



# Transmute: an Interactive Tool for Assisting Knowledge Discovery in Interaction Traces

Pierre-Loup Barazzutti, Amélie Cordier, Béatrice Fuchs

## ► To cite this version:

Pierre-Loup Barazzutti, Amélie Cordier, Béatrice Fuchs. Transmute: an Interactive Tool for Assisting Knowledge Discovery in Interaction Traces. [Research Report] Université Claude Bernard Lyon 1; Université Jean Moulin Lyon 3. 2015. <hal-01172013>

**HAL Id: hal-01172013**

**<https://hal.archives-ouvertes.fr/hal-01172013>**

Submitted on 6 Jul 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Transmute: an Interactive Tool for Assisting Knowledge Discovery in Interaction Traces

Pierre-Loup Barazzutti<sup>1,2</sup>, Amélie Cordier<sup>1,2</sup>, Béatrice Fuchs<sup>1,3</sup>.

<sup>1</sup> LIRIS CNRS, UMR 5202,

<sup>2</sup> Université Lyon 1,

<sup>3</sup> Université Lyon 3, France.

**Abstract.** Analysis of traces is an increasingly active issue in knowledge discovery (KD). Unfortunately, to date, few tools are designed to help interactively the analyst during the knowledge discovery process. In this paper, we present TRANSMUTE, a tool to assist the analyst throughout an interactive KD process in traces. TRANSMUTE displays the results of the mining together with the source trace so that it is easier for the analyst to understand them and thus to refine the parameters of the algorithm accordingly. TRANSMUTE is developed as a web platform and relies on a Python implementation of the KD process.

## 1 Introduction

In this paper, we introduce TRANSMUTE, an interactive tool for supporting knowledge discovery (KD) from data represented as traces. A trace is a record of temporally situated observed elements.

KD aims to analyze data through an iterative process composed of several steps. During this process, interactions with analysts (users, domain experts, etc.) play a leading role. Many tools provide all kinds of visualization of the results of the mining, but their interaction abilities often remain limited to graphical manipulation.

We developed TRANSMUTE to enhance interactivity in the KD process, and to bring assistance to the expert in all possible steps. In this paper, we focus on the interpretation step. We propose a scenario in which the analyst can dynamically observe the effects of his actions and choices as his work progresses.

In the remaining of this paper, we describe our application context and we outline some of the interactive features of TRANSMUTE. We report on the current state of the development of TRANSMUTE in the conclusion.

## 2 Knowledge Discovery from Traces

The context of our work is that of knowledge extraction from traces. A trace is a set of observed elements temporally situated called *obsels*. A trace is associated with a trace model describing the types of obsels, their attributes and their

relationships with other obsels. This model allows to interpret the content of the trace and to facilitate its subsequent exploitation [1].

*Transformations* are operations that apply on traces and produce new traces called *transformed traces*. Several types of transformations exist. Among them, the *rewrite* transformation enables one to rewrite a trace at a higher level of abstraction, thus gradually increasing its level of understandability. Therefore, the rewriting of  $t_1$  consists in producing a trace  $t_2$  in which some sequences of obsels (patterns) of  $t_1$  are replaced by new obsels resuming each pattern. The rewriting of traces is one of the central features of TRANSMUTE.

The KD process implemented here uses traces in a cycle composed of three main phases: pre-processing (collect, data selection, transformations), mining, and post-processing (visualization, interpretation). The mining step relies on a sequence mining algorithm called DMT4SP<sup>4</sup> which detects frequent sequential episodes given constraints specified by the analyst. During the post-processing, the results of the mining are given to the analyst who chooses the patterns which are the most relevant regarding its knowledge of the domain.

### 3 Transmute

TRANSMUTE architecture is organized around several modules (cf. Figure 1).

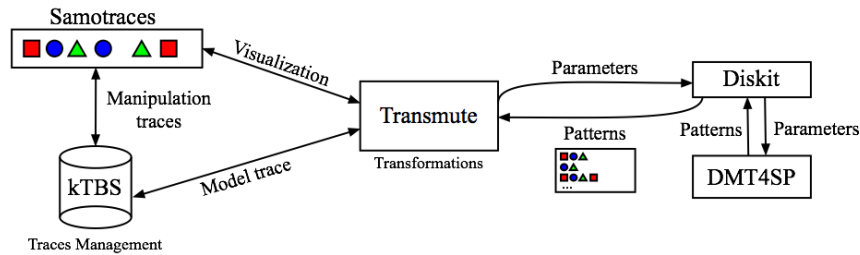


Fig. 1. TRANSMUTE: architecture

TRANSMUTE relies on Samotraces<sup>5</sup>, a Javascript framework for building trace visualizations. Samotraces enables the configuration of the display of obsels depending on some of their characteristics such as their types, attributes, etc. TRANSMUTE also uses Samotraces for communicating with a trace manager called ktBS (kernel for Trace Based System). A ktBS<sup>6</sup> is a system providing all the basic manipulations on the traces: collect, processing, transformation, exportation.

<sup>4</sup> Data Mining Techniques For Sequence Processing <http://liris.cnrs.fr/~crigotti/dmt4sp.html>

<sup>5</sup> <https://github.com/bmathern/samotraces.js>

<sup>6</sup> <http://tbs-platform.org/tbs/doku.php>

The KD process is implemented in the DISKIT<sup>7</sup> module. For the mining step, DISKIT uses DMT4SP to find frequent serial episodes from one or several sequences of events. DMT4SP supports numerous types of constraints such as support, temporal constraints (window, gap), syntactic constraints (length, prefix, suffix), etc. It is completed during the post-processing by other types of constraints such as the search of closed patterns (as in [2]), constraints on presence or absence of specific patterns, in order to reduce the number of result patterns. Currently, DISKIT is limited to the processing of a single trace at a time.

Figure 2 shows the interface of TRANSMUTE which has three main parts. The upper part displays the trace being analyzed. The middle part represents the working trace in which patterns are displayed. The bottom part displays the patterns outputted by the miner. The trace model is displayed on the right hand side.



**Fig. 2.** Main interface of TRANSMUTE

This demonstration focuses on the post-processing phase. The mining step produces a huge number of patterns, often with a high combinatorial redundancy. The analyst has to process these patterns to choose relevant ones regarding his expertise of the domain. We propose a scenario to assist the interpretation step.

The set of patterns is enriched with relevance measures that can be used to sort results and help highlighting the most interesting ones. Every time a pattern is selected by the analyst, the effect of its choice can be observed both on the trace and on the list of patterns. It is possible to visualize and to access

<sup>7</sup> Discovering Knowledge from Interaction Traces

all the occurrences of the patterns on the trace in order to estimate the relevance of this choice. For a given selected pattern, the analyst can create a new obsel type that will be used to replace every occurrence of the selected pattern in the transformed trace. The list of patterns is then dynamically updated by discarding all patterns having at least one obsel in common with the selected one: they are shaded on the interface. The relevance measures of the remaining patterns are then recalculated to take into account the selected pattern and the list of patterns is sorted accordingly. Patterns which measure values do not satisfy constraints anymore are then automatically discarded, leading to a gradual decrease of the number of remaining patterns, thus making easier the next choice by focusing on other patterns. Once the analyst estimates that all interesting patterns have been chosen, he can confirm and trigger the effective creation of a new transformed trace. As a result, all patterns previously discarded are definitively removed.

The analyst has then several possibilities: continue the interpretation from remaining patterns or repeat the KD process by using other constraints on the same trace or on the newly created transformed trace.

## 4 Conclusion and future work

TRANSMUTE was originally designed as a tool for interactively visualizing and transforming traces. It provides many features for creating obsels, manipulating traces and visualizing/storing transformation results. The motivation underlying this work is to provide activity analysts with interactive tools supporting them in their task of interpretation of traces.

In this work, we have enriched TRANSMUTE with KD features. TRANSMUTE now offers, among other possibilities, a set of features for setting the parameters of a mining algorithm, for running it, and for displaying its results in order to build transformed traces.

So far, we focused on the post-processing step of the KD process and the manipulation of the results. In the next step, we will focus on the implementation of assistance features during the pre-processing phase. More precisely, we want to help the analyst in the parameter settings phase. For that, we consider two directions: (1) getting towards a fully interactive KD cycle supported by TRANSMUTE, in which the interactions of the analyst with the result of the mining are used to guide the refining of the parameters and (2) recommending parameters to the user from previous KD experiences by using a Case-Based Reasoning approach.

## References

1. Cordier, A., Lefevre, M., Champin, P.A., Georgeon, O., Mille, A.: Trace-Based Reasoning — Modeling interaction traces for reasoning on experiences. In: The 26th International FLAIRS Conference. (2013)
2. Tatti, N., Cule, B.: Mining closed strict episodes. In: ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010. (2010) 501–510