ToscanaJ – An Open Source Tool for Qualitative Data Analysis

Peter Becker[1] and **Joachim Hereth**[2] and **Gerd Stumme**[3]

Abstract. In this paper, we report about the current state of ToscanaJ, the latest member of the family of TOSCANA programs, that have been used for conceptual data analysis and knowledge discovery. After presenting the tool itself, we shortly describe the planned Tockit framework surrounding ToscanaJ, that will provide an Open Source toolset for programs based on Formal Concept Analysis and the planning for future development.

1 Introduction

Grounded on the idea of restructuring lattice theory, the development of Formal Concept Analysis (FCA) was started (cf. [16, 2]) about twenty years ago. FCA is based on the conceptualization of the concept 'concept' and supports –in a mathematically founded way– the conceptual representation and visualization of data and knowledge. In the context of research and development of this theory, many ideas have been applied in projects in practice, resulting in the implementation of several programs by the Darmstadt research group on FCA. One of the most widely used programs is TOSCANA, a frontend for conceptual information systems. Basically, it displays predefined diagrams of conceptual structures, allowing to browse and navigate through complex data sets, using a simple graphical interface.

The screenshot of ToscanaJ in Fig. 1 exemplifies this interface, the original TOSCANA interface is very similar. The screenshot shows two diagrams nested one into the other. Both have been defined by the knowledge engineer who has built the system. In the analysis process the user can use any predefined diagram or combinations of two (or more) diagrams to get more detailed information. Clicking on one node results in a filter process, thus that then only the objects belonging to the selected node will be used for the following analysis. A more detailed description can be found, for instance, in [14].

In the sequel, we will discuss some shortcomings of the implementation of the original TOSCANA, together with our attempt to overcome them.

2 Conceptual Knowledge Discovery

TOSCANA systems have been successfully applied for many different purposes, including knowledge discovery. The connections between Knowledge Discovery in Databases and the principle ideas underlying TOSCANA systems have been elaborated in [12, 5]. The principle goal of *Conceptual Knowledge Discovery* (CKDD) is to support a human-centered process of knowledge discovery by providing a visualization of the data based on a visualization of underlying conceptual structures. This idea is explained in more detail in [15, 4].

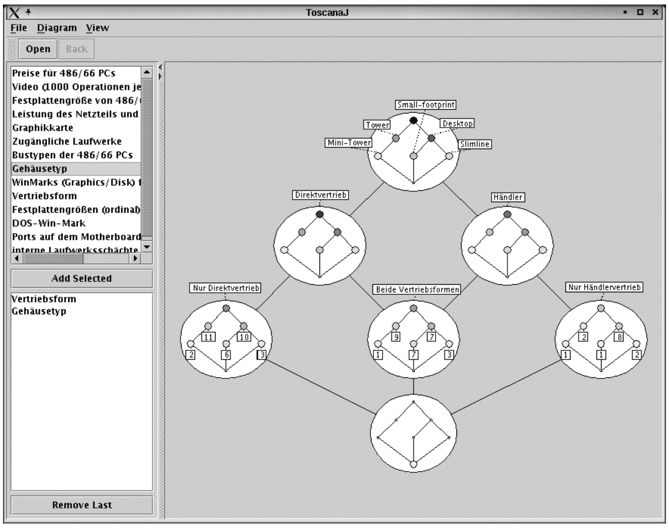

Figure 1. Screenshot of ToscanaJ with nested diagram and highlighting

Most notably in this CKDD context, some shortcomings of the older versions of TOSCANA became apparent. While they have been extremely useful in many projects, it became impossible to add new features, due to the complexity of the code grown over the years without any major refactoring. One such requirement has already been stated in [13], the support of numerical operations in TOSCANA. For related goals, such as on-line analytical processing, this feature is also very important (cf. [11, 3]). With TOSCANA 3, an attempt for a new version based on the old code was made, but although it was successful in its way, some problems with the old design and the underlying libraries could not be overcome.

3 The new approach

Due to these problems the KVO group at Griffith University initiated a new TOSCANA project from scratch. The first code was written during a three-day session at the KVO laboratories. Afterwards the project was ported to Sourceforge where it was put under a BSD-style license, to make the exchange of code as simple as the exchange of

[1] Distributed System Technology Centre (DSTC), Level 12 S Block, QUT Gardens Point, GPO Box 2434, Brisbane QLD 4001, Australia, peter@peterbecker.de

[2] Department of Mathematics, Darmstadt University of Technology, Schloßgartenstr. 7, D-64289 Darmstadt, Germany, hereth@mathematik.tu-darmstadt.de

[3] Institut für Angewandte Informatik und Formale Beschreibungsverfahren (AIFB), Karlsruhe University, D-76128 Karlsruhe, Germany, stumme@aifb.uni-karlsruhe.de

ideas has already been in the past.

As platform for the new development Java was chosen. Java offers virtual machines for most common operating systems and since it is very common as programming language in universities, it makes the project more accessible for many students. Although Javas main GUI library has a number of technical and conceptual drawbacks, the existence of a large set of standard libraries for different purposes is easing development.

The new file format for ToscanaJ is based on the W3C specification XML [10]. XML allows to define formats that are human-readable, easy to parse and easy to transform into other formats. In addition it allows offering compatibility between different versions of the file format in both directions.

4 Current Status

At the moment, the implementation offers most of the important features from TOSCANA 2 and 3, adding some new ones which will be described in the following section. It does draw line diagrams with up to one level of nesting and is able to handle data either given as part of the TOSCANA file itself or by accessing external data sources using the JDBC interface.JDBC allows connecting to all common database management systems either by using a native JDBC driver or by using the JDBC-ODBC bridge.

The user interface has been simplified in many ways, mostly by dropping features that turned out to be irrelevant in practice for common TOSCANA projects. A packaged distribution including examples allows for easy setup of the tool. It has been tested on a number of different Java Runtime Environments and can be downloaded from the ToscanaJ homepage [9].

5 Advanced Features in ToscanaJ

Some of the most interesting new features in ToscanaJ are:

- color is used to indicate the size of the concept extent (number of objects belonging to the corresponding node),
- the set of nodes above and below a selected node can be highlighted by clicking on it, even in nested diagrams,
- diagrams can be exported in the SVG vector graphic format,which can be converted into other common graphic formats, e. g. Encapsulated Postscript (eps) or Portable Networks Graphic (png),
- not only simple lists or object counts can be queried from the underlying database, but also arbitrary aggregates and formatted combinations of columns, thus fulfilling certain requirements for more complex CKDD,
- and last but not least easy extensibility for new features.

In addition to this, ToscanaJ is the first TOSCANA running on multiple platforms and it can be run without a database – a feature that got lost early in the development of TOSCANA 2. Finally, the user interface has become more intuitive while offering the same relevant features.

6 Outlook

The further development of ToscanaJ is meant to be project-driven. The most basic features are there so the program offers enough functionality to be useful. Due to the accessible structure of the code and the open development process, new features should be rather easy to add as long as they fit into the TOSCANA workflow model. ToscanaJ is supposed to supersede the currently used versions TOSCANA 2 and TOSCANA 3 by offering a superset of the relevant features of both programs.

While ToscanaJ is at the moment still an independent project, it is considered to be a first step in the direction of a new Java-based framework called Tockit [8] with reusable components to build FCA-related programs. It is also planned as Open Source software and shall give researchers the option to implement prototypes more easily, and end users the option to create in a straightforward manner conceptual information systems according to their respective projects needs.

One of the promising upcoming research topics is the combination of FCA with Ontology Engineering. Our aim is to establish links between both domains both on the theoretical as on a practical level. As a basis for future implementations, the combination of Tockit with the Karlsruhe Ontology framework KAON [7] is planed. As Tockit, KAON provides an Open Source development framework, called OntoMat [6]. We are currently discussing possibilities to integrate Tockit and OntoMat on a technical level, accompanying our research about a theoretical framework for integrating Formal Concept Analysis and Ontology Engineering.

REFERENCES

[1] Bernhard Ganter and Rudolf Wille, *Formale Begriffsanalyse: Mathematische Grundlagen*, Springer–Verlag, Berlin–Heidelberg, 1996.

[2] Bernhard Ganter and Rudolf Wille, *Formal Concept Analysis: Mathematical Foundations*, Springer–Verlag, Berlin–Heidelberg–New York, 1999. Translation of [1].

[3] Joachim Hereth, *Formale Begriffsanalyse im Data Warehousing*, Diploma thesis, TU Darmstadt, April 2000.

[4] Joachim Hereth, Gerd Stumme, Uta Wille, and Rudolf Wille, 'Conceptual knowledge discovery – a human-centered approach'. To appear in the JETAI special issue on Concept Lattices for KDD.

[5] Joachim Hereth, Gerd Stumme, Uta Wille, and Rudolf Wille, 'Conceptual knowledge discovery and data analysis', in *Conceptual Structures: Logical, Linguistic, and Computational Issues*, eds., Bernhard Ganter and Guy W. Mineau, volume 1867 of *Lecture Notes in Computer Science*, pp. 421–437, Berlin–Heidelberg–New York, (2000). Springer–Verlag. Proceedings on the 8th International Conference on Conceptual Structures, Darmstadt, Germany, August 14–18, 2000.

[6] http://kaon.aifb.uni karlsruhe.de/IDE. OntoMat– the KAON GUI framework.

[7] http://kaon.semanticweb.de. KAON - the karlsruhe ontology and semantic web tool suite.

[8] http://tockit.sourceforge.net. Tockit project pages.

[9] http://toscanaj.sourceforge.net. ToscanaJ homepage.

[10] http://www.w3.org/XML. XML homepage of the W3C.

[11] Gerd Stumme, 'Conceptual on-line analytical processing', in *Information Organization and Databases*, ed., Y. Kambayashi K. Tanaka, S. Ghandeharizadeh, Boston–Dordrecht–London, (2000). Kluwer.

[12] Gerd Stumme, Uta Wille, and Rudolf Wille, 'Conceptual knowledge discovery in databases using formal concept analysis methods', in *Principles of Data Mining and Knowledge Discovery*, eds., J. M. Zytkow and M. Quafofou, volume 1510 of *Lecture Notes in Computer Science*, pp. 450–458. Springer–Verlag, (1998).

[13] Gerd Stumme and Karl Erich Wolff, 'Computing in conceptual data systems with relational structures', in *Proceedings of the International Symposium on Knowledge Representation, Use and Storage Efficiency*, Vancouver, (1997).

[14] Frank Vogt and Rudolf Wille, 'Toscana — a graphical tool for analyzing and exploring data', in *Graph Drawing*, eds., Roberto Tamassia and Ioannis G. Tollis, pp. 226–233, Heidelberg, (1995). Springer–Verlag.

[15] Rudolf Wille, 'Why can concept lattices support knowledge discovery in databases?'. To appear in the JETAI special issue on Concept Lattices for KDD.

[16] Rudolf Wille, 'Restructuring lattice theory: an approach based on hierarchies of concepts.', in *Ordered sets*, ed., Ivan Rival, pp. 445–470, Dordrecht–Boston, (1982). Reidel.