeaPie (right).

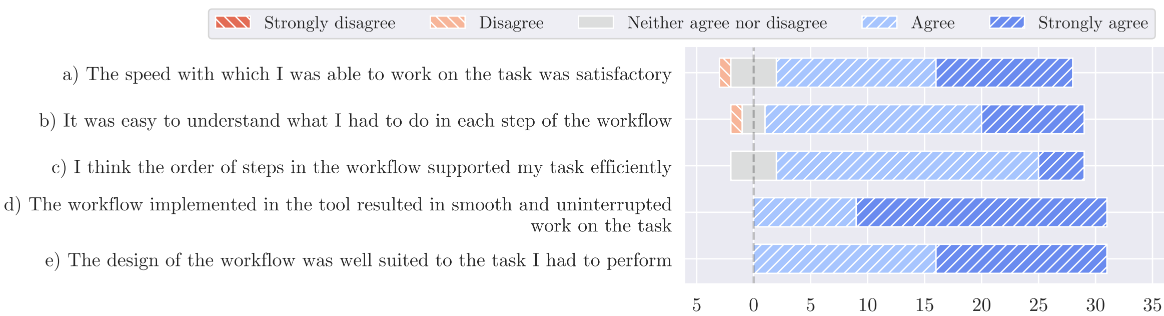


Figure 4: User feedback regarding the workflow implemented in TeaPie

step (Fig. 4b), and agreed with the order of extraction steps (Fig. 4c). All users were satisfied with the way the workflow was implemented (Fig. 4d,e). This is especially encouraging, as first time BPMN users agree with experts in this matter, leading us to believe the workflow provides good guidelines for novice users, while not being overly restrictive for experienced ones.

Limitations. We currently see two main limitations with TeaPie. First, the process generation algorithm we use is a very rough prototype and sometimes results in confusing or incomplete process models. This results in many users rating the visualization as less useful, preferring annotation recommendations over the visualization of currently extracted information (see Fig. 5c). We plan to use an improved visualization algorithm to further improve the usefulness of the process model visualization. Second, TeaPie only supports the PET data annotation schema. We are actively working on the dynamic definition of annotation schemas in a graphical user interface integrated into TeaPie. This will make TeaPie useful in data annotation projects for various modelling languages, e.g., DCR graphs [11], as well as different paradigms, e.g., Object-Centric modelling.

Future Work. Besides the future work mentioned during our discussion of current limitations, we plan to extend TeaPie with additional features for large-scale data annotation projects. First, we want to integrate features to support the collaboration of multiple annotators. These features include, among others, the automatic calculation of inter-annotator agreement, i.e., how well the annotations of two or more annotators align. Process descriptions where annotators disagree, will be assigned to a referee annotator. This concept proved useful for other text annotation projects [12] and results in higher data quality. Next, TeaPie will provide annotation statistics, such as linguistic variability of process elements, preliminary results of training extraction models, or the percentage of process relevant

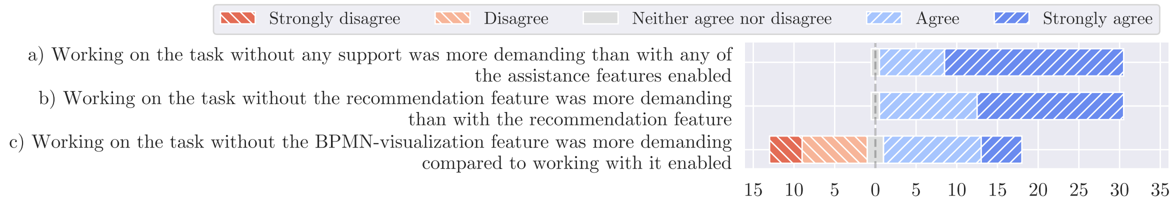


Figure 5: User preferences regarding the supporting features of TeaPie.

text in documents. Such statistics are often included in articles presenting new datasets (cf. PET [7]). Additionally, providing automatic conversion of other modalities to text, such as image-to-text, or audio-to-text, could enable new applications of TeaPie. Furthermore, we plan to experiment with different approaches towards generating annotation recommendations. The current approach uses very few learnt parameters, which makes it efficient, but less effective compared to LLMs, which outperform shallow machine learning approaches [13]. Finally, we plan to provide a publicly accessible instance of TeaPie for use in annotation projects of the BPM research community.

References

[1] L. Ackermann, J. Neuberger, S. Jablonski, Data-driven annotation of textual process descriptions based on formal meaning representations, in: CAiSE, 2021.

[2] L. Ackermann, M. Käppel, L. Marcus, L. Moder, S. Dunzer, M. Hornsteiner, A. Liessmann, Y. Zisgen, P. Empl, L.-V. Herm, N. Neis, J. Neuberger, L. Poss, M. Schaschek, S. Weinzierl, N. Wördehoff, S. Jablonski, A. Koschmider, W. Kratsch, M. Matzner, S. Rinderle-Ma, M. Röglinger, S. Schönig, A. Winkelmann, Recent advances in data-driven business process management, 2024. arXiv:2406.01786.

[3] P. Bellan, M. Dragoni, C. Ghidini, Extracting business process entities and relations from text using pre-trained language models and in-context learning, in: EDOC, 2022.

[4] J. Neuberger, L. Ackermann, S. Jablonski, Beyond rule-based named entity recognition and relation extraction for process model generation from natural language text, in: CoopIS, 2023.

[5] L. Ackermann, J. Neuberger, M. Käppel, S. Jablonski, Bridging research fields: An empirical study on joint, neural relation extraction techniques, in: CAiSE, 2023.

[6] M. Käppel, S. Schönig, S. Jablonski, Leveraging small sample learning for business process management, Information and Software Technology 132 (2021) 106472.

[7] P. Bellan, C. Ghidini, M. Dragoni, S. P. Ponzetto, H. van der Aa, Process extraction from natural language text: the PET dataset and annotation guidelines, in: NL4AI, 2022.

[8] L. Delicado Alcántara, J. Sanchez-Ferreres, J. Carmona Vargas, L. Padró, The model judge: a tool for supporting novices in learning process modeling, in: BPMTracks, CEUR-WS. org, 2018.

[9] J. Neuberger, H. van der Aa, L. Ackermann, D. Buschek, J. Herrmann, S. Jablonski, Assisted data annotation for business process information extraction from textual documents, 2024. arXiv:2410.01356.

[10] H. van der Aa, H. Leopold, F. Mannhardt, H. A. Reijers, On the fragmentation of process information: Challenges, solutions, and outlook, in: BPMDS, 2015.

[11] H. A. López, R. Strømsted, J.-M. Niyodusenga, M. Marquard, Declarative process discovery: Linking process and textual views, in: CAiSE, 2021.

[12] T. Brants, Inter-annotator agreement for a german newspaper corpus., in: LREC, Citeseer, 2000.

[13] J. Neuberger, L. Ackermann, H. van der Aa, S. Jablonski, A universal prompting strategy for extracting process model information from natural language text using large language models, arXiv preprint arXiv:2407.18540 (2024).