Multi-Log Comparison Using Transition Systems

Steffen Anhäuser[†], Lukas Häußler[†], Lasse Jacob, Harry H. Beyel[*] and
Wil M. P. van der Aalst

RWTH Aachen University, Aachen, Germany

Abstract

Process comparison provides vital insights for practitioners. However, existing approaches for log comparison are mostly limited to two event logs, and multiple comparisons increase the risk of false-positive findings. We present our tool COMET for visually comparing multiple event logs. The approach abstracts logs using transition systems and visualizes merged and individual behavior. Statistical tests, post-hoc analysis, and effect sizes are used to detect significant differences in occurrences and elapsed-time metrics. The results are directly integrated into the visualization through color scales and element thickness, allowing users to identify relevant deviations.

Keywords

Process Mining, Transition System, Event-Log Comparison, Visual Analysis

1. Introduction

Within process mining [1], process comparison is used to detect differences between processes. Such comparisons can be used to identify differences between seasonal executions or executions across an organization with multiple locations. For this task, visual support is important. A prominent approach is presented in [2]. In this work, two event logs are combined to create a Transition System (TS), a model whose states represent the stages a case can be in and whose transitions represent the activities that move a case from one stage to the next. Each log is then used to annotate the mined TS, labeling its states and transitions with how often and how quickly cases pass through them. Based on these annotations, two tests are applied. First, the two-tailed *Welch's T-test* (the unequal-variances t-test) [3] is used to check for statistically significant differences. Second, *Cohen's d* [4] is used to measure the effect size. A clear limitation is the comparison of only two event logs at a time. If one wants to compare more than two event logs, comparing all possible pairs inflates the risk of Type-I errors, i.e., reporting differences that arise by chance rather than because they exist. To overcome this limitation, we present our tool *COMET* (Comparison Of Multiple Event logs with Transition systems). First, as before, we allow the user to mine a TS. TSs provide a useful compromise for interpretability between over-generalizing directly-follows graphs and complex Petri nets. As in the earlier work, annotations are used to add frequency and elapsed-time information. Second, we employ a test pipeline to assess whether observed differences between logs are statistically relevant. The pipeline first checks the assumptions of the data and then applies an appropriate group comparison test. If differences are found, branch-appropriate post-hoc tests identify the pairs of logs that differ, and an effect size quantifies how large each difference is. The p-values are displayed as a heatmap to make statistically different pairs easy to identify, and the effect sizes are used to color states and transitions of the TS. We provide an interactive variant selection that displays the frequency of each variant per event log during selection, which occurs before mining a TS. We display a TS for each event log to enable an intuitive control-flow comparison. Shared elements in individual TS and merged TS are highlighted when interacting with one of them.

Proceedings of the BPM 2026 Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Forum (BPM 2026) , Toronto, Canada, September 27th to October 2nd, 2026.

[*]Corresponding author.
[†]These authors contributed equally.
steffen.anhaeuser@rwth-aachen.de (S. Anhäuser); lukas.haeussler@rwth-aachen.de (L. Häußler);
lasse.jacob@rwth-aachen.de (L. Jacob); beyel@pads.rwth-aachen.de (H. H. Beyel); wvdaalst@pads.rwth-aachen.de
(W. M. P. van der Aalst)
0000-0002-6541-3848 (H. H. Beyel); 0000-0002-0955-6940 (W. M. P. van der Aalst)

Figure 1: The workflow of the tool: (1, red) upload logs → (2, orange) choose settings → (3, yellow) select variants → (4, green) analyze the combined transition system, and (5, purple) individual transition systems.

2. Tool

The tool, an introduction video, and a small case study conducted on the Road Traffic Fine Management log [5] are available at https://git.rwth-aachen.de/lukas_haeussler/blc. An overview of the workflow of our tool is shown in Figure 1. After uploading event logs, a user can select different parameters to mine a TS. Then, a user can select variants across the uploaded event logs that should be considered for the analysis. Finally, the results, including the combined TS, can be inspected. In the following, we discuss the settings, the functionality and usage of the variant selector, and the combined and individual TSs in greater detail.

2.1. Settings

In the settings, it is possible to upload the logs the user wants to analyze via drag-and-drop. Furthermore, there are various options for the TSs:

- The abstraction can be switched to either *Set*, *Multiset* or *Sequence*.
- The metric that is measured can be switched between *Number of Occurrences* or *Elapsed Time Between An Event and Start*.
- The direction setting dictates whether the TS defines a case's current state based on the historical path taken to get there (*past*) or the remaining steps required to finish the process (*future*).
- The window setting, where the user can choose how many steps a state will represent.
- Configuration of the statistical tests.

2.2. Variant Selector

The variant selector shows every variant that can be found in all logs that have been uploaded. They are listed and sorted by their overall frequency, enabling users to quickly identify the most relevant ones. For readability reasons, activity names within variants are displayed in an abbreviated form using the first three letters. The full activity names can be viewed via tooltips when hovering over the labels. In addition, a bar chart visualizes the distribution of each variant across the individual logs, supporting an initial comparison before analyzing the TSs. Hovering the cursor over the bar chart reveals the absolute and relative frequencies of the current variant in each log, where the relative frequency is the share of that log's cases which follow the variant. The user can select and deselect variants to be included in the analysis. To select a variant, the corresponding checkbox is clicked and the variant is then included in the TS generation as well as the statistical analysis. Unchecking the corresponding checkbox excludes the variant from the generation and analysis. When selecting or deselecting any variant, the TS generation and statistical analysis automatically run.

2.3. Combined Transition System

The combined TS provides an overview of the analyzed process behavior across all event logs. It is constructed by merging the selected variants from all logs into a single log, used as input for mining the TS, which is the main visualization for identifying differences between logs. Nodes and edges of the TS represent states and transitions derived from the PM4Py [6] discovery algorithm using the configured abstraction and direction settings. To ensure consistency and comparability between TSs, the layout is fixed, meaning states share the same position across all individual TSs.

The combined TS has the following features:

- The **thickness** of edges indicates how frequently they occur in the merged log.
- All elements are **color-coded** by effect size, so that visually stronger differences stand out. Elements with a negligible effect are shown in blue. From small to strong, the color is interpolated on a green-to-red scale, with redder elements indicating larger differences. The same scale is applied regardless of which statistical test produced the effect size, so equal colors denote equal effect sizes across the graph. The exact thresholds are described in Section 3.
- All elements are also **interactive**: clicking on them displays additional information and statistics for the selected state or transition in a popup window. Clicking the same element again closes the popup and returns to the default view. Clicked elements are also highlighted across all TSs.
- The view of all TSs can be moved using mouse-drag and can be zoomed in and out using the mouse wheel.

2.4. Individual Transition System

On the bottom of the dashboard view (see Figure 1), the individual TSs for the selected variants are shown for every log that has been uploaded. It is possible to scroll to the right in case not all TSs fit onto the screen. As in the combined TS, the thickness of the edges indicates the occurrence of the transition.

3. Implementation

The implementation of our tool is based on a dockerized Streamlit setup. Streamlit provides a lightweight web-based user interface, while Docker bundles all required dependencies into a reliable execution environment. Hence, the tool can be used as a standalone offline web application without requiring users to manually install Python packages or configure the runtime environment. This setup simplifies deployment and keeps all analysis local to the user's machine. After uploading multiple event logs in XES or CSV format, the tool performs a compatibility check based on common activity names. Performing the analysis requires event logs to share at least one common activity. If this prerequisite is not fulfilled, a warning will appear and further steps are prohibited to prevent errors. The user can then select the variants to be included in the analysis through the variant selector on the main dashboard. Based on the selected variants, TSs are then generated using PM4Py [6]. Individual and combined TSs are restricted to this subset of variants, and shared states across all TSs are displayed at the same position in the PyVis graph, ensuring consistency across the visualizations and enhancing readability.

The statistical analysis pipeline is executed after the user has selected any variant to be included or excluded in the comparison and is restricted to the selected variants. Every change in the variant selector directly modifies the underlying data basis for the statistical tests, i.e., changing frequencies as well as time measurements, making previously computed statistical results invalid. Therefore, recomputing statistics is necessary after every change in the variant selection.

For the given subset and metric, the Shapiro-Wilk test [7] is used first to assess normality. If normality is violated, the tool applies the non-parametric Kruskal-Wallis test [8]. If significant differences are detected, the tool continues with Bonferroni-corrected Dunn's test [9] to identify pairwise differences and computes the bias-corrected rank-based effect size $\varepsilon^2 = (H - k + 1)/(n - k)$ [10], where H is the Kruskal-Wallis statistic, k the number of logs, and n the total number of cases. If normality is not violated, the tool applies Levene's test [11] to test homogeneity of variances. Depending on the result, the tool then applies either one-way ANOVA [12] (homogeneous variances) or Welch's ANOVA [13] (unequal variances). If significant differences are detected, the tool continues with a branch-specific post-hoc test to identify pairwise differences, namely Tukey's HSD [14] after one-way ANOVA and Games-Howell [15] after Welch's ANOVA. For the parametric branch, the tool computes a matching effect size, namely η^2 [16] or ω^2 [17] after ANOVA. Both measures express the proportion of (rank-)variance in the metric attributable to the log a case belongs to and are bounded in $[0, 1]$, which lets us interpret them on a common scale derived from Cohen's conventional variance-based benchmarks [4]: negligible (< 0.01), small ($0.01 \leq e < 0.06$), medium ($0.06 \leq e < 0.14$), and strong ($e \geq 0.14$), where e denotes the effect size. Applying the η^2 benchmarks to ε^2 is an interpretation by analogy, as the two measures share the same variance-explained meaning but not their exact sampling behavior, particularly for small or strongly unbalanced samples. Elements with an effect size below 0.01 are colored blue; above this value, the element color is interpolated linearly from green (small) to red (strong). The results are stored for each element of the TS and are directly applied to the visualization.

4. Conclusion

In this paper, we presented a visual analysis tool for comparing more than two event logs at once using TSs, without the inflated false-positive risk of repeated pairwise testing. The tool combines statistical analysis with an intuitive dashboard-based visualization, enabling users to identify significant differences between several event logs in terms of frequencies and elapsed time.

Maturity. The tool has reached the stage of a functional prototype and implements the complete workflow for visually comparing multiple event logs. During development, it was tested with exemplary real-life event logs that were split into different groups to validate the analysis pipeline and visual output. These example logs as well as a video demonstration are available in the Git repository (see Section 2). While no extensive real-life case study or scalability benchmark has been conducted yet, the Docker-based deployment supports reproducible execution as a standalone application.

Limitations. Since it is mainly designed as a visual analysis tool for human users, large event logs and complex or large TSs can quickly become difficult to interpret. In such cases, the visualization may contain too many nodes, edges, and statistical results to extract valuable insights efficiently. Also, the tool does not provide preprocessing functionality such as pruning or splitting, so the event logs have to be prepared by the user.

Future work. Future work could address these limitations by adding preprocessing and log-splitting functionality directly to the tool. Event logs could be automatically split based on case attributes such as patient group, location, department, or time windows to provide users with relevant comparison groups without preparing separate event logs manually. In addition, the tool could be extended with additional metrics to support a broader range of comparisons. Also, a TS can be configured in many ways, each having an impact on the result. As such, an automatic way or a guideline for finding an appropriate abstraction is needed. Moreover, a minimization technique that preserves comparability between TSs can increase the usability of COMET.

Declaration on Generative AI

During the preparation of this work the authors did not use any generative AI for writing this paper. For developing the tool, generative AI was used.

References

[1] W. M. P. van der Aalst, Process Mining - Data Science in Action, Second Edition, Springer, 2016. doi:10.1007/978-3-662-49851-4.

[2] A. Bolt, M. de Leoni, W. M. P. van der Aalst, A visual approach to spot statistically-significant differences in event logs based on process metrics, in: CAiSE, Springer, 2016, pp. 151–166. doi:10.1007/978-3-319-39696-5_10.

[3] B. L. Welch, The generalization of 'student's' problem when several different population variances are involved, Biometrika 34 (1947) 28–35.

[4] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, Routledge, New York, 1988.

[5] M. M. de Leoni, F. Mannhardt, Road traffic fine management process, 2015. doi:10.4121/UUID:270FD440-1057-4FB9-89A9-B699B47990F5.

[6] A. Berti, S. van Zelst, D. Schuster, Pm4py: A process mining library for python, Software Impacts 17 (2023) 100556. doi:10.1016/j.simpa.2023.100556.

[7] S. S. Shapiro, M. B. Wilk, An analysis of variance test for normality (complete samples), Biometrika 52 (1965) 591–611. doi:10.1093/biomet/52.3-4.591.

[8] W. H. Kruskal, W. A. Wallis, Use of ranks in one-criterion variance analysis, Journal of the American Statistical Association 47 (1952) 583–621.

[9] O. J. Dunn, Multiple comparisons using rank sums, Technometrics 6 (1964) 241–252.

[10] M. Tomczak, E. Tomczak, The need to report effect size estimates revisited. an overview of some recommended measures of effect size, Trends in Sport Sciences 21 (2014) 19–25.

[11] H. Levene, Robust tests for equality of variances, 1960.

[12] R. A. Fisher, Statistical Methods for Research Workers, Oliver and Boyd, Edinburgh, 1925.

[13] B. L. Welch, On the comparison of several mean values: An alternative approach, Biometrika 38 (1951) 330–336.

[14] J. W. Tukey, Comparing individual means in the analysis of variance, Biometrics 5 (1949) 99–114. doi:10.2307/3001913.

[15] P. A. Games, J. F. Howell, Pairwise multiple comparison procedures with unequal n's and/or variances: A monte carlo study, Journal of Educational Statistics 1 (1976) 113–125.

[16] J. T. E. Richardson, Eta squared and partial eta squared as measures of effect size in educational research, Educational Research Review 6 (2011) 135–147. doi:10.1016/j.edurev.2010.12.001.

[17] W. L. Hays, Statistics for Psychologists, Holt, Rinehart and Winston, New York, 1963.