Parts of Manual Processes: Tagging Tool User Manual

1st Tom Hohenadl
THI Business School
TH Ingolstadt
Ingolstadt, Germany
0000-0002-7501-9780

***Abstract*—This manual is used with the Parts of Manual Processes tool for user interaction logs. The tool allows users to tag multiple user interaction logs fast and easy. The main objective is to generate enhanced data for task mining or Robotic Process Mining. This document explains the basics to use the POMP tagging tool.**

***Index Terms*—Task Mining, Robotic Process Mining, POMP**

I. Introduction

User interactions are analyzed to better understand and automate manual user processes. To automate such processes, Robotic Process Automation (RPA) is used. RPA techniques allow users the automation of repetitive manual user tasks on computers. To ease the RPA implementation the Robotic Process Mining (RPM) framework and task mining are used. Both research areas describe techniques to record manual user actions on the GUI level. The recorded low-level mouse clicks and keystrokes can be used to generate RPA bots. Yet, the research artifacts created in task mining and RPM have different names and concepts for equal user interactions. It is difficult for end users not familiar with all tools of the RPM domain to understand the differences. Therefore a taxonomy of what user actions are recorded is necessary. This taxonomy is created under the term ”Parts of Manual Processes” (POMP).

This manual is an addition to the literature review categorizing the actions into the POMP taxonomy. The following sections describe the user interface and the methods developed to easily tag user interaction logs with the POMP categories. In section II a short summary of the POMP categories is presented. The section III the setup of the tagging tool is described. Section IV is a detailed description of the tagging method and usage. In the final section V we describe common errors, when using the tool and how to fix these.

II. The POMP categories

The following actions were identified in the literature review:

- Open: Opening of applications or windows within an application
- Navigate: Navigation to an already open application or window
- Transform: Change of GUI elements, e.g. activating, selecting, or typing content
- Transfer: Copying content in to the clipboard
- Conclude: Clicking a button after changing content on a page, e.g. sending a form or confirming a selection
- Close: Closing an application or window
- Empty: Actions changing nothing on screen, e.g. clicking on the background of an already active window

III. Set up of the tool

Setting up the tool consists of three steps: 1. Installing and ensuring a working Python environment, 2. installing all necessary requirements, and 3. storing the logs that should be tagged in the right directory.

A. Installing Python

To install Python we refer the user to w3schools Python getting started.

B. Setting up the POMP tool

The POMP tool is available on GitHub. First, you have to download the source code or check out the code into your development environment. Afterwards, open a terminal window and navigate to the code repository. In the repository you run **’pip install -r requirements.txt’**. All necessary requirements will be installed. If there are any outdated libraries in the *requirements.txt* you can update the file.

C. Adding files for tagging

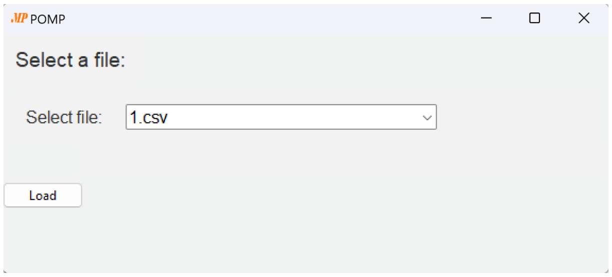
When the libraries are installed and before you run the tool you can and should add your logs in the folder **’logs/pompTagged’**. The files in this folder will be displayed in the selection window of the POMP tagging tool.

Note! The files have to be in .csv or .xes format. The tool currently only supports logs created from **smartRPA** by Agostinelli et al [1] or **Action Logger** by Leno et al [2].

IV. How to use

The tool can be started with running **’python main.py’**. Make sure you are in the correct repository, when running the command.

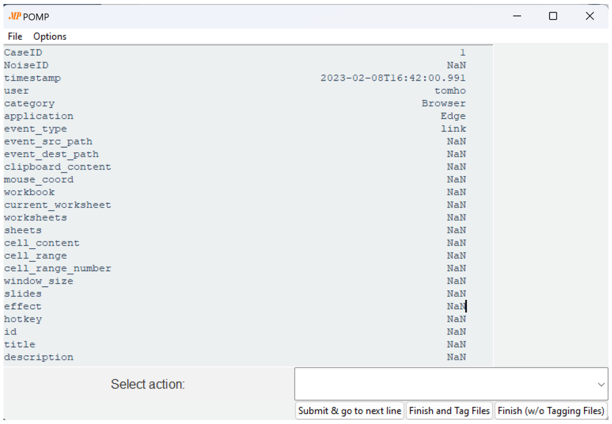
Thanks to Jules Rau for development on the front end and user interface of this tool.



Select a file:

Select file: 1.csv

Fig. 1. File Selection



```
CaseID                                              1
NoiseID                                           NaN
timestamp                     2023-02-08T16:42:00.991
user                                            tomho
category                                      Browser
application                                      Edge
event_type                                       link
event_src_path                                    NaN
event_dest_path                                   NaN
clipboard_content                                 NaN
mouse_coord                                       NaN
workbook                                          NaN
current_worksheet                                 NaN
worksheets                                        NaN
sheets                                            NaN
cell_content                                      NaN
cell_range                                        NaN
cell_range_number                                 NaN
window_size                                       NaN
slides                                            NaN
effect                                            NaN
hotkey                                            NaN
id                                                NaN
title                                             NaN
description                                       NaN
```

Select action:

Fig. 2. Initial tagging screen

a) File Selection: The tool will start and display a file selection. This is presented in figure 1. Now you can select the file you want tag. After selecting the file from the drop-down menu, which displays all files from the folder **'logs/pompTagged'**, you can press the load button. During the loading process the tool will go through the file and check if the file was tagged already.

Note! If the file was tagged before and all rows have been tagged with a category from the POMP taxonomy, the following text is displayed: **All rows in selected file have been tagged.** To remove tags you have to open the .csv file manually.

b) Tagging Method: When the file is loaded a single user interaction is shown. The interaction is shown as a list of all its parameters, i.e. all parameters the log file provides. The application should look as displayed in 2.

To ease the use of this screen you can hide all unnecessary parameters. To do so you open the options menu and you can select **Hide NaN** or **Context Values only**, see figure 3. The 'Hide NaN' values option will remove all empty values from the screen and shorten the list of parameters presented. The 'Context Values only' option will remove all values that are not context parameters. As defined in the original paper the context values are all parameters describing the object a user is interacting with. The tool is currently limited to the context parameters from [1] and [2].

After reviewing the parameters of the action you can decide on the action type. To select the action type the drop down menu on the bottom right is used. The menu presents all options from the POMP categories, see figure 4.

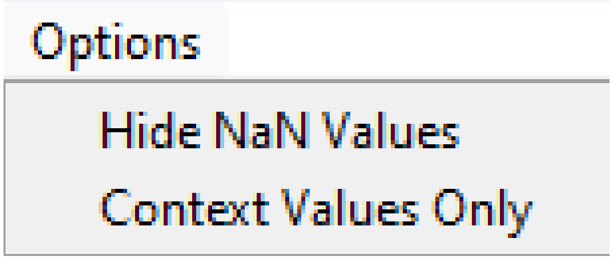


Fig. 3. Options Menu

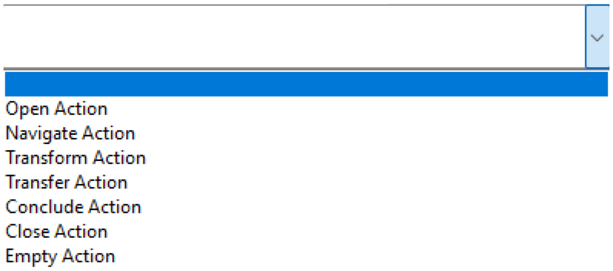


Fig. 4. Available actions

After selecting the action you have two possible choices to tag the file and one to aboard the tagging process, see figure 5 and the following explanation:

- **Submit & go to next line:** This button stores the tag and displays the next untagged action from the user interaction log. You can repeat this until all untagged, unique actions have a POMP tag assigned. **Note!** Only if you store the tags by clicking the second button (Finish and Tag files) the tags will be stored in the .csv file.
- **Finish and Tag files:** This button will use the tagged actions and assign the same tag to all equal user actions. An equal user action is every action with the same context parameters. **Note!** Only this button does store the tags into the .csv or .xes file.
- **Finish (w/o Tagging Files):** This button does discard all changes you have done to the log and will not store the changes into your file.

c) Some more functions: We have added two useful Python functions for working with user interaction logs:

Running **'python concat.py'** will create a single file from all log files stored in the folder that is specified in the concat.py file, e.g. *'logs/uilogs'*.

Running **'python stats.py'** will provide you with a terminal response about the files in the folder *'logs/uilogs*. **Note!** This folder is stored as constant in the *'const.py'* as variable named *path_to_untagged*. Next to the terminal response, there will be an *.xml* file and a *.pickle* file stored in the folder **uiobjects**. The .xml file contains the stats of the processed files, see figure 6. The .pickle file contains a set-type Python object that can be used for future processing, especially for time consuming operations, e.g. calculation of all unique user actions from long

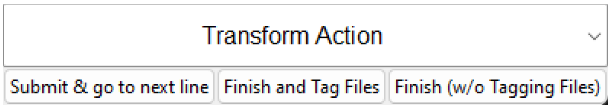
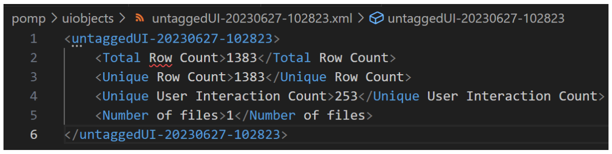


Fig. 5. Buttons for tagging


pomp > uiobjects > untaggedUI-20230627-102823.xml > untaggedUI-20230627-102823

```xml
<untaggedUI-20230627-102823>
    <Total Row Count>1383</Total Row Count>
    <Unique Row Count>1383</Unique Row Count>
    <Unique User Interaction Count>253</Unique User Interaction Count>
    <Number of files>1</Number of files>
</untaggedUI-20230627-102823>
```

Fig. 6. .xml-Statistics-file created by running stats.py

user interaction logs. The .pickle can be loaded again. If the folder **uiobjects**, which should contain the .xml and .pickle file, does not exist it will be created automatically.

V. Error Log

Currently there are no known errors.

References

[1] Simone Agostinelli, Marco Lupia, Andrea Marrella, and Massimo Mecella. Smartrpa: A tool to reactively synthesize software robots from user interface logs. In Selmin Nurcan and Axel Korthaus, editors, *Intelligent Information Systems*, volume 424 of *Lecture Notes in Business Information Processing*, pages 137–145. Springer International Publishing and Imprint Springer, Cham, 2021.

[2] V. Leno, A. Polyvyanyy, M. La Rosa, M. Dumas, and Fabrizio Maria Maggi. Action logger: Enabling process mining for robotic process automation. In *Proceedings of the Dissertation Award, Doctoral Consortium, and Demonstration Track at BPM 2019*, volume 2420 of *CEUR Workshop Proceedings*, page 5. CEUR-WS, 2019.