FAKULTÄT
FÜR !NFORMATIK

Faculty of Informatics

TempMunger

A Visual Analytics Approach Supporting Transformations of Time-Oriented Data

MASTER'S THESIS PROPOSAL

for the degree of

Diplom-Ingenieur

in

Computer Science – Master – 066 933
Information & Knowledge Management

by

Robert Thurnher
Registration Number 0004297

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Univ.-Prof. Silvia Miksch
Assistance: Dr. Theresia Gschwandtner

Wien, 18.11.2012

(Signature of Advisor)

Contents

CHAPTER 1

Abstract

1.1 Problem Description

Applied work within **Visual Analytics**, *"the science of analytical reasoning facilitated by interactive visual interfaces"* [10], can be seen to roughly consist of three main basic building blocks[1]:

1. Statistics & machine learning
2. **Data wrangling** a.k.a. **munging**
3. Visualization & analysis itself

While the first and last mentioned fields are constantly evolving related tools and techniques, the second one is comparatively still a bit lacking. This is when it comes down to actually wrangle usually messy real-world data into a format prepping it useful for analysis.

Currently, it mostly means fiddling around with the data manually, applying hand-crafted transformation scripts. This is a tedious task and discourages analyzing data altogether, especially if the ones intending to work on are not technically expertized (for instance, journalists).

However, combining contemporary knowledge and technology from the domains of **Human-Computer Interaction** (HCI) & **User Experience** (UX) as well as **Information Retrieval** (IR) [6], **Data Mining / Machine Learning** (ML) [11], and **Visual Analytics** (VA) could yield substantial improvements here. Namely, making data wrangling accessible to a wider audience, mainly by providing a combination of analytical and visual methods to support the task of data munging. In this thesis we are going to scientifically investigate the field of data wrangling, plus, design, and prototypically implement a Visual Analytics approach to support this task.

That is, iteratively creating a software prototype which enables users to munge data suitable for analysis in a way which is as agile, intuitive, interactive, and overall visual as possible, respectively reasonable [7].

[1] Cf. http://www.dataspora.com/2009/05/sexy-data-geeks/

Additionally, the to-be-developed prototype shall, in particular, make it convenient to work on **time-oriented datasets**. Time and time-oriented data have distinct characteristics that make it worthwhile to treat it as a separate data type [1].

Research Questions

So, the main research question is:
In what way can we support data munging with Visual Analytics techniques?

Plus, sub hypotheses being connected to design/implementation details. E.g.:

- *Which data transformations are best supported by analytical methods and for which transformations is visual support beneficial?*
- *How do concrete data munging workflow processes look like and and how can these processes be supported by VA methods?*
- *What data munging tasks need to be tackled in particular when dealing with time-oriented data and how can we support them with VA methods?*

The emphasis of this thesis lays on evaluating the feasibility of corresponding concepts via iterative design, implementation, and evaluation of a software prototype.

1.2 Expected Results

Results to be achieved will be:

- Design and implementation of a research prototype
- Related evaluation and findings
- Detailed documentation of these

At the end of the project, it should be known whether developing such a tool in the described context is feasible and if so, how in detail. As mentioned above, the **prototype** shall combine concepts from HCI & UX with ones from IR, ML, & VA. Special focus will be put on crafting the **UI** and an underlying **analytical inference engine** interactively providing the user with respective data transform suggestions visually. Furthermore, **direct manipulation** of data should be easy. Plus, transformations shall, generally, be easily repeatable/-usable. A central challenge is making this all work in the context of time-oriented data (see above).

Answers to the stated research questions should be given, by designing, implementing, and evaluating a research prototype which provides VA techniques to improve and support data munging tasks.

1.3 Method

In order to answer our research questions we go for these concrete scientific methods:

1. **Requirements analysis**
2. **Design of UI & interactions**
3. **Iterative prototypical implementation**
4. **Qualitative evaluation of results**

Thus, the whole development process of the prototype should be conducted iteratively in an **agile** manner until satisfying results are achieved.

In the end, it should all be thoroughly documented emphasizing findings of the evaluation and lessons learned. The thesis will cover all aspects and findings of the development and evaluation of the prototype from UI/interaction mockups to architectural diagrams.

Implementation

Technically, the prototype will be a **web-based** application which can be run locally as well.

The backend will at its core be powered by **Elasticsearch**[2], a high-performance search and data storage engine with strong, real-time analytics capabilities. Inference and probably transform operations of the backend will be based on **Apache Spark**[3], a fast engine for large-scale data processing with convenient access to ML algorithms via its **MLlib**. Spark integrates nicely with Elasticsearch via native support provided by ES for Hadoop project. The backend will expose a RESTful API built with **Kotlin**[4], Gradle build system, and **Spring Boot** framework.

The frontend will be a modern universal web browser app utilizing HTML5 with interactive charts via **SVG**, transpiled CSS, and **ES6**(+)-flavored JS. Probable technologies and patterns: Node.js, Gulp tasks, Webpack, Babel, **Redux/React** architecture, material design UI kit, **D3.js**...

1.4 State of the Art

Basics of the field are laid out in [2], mainly related to classic extract, transform, and load (**ETL**) processes as known from **data warehousing**. General theoretical foundation of transforming large amounts of data interactively can be found in [3], and more recently [4].

A system pioneering this area is *Potter's Wheel* [9] from 2001. Yet, among other things, it doesn't support "fill" transform operations (i.e., automatically filling certain fields with certain data, batch-wise) and, naturally, its usability is not up to modern standards. So, this tool resembles quite much the look & feel of a Java Swing GUI application from the late 1990s, early 2000s (which, as a matter of fact, it happens to be).

[2] https://www.elastic.co/products/elasticsearch

[3] https://spark.apache.org/

[4] http://kotlinlang.org/

The two systems which can claim to represent the current state of the art here are:

1. ***DataWrangler*** [5] (Stanford Visualization Group research project[5])
2. ***OpenRefine*** [8] (open-sourced product f.k.a. *Google Refine*, f.k.a. *Freebase Gridworks*)

DataWrangler is a web-browser-based app very much oriented towards a visually interactive approach to some extent similar to our proposed one. It contains an inference engine suggesting transforms and data cleaning sessions can be exported and reused as scripts. One thing which is not really supported by Wrangler is direct manipulation of data. In addition to that, the UI is sort of limited which can also be ascribed to the web-based nature of the tool. Things like not truly responsively and richly interactive UX, especially when amounts of to be wrangled data grow.

OpenRefine is a browser-based app as well (but running locally on the user's machine, mainly due to data privacy reasons). It allows for direct manipulation of data, yet, doesn't support as purely by interaction driven transform operations as Wrangler (receiving very appropriate transformation suggestions solely by pointing the cursor to data in a certain way). A nice feature is its visual statistical analytics of data distributions via histograms etc. But it generally lacks some data transformation operations which Wrangler has (especially reshaping-related, like un/folding – extracting/merging specific parts of data within a column into/from additional ones).

Where we're intending to excel with ***TempMunger*** here is by bringing concepts from both Wrangler and Refine together with our own ideas and improved UX with respect to VA techniques, plus, focusing on specific munging tasks and support needed when dealing with time-oriented data.

Intended features of our prototype include:

- Drag & drop column merging
- Visualizing data structures via meaningful charts
- Directly manipulative interaction with these charts to transform underlying data

We think that further reasonable functionality will emerge from the iterative *design – implementation – evaluation* process.

1.5 Topic Match

The thesis topic is a good match for the master's studies in computer science of *"Information & Knowledge Management"* because this program can somehow be seen as a blend of the master's of *"Computational Intelligence"* with the one of *"Software Engineering"*.

The topic relates closely to Software Engineering as it's in its essence an advanced, extensive software development project focusing on UI features, the efficient communication of information, and efficient support of user tasks, intending to answer given research questions in the field of VA. Moreover, it relates to Computational Intelligence due to its close relationship with applied, contemporary Data Mining / ML, IR, analytical, and visual methods.

[5] http://vis.stanford.edu/wrangler/

Bibliography

[1] Wolfgang Aigner, Silvia Miksch, H. Schumann, and C. Tominski. *Visualization of Time-Oriented Data*. Human-Computer Interaction. Springer Verlag, 1st edition, 2011.

[2] T Dasu and T Johnson. *Exploratory Data Mining and Data Cleaning*. Wiley Series in Probability and Statistics. Wiley, 2003.

[3] Tamraparni Dasu, Theodore Johnson, S Muthukrishnan, and Vladislav Shkapenyuk. Mining Database Structure; Or, How to Build a Data Quality Browser. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of Data*, SIGMOD '02, pages 240–251, New York, NY, USA, 2002. ACM.

[4] Joseph M Hellerstein. Quantitative Data Cleaning for Large Databases. United Nations Economic Commission for Europe (UNECE), 2008.

[5] Sean Kandel, Andreas Paepcke, Joseph Hellerstein, and Jeffrey Heer. Wrangler: Interactive Visual Specification of Data Transformation Scripts. *Human Factors*, pages 3363–3372, 2011.

[6] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*, volume 1. Cambridge University Press, 2008.

[7] Donald A Norman. *The Design of Everyday Things*. Basic Books, 2002.

[8] OpenRefine Git Repo. https://github.com/openrefine/openrefine. Accessed: 2012-11-03.

[9] Vijayshankar Raman and Joseph M Hellerstein. Potter's Wheel: An Interactive Data Cleaning System. *Data Base*, 01:381–390, 2001.

[10] James J Thomas and Kristin A Cook. A Visual Analytics Agenda. *IEEE Computer Graphics and Applications*, 26(1):10–13, 2006.

[11] Ian H Witten, Eibe Frank, and Mark A Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Number 2 in Morgan Kaufmann series in data management systems. Morgan Kaufmann, 3rd edition, 2011.